IMAGE PROCESSING RESEARCH BASED ON DEEP NEURAL NETWORK

LeiBin Wu[1], JuYang Lei[2]

[1,2]*College of Mechanical Engineering, Shanghai University of Engineering Science, Shanghai, China*

Abstract

Machine Learning is always the focus in research of pattern recognition in the field of artificial intelligence . Deep Learning the latest rising algorithm is a new type of machine learning methods in recent years .Nowadays a lot of machine learning methods are based on the shallow network structure , it's hard to express and generalize some of the complex nonlinear functions by this shallow network structure . Deep Learning can realize the approximation of the complex functions by one of the deep nonlinear neural network . The paper will mainly introduce two depth network model and combine them with each other in order to improve the quality of the network .Last we will verify the feasibility though the experiment.

Keywords: *Deep Learning, Image Processing, Machine Learning, Neural Network*

----------***----------

1. INTRODUCTION

Image is one of the most important source of information . With the development of the scinece and technology , the application of image has attracted more and more attention of the humans and image processing has become the focus research for the several years . Due to the rich information contained in the image , how to realize the extraction and recognition of the effective information contained in the image is always a hot area for the researchers . There are a number of related research in the field of image such as the method of image processing with the neural network , by using multilayer feed forward network with BP algorithm which can approximate any nonlinear mapping relation .

Although the research of image processing mentioned above has made some achievements but there are still some problems , for instance , the training for the multilayer network is not very ideal . In addition , it will be trapped in the plight of local minimum because of the nonlinear mapping between the input and output of the network that make the error function is a nonlinear space containing multiple minimum point while using the training algorithm , randomly select the initial value of multilayer network .

To solve the problems above , we will adopt deep learning for image processing . Deep Learning is a new emerging learning algorithm of the neural network , it detects the distributed features of the data mainly through learning low-level features to form abstract executive high-level expression (attribute category or features) . This paper will combine with improved depth model and validate the feasibility by experiments

2. DEEP LEARNING MODELS AND THE RELATED IMPROVEMENT

2.1 Sparse Auto Encode

Auto Encoder (AE) can be used for dimensionality reduction of high-dimensional data in order to get a low dimensional feature vector data .It's model generally contains an input layer , one hidden layer and an output layer . If we set the input vector of samples to x , we can get the active situation of the hidden layer and the output layer though the following formulas.

$$y = f(wx + b) \quad (1)$$

$$z = f(w^T y + b') \quad (2)$$

In the math above f is an S function: $f = 1/(1 + e^{-x})$ The purpose of the AE is to make the output is equal to the input as far as possible , it means that the output z is same as the input x . The loss function of the AE.

$$l(x, z) = \sum_{i=1}^{n} KL(x_i \parallel z_i) \quad (3)$$

In the math above , KL means the divergence of input vector X_i and output vector z_i which can be used to measure the difference between x and z . The hidden layer y can learn low dimensional representation of samples because of the dimension of the layer which is smaller than the input x .While training the models , stochastic gradient descent will be always used for weight training , we usually add constraint conditions and limit the number of the activated neurons in hidden layer , only a small number of neurons can be activated . This is why we call it Sparse Auto Encoder (SAE) and the structure is shown as Fig.1.

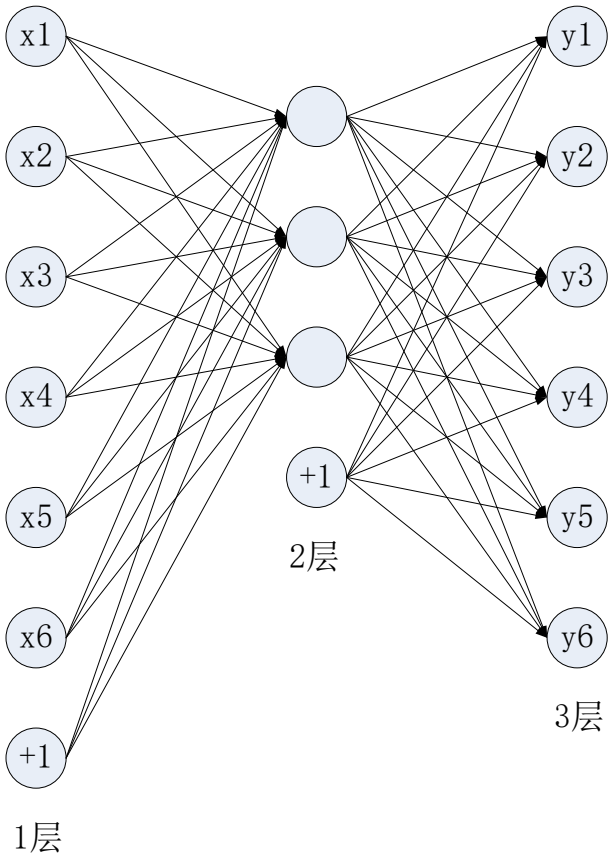

Fig.1: the structure of SAE

The loss function of the SAE is :

$$J_{sparse}(W,b) = J(w,b) + \beta\sum_{j=1}^{s_2} KL(\rho \parallel \bar{\rho}_j) \quad (4)$$

