Genetic Programming Based ECOC for Multiclass Microarray Data Classification

Wang JiaJun, Liu KunHong(✉), Sun MengXin, Hong QingQi
Software School of Xiamen University, Xiamen, China
wangjiajun422@gmail.com; lkhqz@xmu.edu.cn; mengxinsun.xmu@gmail.com; hongqq@xmu.edu.cn

***Abstract*—A genetic programming (GP) based algorithm is proposed to improve the performance of ECOC algorithms for multiclass microarray datasets. The individual of our GP is revised to solve a set of binary class problems decomposed by an ECOC algorithm directly, which picking up genes with biological significance simultaneously. Experimental results prove the effectiveness of our algorithm in five data sets.**

Keywords-genetic programming; multiclass microarray; ECOC;

I. INTRODUCTION

The development of microarray technology allows people to classify some particular cancers using microarray datasets, so many researchers are devoted to developing algorithms for microarray data in recent years. However, as the number of samples in microarray data tends to be much smaller than the number of genes, the generalization ability of data mining methods is decreased. The multiclass problem is even harder because of heavy overlaps among different classes[1].

An efficient way to deal with a multiclass classification problem is to break it down into several binary classification problems, and the final solution is made based on the fusion of these problems. A typical algorithm is ECOC, whose framework contains a coding process to decompose a multiclass problem into a set of binary-class problems, and a decoding process to combine all the binary problems[2]. Even for the same encoding and decoding scheme, different feature subspaces can lead to difference class splitting schemes for a data dependent ECOC algorithm, so as to result in different final results. Some researchers used Genetic Algorithms to obtain higher accuracy in ECOC algorithms by optimizing ECOC coding matrix[3]. Although there are already a lot of works discussing the design of ECOC algorithms and its applications, the study of ECOC in microarray analysis research field is just in the beginning[4].

Genetic programming (GP) has been successfully applied in pattern recognition research field owing to its great ability in the discovery of implicit relationships among data[5]. And the individuals of GP can be applied as classifiers by combining important features with some special operators. Although there are some studies discussing the search of optimal coding matrix and decoding schemes using different algorithms, there is still no research work on search optimal subspace by GP. This paper proposes a GP based framework to select optimal feature subsets and produce robust classifier simultaneously for different ECOC coding schemes. The individual in our GP is adjusted to contain a set of trees (named as Forest), so as to be matched an ECOC matrix directly. Experiments are carried out for comparisons, and results prove that our GP based scheme can beat other classifiers in most cases.

In this paper, section 2 reviews ECOC and GP, and descripts the design of GP for ECOC. Section 3 shows the settings of experiments, and compares and discusses the results. Section 4 concludes this paper.

II. METHOD

A. ECOC

ECOC is a widely deployed framework for multiclass problems[2]. ECOC algorithm builds a unique "code word" for each class in the encoding process. The elements in the coding matrix of size N×M belong to the set {-1, 0, +1}. Each row represents a class, and there are N classes (N > 2). Let K denotes a set of class labels, K={k_1, k_2, …, k_N}. Let S={(X_1, y_1), (X_2, y_2), …, (X_L, y_L) }, where S represents a set of samples, X_i is the features vector representing sample S_i, and y_i is the class label to which X_i belongs, y_i∈K. L represents the number of samples. Meanwhile, each column is interpreted as a binary classifier, and the original class label is re-calibrated into binary classes, requireing a dichotomizer. Let D be a set of dichotomizers matching rows in a ECOC coding matrix, D={d_1, d_2, …, d_M}. For a sample X*, an ECOC output makes up a vector V* with length L. Then, the distance between the output vector and code words is calculated, and the code word with the minimum distance will determined as its class label.

