Active Diverse Learning Neural Network Ensemble Approach for Power Transformer Fault Diagnosis

Yu Xu
College of Information and Engineering, Xiangtan University, Xiangtan, China
Email: rule19852003@yahoo.com.cn

Dongbo Zhang and Yaonan Wang
a College of Information and Engineering, Xiangtan University, Xiangtan, China
b College of electrical and information engineering, Hunnan University, Changsha, China
Email: zhadonbo@yahoo.com.cn, yaonan@hnu.cn

Abstract—An ensemble learning algorithm was proposed in this paper by analyzing the error function of neural network ensembles, by which, individual neural networks were actively guided to learn diversity. By decomposing the ensemble error function, error correlation terms were included in the learning criterion function of individual networks. And all the individual networks in the ensemble were leaded to learn diversity through cooperative training. The method was applied in Dissolved Gas Analysis based fault diagnosis of power transformer. Experiment results show that, the algorithm has higher accuracy than IEC method and BP network. In addition, the performance is more stable than conventional ensemble method, i.e., Bagging and Boosting.

Index Terms—power transformer; fault diagnosis; diversity; neural network ensemble

I. INTRODUCTION

Transformer is very important in power system, and its healthy operation is the basis of the power supply and normal social life. Detecting incipient fault and estimating the fault’s level could prevent further accidents. Therefore, fault diagnosis research has the vital practical significance for improving the transformer’s operation. Dissolved Gas Analysis (DGA) technique is the most effective practice in transformer fault diagnosis, especially to oil-immersed transformer. Under the normal condition, the insulating oil and organic insulating material in oil-filled equipment generate a small amount of gases, such as low-molecular-hydrocarbons, carbon monoxide (CO) and carbon dioxide (CO2), causing by the gradual degradation and decomposition. Most of them can dissolve in the oil. While exist latent thermal and discharging fault, the gas generation rates will be immediately accelerated, as well as methane and hydrogen. Consequently, gas-in-oil concentrations and gas generation rates are usually used to diagnose the type and degree of transformer fault. The three ratio method has been the most typical methods of diagnosis, which is simple, practical and has high accuracy. However, the code table involved in fault symptoms and fault types clearly depend on expert experience [1, 2] in this method. And in engineering application, it has difficult problems, such as incomplete code, too absolute code border and unavailable to multi-fault diagnosis. Even for the same experiment data, different fault diagnosis results may be achieved when different diagnosis methods are adopted. With the development of computer science, artificial intelligent methods are extensively applied to DGA based fault diagnosis of power transformer. Available techniques include expert system [3], artificial neural network [4, 5, 6], fuzzy logic [7], rough set [8, 9, 10] and support vector machine [11]. Although these methods have achieved promising results, however, due to the complexity of problem, corresponding methods are still needed to be improved to achieve higher adaptability and stability.

Actually, there exist complex and uncertain mapping relationship between gas content and fault category, so it is a very difficult learning problem to build accurate mathematical model of power transformer fault diagnosis. With the powerful learning ability, excellent generalization capability and infinite non-linear approximation characteristic, Neural Network (NN) has become a valid method for fault diagnosis. The earlier literatures introducing the idea of NN to fault diagnosis of power transformer were O. Vanegas [12] and Y. Zhang [13]. The experimental results show that the performance of NN method is better than conventional three ratio method. Because NN usually has slow convergence rate and easily fall into local minimum point, many researchers introduce more complex network structures and learning algorithm to improve NN’s performance. An efficient learning algorithm for self-adaptive BP was presented by S. Yan-jing [14], which has faster convergence, and via adjustable training rate and self-
adaptive momentum factor, the global optimum may be achieved. P. Chong [15] presented a fault diagnostic method based on genetic algorithm, which is realized by evolving wavelet neural network. The confirmation of network structure and parameters are decided by real value encoding genetic algorithm instead of artificial specified. And a good balance is satisfactorily reached about the network complexity, convergence rate and generalization ability. A novel fuzzy neural classifier and learning algorithm were proposed based on EM algorithm by S. Hong-sheng[16]. Where, rough set is used to measure the importance of the features. And the parameters of fuzzy membership function are optimized with EM algorithm. Although these methods mentioned above improve NN's diagnostic performance in a certain extent, it also results in inevitable high computational cost, while the performance’s improvement is finite and unstable.

