

Analog Circuit Fault Diagnosis Based on Adaptive Gaussian Deep Belief Network

WANG Ying-chen¹, DUAN Xiu-sheng²

¹Department of Electronic and Optical Engineering, Army Engineering University of PLA, Shijiazhuang, China

²School of mechanical engineering, Shijiazhuang Tiedao University, Shijiazhuang, China

Abstract. Aiming at the problem that the traditional intelligent fault diagnosis method is overly dependent on feature extraction and the lack of generalization ability, deep belief network is proposed for the fault diagnosis of the analog circuit; Then, by analyzing the deficiency of deep belief network application, a Gaussian deep belief network based on adaptive learning rate is proposed. The automatic adjustment learning step is adopted to further improve fault diagnosis efficiency and diagnosis accuracy; Finally, particle swarm support vector machine to extract the fault characteristics to identify. The simulation results of circuit fault diagnosis show that the algorithm has faster convergence speed and higher fault diagnosis accuracy.

1 Introduction

Feature extraction is a key link in fault diagnosis of analog circuits and has always been a major problem. At present, the feature extraction methods commonly used in analog circuit fault diagnosis include principal component analysis(PCA)[1], wavelet (packet) analysis [2] and nuclear analysis. These methods have their limitations. For example, the PCA method is only suitable for linear feature extraction, wavelet (packet) analysis and nuclear analysis involve the selection and consideration of many factors such as wavelet basis and nuclear parameters. It is influenced by experience and is essential. These analytical methods and the data itself are isolated from each other, and it is difficult to ensure that the extracted features are the essential characteristics of the data. Deep learning shows special advantages in feature extraction. It can learn features from the data autonomously. The learned features can better reflect the nature of data. At present, the application of deep learning in the field of analog circuit fault diagnosis is almost blank. This paper attempts to use adaptive gaussian deep belief network(AGDBN), the main model of deep learning, to extract fault features of analog circuits, and demonstrates its effectiveness through experiments.

2 Deep learning feature extraction model

Deep belief network (DBN) is constructed by stacking several restricted Boltzmann machines (RBM) [3]. The architecture of a DBN with three stacked RBMs is shown in Figure

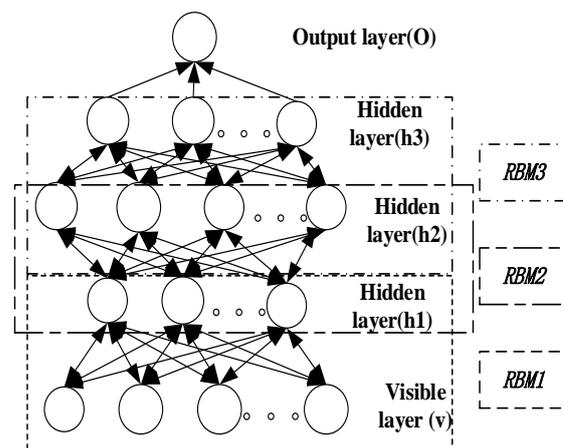


Figure 1. The structure of DBN.

2.1 Restricted Boltzmann Machine

RBM is a double-layer neural network consisting of a visible layer and an implicit layer. There is a weighted connection between each visible and hidden unit, but there is no connection between its peer nodes.

Corresponding author: f_author, 791191357@qq.com

In RBM, each node is a binary value, and the fault signal is a continuous value, and the binary unit RBM does not simulate well. Therefore, gaussian restricted boltzmann machine (GRBM) is used in this paper to replace the binary visual layer node value with a continuous real number with a Gaussian distribution, so that the input layer can accept continuous signals. The hidden layer is still used. Obeying the binary neuron node of the Bernoulli distribution, the energy function of a Gaussian-restricted Boltzmann machine is

$$E(v, h; \theta) = \sum_{i=1}^m \frac{(v_i - a_i)^2}{2\sigma_i^2} - \sum_{i=1}^m \sum_{j=1}^n \frac{v_i}{\sigma_i} w_{ij} h_j - \sum_{j=1}^n b_j h_j \quad (1)$$

The model parameter is $\theta_{GRBM} = \{w_{ij}, a_i, b_j\}$, Among them w_{ij} are the weights between the visible v and hidden units h , a_i and b_j are offsets, n and m the number of visible and hidden units, respectively. Based on this energy function, get the (v, h) joint probability

$$p(v, h; \theta) = \frac{1}{Z(\theta)} \exp(-E(v, h; \theta)) \quad (2)$$

Since there is no connection between the hidden layer and the visible layer, the nodes at the same layer are independent of each other. When a visible element state is given, the activation probability of the first element of the hidden layer is

$$P(h_j = 1|v; \theta) = \sigma\left(\sum_{i=1}^m w_{ij} \frac{v_i}{\sigma_i} + b_j\right) \quad (3)$$

The activation probability of the visual unit is

$$P(v_i = v|h; \theta) = N\left(a_i + \sigma_i \sum_{j=1}^n w_{ij} h_j, \sigma_i^2\right) \quad (4)$$

In the formula, $\sigma = 1/(1 + e^{-x})$ is a sigmoid function, $N(\mu, \sigma^2)$ means a Gaussian distribution of variance. In order to make the model simpler to implement, each component of the input data is usually normalized to zero mean and unit variance.