B. Genetic Programming

GP is an extension of genetic algorithm (GA), and the syntax tree structure of its individual is the key to its achievements in many different research fields. In terms of classification, GP has been applied to analyze binary class microarray data because each individual can produce a yes/no answer for a classification problem by formulating important features. However, it is obvious that GP can't solve multiclass problems directly due to the limit of individual structure. So far, some authors proposed different methods to apply GP to deal with multiclass problems. [6] applied some base classifiers as leaf nodes, and treated each individuals as an ensemble of base learners, so as to tackle binary class and multiclass problem at the same time. In this paper, we propose a new individual structure that can match different ECOC coding matrices, as described below.

a. *The Individual Structure*

The initialization settings of our GP is the same as [7]. That is, the Ramped Half-and-Half method is applied to produce first generation with an equal number of trees initialized with

2473-3547/17 $31.00 © 2017 IEEE
DOI 10.1109/ISCID.2017.75

depth ranging from 2 to the initial maximum tree depth value. So there are both balanced and unbalanced trees with different depths. The dynamic maximum tree depth technique is used to control the scale of trees by setting the strict depth limit less than the dynamic depth limit. The initial maximum dynamic depth limit is small, so as to lead GP to produce simple trees firstly before trying more complex solutions.

Let *F*/*T* be the set of functions/terminals. *T* contains features and constants, and *F* is composed of arithmetical or logical functions. Terminal/nonterminal nodes of trees are selected from *F*/*T* sets, so trees in our GP are the set of all possible compositions of functions and terminals in *F* and *T*. So the rule embedded in a tree may take the form of: if max(X15,X12) > 0.731 then group 1; otherwise group 2. In this rule, the index number of a gene is indicated by the letter 'X', so X15 and X12 refers to the 15th and 12th gene. The function 'max', two genes, logical operator '>', and the constant 0.731 are produced by GP. The target group is set by one column in an ECOC coding matrix. When the maximum expression value of the two genes is larger than 0.731, the tree generates 'yes' (+1) and assigns the sample to the first group; otherwise, it is assigned to the second group.

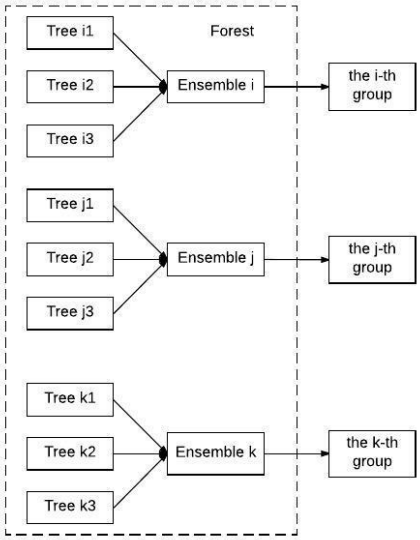

Figure 1. The structure of a Forest

In most cases, the class unbalance problem unavoidably deteriorates performances of classifiers in microarray data, so the fusion of three trees are deployed to the classification task in our algorithm based on majority voting scheme. In this way, 3×M trees are combined as an individual in our GP (named as Forest), as shown in Fig. 1. For the trees marked as i1, i2 and i3, they are combined as the i-th ensemble, solving the binary classification problem matching the i-th column of an ECOC matrix. So each Forest can solve a multiclass problem directly. The combination of answers from a Forest produce a vector, which is then compared with all codewords in the coding matrix. And an unknown sample is assigned to the i-th class if the i-th codeword obtains the shortest distance.

