International Journal of Engineering Sciences & Research Technology

(A Peer Reviewed Online Journal)
Impact Factor: 5.164

IJESRT

RESEARCHERID
THOMSON REUTERS
[Pan *et al.*, 9(7): July, 2020]
IC™ Value: 3.00
ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

IJESRT
INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

STRUCTURE OPTIMIZATION OF DEEP BELIEF NETS IN THE APPLICATIONS OF IMAGE RECOGNITION

Qili CHEN[1,2],Guangyuan PAN[3,*4] & Ming Yu[3]
[1]Faculty of Information Technology, Beijing University of Technology, Beijing, 100124, China
[2]Department of Automation, Beijing Information Science and Technology University, Beijing, 100192, China
[3]Department of Electronic Engineering, The Chinese University of Hong Kong, Hong Kong SAR;
[4]Shenzhen Garry Intelligent Technology co. ltd, China

DOI: https://doi.org/10.29121/ijesrt.v9.i7.2020.17

ABSTRACT
Deep belief network (DBN) has become one of the most important models in deep learning, however, the un-optimized structure leads to wasting too much training resources. To solve this problem and to investigate the connection of depth and accuracy of DBN, an optimization training method that consists of two steps is proposed. Firstly, by using mathematical and biological tools, the significance of supervised training is analyzed, and a theorem, that is on reconstruction error and network energy, is proved. Secondly, based on conclusions of step one, this paper proposes to optimize the structure of DBN (especially hidden layer numbers). Thirdly, this method is applied in two image recognition experiments, and results show increased computing efficiency and accuracies in both tasks.

KEYWORDS: Deep learning, Structure analysis, Unsupervised learning, Image recognition.

1. INTRODUCTION
Deep belief network (DBN) [1] is famous for simulating human brains, which helps to increase training efficiency greatly, by solving curse of dimensionality problem. By now, DBN has been applied in multi signal processing applications, such as but not limited to voice, image, video, text and semantic transmission [2-8].

However, DBN still has many issues to be studied further. At present, due to short of efficient training algorithm in parallel, in its applications, the empirical approach is still used to determine the hidden layers and neurons, resulting in big errors, and preventing its extended network application, hence of high computing cost and low efficiency. Professor Bengio of the University of Montreal argues in his literature [9]: is it possible to define a proper network depth enabling the DBN to solve almost all AI issues similar to that of human beings? This is an open issue, and its research will of great significance in DBN's application in the field of AI. However, given its wide scope involved, it is hard to present a standard answer. As such, in other words, it is to say not to define the network depth manually, but to set up a methodology, enabling the network itself to compute the most appropriate depth. So, the results would vary upon different questions and demands. This is what this work focuses on.

This paper is divided in the following sections. Section 2 analyzes training processes with and without supervision learning in DBN, finding out the connection between network depths and training errors. Based on that, a depth defining method is given based on RBM reconstruction errors. in section 3. In this way, the network self-organizes trainings in computing, suggesting the depth meeting requirements, with satisfactory accuracy and reduced costs. Section 4 presents experimental results while Section 5 concludes the paper.

2. FEATURE ANALYSİS AND BİOLOGİCAL SİGNİFİCANCES
The relationship between initial state and training result is still unknown to researchers. Several studies have found the conclusion that initializing the starting distribution of samples or weights with a same special method will result in different minimum in the stage of fine tuning [10-12]. While in this work, we try to find out the reason of why random initializing net's weights leads to bad minimums, using biological and mathematical tools.

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

By machine learning and previous study of biology[1,13-14], the following facts have been known: (1, Unsupervised learning plays an important role in biological cognition; (2, Since there are very few samples with marking information, containing of limited information, unsupervised learning contributes to increasing prior knowledge, bringing network weights at a preferable initial position, leading to enhanced network performance; (3, In biological system, stable neural networks once are established, and then it would be difficult to change.