The training of GRBM uses the CD algorithm proposed by Hinton [4]: using Gibbs sampling to start from any state of the training sample, calculate the probability of the hidden unit according to equation (3); then, fix the implicit unit according to (4) Reconstruct the visual unit. In this way, approximate adjustment rules for model parameters $\theta_{GRBM} = \{w_{ij}, a_i, b_j\}$ are obtained:

$$\Delta w_{ij} = \varepsilon \left(\left\langle \frac{1}{\sigma} v_i h_j \right\rangle_{data} - \left\langle \frac{1}{\sigma} v_i h_j \right\rangle_{recon} \right) \quad (5)$$

$$\Delta a_i = \varepsilon \left(\left\langle \frac{1}{\sigma} v_i \right\rangle_{data} - \left\langle \frac{1}{\sigma^2} v_i \right\rangle_{recon} \right) \quad (6)$$

$$\Delta b_j = \varepsilon \left(\left\langle h_j \right\rangle_{data} - \left\langle h_j \right\rangle_{recon} \right) \quad (7)$$

ε is the learning rate. In order to better train the GRBM, training samples are usually divided into multiple smaller subsets. After each subset is trained, the model parameters are updated.

2.2 Deep Learning Training Algorithm

Gaussian DBN (GDBN) feature extraction process uses layer-by-layer unsupervised greedy pre-training [5], and the algorithm is performed as follows.

- 1) First fully train the first GRBM to get its weight matrix and offset;
- 2) Fix the weight and offset of the first GRBM, then use the state of its recessive neurons as the input vector for the second GRBM;
- 3) After fully training the second GRBM, stack the second GRBM above the first GRBM;
- 4) Repeat the above three steps any number of times until the last layer.

The low-level GRBM can extract the details of the original data, the high-level GRBM extracts the attribute categories of the data, and abstracts from the lower level to the higher level, abstracting the essence of the data.

2.3 Adaptive Deep Belief Network

The learning rate is a decisive factor influencing the speed of GDBN convergence and extracting feature performance. In the gradient descent algorithm, the learning rate determines the distance that the parameter moves in the direction of the gradient descent during each iteration. If the learning rate is too large, it is likely to cross the optimal value and jump back and forth near the local optimal point. If the learning rate is too small, the optimization efficiency may be too low and the algorithm cannot converge for a long time, so the learning rate is critical to algorithm performance. important.

The standard GRBM uses a global learning rate based on a comparative divergence (CD) fast learning algorithm to train the network [6]. Once selected, the learning rate remains constant throughout the training process and the adaptability is poor. In order to obtain better fault diagnosis performance and speed up the learning process of the network, based on the traditional DBN algorithm, an adaptive learning rate algorithm based on reconstruction error is introduced, which can adjust the learning rate adaptively in each step of training. The reconstruction error is the most widely used judgment criterion in the RBM learning rules and is defined as

$$Err = \frac{1}{N} \sum_{i=1}^N (v_i - v_i') \quad (8)$$

In the formula: v_i is the samples for the visible layer, v_i' is the reconstruction data obtained from the hidden layer, N is the number of samples. The learning rate can be adjusted at each of the GRBM training rules. Dynamic adjustment of learning rate

$$\varepsilon_{now} = \begin{cases} \varepsilon_{old} + \eta & \text{if } \Delta > 0 \\ \varepsilon_{old} * (1 - \eta) & \text{if } \Delta < 0 \end{cases} \quad (9)$$

In the formula: ε_{old} is the original learning rate, ε_{now} is the updated learning rate, $\Delta = Err_{old} - Err_{now}$. Compared with continuous reconstruction error Err_{now} and Err_{now} , when the

reconstruction error is reduced, the learning rate is increased; when the error is increased, the learning rate is decreased.

3 fault diagnosis model

The fault diagnosis model based on the GDBN-SVM model is shown in Figure 2.

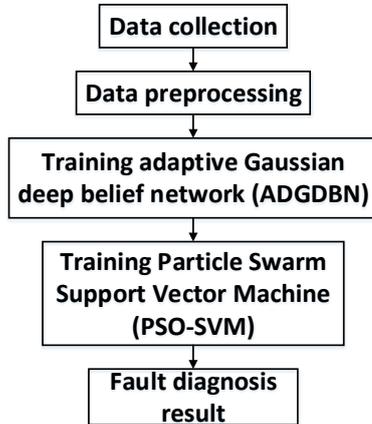


Figure 2. Flow chart of fault diagnosis model.

The steps can be summarized as follows:

- 1) Use the sensor to obtain the signal of the circuit under different fault conditions, preprocess it, and use it as the input of the network;
- 2) Establish a multi-hidden layer adaptive Gaussian depth belief network, unsupervised layer-by-layer pre-training, and extract deep fault features;
- 3) Automatically extract feature input support vector machines for deep network training, optimize the parameters of the support vector machine using the particle swarm algorithm, and output fault categories to complete the fault diagnosis.