b. *The Fitness Function*

Due to the small training sample size, 5-fold cross validation (CV) is applied to evaluate the generalization ability of trees in each subgroup system. For the i-th column, the sample size in each group would be unequal in most cases. To solve the skewing data distribution problem, assume that the sample size in positive/negative group for the is $N_{i,p}/N_{i,n}$, and the correctly classified sample size is $C_{i,p}/C_{i,n}$, then the fitness function is set as:

$$fitness = \sum_{i=1}^{M} \frac{C_{i,p}\times\omega_{i,1}+C_{i,n}\times\omega_{i,2}}{C_{i,p}\times N_{i,p}+C_{i,n}\times N_{i,n}} \quad (1)$$

where $\omega_{i,1}=N_{i,n}/(N_{i,p}+N_{i,n})$, and $\omega_{i,2}=N_{i,p}/(N_{i,p}+N_{i,n})$. $\omega_{i,1}$ and $\omega_{i,2}$ act as weights, encouraging base classifiers to identify more samples in the small-size group. As a result, this fitness function guides GP to evolve towards a balanced covering in respective two-class problems. For each Forest, the final fitness value is the average fitness value of all trees in it. So the individual with high fitness value is preferred. Due to the huge search space, only the individuals with top fitness values are kept, so as to accelerate the evaluation process. When two or more individuals obtain the same fitness score, the one with smaller size would be selected.

III. Experiment Results and Discussions

In our experiments, three ECOC methods in ECOC library[8] are employed for comparisons: DECOC, forest-ECOC, and ECOC-One. Hamming distance is applied for decoding. Default settings are applied for other parameters. To simplify discussion, only t-test is applied by keeping top 100 features. For the i-th class, measures are named as true positives (TP_i), true negatives (TN_i), false positives (FP_i) and false negatives (FN_i). Fscore is defined as shown in formula (2)-(4) with β=1. Accuracy is also used for comparisons, as shown in (5).

$$recall = avg\left(\sum_{i=1}^{n} \frac{TP_i}{P_i}\right) \quad (2)$$

$$precision = avg\left(\sum_{i=1}^{n} \frac{TP_i}{TP_i+FP_i}\right) \quad (3)$$

$$Fscore = avg\left(\sum_{i=1}^{n} \frac{(\beta^2+1)*precision_i*recall_i}{\beta^2*precision_i+recall_i}\right) \quad (4)$$

$$Acuracy = \frac{\sum_{i=1}^{n} TP_i+TN_i}{\sum_{i=1}^{n} TP_i+TN_i+FP_i+FN_i} \quad (5)$$

Table I shows the primary parameters used in our GP, and Table II shows the details about five microarray datasets used in experiments. The datasets and the preprocessing methods are the same as those in [4]. From Table III, it is found that different ECOC algorithms require different numbers of base learners because these ECOC algorithms are data dependent. Usually the larger size of coding matrix requires more base learners, leading to higher performance. However, in our experiments, it is found that even for DECOC that requires only N-1 base learners, can still achieve high performance in most cases, as shown in Table III. This observation proves the great discriminant ability of our algorithm.

TABLE I. THE SETTING OF PRIMARY GP PARAMETERS IN ALL EXPERIMENTS

Parameter	Setting
Terminal set (*T*)	All gene expression values and constants
Function set (*F*)	Boolean/mathematical operators: gt(>), le(<=), max, min, times(×), minus(-), plus(+).
Maximum Generation	100
Population Size	100
Crossover probability	0.7
Mutation probability	0.5
Termination criteria	Fitness score reaches 1 or gets 100 generations
Dynamic maximum tree depth limit	5
Strict depth limit	10

TABLE II. DETAILS ABOUT DATASETS IN EXPERIMENTS

Dataset	No. of classes	No. of genes	No. of training/test samples	References
Breast	5	9216	54/30	[10]
Cancers	11	12,533	100/74	[11]
DLBCL	6	4026	58/30	[12]
Leukemia	3	7129	38/34	[13]
Lung	3	7129	64/32	[14]

TABLE III. COLUMNS NUMBERS PRODUCED BY DIFFERENT ALGORITHMS

Dataset	ECOC-One	Forest-ECOC	DECOC
Breast	8	12	4
Cancers	12	30	10
DLBCL	8	15	5
Leukemia	4	6	2
Lung	12	18	2

The results listed in Table III prove that our GP based classifier guarantees the performance of different ECOC algorithms. From the results, it is obvious that in the case of hard datasets, ECOC algorithms can't always produce balanced results. Take Cancer data as an example, Forest-ECOC and DECOC get very low scores in cancer data because there are less than 10 samples in two classes. However, owing to the great generalization ability, ECOC-One can still achieve high Fscore scores with the application of GP. And in most cases, our algorithm can obtain achieve high performance in both measurements.