The three points are easily comprehensible, for human cognition are unsupervised and supervised learning in parallel. At the initial stage of cognition, loop connection in the human brain is gradually developed, which does not make sense physically for those being seen and heard. At the moment, the brain greedily records them all. Whereas while growing up, the brain guided by marking information (such as books and teachers) may classify and sort out those previously received signals, and form new loop connections of neural networks by brainstorming and memorizing. Such is human cognitive process, learning and growing continuously.

So if without unsupervised learning, by relying on randomly initial supervised learning, it would lead to failure of network trainings. Two assumptions are given in the article as below:

Assumption 1, a Gradient Descent Approach (GDA) of random initialization easily falls in local minimum;
Assumption 2, it is tough to choose proper methods and steps of batch processing to jump out of a local minimum.

The assumption 1 is a theoretical reason. Mathematical methods can be used to analyze training errors in various hidden layers in DBN in a GDA.

In a top-to-down transmission, the errors at the layer 1 (or the top layer) are:

$$e_j = d_j - y_j \qquad (1)$$

where e_j is error of neuron j, d_j is ideal output and y_j is output. By DBN computing rules, we have $y_j^l = \sum w_{ji} y_i^{l-1}$, where w_{ji} is the weight between j and i. And by GDA rules, we have $\delta_j = e_j y_j[1 - y_j]$ in output layer and $\delta_j^l = y_j^l[1 - y_j^l] \sum \delta_k^{l+1} w_{kj}^{l+1}$ in other layers, where δ_j^l presents for the output of unit j of hidden layer l in GDA. So we have

$$e^l = d - \delta^{l+1} w^{l+1} \qquad (2)$$

Then use the transfer formulas above, e^l can be rewritten as

$$e^l = d - y^{l+1}[1 - y^{l+1}]\delta^{l+2} w^{l+2} w^{l+1} \qquad (3)$$

So,

$$e^l = d - e^l(\prod_{l+1}^{L} y[1 - y]w) \qquad (4)$$

Because

$$y_i \in [0,1] \qquad (5)$$

So for every y_i in hidden layer l and $l - 1$, we also have

$$y^{l-1} = y^l w \in [0,1] \qquad (6)$$

Then use the limiting condition in (4), we get the result below,

$$\begin{aligned} e^l &= d - e^L \left(\prod_{l+1}^{L} y[1 - y]w \right) \\ &= d - e^L \left(\prod_{l+1}^{L} yw \right) \prod_{l+1}^{L} (1 - y) \\ &> d - e^L \left(\prod_{l+2}^{L} y[1 - y]w \right) \end{aligned}$$

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

where
$d - e^L(\prod_{l+2}^{L} y[1-y]w) = e^{l+1}$ (7)
so
$e^l > e^{l+1}$ (8)

It can be seen that in DBN, a reversely adjusted GDA will lead to gradual enlargement of training errors by layers. If adopting random initialized approaches, network weights will be distributed in state space evenly. As initial errors are already big, it would be expanded as the trainings go on by layers, with the trainings ended in failure.

Assumption 2 is a technical reason. As the GDA features of gradient diffusion [15], that is, when calculating derivatives in a reverse transmission approach, threshold value of the reverse transmission gradient will be sharply reduced, with growing depth of networks. So it results in very limited derivatives on loss functions compared with the initial layers' weights. Thus, when using the GDA, the initial layers see a rather slow change in weights, so that it is incapable to learn effectively from samples.

2.1 Experimental Verification
The article designs handwritten digital noise-reducing experiments, in an attempt to verify the two assumptions. The experiments go as follows:

In the MNIST [16] database, we select 5000 samples for unsupervised learning, and 1000 for testing. Add 10% of background noise in the tested images, with each image of pixels of 28*28. Then 1000 samples are grouped into 10 batches, with 100 samples each. As temporary no relevant research is available to define numbers of neurons at each layer, and each batch consists of 100 images, so assuming 100 selected neurons at each layer, in 3 hidden layers. The experimental effects are shown in Fig 1.

