Journal of Physics: Conference Series

PAPER • **OPEN ACCESS**

Optimization of DBN Network Structure Based on Information Entropy

To cite this article: Ziliang Huang *et al* 2019 *J. Phys.: Conf. Ser.* **1176** 032046

Optimization of DBN Network Structure Based on Information Entropy

Ziliang Huang[1], Yan Cao[1], Tianbao Wang[1,*]

[1]Chengdu University of Information Technology, Shuangliu Chengdu Sichuan, China

*Corresponding author e-mail: 819086031@qq.com

Abstract. To determine the appropriate depth of DBN network and hidden layer neurons at the same time, the information entropy of the input layer is analyzed on the basis of information entropy and the traditional reconstruction error calculation and the decision of network depth. Thus, according to the relationship between information entropy and hidden layers, an optimization method based on information entropy to determine the number of hidden neurons is proposed, which makes the structure of the DBN network model tend to be better. Experimental results on the handwritten numeral recognition demonstrated that the proposed method is capable of self-organizing the depth of the network and hide the number of neurons in the hidden layer, effectively optimize the DBN network structure, reduce the training time of the network, as well as improve the network accuracy and recognition accuracy.

1. Introduction

With the growing interest of artificial intelligence research, the artificial neural network [1] has become an indispensable field in artificial intelligence with its excellent performance. However, with the continuous expansion of the artificial neural network [2], the training time is longer and the work efficiency is lower. In order to improve the adverse effects caused by the deep structure, Professor Hinton of the University of Toronto proposed the Deep Belief Network (DBN) [3], which obtained breakthrough in the artificial neural network in the establishment of multiple hidden layers.

At present, DBN has been successfully deployed in many fields [4,5], such as speech recognition [6], computer vision [7], information retrieval [8], etc. However, it is still in the process of development and optimization, and many problems have not been effectively solved. For example, for a specific application scenario, how do we set the appropriate network depth and hidden layer neurons to effectively represent the information or features of the original data？

Some researchers initialized with the mechanism of probabilistic model evolutionary algorithm [9,10], and then proposed a DBN structure optimization algorithm based on BOA [12]. Some academic researchers have analyzed the training process of supervised learning and unsupervised learning in DBN, and obtained the relationship between network depth and training error. Based on this research, a specific deep determination method based on RBM reconstruction error is proposed and investigated[13]. However, it only addresses the problem of network depth. For the problem of choosing the amount of neurons in the hidden layer, there is still no effective solution.

On this basis, from the point of view of information expression, this paper investigates the internal relationship between information entropy, input layer and hidden layers. Then, according to the relationship between information entropy [14] and hidden layer, an optimization method based on information entropy to determine the number of neurons in hidden layer is proposed. This proposed method is capable of computing the accuracy that complies with the requirements, reducing the network computing cost, and improving the learning and convergence efficiency of the network.

2. Principle Analysis of the DBN

The deep belief network consists of RBMs and a BP network stacked together. The training process can be divided into supervised fine-tuning and unsupervised pre-training. Firstly, through the CD fast algorithm, each layer of RBM is trained unsupervised, and the weight and bias of RBM are initialized. Then, the trained RBM network is connected to the BP network to form a DBN, and the DBN output error is propagated from one layer to another layer, the weight and bias of the entire network are fine-tuned depending on the error function. Its structure is shown in Fig. 1.

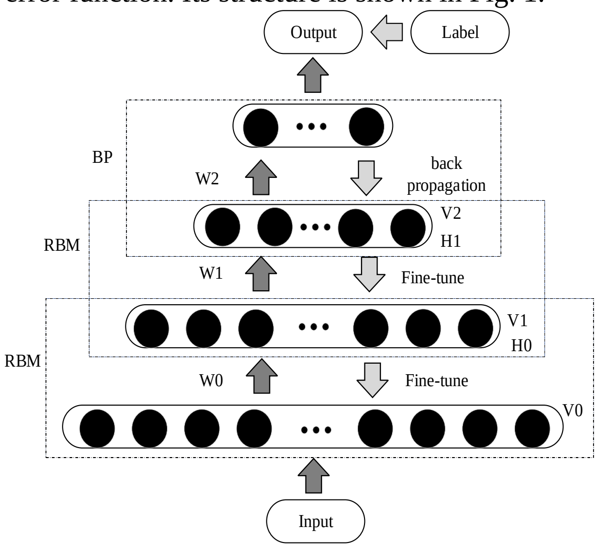

Figure 1. DBN network structure.

2.1. Pre-training

Specifically, we first use training set to train the first level RBM of DBN network. Through the machine learning, the weight and bias of the first layer RBM can be quickly initialized. After the first level of RBM training is completed, its output is input to the second level of RBM, and the second level of RBM training is performed. Then the training of the RBM is similar to this process, and layer by layer is superimposed to obtain a n-layer RBM network and its parameters.