In machine learning field, ensemble learning has become a novel approach to improve the generalization ability of difficult learning problems. In multiple classifiers ensemble, the individual classifiers might not require high accuracy and each component only learn part of the system. And ensemble generalization performance can be greatly improved while each classifier is constructed with simple structure and training strategy. Neural network ensemble (NNE) is a learning paradigm where a collection of finite number neural networks are trained for the same task. It originates from Hansen [17] and Sollich’s work, which show that the generalization ability of neural network ensemble system can be significantly improved by integrating a number of neural networks, i.e., training a number of component neural networks and then combining their predictions. But how to generate individual network in the ensemble is always a hard problem. According to theoretical analysis, the diversity of individual networks is the most important factor to affect NNE’s performance. However, in conventional NNE method, the individual NNs are often trained independently, and the diversity of individual network comes from the training subset, different network structure and the difference and randomness of learning algorithm. One of the defects of such an approach is absent interaction and positive correlation individual networks are always present. Among them, the most classical approaches are Bagging [19] and Boosting [20]. They generate several training sets from the original training sets by random resampling. Because of randomness and blindness, plenty of individual network candidates are required to ensure individual networks diversity and error space coverage, which leads to extensive network redundancy and huge size of ensemble. To solve this problem, we believe that the key factor lies in how to evaluate the diversity of ensemble networks, and how to decompose ensemble error function into individual network’s training target function, which will automatically make individual networks maintain sufficient diversity. Based on this idea, an active diverse learning (ADL) algorithm is proposed in this paper by theoretical analysis of neural network ensemble error function. And a cooperative learning algorithm is presented, by which individual neural networks are actively guided to learn diversity.

II. Active Diverse Learning (ADL)

A. Corresponding theory analysis

Suppose that there are a training set $D = \{ (x(1), d(1)), ..., (x(N), d(N)) \}$, where $x \in \mathbb{R}^p$, $d$ is a scalar, and $N$ is the size of the training set. The assumption that the output $d$ is a scalar has been made merely to simplify exposition of ideas without loss of generality. The ensemble output is a simple averaging of outputs of a set of neural networks

$$f(n) = \frac{1}{M} \sum_{i=1}^{M} f_i(n).$$

Where, $M$ is the number of the individual neural networks in the ensemble, $f_i(n)$ is the output of network $i$ on the $n$th training pattern, and $f(n)$ is the output of the ensemble on the $n$th training pattern.

According to the theoretical analysis [21], following ensemble error function is induced

$$E = \overline{E} - \overline{A}.$$  

where

$$E = \sum_{n=1}^{N} (d(n) - f(n))^2$$

is the ensemble error function on the training set $D$.

$$\overline{E} = \sum_{i=1}^{M} \left[ \frac{1}{M} \sum_{n=1}^{N} (d(n) - f_i(n))^2 \right]$$

is simple averaging of the error function of individual networks in the ensemble on the training set $D$.

$$\overline{A} = \sum_{i=1}^{M} \left[ \frac{1}{M} \sum_{n=1}^{N} (f_i(n) - f(n))^2 \right]$$

is simple averaging of the diversity function of individual networks in the ensemble on the training set $D$.

From formula (2), we know that if ensemble error $E$ need to be minimized, the average component networks error $\overline{E}$ in the ensemble should minimum, and the diversities among component networks should be as large as possible. It is apparent that if component networks in the ensemble are positive correlation, that means each individual network in the ensemble has the same or similar output for the same input, then the diversity $\overline{A}$ of ensemble is close to 0, and the generalization error $E$ is close to the simple averaging generalization error $\overline{E}$ of individual networks in the ensemble. If each component network is independent (error is uncorrelated), in this...
By decomposing the ensemble error function, ADL introduces an error correlation term into the learning criterion function of each individual network in the ensembles, so that all the networks can be trained simultaneously and interactively on the same training data set. The ensemble error function $E$ has the form:

$$E = E - A = \frac{1}{M} \sum_{i=1}^{M} (E_i - A_i)$$

$$= \frac{1}{M} \sum_{i=1}^{M} \left\{ \sum_{n=1}^{N} (f_i(n) - d(n))^2 - \sum_{n=1}^{N} [f_i(n) - f(n)]^2 \right\}$$

$$= \frac{1}{M} \sum_{i=1}^{M} \left\{ \sum_{n=1}^{N} 2[f_i(n) - d(n)] - [f(n) - d(n)][f(n) - d(n)] \right\}$$

$$= \frac{1}{M} \sum_{i=1}^{M} \left\{ \sum_{n=1}^{N} [f_i(n) - d(n)][f(n) - d(n)] - [f(n) - d(n)]^2 \right\}$$

$$= \frac{1}{M} \sum_{i=1}^{M} \left\{ \sum_{n=1}^{N} [f_i(n) - d(n)][f(n) - d(n)] \right\} - E$$

thus,

$$E = \frac{1}{M} \sum_{i=1}^{M} \sum_{n=1}^{N} (f_i(n) - d(n))[f_i(n) - d(n)]$$

$$= \frac{1}{M} \sum_{i=1}^{M} \sum_{n=1}^{N} [f_i(n) - d(n)]^2 + \sum_{j=1}^{M} [f_j(n) - d(n)]$$

$$= \frac{1}{M} \sum_{i=1}^{M} \sum_{n=1}^{N} [f_i(n) - d(n)][f_i(n) - d(n)]$$

$$= \frac{1}{M} \sum_{n=1}^{N} [f_i(n) - d(n)][\sum_{j=1}^{M} [f_j(n) - d(n)]]$$

where the first term in the brace of the left side of (7), is the error function of network $i$. The second term is error correlation function between network $i$ and the rest of the ensembles, which can be denoted as $C_i$.