In the math, $J(w,b)$ is the loss function of the network without the constraint of sparse coding and the representation of KL is as follows:

$$KL\left(\rho \parallel \bar{\rho}_j\right) = \rho\log\frac{\rho}{\rho_j} + (1-\rho)\log\frac{1-\rho}{1-\rho_j} \quad (5)$$

The expression of the nodes in hidden layer is:

$$\hat{\rho_j} = \frac{1}{m}\sum_{i=1}^{m}\left[a_j^{(2)}\left(x^{(i)}\right)\right] \quad (6)$$

After calculating the loss function , we can get the partial derivative of the loss function . The results are as follows:

$$\frac{\partial}{\partial W_i^{(l)}}J(W,b;x,y) = \alpha_j^{(l)}\delta_i^{(l+1)} \quad (7)$$

$$\frac{\partial}{\partial b_i^{(l)}}J(W,b;x,y) = \delta_i^{(l+1)} \quad (8)$$

Then we will calculate the parameters by the method of gradient descent and the general process of the algorithm is as follows :

(1)Giving the training samples and make the settings as follows:

$$\Delta W^{(l)} = 0; \Delta b^{(l)} = 0$$

(2)Though the comparison between the input and the output $\Delta W^{(l)} = \Delta W^{(l)} + \nabla_{W(l)}J(w,b;x,y)$ and the BP algorithm , we can work out that:

$$\nabla_{W(l)}J(w,b;x,y)和\Delta b^{(l)} = \Delta b^{(l)} + \nabla_{b(l)}J(w,b;x,y)$$

(3)Updating the parameters:

$$W^{(l)} = W(l) - a[\left(\frac{1}{m}\Delta W^{(l)}\right) + \lambda W^{(l)}]$$

$$b^{(l)} = b^{(l)} - a[\frac{1}{m}\Delta b^{(l)}]$$

In conclusion , there is no relationship between the loss function , which is calculated by the superposition of the loss of every samples , and the precedence relationship of the input .

2.2 Restricted Boltzmann Machine

Restricted Boltzmann Machine (RBM) has only a visible layer and a hidden layer and there is no connection of the structure between the two layers . It can be showed as follows:

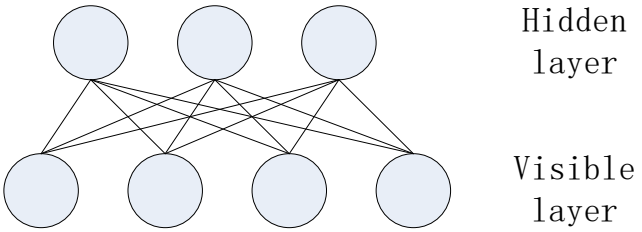

Fig.2: the structure of RBM

If there are n visible units and m hidden units in a RBM and the vector v and h respectively represent the visible and hidden units. The v_i represents the state of ith visible units and h_J represents the state of jth hidden units . For the given state (v,h) the power of the RBM can be defined as follows:

$$E(v,h|\theta) = -\sum_{i=1}^{n}a_i v_i - \sum_{j=1}^{m}b_j h_j - \sum_{i=1}^{n}\sum_{j=1}^{m}v_i W_{ij} h_j \quad (9)$$

In the math above, $\theta = \{W_{ij}, a_i, b_j\}$ is the parameter of RBM, W_{ij} is the weight between the visible layer and the hidden layer, a_i and b_j are the bias of visible layer and hidden layer . Then we can get the joint probability distribution based on the state (v,h):

$$P(v,h|\theta) = \frac{e^{-E(v,h|\theta)}}{Z(\theta)}, Z(\theta) = \sum_{v,h}e^{-E(v,h|\theta)} \quad (10)$$

Among the math, $Z(\theta)$ is the normalization factor.