TABLE IV. RULES AND FITNESS VALUES BASED ON FOREST-ECOC

Learner	Rules in the classifier	Fitness
Ensemble 1	X136< 4.89e3	0.489
	times(min(X190,X188),X136)< 1.93e8	0.500
	min(X143,X131) > 3.13e4	0.846
Ensemble 2	X151 > 3.59e4	0.630
	X152 > 2.55e4	0.648
	min(X169,min(X145,X182)) > 1.650e4	0.853
Ensemble 3	X162 > 4.46e4	0.662
	X143 > 4.26e4	0.769
	min(X119,X131) > 3.54e4	0.807
Ensemble 4	min(X145,X182) > 1.73e4	0.828
	min(max(X137,min(X145,X182)),times(X116,X168)) > 2.42e4	0.829
	min(min(X145,X182),plus(X132,X136)) > 1.71e4	0.834
Ensemble 5	min(X65,plus(X152,X137))< 2.58e4	0.754
	min(X65,plus(X152,plus(X152,X137)))< 3.25e4	0.764
	min(max(X137,min(X169,plus(X137,X182))),times(X116,X168)) > 2.78e4	0.854
Ensemble 6	X138 > 2.41e4	0.845
	min(minus(X80,X170),min(X105,X20)) > -6.80e3	0.861
	min(max(X137,min(X169,plus(X137,X182))),times(X116,X182)) > 2.95e3	0.865

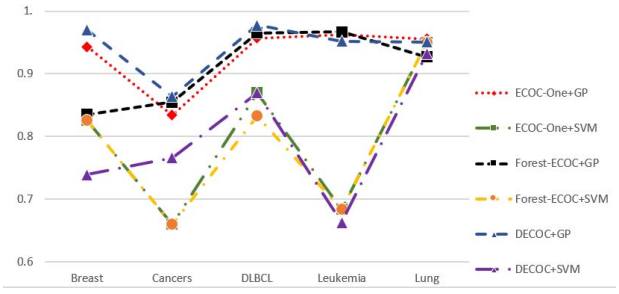

Figure 2. The comparisons of our algorithm with SVM

TABLE III. DETAILS ABOUT DATASETS IN EXPERIMENTS

Dataset	Measures	ECOC-One	Forest-ECOC	DECOC
Breast	Fscore	0.932	0.856	0.906
	Accuracy	0.943	0.835	0.970
Cancers	Fscore	0.976	0.487	0.534
	Accuracy	0.834	0.855	0.864
DLBCL	Fscore	0.987	0.814	0.711
	Accuracy	0.957	0.965	0.978
Leukemia	Fscore	0.981	0.901	0.882
	Accuracy	0.962	0.967	0.952
Lung	Fscore	0.985	0.568	0.769
	Accuracy	0.956	0.927	0.951

For further comparisons, we also illustrate the results of accuracies obtained by applying SVM as base learner for each ECOC algorithm (linear kernel with default settings[9]) in Fig. 2. It is found that our algorithm can always obtain much higher accuracies compared with those obtained by SVM. And the same conclusions can also be drawn in the Fscore values, confirming the robustness of our algorithm.

Table IV gives an example for Leukemia data based on Forest-ECOC. There are 18 rules for this classification task in all. As each rule represents the relationship among gene expression values and constant values, it can be found that gene 182, 152, 137, 136 and 145 are of great biological significance in discriminate some cancer subtypes. Although some rules can't produce high fitness values, as those included in Ensemble 1, this Forest can still reach a high accuracy score 0.881, revealing the power of ensemble and the error correction ability of ECOC algorithms.