$$C_i = \sum_{n=1}^{N} [f_i(n) - d(n)][\sum_{j=1}^{M} [f_j(n) - d(n)]]$$

So, the ensemble error function $E$ can be restated as

$$E = \frac{1}{M} \sum_{i=1}^{M} (E_i + C_i) = \frac{1}{M} \sum_{i=1}^{M} J_i$$

Where,

$$J_i = \sum_{n=1}^{N} (f_i(n) - d(n))^2 + \sum_{n=1}^{N} (f_i(n) - d(n))[\sum_{j=1}^{M} (f_j(n) - d(n))]$$

(10)

is the error criterion function of network $i$.

Consequently, the minimization of the ensemble error function can be realized by minimizing the error criterion function $J_i$ of individual networks. In conventional NNE method, the training of individual networks only considers generalization error. However, here the error criterion function $J_i$ considers not only individual network generalization error, but also error correlation among individual networks. If $\sum_{i=1}^{M} C_i < 0$ (i.e., diverse individual network), then the error of ensemble will always be less than the average error of all component networks. Therefore the ensemble performance improvement is possible.

In convenience of analysis, the general form of $J_i$ on the $n$th training pattern is adopted,

$$J_i(n) = \frac{1}{2} [f_i(n) - d(n)]^2 + \lambda ([f_i(n) - d(n)] \sum_{j=1}^{M} (f_j(n) - d(n))]$$

$$= \frac{1}{2} [f_i(n) - d(n)]^2 + \lambda \sum_{j=1}^{M} (f_j(n) - d(n))$$

(11)

The parameter $0 \leq \lambda \leq 1$ is used to adjust the balance of accuracy and diversity of networks, which corresponds to the weight of error correlation degree between individual network and others network in the ensemble.

The partial derivative of $J_i(n)$ with respect to the output of network $i$ on the $n$th training pattern is

$$\frac{\partial J_i(n)}{\partial f_i(n)} = f_i(n) - d(n) + \lambda \sum_{j=1}^{M} (f_j(n) - d(n))$$

$$= f_i(n) - d(n) + \lambda M(f(n) - d(n)) - (f_i(n) - d(n))$$

$$= (1 - \lambda) [f_i(n) - d(n)] + \lambda M(f(n) - d(n))$$

(12)

1) For $\lambda = 0$, the individual networks’ learning only considers the difference between $f_i(n)$ and $d(n)$, and all networks are just trained independently. That is a special case in this paper.

2) For $0 < \lambda \leq 1$, the individual networks’ learning considers not only the difference between $f_i(n)$ and $d(n)$, but also the difference between $f(n)$ and $d(n)$. During the learning process, all networks will be trained interactively through the error correlation term $C_i$. It ensures that each individual network tends to learn the different parts or aspects of the problem. Especially for $\lambda = 1$, from (12), we get

$$\frac{\partial J_i(n)}{\partial f_i(n)} = M(f(n) - d(n))$$

(13)

Note that the empirical error function of the ensemble for the $n$th training pattern is defined by

$$J_{ene}(n) = \frac{1}{2} \sum_{i=1}^{M} [f_i(n) - d(n)]$$

(14)
\[
\frac{\partial J_{\text{out}}(n)}{\partial f_i(n)} = \frac{1}{M} \left( \frac{1}{M} \sum_{j=1}^{M} f_j(n) - d(n) \right) = \frac{1}{M} (f(n) - d(n))
\]

(15)

In the case, we get

\[
\frac{\partial J_i(n)}{\partial f_i(n)} \propto \frac{\partial J_{\text{out}}(n)}{\partial f_i(n)}.
\]

(16)

The minimization of the empirical error function of the ensemble is achieved by minimizing the error functions of the individual networks. From this point of view, active diverse learning provides a novel way to decompose the learning task of the ensemble into a number of subtasks for different individual networks.

B. learning algorithm

Suppose one component network \( f_i \) in the ensemble is a back-propagation (BP) network with three-layers, i.e. input layer, hidden layer and output layer (Fig.1).