The activation probability of the jth hidden unit is defined as:

$$p(h_j = 1) = \frac{1}{1 + \exp(-b_j - \sum_i v_i w_{ij})} \tag{1}$$

Since the RBM is a symmetric network, the activation probability of ith visible unit according to the reconstructing of the hidden layer is:

$$p(v_i = 1) = \frac{1}{1 + \exp(-c_j - \sum_j h_j w_{ji})} \tag{2}$$

v_i and h_j respectively represent the ith and jth node values of the visible layer and the hidden layer, b and c are the bias values of the two layers, W_{ij} can be viewed as the connection weights of the visible unit i and the hidden unit j . Through greedy unsupervised training layer by layer, the feature representation of the samples is transformed into a new feature space, and higher-level representations are expected to unearth more abstract features, making classification or prediction easier.

2.2. Fine-tuning
The unsupervised pre-training process completes the initialization of the network parameters and sends the output of the top-level RBM as input to the BP network. Then, according to the error function, the backward propagation algorithm is used to fine tune the ownership value of DBN.

3. DBN network structure optimization
Based on the calculation of the reconstruction error and the decision of the network's depth, this paper proposed a method to determine the number of hidden layer neurons based on information entropy. In specific applications, appropriate network depth and hidden layer neurons are deployed to optimize the whole DBN network structure, improving the accuracy, reduce the computing costs and improve the prediction efficiency.

3.1. Selection of hidden layer neurons based on the information entropy
To compute the number of hidden layer neurons, the concept of information entropy was introduced, which solved the problem of quantitative measurement of information. Information entropy is used to measure the information quantity in information theory [9]. The amount of information is inversely proportional to the information entropy of a system. Therefore, information entropy can be viewed as a measure of the size of system information.

Suppose a system X in different states such as $x_1, x_2, \ldots x_n$, $p(x_i)$ represents the probability of $x_i\,(i=1,2\ldots,n)$, then the information entropy $H(x)$ of the system is defined as:

$$H(x) = -\sum_{i=1}^{n} p(x_i)\log(p(x_i)) \tag{3}$$

Where $0 \le p(x_i) \le 1$ and $\sum_{i=1}^{n} p(x_i) = 1$, and $0\log 0 = 0$ when $p(x_i) = 0$.

When the sample size of the original data is consistent with the amount of neurons in the input layer, even if the amount of information carried by each sample is different, it can be effectively characterized by the neurons of the input layer, and there is no loss of the information. It can be considered that the information of the input layer cannot be expressed by the information quantity of the fixed value.

To solve this problem, from the point of view that "information is defined by the measure of uncertainty (i.e, information entropy)" proposed by C. E. Shannon in information theory, another form of expression of information was proposed for information entropy. Although the amount of information represented by the input layer is variable, the information entropy that the input layer can express has a maximum value, that is:

$$\begin{aligned} H(n) &= \log_2(n) \\ &= H(1/n, 1/n, \ldots, 1/n) \\ &\ge H(p_1, p_2, \ldots, p_n) \end{aligned} \tag{4}$$

Where n represents n different states of the input data. This equation demonstrates that the entropy value reaches the maximum value when the input data is in an equal probability distribution.

It can be observed from the construction process of the network that the output of the RBM hidden layer can be represented by the input of the visible layer.

$$h_j = b_j + v_i * w_{ij} \tag{5}$$

Where v_i and h_j respectively represent the ith and jth node values of the visible layer and the hidden layer, b represents the bias of the two layers, w_{ij} represents the connection weight of the visible unit i and the hidden unit j .

To maintain consistency with the way the input layer represents the amount of information, the hidden layer also deploys the information entropy of the original data to represent the amount of information. Secondly, in order that the hidden layer can effectively represent the information or features of the original data, it is necessary to deploy an appropriate number of neurons to represent information. The number of neurons in the hidden layer depends on the proportion of the amount of information that the original data can express in the input layer. Using information entropy to express is: the proportion of the information entropy of the original data in the maximum entropy determines the ratio of neurons in the hidden layer to neurons in the input layer, that is:

$$C(V) = N * \frac{H(V)}{H(n)} \tag{6}$$

Where $C(V)$ is the number of hidden layer neurons, N is the dimension of the input layer or the number of neurons, $H(n)$ is the maximum entropy, and $H(V)$ is the information entropy of the original data.

The amount of hidden layer neurons is computed to optimize the DBN network structure, so that the entire network model can better represent the input data, effectively avoiding the occurrence of under-fitting or over-fitting, reducing network cost, and improving the efficiency and precision of the network learning.