IV. Conclusion

In this paper, we propose a Forest based individual structure to match different ECOC algorithm. Since each individual contains a small scale ensemble, this algorithm can help to enhance the generalization ability of ECOC algorithms. Experiment results prove that our algorithm can obtain balanced results in most cases despite of the imbalanced microarray data. And the comparisons with SVM further confirm the performance of our algorithm.

Acknowledgment

This work is supported by National Key Technology Research and Development Program of the Ministry of Science and Technology of China (No. 2015BAH55F05); Natural Science Foundation of Fujian Province (No. 2016J01320 and 2015J05129), and National Natural Science Foundation of China (No.61502402 and 61772023).

References

[1] G. Y. Zheng *et al.*, "The combination approach of SVM and ECOC for powerful identification and classification of transcription factor," *Bmc Bioinformatics,* vol. 9, 2008.
[2] X. L. Zhang, "Heuristic Ternary Error-Correcting Output Codes Via Weight Optimization and Layered Clustering-Based Approach," *IEEE Transactions on Cybernetics,* vol. 45, no. 2, pp. 289-301, Feb 2015.
[3] M. A. Bautista, S. Escalera, X. Baro, and O. Pujol, "On the design of an ECOC-Compliant Genetic Algorithm," *Pattern Recognition,* vol. 47, no. 2, pp. 865-884, Feb 2014.
[4] K. H. Liu, Z. H. Zeng, and V. T. Y. Ng, "A Hierarchical Ensemble of ECOC for Cancer Classification Based on Multi-Class Microarray Data," *Information Sciences,* vol. 349, pp. 102-118, 2016.
[5] P. G. Espejo, et al., "A Survey on the Application of Genetic Programming to Classification," *IEEE Transactions on Systems, Man, and Cybernetics, Part C,* vol. 40, no. 2, pp. 121-144, 2010.
[6] K. H. Liu, M. Tong, S. T. Xie, and V. T. Yee Ng, "Genetic programming based ensemble system for microarray data classification," *Computational & Mathematical Methods in Medicine,* vol. 2015, p. 193406, 2015.
[7] K. H. Liu and C. G. Xu, "A genetic programming-based approach to the classification of multiclass microarray datasets," *Bioinformatics,* vol. 25, no. 3, pp. 331-337, 2009.
[8] O.P.S. Escalera, P. Radeva, "Error-Correcting Output Codes Library," *Journal of Machine Learning Research,* vol. 11, pp. 661−664,, 2010.
[9] Pedregosa et al., "Scikit-learn: Machine Learning in Python," *JMLR,* vol. 12, pp. 2825-2830, 2011.
[10]C. M. Perou, T. Sørlie, M. B. Eisen, M. v. d. Rijn, S. S. Jeffrey, C. a. Rees, et al., "Molecular portraits of human breast tumours," Nature, vol. 406, pp. 747-752, 2000.
[11]A. I. Su, J. B. Welsh, L. M. Sapinoso, S. G. Kern, P. Dimitrov, H. Lapp, et al., "Molecular classification of human carcinomas by use of gene expression signatures," Cancer research, vol. 61, pp. 7388-7393, 2001.
[12]M. A. Shipp and e. al., "Diffuse Large B-cell Lymphoma Outcome Prediction by Gene-expression Profiling and Supervised Machine Learning," Nature Medicine, vol. 8, pp. 68-74, 2002.
[13]A. Ben-Dor, L. Bryhn, N. Friedman, I. nachman, M. Schum-mer, and Z. Yakhini, "Tissue classification with gene expression profiles," Journal of Computational Biology, vol. 4, pp. 290 - 2301, 2000.
[14]Z. Q. Hong and J. Y. Yang, "Optimal Discriminant Plane for a Small Number of Samples and Design Method of Classifier on the Plane," Pattern Recognition, vol. 24, pp. 317-324, 1991.