![Network Diagram](image)

Figure 2. The structure of the BP neural network

To component network \( f_i \) (\( i = 1, 2, \ldots, M \)), assume we have \( n \) dimensions input, so the inputs of the network can be denoted as \( X = [x(1), x(2), \ldots, x(n)] \). \( Y_i \) and \( Z_i \) is the outputs of hidden layer and output layer, respectively. \( W_i \) represents the weight matrix between input layers and hidden layers, \( V_i \) denotes weight matrix between hidden layer and output layer. If \( d \) is the expected target output, then the output of hidden layers in component network \( f_i \) is

\[
Y_i = G_i(W_iX) = G_i(\text{net}_{Y_i}).
\]

(17)

Where, \( G_i \) denotes the activation function of hidden layers, and

\[
\text{net}_{Y_i} = W_iX.
\]

(18)

The output of component network \( f_i \) is

\[
Z_i = G_2(V_iY_i) = G_2(\text{net}_{Z_i}).
\]

(19)

Where, \( G_2 \) denotes the activation function of output layers, and

\[
\text{net}_{Z_i} = V_iY_i
\]

(20)

Both the hidden activation function and the output activation function can be chosen as common function such as logsig function, tansig function or pureline function. According to (11), the error criterion function of component network \( f_i \) is

\[
J_i = \frac{1}{2}(Z_i - d)^2 + \lambda_i(\sum_{j=1}^{M} (Z_j - d))^2. \quad (21)
\]

With standard back-propagation (BP) algorithm, the weight should be tuned along the negative gradient direction. Thus, the partial derivative of \( J_i \) with respect to output weight matrix \( V_i \) is

\[
\frac{\partial J_i}{\partial V_i} = \frac{\partial J_i}{\partial Z_i} \ast \frac{\partial Z_i}{\partial V_i} = (Z_i - d) \ast G_i'(\text{net}_{Z_i}) \ast Y_i \quad (22)
\]

where

\[
\delta_i = (Z_i - d) \ast G_i'(\text{net}_{Z_i}). \quad (23)
\]

And the partial derivative of \( J_i \) with respect to the hidden weight matrix \( W_i \) is

\[
\frac{\partial J_i}{\partial W_i} = \frac{\partial J_i}{\partial Z_i} \ast \frac{\partial Z_i}{\partial W_i} = (Z_i - d) \ast G_i'(\text{net}_{Z_i}) \ast V_i \ast G_z'(\text{net}_{Y_i}) \ast X \quad (24)
\]

where

\[
\delta_w = \delta_i \ast V_i \ast G_z'(\text{net}_{Y_i}). \quad (25)
\]

Due to the weight matrix corrected value \( \Delta W_i \) and \( \Delta V_i \) are proportional to the error criterion function negative gradient, the corrected value of output layer weight matrix is

\[
\Delta V_i = -lr \ast \frac{\partial J_i}{\partial V_i} = lr \ast \delta_i \ast Y_i, \quad (26)
\]

\[
V_i = V_i + \Delta V_i. \quad (27)
\]

And the corrected value of hidden layer weight matrix is

\[
\Delta W_i = -lr \ast \frac{\partial J_i}{\partial W_i} = lr \ast \delta_w \ast X, \quad (28)
\]

\[
W_i = W_i + \Delta W_i. \quad (29)
\]

Where, \( lr \) is learning rate.
III. APPLICATION TO TRANSFORMER FAULT DIAGNOSIS

A. data processing and fault encoding

40 power transformer fault samples are used in our experiment. According to cross-validation technology, the data are divided into 5 subsets. Each time, one subset is selected as testing set, and the remaining 4 subsets are left as training set. The corresponding results are the statistical results that are evaluated over 20 runs on each subset with cross-validation technology.

Five characteristic gases (H₂, CH₄, C₂H₆, C₂H₄, C₂H₂) are used as input feature for neural network. In order to reduce the impacts of range difference, normalization processing is adopted to original DGA data.

\[
x' = \frac{x_p - x_{\text{min}}}{x_{\text{max}} - x_{\text{min}}},
\]

(30)

where, \(x_p\) denotes the concentration of original gases, \(x_{\text{max}}\) and \(x_{\text{min}}\) denote the maximum and minimum value of the concentration of corresponding gases in the samples, respectively. Therefore, \(x'\) is the normalization value.

We divide the fault into five fault categories, i.e. High Temperature overheating, Low-energy Discharge, Medium Temperature overheating, Arc Discharge and Normal. And corresponding codes are 10000 (High Temperature overheating), 01000 (Low-energy Discharge), 00100 (Medium Temperature overheating), 00010 (Arc Discharge), and 00001 (Normal).

In the experiments, BP neural network with single hidden layer is adopted, which meets non-linear mapping relation between phenomenon and cause of the fault with a minimum computation cost. According to trail and error, single hidden layer BP network with 12 hidden nodes is adopted as component networks in the ensemble.