3.2. The network depth determination method based on the reconstruction error

RBM has a special structure: the layers are connected to each other, and the neurons in the layer are not connected to each other, which makes the relationship between the hidden neurons independent. When the original data is given to the visible layer, the output of the hidden layer is obtained by the calculation of information entropy. Similarly, the reconstructed value of the visible layer can be obtained owing to the symmetry of RBM. According to the difference between the original data and the reconstructed layer, the reconstructed error can be computed.

$$RError = \frac{\sum_{i=1}^{n}\sum_{j=1}^{m}(p_{ij} - d_{ij})}{mnp_x} \tag{7}$$

Among them, n represents the number of samples, m represents pixels, p represents the value computed by the network, d is the real value, p_x is the number or range of values.

Standard of the decision:

$$M = N_{RBM} + 1, RError > \varepsilon \tag{8}$$

$$M = N_{RBM}, RError < \varepsilon \tag{9}$$

Where ε is the target reconstruction error preset value, M is the number of hidden layers.

If the reconfiguration error of the network is less than our expected value after unsupervised training, the weight of the network will be fine-tuned by backward propagation algorithm immediately. Otherwise, the depth of the network will be added by one, and the training process will continue until the reconfiguration error reaches the criterion [13]. The whole DBN network training process is shown in Fig. 2.

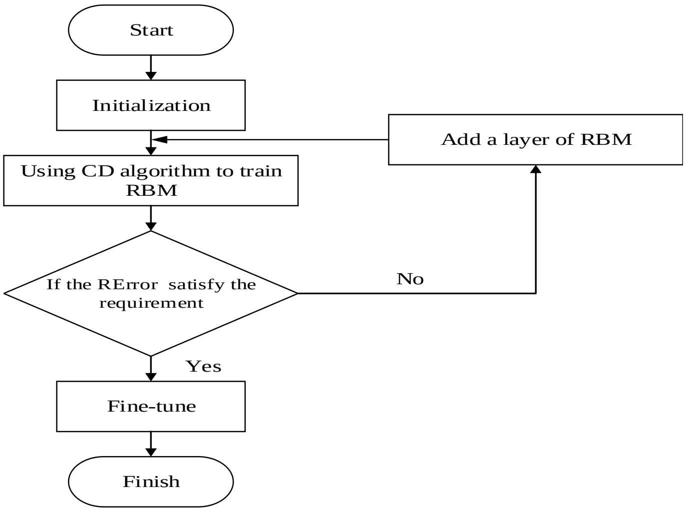

Figure 2. compute DBN depth using RBM reconstruction error.

4. Experiment and analysis

In contrast to the experimental results in [8], the experiment used a handwritten digital database created by Corinna Cortes and Yann LeCun.

The experiment took 60,000 samples for pre-training, 60000 samples for weight fine-tuning, and 10,000 samples for final testing. The gray value of each image has a value range of $[0,255]$, but the actual value of the gray value of the handwritten digital image is not fully covered. After calculation, the average gray value of each image has only 64 values of practical significance, and the rest value is set to zero. To conclude, the image gray value has 64 kinds of probability state distributions, and the actual maximum information entropy of the image can be obtained as 6 by introducing equation (4), and according to the formula (3), the information entropy of the original data can be obtained as 1.945. Using the formula, we can easily find that the number of hidden layer neurons is 255.

The experiment was carried out under the condition that the depth of the network was 2 layers of RBM and the number of hidden layer neurons was different. We got the following results:

Table 1. DBN test data

Network depth	Number of hidden layer nodes	Error	Operation time (s)
2	100*100	6.70%	4.893
2	200*200	5.63%	8.6797
2	255*255	5.00%	11.3974
2	300*300	4.90%	12.7813
2	400*400	5.36%	19.227
2	500*500	5.38%	24.3927

From table 1, we can clearly observe that when the number of hidden layers is small, the network can not express the characteristics of the input data well, and the number of neurons is much higher. It is obvious to express the phenomenon of over-fitting, and the training time is much longer. When the number of hidden layer neurons are calculated by introducing information entropy, the error is small, and the whole network performs better with a low operation cost.

