Hybrid Flexible Neural Tree Approach for Leukemia Cancer Classification

Lei Zhang
Shandong Provincial Key Laboratory of Network Based Intelligent Computing
School of Information science and Engineering, University of Jinan
Jinan, China
e-mail: zhanglei@ujn.edu.cn

Yuehui Chen
Shandong Provincial Key Laboratory of Network Based Intelligent Computing
School of Information science and Engineering, University of Jinan
Jinan, China
e-mail: ychen@ujn.edu.cn

Ajith Abraham
Machine Intelligent Research Labs (MIR Labs),
Scientific Network for Innovation and Research Excellence
USA
e-mail: ajith.abraham@ieee.org

Zhenxiang Chen
Shandong Provincial Key Laboratory of Network Based Intelligent Computing
School of Information science and Engineering, University of Jinan
Jinan, China
e-mail: czx@ujn.edu.cn

***Abstract*—In this paper, a novel method for improving flexible neural tree is proposed to classify the leukemia cancer data. The hybrid flexible neural tree with pre-defined instruction sets can be created and evolved. The structure and parameter of hybrid flexible neural tree are optimized using probabilistic incremental program evolution (PIPE) and particle swarm optimization (PSO) algorithm. The experimental results indicate that the proposed method illustrates feasible and efficient for the classifications of microarray data.**

Keywords- hybrid evolutionary method; flexible neural tree; particle swarm optimization; leukemia cancer data

I. INTRODUCTION

The accurate cancer diseases diagnosis is very important to patients for successful treatment. Microarray data which measure the mRNA expression level of thousands of genes in a cell are used to analyze gene expression data for cancer classification. It is more desirable to applied intelligent methods in this area for analyzing the microarray data which have high dimensionality of genes and small sample size [1]. Recently, some traditional intelligent methods have been applied for microarray data, such as support vector machines (SVM), artificial neural network (ANN), k-nearest neighbor (KNN), decision tree, etc. It is difficult to classify in such a high dimension space using traditional classification methods directly. Classification method should not only provide appropriate measures of variable gene selection, but also present features that is well suited for classification problem.

Chen has proposed the flexible neural tree (FNT) [2-5]. The flexible neural tree is a special multi-layer feed-forward neural network and allows over-layer connections, input variables selection and different activation functions for different nodes. In order to reduce the size of the searching space and improve searching efficiency, we improve the flexible tree model adding additive instructor.

In this paper, a hybrid evolutionary method is proposed to fulfill obtaining the useful features. The tree structure model based PIPE and particle swarm optimization (PSO) are employed to evolve the architecture and the parameters to analyze leukemia cancer via microarray data. Starting with random structures and corresponding parameters, it tries to improve the structure and then an improved structure is found, its parameters are tuned accordingly. The paper is organized as follows. Section 2 is shown the representation of hybrid flexible neuron instructor and hybrid FNT model. In Section 3, we describe the details of the proposed method. The experiment result is showed in Section 4 and the conclusion is in Section 5.

II. REPRESENTATION OF HYBRID FLEXIBLE NEURON INSTRUCTOR AND HYBRID FNT MODEL

Hybrid neural tree model is a tree-based encoding neural network encoding with a specific instruction set which is selected for representing a model. There are two types of sets in the model: the hybrid flexible neuron instructor set and the terminal set. The hybrid flexible neuron instructor is used to join the subtree of non-leaf's node, the terminal set is used to input leaf node instructor. The function set I_1, I_2 and terminal set T for generating a hybrid neural tree are described as follows:

$$S=I_1 \cup I_2 \cup T=\{+_2,+_3,...,+_{N1}\}\cup\{*_2,*_3,...,*_{N2}\}\cup\{x_1,x_2,...,x_n\} \quad (1)$$

Where $+_i$, $*_i$ denote the non-leaf's node and have i arguments. $x_1, x_2, \ldots, x_n$ are leaf nodes' instructions and have no arguments, in fact are input variables.

The output of a non-leaf node is calculated as hybrid neuron model showed by figure 1 and figure 2.

In the constructing process of hybrid neural tree, if a non-leaf node instruction, i.e. +i, or *i (i = 2, 3, … , N) is selected, i real values are randomly generated and acted as representing the connection strength between the sub-nodes. In addition, two adjustable parameters a and b are randomly created as flexible activation function parameters. Table1 show some examples of flexible activation functions.