B. active diverse learning (ADL) algorithm

All component networks have the same structure, that is to say, an individual BP NN with three layers, which includes one hidden layer with 12 nodes and an output layer with 5 nodes. We adopt \(tansig\) transfer function for hidden layer and \(losig\) transfer function for output layer. Corresponding parameters are as follows: allowable maximum population size \(M_{\text{max}}\) (8), maximum number of epochs \(\text{max-epoch}\) (1000), the target error of the ensemble \(e\) (0.001), the learning rate \(lr\) (0.02). The number of component networks can be determined by algorithm dynamically. Detail steps of ADL algorithm is follows:

Step1: Randomly initialize all connection weights near zero, and build the ensemble network whose original ensemble size is \(M = 2\). Then train the NNE by diverse learning, and calculate \(J_{\text{ens}}(n)\) according to (12);

Step2: If \(J_{\text{ens}} > e\), build a new network \(f'\); else, turn to Step 6;

Step3: Let network \(f'\) join into the ensemble, and revise \(M = M + 1\). Train the new NNE by ADL algorithm, and then calculate \(J'_{\text{ens}}\) with (12);

Step4: If the new ensemble, which combines network \(f'\), can improve the generalization performance, i.e., \(J'_{\text{ens}} < J_{\text{ens}}\), then accept the network \(f'\) and \(J_{\text{ens}} = J'_{\text{ens}}\), else, \(M = M - 1\), turn to Step2 to build a new network \(f'\) again;

Step5: If \(J_{\text{ens}} < e\), or \(M > M_{\text{max}}\), or the number of epochs \(\text{epoch} > \text{max-epoch}\), save the generated NNE; else, turn to Step2;

Step6: Test the fault category of the test samples by constructed NNE.

IV. EXPERIMENTS RESULTS AND ANALYSIS

To evaluate the performance of our ADL algorithm, several experiments were conducted, which include diversity measure among individual networks, the comparison between single BPN and NNE methods, the comparison between independent learning (\(\lambda = 0\)) and diversity learning (\(\lambda = 1\)), and the comparison between our ADL and conventional NNE methods.

A. diversity measure among individual networks

To properly evaluate the diversity among individual networks, the diversity measure [22, 23] as follow is adopted, it has been widely used in other literatures.

\[
\text{Dis}_{i,j} = \frac{N_{i1}^{01} + N_{i1}^{10}}{N_{i1}^{01} + N_{i1}^{10} + N_{i1}^{11} + N_{i1}^{00}},
\]

(31)

where \(\text{Dis}_{i,j}\) denotes the diversity between network \(i\) and network \(j\), \(N_{ab}\) refers to the number of examples that network \(i\) classifies correctly (\(a = 1\)) or incorrectly (\(a = 0\)), and network \(j\) classifies correctly (\(b = 1\)) or incorrectly (\(b = 0\)) on sample sets. To observe the effect of error correlation term \(C_{ij}\), Table I shows the diversity among the individual networks trained by ADL with different \(\lambda\) values. According to the running result of ADL, 4 individual networks are proper for the ensemble.

| \(\lambda\) | \(\text{Dis}_{i,j} = 0.0667\) | \(\text{Dis}_{i,j} = 0.2000\) | \(\text{Dis}_{i,j} = 0.0667\) |
| 0 | \(\text{Dis}_{i,j} = 0.1333\) | \(\text{Dis}_{i,j} = 0.0000\) | \(\text{Dis}_{i,j} = 0.1333\) |
| 0.5 | \(\text{Dis}_{i,j} = 0.3333\) | \(\text{Dis}_{i,j} = 0.0400\) | \(\text{Dis}_{i,j} = 0.4667\) |
| 1 | \(\text{Dis}_{i,j} = 0.6000\) | \(\text{Dis}_{i,j} = 0.0400\) | \(\text{Dis}_{i,j} = 0.5333\) |

The diversity among individual networks in NNE is relatively small when \(\lambda\) is 0, mostly close to 0. It means that the correlations among them are generally positive. When \(\lambda\) increases to 0.5, the values of the diversity are
obviously enhanced. All of them became relatively larger when \( \lambda \) is further increased to 1. In this case, there is smaller correlation between different individual networks. The empirical results indicate that ADL algorithm is effective, and it can guide the diversity learning among individual networks. Each individual network learns different parts or aspects of the training sets.

**B. the performance comparison among single BPNN, independent NNE and diversity NNE**

Under the same condition, Table II shows the minimum (Min), maximum (Max) and statistical mean (Mean) accuracy among single BPNN, independent NNE and diversity NNE. They are averaged over 20 runs based on cross-validation. All network parameters are the same, epoch is 1000 and learning rate (lr) is 0.02.

The single BPNN’s accuracy is about 82.47%, and when independent NNE(\( \lambda = 0 \)) is applied, due to the independent learning of individual networks, the component networks present larger positive correlation, the accuracy is only 83.53%, which has no superiority to single BPNN. However, the accuracy of diverse NNE (\( \lambda = 1 \)) can achieve 92%. The results show that diversity NNE is superior to single BPNN and independent NNE.