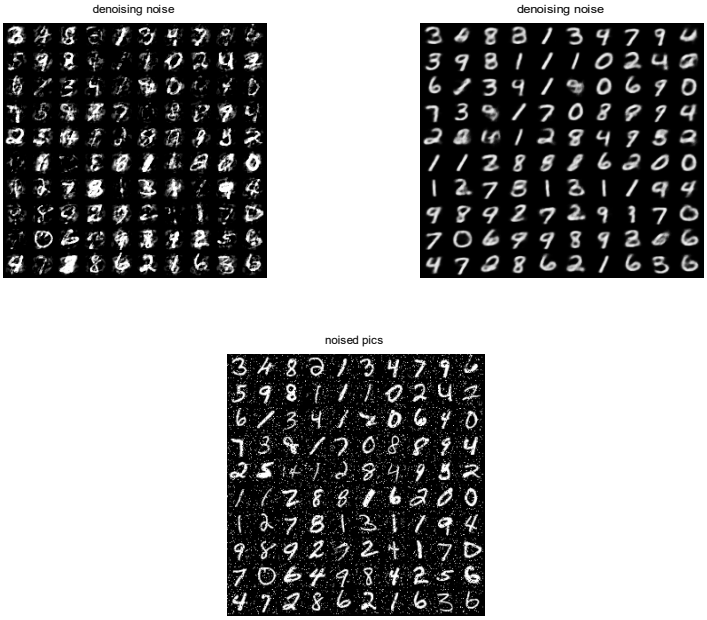


Fig.1 Noised pics

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

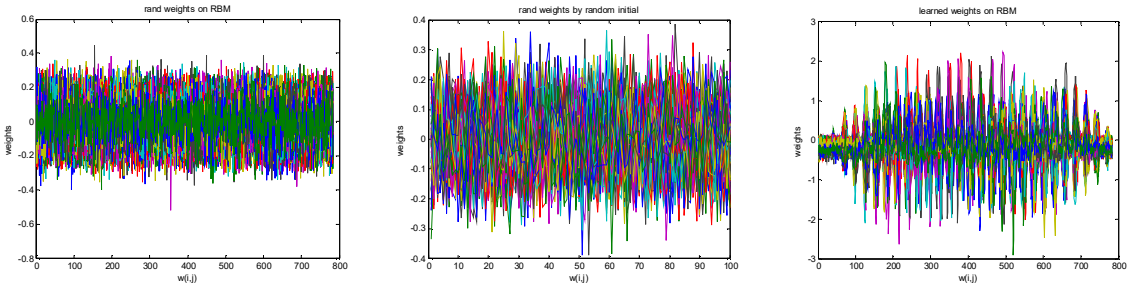


Fig.2 Learned weights in DBN

The left picture of Fig 1 is the original image with 10% noise added; the center one is noise reducing effect with random initial descent while the right is the initial noise reducing effect with unsupervised learning on gradient descent. Apparently, the randomly initial GDA does not work well for noise reduction, but blurring the image instead, while adopting unsupervised learning; initialized gradient descent can effectively reduce the background noise. Fig 2 is the network weight. The left image is its randomly initial state, which is assumed of an initial value as [-0.5, 0.5], and various colors are selected to demonstrate different weights connection. The center is the network weight after descending training of random initialization, suggesting that the weight at the moment still shows a messy and irregular state, or the network has not been well trained. On the right image is the unsupervised initial GDA, with the network weight showing regular ups and downs, or some learning is stressed, and some weakened. Such weight states indicate that the network is capable of some cognitive and identification for identifying image correctly and reducing noises.

From the experiment above, it concludes:

1) In the cognitive process, unsupervised and supervised learning may be helpful for better training results. In the training, there is a shortage of marking samples, but unmarked samples contain of more information.

2) Randomly initial GDA will easily lead to the failure of network training, with reasons as:

- Insufficient prior knowledge, limited knowledge reserves of network, difficult to apply restricted information to process mass data;
- As of difficulties to determine proper training methods and algorithm steps for batch processing, it is hard to jump out when falling into the local minimum.

3) Unsupervised learning contributes to finding out more information for network, playing following roles in the training:

- Unsupervised learning is capable of bringing more features for network learning.
- Unsupervised learning works well to initialize reversely adjusted parameters of the algorithm.

4) Unsupervised learning reduces errors with growing depth in hidden layers.

