An Integrated Pattern Recognition System and Its Application

WANG, Lixin
School of Mathematics
University of Minnesota, Twin Cities Campus
MN 55455, USA

HUA, Jing DAI, Ruwei
AI Lab, Institute of Automation
Chinese Academy of Sciences
Beijing, 100080, CHINA

Abstract

In the paper we design an integrated pattern recognition system. Instead of design a classifier for pattern recognition, a finite number of classifiers are simultaneously applied, and a multi-layer artificial neural network with feedback is employed to process all outputs of the individuals to get more accurate classification rate. Because of the introduction of the feedback loop, the pattern recognition system becomes a nonlinear dynamic system, not a nonlinear mapping any more. We obtain the sufficient condition on the absolute stability for the integrated network and derive a corresponding learning algorithm to ensure its stability. The system has been applied in totally unconstrained handwritten numeral recognition and its performance is excellent.

1. Introduction

Recently, many researchers are interested in combining classifiers because they note that different classifier designs potentially offered complementary information about the patterns to be classified. In order to combine multiple classifiers to obtain more accurate classification rate, we design an **Integrated Pattern Recognition System (IPRS)** (Fig 1). The integrated classifier can be fulfilled by different approaches, such as the voting method [1] and MLP [2]. If we consider the integrated layer a nonlinear mapping, perhaps the artificial neural network is the effective tool [2] for its nonlinear classification ability is superiority. From the view of cybernetics [3], the performance of the close loop structure is often superior to that of the open loop structure. Then why not try introducing the close loop structure to pattern recognition system?

In this paper, enlightened by the idea of metasynthesis [2][4], we employ artificial neural network to design several classifiers and by means of an artificial neural network with feedback several classifiers are integrated to form a new classifier in order to get high performance. The supervised learning was used in the training of single and integrated classifiers. One can find that the pattern recognition system is not only a nonlinear mapping, it is a nonlinear dynamic system so far as the feedback mechanism is introduced to neural network for integration. So the integrated pattern recognition system is also a nonlinear dynamic system.

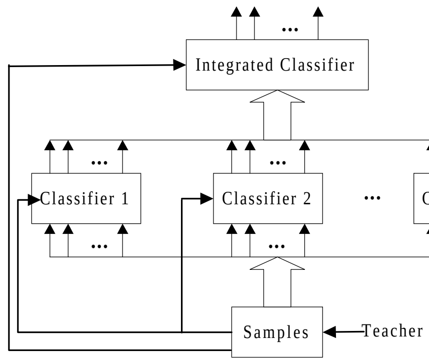

Fig. 1 Architecture of IPRS

2. An integrated neural network with feedback (INNF)

As we have described in the Introduction, from the view of cybernetics, the addition of the feedback loop can improve the performance of the system, but the model must be reasonable. We propose a Feedback Model based on the combination of multiple classifiers and use **an integrated neural network with feedback (INNF)** in the integrated classifier (Fig 2). It connects same outputs of the output layer to the hidden unit layer as a feedback loop. And the dimension of output and both the dimension of the input layer and that of the hidden unit layer are all “Category”. So such feedback is meaningful. We resolve the dimension conflict between the feedback information and input feature in the pattern recognition system.

In Fig. 2, $\boldsymbol{W}_{hi}$, $\boldsymbol{W}_{oh}$ and $\boldsymbol{W}_{io}$ are the connection matrixes from the input layer to the hidden layer, from the hidden layer to the output layer and from the output layer back to the input layer, respectively. We add some virtual nodes in the input layer, which are corresponding to those in the output layer. The connection matrix among the virtual ones and those in the hidden layer is $\boldsymbol{W}_{ho}$. For n-th iteration, the values of the virtual nodes are those of the corresponding output nodes at (n-1)th iteration. As the above description, because the input and output in the INNF both stand for the class attribute of a sample, and they are identical in dimension, we say that the feedback is appropriate and reasonable.