**C. the comparison between independent learning and diverse learning**

To further present the characteristic of independent learning and diverse learning, the training error curves of these two cases are illustrated in Fig. 2 and Fig. 3.

We can distinguish the error curves of component networks and network ensemble in Fig. 2, when epochs range from 0 to 250, then after 250 epochs, the error curves become convergent status, all of the error curves are mostly overlap and it is hardly impossible to distinguish. It shows that diversity among component networks in independent learning is relatively small, and independent NNE’s ensemble error is almost identical to the average of individual networks’ error. However, Fig. 3 indicates that the ADL’s error is always less than that of individual networks. Moreover individual network’s accuracy may not very high. It presents that the diversity among individual networks is more important than the accuracy of individual network in ensemble, and it also means that not all individual networks must be trained to high precision, which implies that we can reduce the number of iterations and the training time of individual neural networks.

**TABLE III.**

THE SIZE OF THE CORRECTLY CLASSIFIED SAMPLES OF INDIVIDUAL NETWORKS CREATED RESPECTIVELY BY DIVERSE NNE (\( \lambda = 1 \)) AND INDEPENDENT NNE (\( \lambda = 0 \)) ON THE TEST SET

<table>
<thead>
<tr>
<th>( \lambda = 1 )</th>
<th>( N_{i=11} )</th>
<th>( N_{i=8} )</th>
<th>( N_{i=13} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( N_{i=11} )</td>
<td>( N_{i=11} )</td>
<td>( N_{i=9} )</td>
<td></td>
</tr>
<tr>
<td>( N_{i=7} )</td>
<td>( N_{i=7} )</td>
<td>( N_{i=8} )</td>
<td></td>
</tr>
<tr>
<td>( N_{i=9} )</td>
<td>( N_{i=3} )</td>
<td>( N_{i=5} )</td>
<td></td>
</tr>
<tr>
<td>( N_{i=5} )</td>
<td>( N_{i=8} )</td>
<td>( N_{i=3} )</td>
<td></td>
</tr>
<tr>
<td>( N_{i=2} )</td>
<td>( N_{i=6} )</td>
<td>( N_{i=8} )</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>( \lambda = 0 )</th>
<th>( N_{i=13} )</th>
<th>( N_{i=14} )</th>
<th>( N_{i=12} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( N_{i=13} )</td>
<td>( N_{i=13} )</td>
<td>( N_{i=11} )</td>
<td></td>
</tr>
<tr>
<td>( N_{i=12} )</td>
<td>( N_{i=12} )</td>
<td>( N_{i=13} )</td>
<td></td>
</tr>
<tr>
<td>( N_{i=12} )</td>
<td>( N_{i=12} )</td>
<td>( N_{i=11} )</td>
<td></td>
</tr>
</tbody>
</table>

Table III shows the statistic samples, which could be correctly classified by independent learning or diverse learning in a test subset (15 test samples). Suppose that \( S_i \) is the set of samples that correctly classified by individual network \( i \). \( N_i \) is the size of set \( S_i \). Then \( N_j \) means the size of samples that can be correctly classified by both individual networks \( i \) and \( j \), i.e., the
size of set $S_i \cap S_j$. And so forth, $N_{(i,j)}$ denotes the size of set $S_i \cap S_j \cap \cdots \cap S_k$.

It is evident from Table III that the different individual networks created by diverse learning are able to specialize to different parts of the testing set. For instance, in Table III the size of correct response sets $S_1$, $S_2$, $S_3$ and $S_4$ at $\lambda=1$ are 11, 8, 13, 11, respectively, but the size of the intersection $S_1 \cap S_2$ is 5, and the size of $S_1 \cap S_2 \cap S_3 \cap S_4$ is only 3. It indicates that the individual network has great diversity. In contrast, the size of set $S_i \cap S_j \cap \cdots \cap S_k$ also reaches 11. It implies that the diversity among individual networks is very small.

D. The comparison between ADL and other conventional methods

To verify the performance of our ADL method, other conventional methods such as IEC three ration method, BP network, conventional Bagging and Boosting algorithm, are conducted in this power fault diagnosis problem. Table IV illustrates the statistical results on cross-validation over 20 runs. Among them, BPNN has the same network structure and training iterations. Bagging and Boosting consist of 4 component networks too in their ensemble.