3. DETERMİNATİON OF NETWORK DEPTH

Depth refers to the longest path from the input to the output layer [1, 17]. The following formula can be used for depth depiction.

$$Depth = H + 1 \qquad (9)$$

H stands for the number of hidden layers.

The experiment in Chapter 2 shows that unsupervised learning is at the kernel of network training, with errors varying from depths of the training, in which costs correspond with depths positively. So when solving practical issues, it is necessary to choose network of proper depth with satisfactory accuracy and saving costs in every possible endeavor.

At present, there is an important issue in DBN application that is, focusing on various problems, DBN has to set up network depth in advance, then to compare with accuracies and training results of different depths by the empirical approach. It greatly inhibits efficiency of problem solving of network, resulting in a more restricted control on expanding DBN further. In fact, Bengio proposed a similar question in 2009 [9], describing as, "Is there a depth that is mostly sufficient for the computations necessary to approach human-level performance of AI tasks?" As the general nature of the question, involving a broad scope of disciplines, it is difficult to perform tasks

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

by finding out a proper mathematical method, much less to design a representative experiment composed of many characteristics for verification. So the question cannot be answered at present.

Even though, the experiment and analysis of Chapter 2 show that DBN features of accuracy by layers under data training, by which, Bengio's question can be translated as a self-training issue for network. In other words, without deploying any tasks to DBN, focusing on a specific question, by setting up targets, enable the network to judge by itself if with a proper depth. Thus, the original question is translated into a self-organized training issue on network depth, with a roughly common result of the two. In addition, by performing self-defining depth, it acts as an important reference to the original question.

At the present DBN studies, network depth and unit numbers of each hidden layer are selected empirically [18-20], unfavorable for using network advantages, easily leading to high computing cost, negative to increase efficiency. Hinton pointed out two principles in the literature [14], which will act as the lemmas of the coming work, described as below:

Lemma 1: training accuracy of RBM will be enhanced with growing depths.

Lemma 2: in DBN network trainings, by unsupervised learning, the network weights are already properly positioned, while the GDA based reversely computing is capable of adjusting weights in some small aspects.

This article suggests a method of reconstruction errors, used for computing and defining DBN depth, with the two lemmas as an important basis. Also, network energy functions and accurate identification are the important theoretical basis.

Reconstruction error is with training data as its initial state, the variance after distributed by RBM and transferred by Gibbs from the original data (generally evaluated by the first-order normal forms or the second-order normal forms) with formula shown as below:

$$RError = \frac{\sum_{i=1}^{n}\sum_{j=1}^{m}(p_{i,j}-d_{i,j})}{n\times m\times px} \quad (10)$$

In the formula, n is sample number; m, pixel number; p, network computing value; d, real value; px, value number or scope.

With rules as below:

$$\begin{cases} if \quad RError > \varepsilon, \quad L = N_{RBM} + 1 \\ if \quad RError < \varepsilon, \quad L = N_{RBM} \end{cases} \quad (11)$$

In the formula, ε is for pre-set value objective reconstruction errors, and L is for the number of hidden layers. Then, if at the moment, the trained network sees satisfactory construction errors, or less than the present value, then start reversely gradient adjustment. Otherwise, let the network depth adding one automatically, to continue training. As it can be known in advance of the value scope and true value of test samples, so in the application, the value of reconstruction errors can be obtained by computing (generally, the value enables a 95% plus accuracy).

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

Fig.3 flow chart of calculation the depth of DBN by using R-Error of RBM

Though construction errors are not reliable sometimes, for instance, some data even classified correctly may produce certain construction errors. However, reconstruction errors reflect RBM's likelihood of training data in some degree, with simple computing and small costs, resulting in a useful practice.