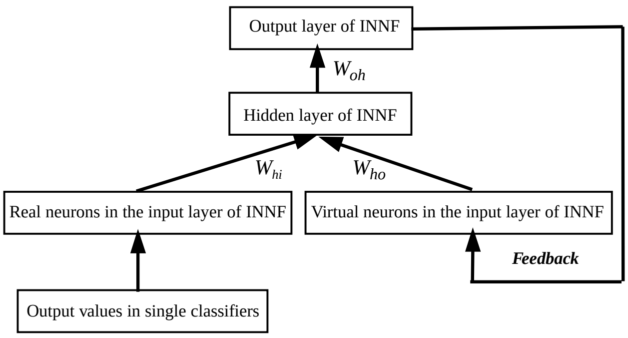

Fig. 2 Integrated neural network with feedback

The parameters of the INNF are defined as follows:

$\boldsymbol{W}_{hi}$, $\boldsymbol{W}_{oh}$ and $\boldsymbol{W}_{ho}$ are $H \times I$, $O \times H$ and $H \times O$, respectively.

I : number of the nodes in the input layer;

O: number of the nodes in the output layer;

H : number of the nodes in the hidden layer;

k : recurrent time of the neural network;

Ω: number of the training samples;

$\boldsymbol{i}_{I\times 1}$: the input vector in the input layer;

$\mathbf{y}_{O\times 1}$: the output vector in the output layer;

$\boldsymbol{g} = (g_1, \cdots, g_H)^t$: the activation vector function in the hidden layer, satisfying

$$\boldsymbol{g}((x_1, \cdots, x_H)^t) = (g_1(x_1), \cdots, g_H(x_H))^t .$$

$\boldsymbol{f} = (f_1, \cdots, f_O)^t$: the activation vector function in the output layer, satisfying

$$\boldsymbol{f}((x_1, \cdots, x_O)^t) = (f_1(x_1), \cdots, f_O(x_O))^t .$$

The coordinate functions of activation functions are sigmoid ones, such as $\dfrac{1}{1+e^{-x}}$, which have the following properties: 1) monotonously increasing; 2) the function and its derivative are continuous and bounded.

$\{\boldsymbol{i}(a), \quad \boldsymbol{y}^e(a)\}$ $(a = 1, \cdots, \Omega)$ is the training set. $\boldsymbol{i}(a)$ is the training sample; $\boldsymbol{y}^e(a)$ is the corresponding expectation output.

$$E(a) = \sum_{i=1}^{O} (y_i^e(a) - y_i(a))^2 \Big/ 2 ,$$

$E = \sum_{a=1}^{\Omega} E(a) \Big/ \Omega$ is the objective function.

ε is the accuracy of the objective function after learning. The variance between real and the corresponding expectation outputs should be less than ε when the samples are all trained, otherwise the training process continues.

The recurrent equation of the INNF is given as following:

$$\begin{cases} \boldsymbol{y}(k+1) = \boldsymbol{f}(\boldsymbol{W}_{oh} \cdot \boldsymbol{g}(\boldsymbol{W}_{ho} \cdot \boldsymbol{y}(k) + \boldsymbol{W}_{hi} \cdot \boldsymbol{i}))(k \ge 0) \\ \boldsymbol{y}(0) = \boldsymbol{\alpha} \end{cases} \quad (2.1)$$

where $\boldsymbol{\alpha}$ is an arbitrary initial value of the output layer. Generally, it is set to be a zero vector.

A sufficient condition of absolute stability for the integrated network described by recurrent equation (2.1) has been proved by us. Due to the limited length of this paper, we directly give it as following:

$$d_1 d_2 R\max(\boldsymbol{W}_{ho}) R\max(\boldsymbol{W}_{oh}) < 1, \quad (2.2)$$

where $d_1 = \max\limits_{i=1,\cdots,O}\{ f_i^{'} \} < \infty$, $d_2 = \max\limits_{j=1,\cdots,H}\{ g_j^{'} \} < \infty$.

$\underset{\boldsymbol{x}\in\Re^s}{R\max}$ is an operator which is defined on $n \times m$ function matrix $\boldsymbol{C} = (c_{ij}(\boldsymbol{x}))_{ij}$,

$$\underset{\boldsymbol{x}\in\Re^s}{R\max}(\boldsymbol{C}) = \max_{i,\boldsymbol{x}\in\Re^s}(\sum_{l=1}^{m} |c_{il}(\boldsymbol{x})|) .$$

3. The supervised learning algorithm for the INNF

In last section, we show the sufficient condition on the absolute stability of INNF. It is natural to wonder how to ensure it satisfying the mentioned condition. That is, how can we implement it into the learning algorithm? Although there are methods in constrained optimization theory, such as the penalty function method, we found it was not always efficient. For this reason, we directly add the "regularization" in the algorithm so that the INNF readily satisfies one (2.2).