Table V. Comparison of the accuracy among different conventional fault diagnosis.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Accuracy</th>
<th>Methods</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bagging</td>
<td>86.67</td>
<td>BPNN</td>
<td>82.47</td>
</tr>
<tr>
<td>Boosting</td>
<td>86.67</td>
<td>Independent NNE ($\lambda = 0$)</td>
<td>83.53</td>
</tr>
<tr>
<td>IEC</td>
<td>66.67</td>
<td>Diverse NNE ($\lambda = 1$)</td>
<td>92.00</td>
</tr>
</tbody>
</table>

The performance of IEC is worst, it is only 66.67%. Single BPNN is 82.47%, while Bagging and Boosting have approximate performance, the accuracy are both about 86.67%. Compared with single BPNN, the performance is improved but it is finite. The accuracy of independent NNE is 83.53%, which is a little better than that of single BPNN, but lower than that of Bagging and Boosting. We believe the reason is the individual network diversity of independent NNE originates from the variety weights of the network, while the diversities of Bagging and Boosting methods are not only originate from weights diversity, but also the disturbance of training samples. Thus, it is more possible to generate more diverse individual networks. The results show that the performance of diverse NNE is greatly improved compared with other methods. And it achieves best accuracy 92%.

In addition, the accuracy of component networks and ensemble network with different ensemble methods are listed from Table V to Table VIII, which is the statistical result on one testing subset over 10 runs.

The data in Table VI and Table VII show that Bagging and Boosting have approximate performance. Bagging has four cases (index 4, 6, 7, 10), where the performance of ensemble is better than that of best individual network. And four cases (index 2, 3, 5, 8), the ensemble performance are identical to best individual network. And only two cases are less (index 1, 9). In Boosting, the performance of ensemble is better than the best individual network presents three times (index 3, 8, 10), and five times for equal (index 1, 4, 5, 6, 7), two times for less (index 2, 9).

Most of the performance of independent learning ensemble in ten times operation are less than or equal to the best individual network. there are only one time (index 9) that the ensemble performance is better than the
best individual network, two times for equal (index 3, 7), and seven times for less.

The ensemble performance of diverse learning in ten operations is always better than or equal to the best individual. The equal case happens only two times (index 6 and 7), while eight times for better.

### TABLE VII

<table>
<thead>
<tr>
<th>No.</th>
<th>Component 1</th>
<th>Component 2</th>
<th>Component 3</th>
<th>Component 4</th>
<th>Ensemble network</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>80.00</td>
<td>80.00</td>
<td>73.33</td>
<td>80.00</td>
<td>73.33</td>
</tr>
<tr>
<td>2</td>
<td>86.67</td>
<td>73.33</td>
<td>73.33</td>
<td>73.33</td>
<td>73.33</td>
</tr>
<tr>
<td>3</td>
<td>80.00</td>
<td>86.67</td>
<td>80.00</td>
<td>80.00</td>
<td>86.67</td>
</tr>
<tr>
<td>4</td>
<td>86.67</td>
<td>73.33</td>
<td>80.00</td>
<td>73.33</td>
<td>80.00</td>
</tr>
<tr>
<td>5</td>
<td>73.33</td>
<td>80.00</td>
<td>73.33</td>
<td>86.67</td>
<td>73.33</td>
</tr>
<tr>
<td>6</td>
<td>73.33</td>
<td>73.33</td>
<td>80.00</td>
<td>73.33</td>
<td>73.33</td>
</tr>
<tr>
<td>7</td>
<td>80.00</td>
<td>80.00</td>
<td>80.00</td>
<td>73.33</td>
<td>80.00</td>
</tr>
<tr>
<td>8</td>
<td>80.00</td>
<td>93.33</td>
<td>80.00</td>
<td>80.00</td>
<td>80.00</td>
</tr>
<tr>
<td>9</td>
<td>80.00</td>
<td>80.00</td>
<td>80.00</td>
<td>73.33</td>
<td>86.67</td>
</tr>
<tr>
<td>10</td>
<td>73.33</td>
<td>73.33</td>
<td>86.67</td>
<td>80.00</td>
<td>80.00</td>
</tr>
</tbody>
</table>

### TABLE VIII

<table>
<thead>
<tr>
<th>No.</th>
<th>Component 1</th>
<th>Component 2</th>
<th>Component 3</th>
<th>Component 4</th>
<th>Ensemble network</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>80.00</td>
<td>80.00</td>
<td>73.33</td>
<td>80.00</td>
<td>73.33</td>
</tr>
<tr>
<td>2</td>
<td>86.67</td>
<td>73.33</td>
<td>73.33</td>
<td>73.33</td>
<td>73.33</td>
</tr>
<tr>
<td>3</td>
<td>80.00</td>
<td>86.67</td>
<td>80.00</td>
<td>80.00</td>
<td>86.67</td>
</tr>
<tr>
<td>4</td>
<td>86.67</td>
<td>73.33</td>
<td>80.00</td>
<td>73.33</td>
<td>80.00</td>
</tr>
<tr>
<td>5</td>
<td>73.33</td>
<td>80.00</td>
<td>73.33</td>
<td>86.67</td>
<td>73.33</td>
</tr>
<tr>
<td>6</td>
<td>73.33</td>
<td>73.33</td>
<td>80.00</td>
<td>73.33</td>
<td>73.33</td>
</tr>
<tr>
<td>7</td>
<td>80.00</td>
<td>80.00</td>
<td>80.00</td>
<td>73.33</td>
<td>80.00</td>
</tr>
<tr>
<td>8</td>
<td>80.00</td>
<td>93.33</td>
<td>80.00</td>
<td>80.00</td>
<td>80.00</td>
</tr>
<tr>
<td>9</td>
<td>80.00</td>
<td>80.00</td>
<td>80.00</td>
<td>73.33</td>
<td>86.67</td>
</tr>
<tr>
<td>10</td>
<td>73.33</td>
<td>73.33</td>
<td>86.67</td>
<td>80.00</td>
<td>80.00</td>
</tr>
</tbody>
</table>