As RBM training is based on simulated annealing algorithm, DBN features of characteristics on simulated annealing. So seek for coupling relation between reconstruction errors and network energy, which can be substantiated theoretically the validity of the method. Expectation of RBM eigenvector can be used to describe network energy, in formula as:

$$E(v,h) = -h^T W v - b^T v - c^T h \quad (12)$$

The article suggests the theorem: *Reconstruction errors positively correlate to network energy.* It can be done by the whole and conditional probability formula, with the proofs as below:

Firstly, we define P is the computed result, and D is the ideal output. Then,

$$P = P(v), D = P(v_0) \quad (13)$$

We use conditional probability formula in (13), and get

$$P = P(v) = P(v_0)P(h|v_0)P(v|h) \quad (14)$$

According to the formula that

$$P(v|h) = \frac{P(v,h)}{P(h)} \quad (15)$$

We will get,

$$P = P(v_0) \cdot \frac{P(v_o,h)}{P(v_o)} \cdot \frac{P(v,h)}{P(h)} \quad (16)$$

Eliminate $P(v_o)$, so,

$$P = P(v_o,h) \cdot \frac{P(v,h)}{P(h)} \quad (17)$$

By using conditional probability formula again, the below result will be realized,

$$P = P(v_0|h) \cdot P(h) \cdot \frac{P(v,h)}{P(h)} \quad (18)$$

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

Eliminate $P(v)$, so,
$P = P(v_0|h) \cdot P(v,h)$ (19)

Then, according to the definition in (20), RE should be
$RE = \frac{\sum_{i=1}^{n}\sum_{j=1}^{m}(p_{i,j}-d_{i,j})}{n \times m \times px} = P - D$ (20)

Use (13) and (19) here,
$RE = P(v_0|h) \cdot P(v,h) - P(v_0)$ (21)

Then,
$RE = P(v_0) \cdot (P(v,h) - 1)$ (22)

Use (13) and the objective function again, then there will be,
$RE \propto D \cdot (E(v,h) - 1) \propto E(v,h)$ (23)

Reconstruction errors positively correlate to network energy, QED.

According to the theorem, the method of reconstruction errors ensures a simple computing and an easy practice. Also, due to coupling network energy, a reasonable judgment methodology is given from the network system, bringing persuasive computing results.