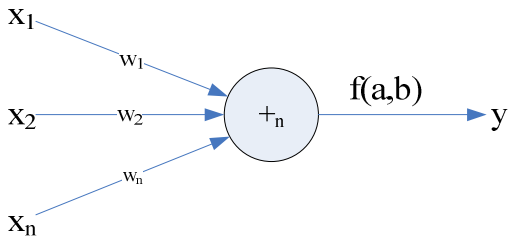

Figure 1. A flexible neuron operator with $+_n$ instructor

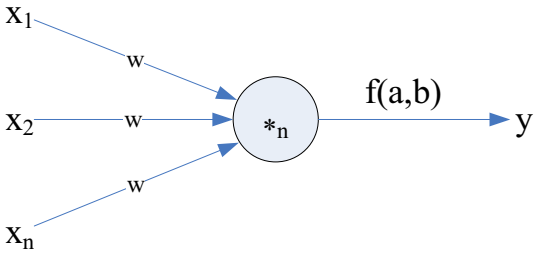

Figure 2. A flexible neuron operator with $*_n$ instructor

Table 1 The activation functions

Gaussian Function	$f(x,a,b) = e^{-\frac{(x-a)^2}{b^2}}$
Flexible unipolar sigmoid function	$f(x,a) = \frac{2\mid a \mid}{1+e^{-2\mid a \mid x}}\ f(x,a)$
Flexible bipolar sigmoid function	$f(x,a) = \frac{1-e^{-2xa}}{a(1+e^{-2xa})}$

In this paper research, we use the following flexible activation function.

$$f(x,a,b) = e^{-\frac{(x-a)^2}{b^2}} \tag{2}$$

The output of flexible neuron $+_n$ is calculated as the following.

$$net_n = \sum_{i=1}^{n} w_i * x_i \tag{3}$$

The output of a flexible neuron $*_n$ can be calculated as follows. The total excitation of $*_n$ is

$$net_n = \prod_{i=1}^{n} w * x_i \tag{4}$$

Where $x_i (i = 1, 2, \ldots, n)$ is the input variable of flexible neuron, the total activation of the node is calculated as the follow.

$$out_n = f(net_n, a_n, b_n) = e^{-(\frac{net_n - a_n}{b_n})^2} \tag{5}$$

Figure3 is an example hybrid flexible neural tree model, the total output of neural tree is computed from left to right by depth-first method.

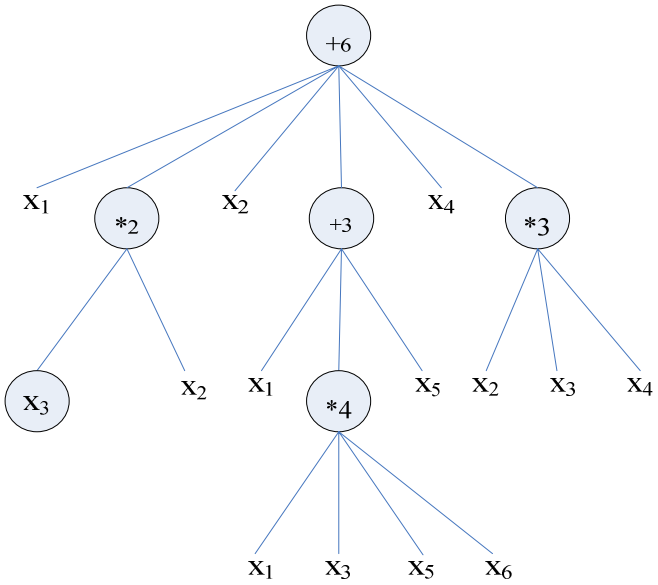

Figure 3. An example of hybrid FNT model

III. THE PROPOSED HYBRID EVOLUTIONARY ALGORITHM

A. *Neural Tree Model Optimization*

The optimization of neural tree model includes the structure and the parameter optimization. The process of searching the optimal or near optimal neural tree is accomplished by using probabilistic incremental program evolution algorithm (PIPE) and the parameters optimized using PSO algorithm. We use the additive operators as follows:

(1) Mutation

In the process of generating the offspring from the parents of hybrid flexible neural tree model, we choose four mutation operators:

1) Change one leaf node: Randomly select one leaf mode in the neural tree and replace it with another leaf node which is generated randomly.

2) Change all leaf nodes.

3) Grow: Randomly select a leaf node in the hidden layer of the neural tree and replace it with a newly generated subtree.

4) Prone: Randomly select a non-leaf node and replace it with a leaf node.

(2) Crossover

Two parents are randomly selected from neural trees according to the predefined crossover probability and one non-leaf node in the hidden layer is randomly selected, then swap the selected subtree.