### TABLE IX

<table>
<thead>
<tr>
<th>No.</th>
<th>Component 1</th>
<th>Component 2</th>
<th>Component 3</th>
<th>Component 4</th>
<th>Ensemble network</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>80.00</td>
<td>80.00</td>
<td>73.33</td>
<td>80.00</td>
<td>73.33</td>
</tr>
<tr>
<td>2</td>
<td>86.67</td>
<td>73.33</td>
<td>73.33</td>
<td>73.33</td>
<td>73.33</td>
</tr>
<tr>
<td>3</td>
<td>80.00</td>
<td>86.67</td>
<td>80.00</td>
<td>80.00</td>
<td>86.67</td>
</tr>
<tr>
<td>4</td>
<td>86.67</td>
<td>73.33</td>
<td>80.00</td>
<td>73.33</td>
<td>80.00</td>
</tr>
<tr>
<td>5</td>
<td>73.33</td>
<td>80.00</td>
<td>73.33</td>
<td>86.67</td>
<td>73.33</td>
</tr>
<tr>
<td>6</td>
<td>73.33</td>
<td>73.33</td>
<td>80.00</td>
<td>73.33</td>
<td>73.33</td>
</tr>
<tr>
<td>7</td>
<td>80.00</td>
<td>80.00</td>
<td>80.00</td>
<td>73.33</td>
<td>80.00</td>
</tr>
<tr>
<td>8</td>
<td>80.00</td>
<td>93.33</td>
<td>80.00</td>
<td>80.00</td>
<td>80.00</td>
</tr>
<tr>
<td>9</td>
<td>80.00</td>
<td>80.00</td>
<td>80.00</td>
<td>73.33</td>
<td>86.67</td>
</tr>
<tr>
<td>10</td>
<td>73.33</td>
<td>73.33</td>
<td>86.67</td>
<td>80.00</td>
<td>80.00</td>
</tr>
</tbody>
</table>

### CONCLUSIONS

The key factor of generalization ability improvement lies in the diversity of the individual network in ensemble. For conventional ensemble methods, whose diversities originate from the difference of the weights, samples, learning algorithm and structures of the network, due to the randomness of the method, the diversity can not be guaranteed. This paper introduces an active approach to design individual neural network in ensemble, by added an error correlation term, our ADL method can realize diverse learning, which improves the ensemble’s generalization ability with a more stable and reliable way. As a universal method, our method can be applied to other classification problems. How to actively design the network structure and select training samples are our future work.

### ACKNOWLEDGMENT

The authors are grateful to the anonymous reviewers for their constructive comments, which helped to improve the clarity of this paper greatly. This work is supported by National Natural Science Foundation of China (Grant no. 60775047, 60835004), Natural Science Foundation of Hunan Province (Grant no. 06JJ50112).

### REFERENCES


Yu Xu was born in Yueyang, China, in 1985. She received her B.E. degree in electronic information engineering from Xiangtan University, China in 2007. She is a Master student of power electronic and electrical drive, Xiangtan University, in China. Her research interests are neural network ensemble and fault diagnosis on power transformer.

Dongbo Zhang was born in Shaoyang, China, in 1973. He received his B.S. and M.S. degree in Computer Science from Xiangtan University, China in 1996 and 2001, respectively, and received Ph.D. degree in Control Science and Technology from Hunan University, China, in 2007. From 1996 he worked in Xiangtan University, and has been an associate professor in Xiangtan University since 2006. His research interests include rough sets, neural network, digital image processing, pattern recognition and artificial intelligence.

Yaonan Wang was born in Jiangxi, China, in 1957. He received the B.S. degree in Computer Engineering from East China Technology Institute, China in 1981. He received the M.S. and Ph.D. degrees in Electrical Engineering from Hunan University, China, in 1990 and 1994, respectively. From 1994 to 1995 he was a Postdoctoral Research Fellow in National University of Defense Technology. From 1981 to 1994 he worked in East China Technology Institute. Since 1995, he has been a professor in Hunan University, China. His research interests are image processing, pattern recognition, and industrial process control.