Network training flowchart is shown as fig 3

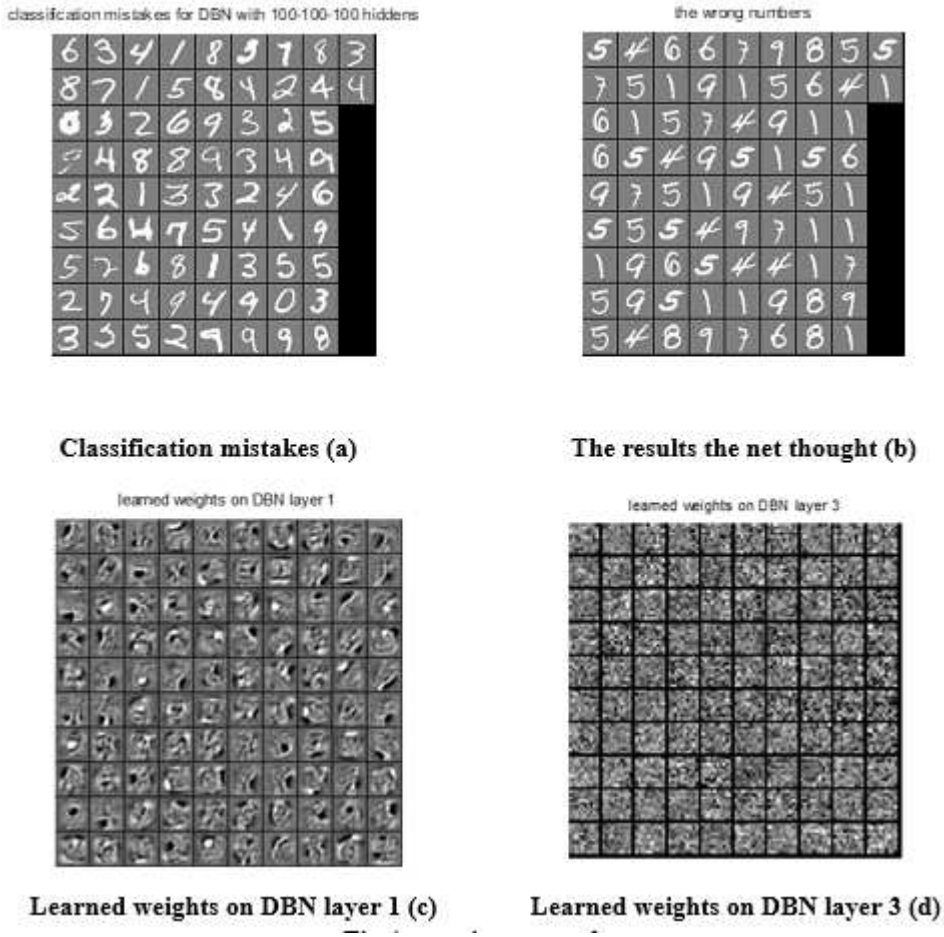


Fig.4 experiment results

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

4. EXPERİMENTS AND ANALYSİS

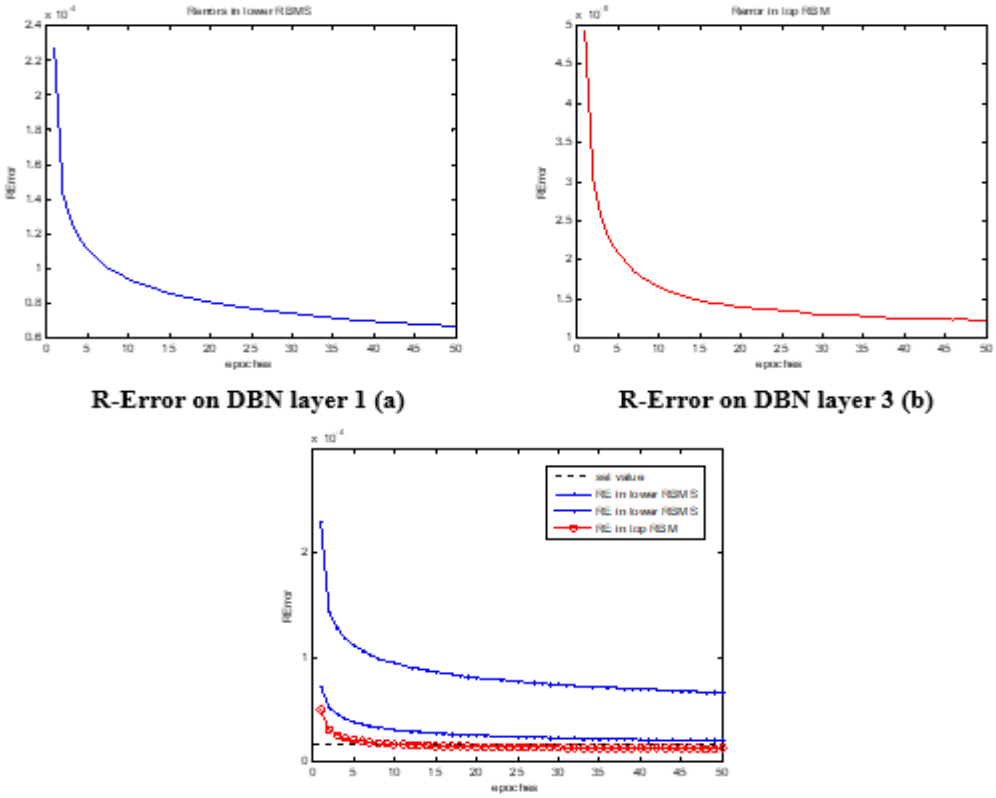
A digital recognition experiment is designed in the article for testing effectiveness of the method, which uses MNIST handwritten database. The database boasts of 60,000 training images and 10,000 testing images, all in handwritten, and each number is displayed in many handwritten ways, with quite a few modes of recognition technology applied in the database, so the database is known as an ideal evaluation of new methods. For basic version of MNIST learning tasks, as no geometric knowledge available, nor any special pre-processing or training group enhancement, random but fixed arrangement of pixel will not affect the learning algorithm. Taking 5000 samples for training, and another 1000 samples for testing, the database composes of samples of Arabic numbers (0-9), all in handwritten. The 5000 samples are grouped into 50 batches, 100 samples each, so assuming 100 neurons on each layer, with accuracy of 99% above for reconstruction error conditions. So the result is RError =1.59e-005.