For simplicity, denote w_{in}^{h} as the weight from the n-th neuron in the hidden layer to the i-th one in the output layer, w_{nm}^{o} as the weight from the m-th one in the output back to the n-th one in the hidden layer. The weight w_{ps} connects the output of neuron s to the input of neuron of p. The sum of neuron p's inputs is net_p and the output is o_p.

$$net_p = \sum_s w_{ps} o_s \text{ and } o_p = f(net_p),$$

where f is the activation function.

The supervised learning algorithm is as follows.

Step 1. Initialization of $\boldsymbol{W}_{hi}$, $\boldsymbol{W}_{oh}$ and $\boldsymbol{W}_{io}$, let

$$\Delta w_{ps}(0) = 0.$$

Step 2. Let the training time $u = 1$.

Step 3. Let the sample number $a = 1$, error $E = 0$.

Step 4. Let the recurrent time $k = 0$, and $\dfrac{\partial y_i(0)}{\partial w_{ps}} = 0$.

Step 5. Compute $\boldsymbol{y}(k+1)$ and

$$EStable = \sum_{i=1}^{O} \left| y_i(k+1) - y_i(k) \right| \text{ according to}$$

$$\begin{cases} \boldsymbol{y}(k+1) = \boldsymbol{f}(\boldsymbol{W}_{oh} \cdot \boldsymbol{g}(\boldsymbol{W}_{ho} \cdot \boldsymbol{y}(k) + \boldsymbol{W}_{hi} \cdot \boldsymbol{i}))(k \geq 0) \\ \boldsymbol{y}(0) = \alpha \end{cases}$$

Step 6. Compute $\dfrac{\partial y_i(k+1)}{\partial w_{ps}}$ $(i = 1, \cdots, O)$.

$$= \left. \frac{\partial y_i(k+1)}{\partial w_{ps}} \right|_{y(k)} + \sum_{m=1}^{O} \frac{\partial y_i(k+1)}{\partial y_m(k)} \cdot \frac{\partial y_m(k)}{\partial w_{ps}}$$

$\left. \dfrac{\partial y_i(k+1)}{\partial w_{ps}} \right|_{y(k)}$ is the partial differentiation on the assumption that $y(k)$ is constant, which can be calculated by a process similar to back propagation.

$$\frac{\partial y_i(k+1)}{\partial y_m(k)} = f_i^{'}(net_i) \sum_n w_{in}^{h} g_n^{'}(net_n) w_{nm}^{o}$$

Step 7. Let $k = k+1$. If $EStable$ is smaller than a experiential constant or k is larger than the specified recurrent time, continue; Otherwise return to step 5.

Step 8. Compute

$$\frac{\partial E(a)}{\partial w_{ps}} = \sum_{i=1}^{O} (y_i(k) - y_i^{e}(a)) \frac{\partial y_i(k)}{\partial w_{ps}}.$$

Step 9. Change the weight according to

$$\Delta w_{ps}(u) = -\alpha \frac{\partial E(a)}{\partial w_{ps}} + \beta \Delta w_{ps}(u-1)$$

where α and β are positive constants.

Step 10. Let $Max1 = Rmax(\boldsymbol{W}_{oh})$,

$Max2 = Rmax(\boldsymbol{W}_{ho})$.

Step 11. Compute $Abs1(i) = \sum_{n=1}^{H} \left| w_{in}^{h} \right|$ and

$$Abs2(n) = \sum_{m=1}^{O} \left| w_{nm}^{o} \right|.$$

Step 12. Regularize weights w_{in}^{h} and w_{nm}^{o},

$$w_{in}^{h} = \frac{\lambda_1 w_{in}^{h}}{\sqrt{Abs1(i) \cdot Max2}},$$

$$w_{nm}^{o} = \frac{\lambda_2 w_{nm}^{o}}{\sqrt{Abs2(n) \cdot Max1}},$$

λ_1 and λ_2 is positive and satisfy

$$\lambda_1 \lambda_2 < 1/ d_1 d_2.$$

Step 13. Compute the error $E(a)$. Let

$E = E + E(a)$, and $u = u + 1$.

Step 14. When a is less than the sample amount Ω, let $a = a + 1$ and return to step 4. Otherwise continue.

Step 15. Let $E = E/\Omega$. If E is less than ε, or u is greater than the specified training time, finish the training. Otherwise return to step 3 and begin a new training.