From the structure of the RBM , we can know that the activated state of the hidden units is independent when the

state of visible units is given to us . So the probability for activating the jth hidden units is :

$$P(h_i = 1|v,\theta) = \sigma\left(b_j + \sum_i v_i W_{ij}\right) \quad (11)$$

Among them σ is the S function . In the same way , if given the state of hidden units , the probability for activating the visible layer is:

$$P(v_i = 1|h,\theta) = \sigma\left(a_j + \sum_j W_{ij} h_j\right) \quad (12)$$

The parameter θ of the RBM network can be obtained by learning from the training set through maximum likelihood function .It can be showed as follows:

$$\theta^{*} = \arg\max_{\theta} \sum_{t=1}^{T} \log P\left(v^{(t)}|\theta\right) \quad (13)$$

After solving the formula , the gradient of θ is

$$\sum_{t=1}^{T} \frac{\partial}{\partial\theta}\left(\log \sum_{h} \exp\left[-E\left(v^{(t)},h|\theta\right)\right] - \log \sum_{v}\sum_{h} \exp[-E(v,h|\theta)]\right)$$

$$= \sum_{t=1}^{T}\left(< \frac{\partial\left(-E\left(v^{(t)},h|\theta\right)\right)}{\partial\theta} >_{P\left(h|v^{(t)},\theta\right)} - < \frac{\partial\left(-E(v,h|\theta)\right)}{\partial\theta} >_{P(v,h|\theta)}\right)$$

Because of the existence of normalization factor Z(θ), the joint probability distribution on the right side of the equation is hard to be worked out . To solving the problem , we take some sampling methods like Gibbs sampling . Due to the higher sampling frequency , the efficiency of training the RBM is not high , especially when the feature dimension is higher . So the RBM generally adopt a fast learning method called Contrastive Divergence (CD) . The main steps of the CD algorithm are as follows:

(1) Input : training sample --x ; hidden units -- m ; learning rate – ε ; iteration T
(2) Output : weight -- W ; hidden layer bias -- b ; visible layer bias – a.

Initializing the state of the visible layer: $v_i = x_0$
w , a and b are smaller random number
While (t<=T)
calculating the probability of hidden layer (equation 11) ;
extracting the probability of all visible layers (equation 12) ;
calculating the activation probability of hidden layer .
Update of the parameters:

$$W \leftarrow W + \in \left(P(h_1 = 1|v_1)v_1^{T} - P(h_2 = 1|v_2)v_2^{T}\right)$$

$$a \leftarrow a + \in (v_1 - v_2)$$

$$b \leftarrow b + \in \left(P\left((h_1 = 1)|v_1\right) - P\left((h_2 = 1)|v_2\right)\right)$$

This paper mainly combines the two above models to recognize and analyze the image , first extracting the characteristics of the data with SA then recognizing the features through the RBM .

3. SIMULATION & ANALYSIS

We will conduct the experiment in MNIST database which has many different methods of pattern recognition already . So this is an accepted and ideal way for evaluation of the new method . The MNIST handwritten sample library a majorized subset from NIST database is adopted as the training sample set and it contains a training set of 60000 examples and a test set of 10000 examples . This experiment conducted in MATLAB use 5000 training examples and 1000 test examples . Each image has been size-normalized and centered in a fixed-size about 28X28 . During the experiment , the parameters of SA are set as follows :
Number of iterations: 300
Learning rate: 0.1
The results of the experiment are shown in Table 1 below:

Table 1: The recognition effect of different training algorithms in MNIST

Training Algorithm	Recognition Rate
BP Network	89.2%
RBM	91.4%
DBN (with two layers)	92.1%
SA+RBM	93.6%

From the results of the experiment we can conclude that the new method maintained in this paper has a better recognition effect than some of the shallow networks and DBN.

4. CONCLUSION

Deeping Learning is a rising research direction in the field of machine learning , its purpose is to let the machine imitate people that can identify and cognize the image , text etc . This paper mainly make the study of combining the SA with RBM and verify the better recognition effect than shallow neural network and DBN through the experiments.

ACKNOWLEDGEMENT

The authors acknowledge Shanghai University of Engineering and Science for giving financial supports to this project with NO. 16KY0111.

REFERENCES

[1]. ERHAN D. BENGIO Y. COUVILLE A. et al. Why does unsupervised pre-training help deep learning [J] . Journal of Machine Learning Research . 2010 . 11 (03) : 625-660 .

[2]. Yoshua bengio and Olivier Delalleau . On the expressive power of deep architectures [C] // Proe of the 14th International Conference on Discovery Science . Berlin : Springer-Verlag . 2011: 18-36 .

[3]. Vincent P . Larochelle H . Bengio Y . et al . Extracting and composing robust features with denoising autoencoders [C] // Proceedings of the 25th international conference on Machine learning . ACM . 2008 : 1096-1103 .

[4]. Hinton G E . Training products of experts by minimizing contrastive divergence [J] . Neural computation . 2002 . 14(08) : 1771-1800.