The original matlab code is provided by Andrej Karpathy [21] and we improve and use it in the calculation of depth. The network ceases to increase when its hidden layers reach at 3 (or the depth of 4). Now by testing the 1,000 samples, 74 errors are produced, with the original images and the errors as shown in Fig 4. By analyzing numbers in images, it can be achieved that the network is easy to judge what kinds of images by statistics, and what kinds of features are drawn to produce errors, which contribute to improving network performance for reference.

Weight image of the network are shown as Fig 4(c) (d). Fig 4(c) is RBM weight at the bottom layer (or the first layer), Fig 4(d) as the top layer (or the last layer). The two figures show that visualized weights of DBN training, suggesting that with growing depth, network weights are more abstract, indicating that network cognitive information is a combination of such abstract data (in fact, the combination features of sparseness [22]).

Fig 5 is the curve chart of network reconstruction errors. Fig 5(a) is RBM reconstruction errors of the first RBM; Fig 5(b) is those at the last layer, suggesting that the errors at each layer tend to be declining. When combining the 3 reconstruction errors into one figure, as shown in Fig 5(c).

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7



R-Error on DBN layer 1 (a)

R-Error on DBN layer 3 (b)

R-Errors on all DBN layers (c)

Fig.5 RE curves

The red curve in the figure, or the RBM reconstruction errors at the last layer, has reached the preset target. Why is each initial value of RBM reconstruction errors higher than the previous final value? When adding each layer of RBM, its initial weights are randomly selected, so the initial reconstruction errors are relatively high. And more efforts should be done to initialize coming RBM, which will contribute to increasing network performance.

For facilitating comparison and data analysis, the network depth is increased to 6, with DBN training data computed, as shown in Table 1.

The table suggests that with growing depth, network reconstruction errors are gradually lowered, and computing time increased, which is in line with network features. The error amount (or accuracy) maximizes at the depth of 4, reaching 92.6%, while if increasing network depth further, at the moment, though reconstruction errors are still declining, the accuracy are reduced, minimized as 88.8% at the depth of 6.

Tab.1 training data for DBN with different depths

DEPTH	RE	MISTAKES	ACCURACY	TIME
2	6.6181e-5	88	91.2%	22.9s
3	2.0742e-5	77	92.3%	26.9s
4	**1.2326e-5**	**74**	**92.6%**	**32.9s**
5	0.7785e-5	89	91.1%	38.6s
6	0.5344e-5	112	88.8%	44.6s

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

Why is the accuracy reduced with growing depth? For this phenomenon, the analysis is given in the article as:

(1 Only at the last layer, RBM reconstruction errors can satisfy the requirements, while those at the previous layer lower the accuracy by accumulated errors;

(2 Increased hidden layers lead to overly accumulated errors in a reversely GDA;

(3 Adding hidden layers results in complicated computing time and lower efficiency.

Therefore, how to set up reconstruction errors reasonably, or seek for connection between networks computing cost and depth, it will contribute to a more intelligent approach of self-learning network.

5. CONCLUSİONS

In this paper, an improved mechanism for optimizing deep belief network's structure is introduced. Furthermore, it suggests that by varying tasks, the network is capable of self-organized self-trainings and defining network depth. It also provides a reconstruction errors based judging method, to define network depth, for self-organized training network in the training process, to perform tasks of hidden layer selection on model cognition for in-depth learning network, resulting in enhanced computing efficiency and reduced computing cost. In addition, as stated in Part 4, the article is possible to be extended further. And in the article, the network depth acts as the only self-organized variable, with fixed units of each RBM hidden layer, neglecting biological and mathematical connections by different hidden layers, which are worthy of more efforts to study.

6. ACKNOWLEDGMENTS

This work is supported by Beijing Postdoctoral Science Foundation under Grant ZZ-2019-65, Chaoyang District Postdoctoral Science Foundation under Grant 2019ZZ-45 and Beijing Municipal Education Commission under grant KM201811232016.