Step 12 is the most important, which ensures the constrained weights satisfy inequality (2.2). When the training finishes, the INNF is absolutely stability. It should be noticed that the algorithm is for the INNF with one hidden layer. If it has more hidden layers, the corresponding algorithm can be obtained through simple modification.

4. Experiments and results

To demonstrate the superiority of the proposed INNF, experiments were performed with the totally unconstrained handwritten numeral database. Fig.3 and Fig. 4 shows some representative samples in training and testing sets.

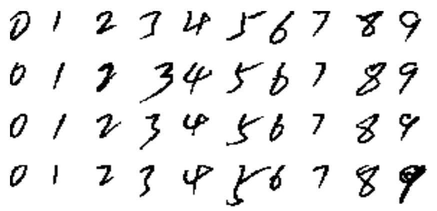

Fig. 3 Representative samples in training set

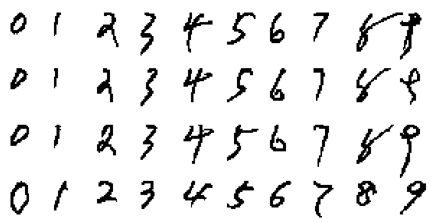

Fig. 4 Representative samples in testing set

Rejection was according to

$$max < \frac{1}{4}(Sum + r)$$

Reliability is due to

$$Reliability = \frac{Recognition \quad with \quad rejection}{1 - Rejection}$$

The integrated pattern recognition system has three single classifiers. C1 uses contour and point features, trained by 8000 samples. C2 uses lattice feature, trained by 10000 samples. C3 uses direction feature, trained by 6000 samples. They are all tested by other 10000 samples. The training algorithm is the competitive supervised learning [5]. The results are shown in table 1.

Table 1. Experimental results of C1, C2 and C3

Classifie	Error	Rec. with Rej.	Rej.	Rel.
C1	<0.08	86.71%	9.60%	95.9%
C2	<0.09	87.72%	9.95%	97.4%
C3	<0.05	91.10%	7.10%	98.1%

The integration is based on C1, C2 and C3. To verify the INNF's effectiveness, it is compared with the MLP. Their input units, hidden units and output units are 30, 10 and 10, respectively. Two experiments were carried out. One was 2000 training samples and 8000 testing samples. The other was 3000 training samples and 7000 testing samples. The results were shown in table 2 and table 3, respectively.

Table 2. Results of the experiment 1

Model of Neural Network	MLP	INNF
Error	<0.01	<0.021
Recognition with rejection	94.29%	96.6%
Rejection	5.03%	2.6%
Reliability	99.3%	99.3%

Table 3. Results of the experiment 2

Model of Neural Network	MLP	INNF
Error	<0.01	<0.012
Recognition with rejection	97.17%	97.91%
Rejection	2.27%	1.5%
Reliability	99.4%	99.4%

Table 2 and 3 showed that the integration of multiple classifiers improved the system's generalization capability greatly. On the other hand, the recognition with rejection of the INNF increased 2.31% and 0.74%. This proved the superiority of the INNF.

5. Conclusion

We can see that the integrated neural network with feedback (INNF) has better performance than the open loop structure. Because of the existing of the feedback, the integrated network is not a simple static nonlinear mapping. The whole pattern recognition system becomes a nonlinear dynamic system and its generalization is improved greatly. The proposed model in this paper denotes a connection between pattern recognition field and control system field.

Reference

[1] C. Y. Suen, C. Nadal, R. Legault, T. A. Mai, L. Lam, Computer recognition of unconstrained handwritten numerals. Proc. of the IEEE, vol. 80, no. 7, 1992, 1162-1180.

[2] Hao Hongwei, Xiao Xuhong and Dai Ruwei, Handwritten Chinese character recognition by metasynthesis approach, Pattern Recognition, 30(8), 1997,1321-1328.

[3] H. S. Tsien, Engineering cybernetics. McGraw-Hill, 1954.

[4] Dai Ruwei and Hao Hongwei, Metasynthesis and its application to pattern recognition, Acta Automation Sinica, 23(3), 1997, 302-307.

[5] Wang Lixin, Fei Yue and Dai Ruwei, Competitive supervised learning based on human-machine combined integration, Chinese J. Pattern Recognition and Artificial Intelligence, 10(3), 1997, 186-195.