(3) Selection

The tournament selection in evolutionary programming is applied to select the parents for next generation. Pairwise comparison is conducted for the union of μ parents and μ offsprings. For each individual, q opponents are randomly chosen from all the parents and offspring. For each comparison, if the individual's fitness is bigger than the opponent's, the individual is selected for the next generation. Some individuals selected from the parents and offspring which has been selected form the new generation. This process is repeated for each generation until the best tree structure is found.

B. Parameter optimization using PSO

The Particle Swarm Optimization (PSO) which is one of the evolutionary optimization techniques, was introduced in 1995 by Kennedy and Eberhart[6]. PSO algorithm is adaptive methods that can be used to solve optimization problem. Conducting searches uses a population of particle which corresponds to individuals in evolutionary algorithm. A flock or swarm of particles is randomly generated initially, each particle's position representing a possible solution point in the problem space. Each particle has an updating position vector x_i and updating velocity vector v_i by moving through the problem space. For each iteration, a fitness function f_i evaluating each particle's quality is calculated by position vector x_i. The vector p_i represents the best ever position of each particle and p_g represents the best position obtained so far in the population.

For each iteration t, following the p_i and $p_g(t)$, the velocity vector of particle i is updated by the follow formula:

$$v_i(t+1)=v_i(t)+c_1\phi_1(p_i(t)-x_i(t))+c_2\phi_2(p_g(t)-x_i(t)) \quad (5)$$

where c_1 and c_2 are positive constant and ϕ_1 and ϕ_2 are uniformly distributed random number in *[0,1]*. The velocity vector v_i is range of $[-V_{max}, V_{max}]$. Updating the velocity vector of particle by this method enables the particle to search around its individual best position and global best position. Based on the updated velocity, each particle changes its position according to the following formula:

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (6)$$

Based on this method, the population of particles tends to cluster together. To avoid premature convergence, the following formula can improve the velocity vector of each particle.

$$v_i(t+1)=\chi(\omega v_i(t)+c_1\phi_1(p_i(t)-x_i(t))+c_2\phi_2(p_g(t)-x_i(t))) \quad (7)$$

Where χ and ω are real numbers. The parameter χ controls the magnitude of v_i , the inertia weight ω controls the magnitude of the velocity vector v_i (t).

C. Fitness function

The fitness function is used to compute the hybrid flexible neural tree model, real value reflect the hybrid flexible neural tree model's performances. The fitness functions are able to clearly reflect the accuracy rate of classification and control to the best scale of neural tree in the non-user-defined objective. Among some of neural tree of having equal fitness values, a smaller scale neural tree is selected. A fitness function in this paper formulating the sum positive and negative classification error is used to design the classification model.

D. The algorithm of hybrid FNT

The learning algorithm procedure for constructing the hybrid neural tree model can be described as follows.

(1) Create an initial population of neural trees including of tree structures and its corresponding parameters;
(2) Structure optimization is accomplished by using the neural tree variation operators as described in subsection A;
(3) If a better structure is found, then go to step (4), otherwise go to step (2);
(4) Parameter optimization of neural tree is accomplished by PSO algorithm which described in subsection B. In this step, the structure of neural tree is fixed and it is the best tree evolved during the end of running of the structure search. The parameters include of weights and flexible activation function parameters encoded in the best tree is defined as a particle.
(5) If the maximum number of local search or the pre-defined time is reached but the better parameter vector is not found, go to step (6); otherwise go to step (4);
(6) If the satisfactory solution is found, the algorithm is stopped; otherwise go to step (2).

IV. EXPERIMENTAL RESULTS AND ANALYSIS

To test the effectiveness of the proposed method, we performed extensive experiments on the Leukemia cancer dataset which consists of 72 samples taken from leukemia patients. This dataset is often used for benchmark for microarray analysis methods. These samples can be divided into two subtypes: 25 samples of acute myeloid leukemia (AML) and 47 samples of acute lymphoblastic leukemia (ALL)[7]. We used a total of 38 out of 72 samples as training data and the remaining samples as test data. Each sample contains 7129 gene expression levels.

The leukemia dataset is divided into training and testing set. The parameter settings in our experiments are shown in Table 2. The number of initial population named by population size is 50. The maximum of generations is 200. The crossover rate and mutation rate showed at table 2 are predefined. C1 and c2 are positive constants used in PSO algorithm. To setup the 50 independent trials, a hybrid neural tree model was constructed using the training data set and used on the test data set. For example, The instruction sets used to construct a hybrid neural tree is $S = I \cup T$ ={+6,x6838,x4140,*3,x6508,x2513,x3237,+6,x2871,x7129,x3281,x6610,x1438,x6569,*4,x1639,x7129,x957,x1788,+3,x2658,x6451,x6204,},which denotes the features

selected in 7129 input variables of the classification model. The hybrid FNT model greatly differs from traditional flexible neural tree in the population initialization and parameter optimization. This proposed model can significantly reduce the search space and improve the efficiency.