4 experimental analysis

In order to verify the correctness and effectiveness of this method, The Sallen-Key circuit in ITC97 was selected as the experimental circuit, and its circuit diagram and components are shown in Figure3.

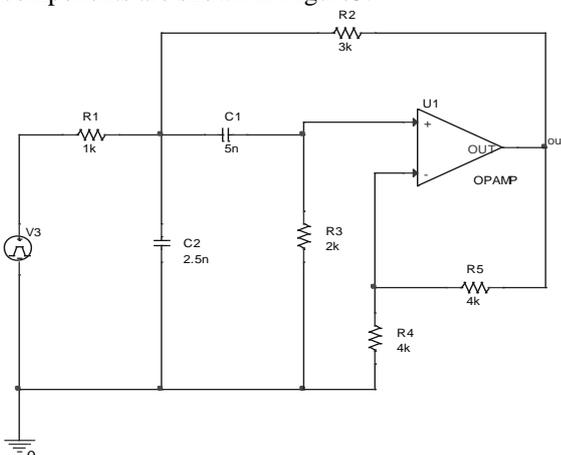


Figure 3. Electric circuit of Sallen-Key.

Through sensitivity analysis, it was found that R2, R3, C1, and C2 had the greatest impact on the band-pass filter circuit, where the allowable tolerance of the resistor and capacitor was set to 5% and 10%, respectively.

Table 1. Fault value of device.

Fault Condition	Nominal Value	Faulty value
C1 ↑	5nF	7.5nF
C1 ↓	5nF	2.5nF
C2 ↑	5nF	7.5nF
C2 ↓	5nF	2.5nF
R2 ↑	3kΩ	4.5kΩ
R2 ↓	3kΩ	1.5kΩ
R3 ↑	2kΩ	3kΩ
R3 ↓	2kΩ	1kΩ

The circuit was simulated with OrCAD/PSpice 10.5. Monte Carlo analysis was performed 1000 times for eight fault conditions and normal conditions, and 320 feature values were extracted at the out end each time. In each case, 1000 sets of fault raw data were formed, of which 700 were used for training and 300 were used for testing.

4.1 Experimental Procedure

1) The original data is 320-dimension, so the number of DBN input layer nodes is 320. Then the hidden layer with the number of nodes 200 and 100 is used to compress the data in order to extract the deep features; the extracted features are input into the SVM classifier and trained. The model tests the model to output fault diagnosis results.

2) In order to verify the superiority of the proposed method, this method is compared with the standard GDBN under the same network structure.

4.2 Analysis of The Experiment

Figure 4 shows the accuracy of the identification of each fault condition of the circuit by the method of this paper.

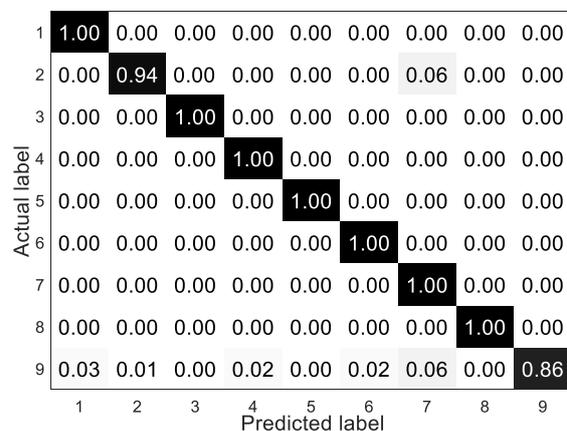


Figure 4. Confusion matrix of the proposed method

Table 2 shows the fault diagnosis capabilities of the proposed method and the standard GDBN model. It can be seen that the accuracy of the proposed fusion model is higher than 98.3%, and the classification performance is better than the standard GDBN.

Table 2. Comparison of diagnostic accuracy.

Method	Diagnostic accuracy
Standard GDBN	92.5%
Adaptive GDBN	98.3%

5 Conclusions

This paper proposes a deep learning model of adaptive Gaussian deep belief network to solve the problem of fault diagnosis of analog circuits. The model directly processes the original vibration signal, does not need time-consuming manual feature extraction process, and adopts an adaptive learning rate to improve the network convergence speed and diagnostic accuracy, and is superior to the traditional single deep learning model.

References

1. Parker M., 2014. *Principal component analysis*.
2. Garcia, C., Zikos, G., & Tziritas, G., 2000. Wavelet packet analysis for face recognition. *Image and Vision computing*, 18(4), 289-297.
3. Hinton G E., 2012. A practical guide to training restricted Boltzmann machines, *Neural networks: Tricks of the trade*. Springer, Berlin, Heidelberg, 599-619.
4. Hinton G E., 2002. Training products of experts by minimizing contrastive divergenc. *Neural computation*, 14(8): 1771-1800.
5. Hinton G., 2014. Where do features come from?. *Cognitive science*, 38(6): 1078-1101.
6. Bengio Y., 2009. Learning deep architectures for AI. *Foundations and trends in Machine Learning*, 2(1): 1-127.