REFERENCES

[1] Hinton, G. E . Reducing the Dimensionality of Data with Neural Networks. Science, 2006, 313(5786): 504-507.
[2] Lecun Y, Bengio Y, Hinton G. Deep learning. Nature, 2015, 521(7553): 436.
[3] Chen.D, Lv.J, and Yi.Z. Graph regularized restricted boltzmann machine. IEEE Transactions on Neural Networks and Learning Systems, 2018, 29(6): 2651-2659.
[4] Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z. Rethinking the inception architecture for computer vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016: 2818-2826.
[5] He K, Zhang X, Ren S, et al. Identity Mappings in Deep Residual Networks. European Conference on Computer Vision. Springer, Cham, 2016, 630-645.
[6] Jia.L.I, David.H.E, and Yong.Q.U. Diagnosis of gear early pitting faults using PSO optimized deep neural network. Journal of Northeastern University (Natural Science), 2019, 40(7): 974-979
[7] Yang. H, Hu.B, Pan.X, Yan.S, Feng.Y and Zhang.X, Deep belief network-based drug identification using near infrared spectroscopy. Journal of Innovative Optical Health Sciences, 2017, 10(2): 545-551.
[8] Soon.F.C, Khaw.H.Y, Chuah.J.H, and Kanesan,J, Hyper-parameters optimisation of deep CNN architecture for vehicle logo recognition, IET Intelligent Transport Systems, 2018, 12(8): 939-946.
[9] Bengio Y. Learning Deep Architectures for AI. Foundations & Trends in Machine Learning, 2009, 2(1):1-127
[10] Erhan D, Bengio Y, Courville A. Why does unsupervised pre-training help deep learning. Journal of Machine Learning Research, 2010, 11: 625-660
[11] Chen Q, Pan G, Qiao J, Yu M. Research on a Continuous Deep Belief Network for Feature Learning of Time Series Prediction. The Chinese Control and Decision Conference, pp. 5977-5983, 2019.
[12] M. F. Hohman, H. Park, C. Robinson, and D. H. Chau. Summit: Scaling deep learning interpretability by visualizing activation and attribution summarizations. IEEE Transactions on Visualization and Computer Graphics (TVCG). Vancouver, Canada, 2020.
[13] Altineay H, Demirelder M. Undesirable Effects of Out-put Normalization in Multiple Classifier Systems. Pattern Recognition Letters, 2003, 24: 1163-1170
[14] Bengio Y, Lamblin P, Popovici D, Larochelle H. Greedy Layer-Wise Training of Deep Networks, Neural Information Processing Systems, MIT Press, 2007: 153-160

ISSN: 2277-9655
Impact Factor: 5.164
CODEN: IJESS7

[15] Hu Y, Yu Y. Learning Restricted Boltzmann Machines using Mode-Hopping MCMC. The 4th International Conference on Machine Learning and Computing, 2012: 105-110
[16] LeCun Y, Corinna Cortes, Christopher J C Burges. THE MNIST DATABASE of handwritten digits. Available: http://yann.lecun.com/exdb/mnist/
[17] Hinton G E, Osindero S, The Y. A fast learning algorithm for deep belief nets. Neural Computation, 2006, 18:1527-1554
[18] Qiao J, Pan G, and Han H. A Regularization-Reinforced DBN for Digital Recognition. Natural Computing, 2019, 18(4): 721-733.
[19] Zhu L, Laptev N. Deep and confident prediction for time series at uber. IEEE International Conference on Data Mining Workshops (ICDMW), 2017, 103-110.
[20] Pan G, Fu L, Thakali L. Development of a global road safety performance function using deep neural networks, International Journal of Transportation Science & Technology, 2017, 6(3), 159-173.
[21] Hinton. Deep learning. Available [2020]: http://deeplearning.net/software_links/
[22] Ranzato M, Poultney C, Chopra S, LeCun Y. Efficient Learning of Sparse Representations with an Energy-Based Model. Neural Information Processing Systems, MIT Press, 2007: 1137-1144.