Table 2 Parameters for experiments

Population size	50
Generation	200
Crossover rate	0.8
Mutation rate	0.2
c_1	2.0
c_2	2.0

Table 3 Comparison of the classification accuracy using different methods for Leukemia cancer dataset

Author	Accuracy
Our method	97.6%~99.8%
Nguyen [8]	94.2%~96.4%
Ben-Dor [9]	91.6%~95.8%
Li[10]	84.6%
Zhao [11]	95.8%~97.2%
Furey [12]	94.1%
Zhenyu Wang [13]	91.50%
Sung-Huai Hsieh[14]	98.10%

A comparison of different features selection methods and different classification methods for leukemia datasets are shown in Table 3. Our method achieve the classification accuracy 97.6%~99.8%, while other methods produce the classification accuracy 84.6%~98.1%. It is show that the hybrid flexible neural tree model gets higher classification accuracy than other methods for leukemia cancer data.

V. CONCLUSIONS

In this paper, we apply the hybrid flexible neural tree model to fulfill the feature selection and classify the leukemia cancer data. The probabilistic incremental program evolution algorithm and particle swarm optimization algorithm are selected to search for the hybrid flexible neural tree structure and its corresponding optimal parameters, respectively. The experiments result of our method demonstrate that the hybrid flexible neural tree model may provide better classification result and better efficiency than other classification models. Compare the results with some other classification methods, the proposed method can improve the classification accuracy for the leukemia cancer data. This illustrates that the proposed hybrid flexible neural tree model can be used as an effective method for classification of microarray data.

ACKNOWLEDGMENT

This work was supported by the Science and Technology Program of Shandong Provincial Education Department (J08LJ18) and the Natural Science Foundation of Shandong Province (ZR2011FL021).

REFERENCES

[1] Golub T R, Slonim D K, Tamayo P, et al. Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring. Science. 1999, 286(5439): 531-537.

[2] Chen, Y., Peng, L., Abraham, A.. Gene expression profiling using flexible neural trees. 2006. LNCS, 2006, 4224, 1121–1128.

[3] Y. Chen, B. Yang, J. Dong, A. Abraham, Time-series forecasting using flexible neural tree model. Information Science.2005, 174, 219–235.

[4] Y. Chen, B. Yang, A. Abraham. Flexible neural trees ensemble for stock index modeling. Neurocomputing.2007, 70, 697–703.

[5] Y. Chen, et al., Small-time scale network traffic prediction based on flexible neural tree. Applied Soft Computing. 2011, doi:10.1016/j.asoc. 2011.08.045.

[6] J. Kennedy and R. C. Eberhard. Particle swarm optimization. Proc. Of IEEE Int'l Conf. 1995, 1942-1948.

[7] Ben-Dor, A., Bruhn, L., Friedman, N., Nachman, I., Schummer, M., Yakhini, N. Tissue classification with gene expression profiles. Computational Biology. 2000, 7, 559–584.

[8] C.A. Harrington, C. Rosenow and J. Retief. Monitoring Gene Expression Using DNA Microarrays. Curr. Opin. Microbiol. 2000, 3, 285-291.

[9] S.B. Cho, Exploring Features and Classifiers to Classify Gene Expression Profiles of acute Lekemia. Artifical Intellegence. 2002, 16(7), 1-13.

[10] M.B. Eisen and B.O. Brown, DNA Arrays for Analysis of Gene Expression. Methods in Enzymology. 1999, 303, 179-205.

[11] Y. Zhao, Y. Chen and X. Zhang. A novel ensemble approach for cancer data classification. LNCS, 2007, 4492, 1211-1220.

[12] T.R. Golub, D.K. Slonim, P. Tamayo, C. Huard, M. GaasenBeek, J.P. Mesirov, H. Coller, M.L. Loh, J.R. Downing, M.A. Caligiuri, C.D. Blomfield and E.S. Lander, Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring. Science 1999, 286(12), 531-537.

[13] Z.Y. Wang, V. Palade, Y. Xu, Nero-fuzzy Ensemble approach for microarray cancer gene expression data analysis. Proceedings of the 2006 International Symposium on Evolving Fuzzy Systems (IEEE), 7–9. 2006, 241–246.

[14] Sung-Huai Hsieh, Zhenyu Wang, Po-Hsun Cheng, I-Shun Lee, Sheau-Ling Hsieh, Feipei Lai, Leukemia Cancer Classification based on Support Vector Machine, Industrial Informatics (INDIN), 2010 8th IEEE International Conference. 2010,819-824.