Based on these results, we used the proposed method of reconstruction error to optimize the network depth which obtained a better performance of the whole network. Information will inevitably lose during the transfer process. Therefore, the preset value of the reconstruction error should not be set too small, nor should it be set too large. If it is too small, the reconstruction error will never reach the target, and the network scale will continue to expand. In contrast, it will not meet the requirements for accuracy. Considering the above reason, we set the default value of the reconstruction error to 7%, after training and testing the network, the results are as follows:

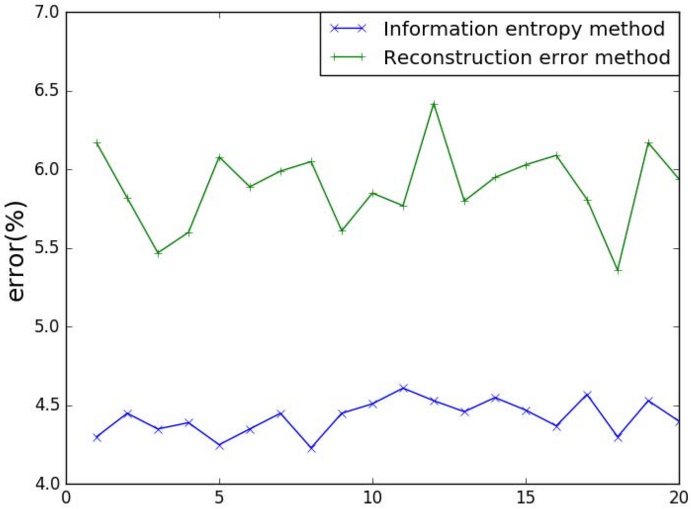

Figure 3. The test data.

In Fig. 3, the blue curve represents a DBN network with a hidden layer neuron number of 100 and a reconstruction error preset value of 7% according to the reconstruction error method, the green curve represents a neural network optimized with information entropy. Under the same conditions, 20 experiments were performed separately to obtain the above data. It can be clearly observed that the high-level neural network can better characterize the information or characteristics of the underlying neural network because of the appropriate network depth and the number of hidden layer neurons. The average error of the test data after network structure optimization is 4.42%, the average error is reduced by 1.47% compared to the test data for determining the network depth based on the reconstruction error method, and the performance of the entire network is significantly improved.

5. Conclusion

To solve the problem that is difficult to extract the appropriate number of layers and hidden layer nodes of DBN network at present, and based on the determination of network depth according to reconstruction error, this paper initialized from the expression of information quantity, then we investigated the relationship among the information entropy, the input layer and the hidden layer, so that we can obtain the number of hidden layer neurons through information entropy, and effectively characterizes the properties of the original information. The handwritten digital image experiment proves that the optimized DBN network is capable of selecting the appropriate number of the network layers and the hidden layer nodes according to the specific application, and then improves the recognition efficiency based on ensuring the reconstruction error.

References

[1] G.Q. Zhang, Forecasting with artificial neural networks: The state of the art, J. International Journal of Forecasting (1998) 14(1): 35 - 62.
[2] Q. Lv, Y. Dou, X. Niu, Remote sensing image classification based on DBN model, J. Computer Research and Development (2014) 51(9): 1911-1918.
[3] G.E. Hinton, S. Osindero, T. Yee-Whye, A fast learning algorithm for deep belief nets, J. Neural computation (2006) 18 (7): 1527 - 1554.
[4] H. Lee, R. Grosse, R. Ranganath, et al. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations, C. Proceedings of the 26th annual international conference on machine learning. ACM, 2009, 609-616.

[5] A. Mohamed, G. Dahl, G.F. Hinton, Deep belief networks for phone recognition, C. Nips workshop on deep learning for speech recognition and related applications, 2009, 1(9): 39.
[6] L. Deng, D. Yu, G.E. Dahl, Deep belief network for large vocabulary continuous speech recognition: U.S. Patent 8,972,253. (2015)
[7] N. Jie, X.Z. B, L. Zhong, et al. An Improved Bilinear Deep Belief Network Algorithm for Image Classification, C. Computational Intelligence and Security (CIS), 2014 Tenth International Conference on, IEEE, 2014, 189-192.
[8] J. Kim, J. Nam, I. Gurevych, Learning Semantics with Deep Belief Network for Cross-Language Information Retrieval, C. COLING (Posters), 2012, 579-588.
[9] P. Martin, E.G. David, G.L. Fernando, A Survey of Optimization by Building and Using Probabilistic Models, J. Computational Optimization and Applications (2002) 21(1): 5-20.
[10] M. Gordon, Probabilistic and genetic algorithms in document retrieval, J. Communications of the Acm (1988) 31(10): 1208-1218.
[11] M. Opper, Generalization performance of Bayes optimal classification algorithm for learning a perceptron, J. Physical Review Letters (1991) 66(20): 2677-2680.
[12] Q.K. Xiao, S. Gao, X.G. Gao, Design of DBN structure optimization algorithm based on bayesian optimization, J. Systems Engineering and Electronics (2007) 29(10): 1732-1737.
[13] G.Y. Pan, W. Chai, J.F. Qiao, Depth determination method of DBN network, J. Control and Decision (2015) 30 (2): 256-260.
[14] R.M. Gray, Entropy and information theory, M. Springer Science & Business Media, 2011.