Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 131 (2018) 977–984

Procedia Computer Science

www.elsevier.com/locate/procedia

8th International Congress of Information and Communication Technology (ICICT-2018)

Research on convolutional neural network based on improved Relu piecewise activation function

Guifang Lin [a], Wei Shen [a]*

[a] School Information Science and Technology, Zhejiang Sci-Tech University, Hangzhou, China

Abstract

With the continuous development of deep learning, convolution neural network with its excellent recognition performance obtains a series of major breakthrough results in target detection, image recognition and other fields. An improved ReLu segmentation correction Activate function is proposed, by improving the traditional convolution neural network, adding the local response normalization layer, and using the maximum stacking and so on. Based on the Google depth learning platform TensorFlow, the activation function is used to construct the modified convolution neural network structure model, using the CIFAR-10 data set as the neural network input for the model training and evaluation. We analyze effects of different neuron activation function on the neural network convergence speed and the accuracy of image recognition. The experimental results show that using the improved unsaturated nonlinear segment activation function SignReLu, the convergence rate is faster, the gradient vanishing problem is effectively alleviated, and the accuracy of neural network identification is improved obviously.

Selection and peer-review under responsibility of the scientific committee of the 8th International Congress of Information and Communication Technology.

Keywords: TensorFlow; Convolutional Neural Networks; Depth Learning; Activation Function.

1. Introduction

Deep Learning (DL) [1] as a new field of machine learning research in recent years, it imitates the working mechanism of human brain to analyse and learn the images, sound, text and other data. Deep learning is a machine learning method based on data representation. Its essence is to construct a multiple hidden layer machine learning architecture model, which is trained by large-scale data, combination of low-level features to form more abstract and more representative feature information, the distribution of data representation is the given, so as to classify and forecast data, then improve the accuracy of classification and prediction.

* Corresponding author. Tel.: +86-15381159168.
E-mail address: latitude@126.com

1877-0509
Selection and peer-review under responsibility of the scientific committee of the 8th International Congress of Information and Communication Technology
10.1016/j.procs.2018.04.239

With the research and development of the deep learning method, convolution neural network [2] and many other excellent machine learning methods have emerged, which have made breakthrough progress in many applications, such as image recognition, target classification and so on. Convolution neural network is a trainable multi-layer network structure composed of multiple stacks of single-layer convolution neural networks. Each single layer convolution neural network consists of three basic stages: convolution feature extraction, nonlinear activation and down-sampling. The basic structure generally includes two layers, one is the feature extraction layer, the input of each neuron connects with the local acceptance domain of the previous layer, and extract the local feature; the other is the feature mapping layer, each computing layer of the network consists of multiple feature maps, and each feature map is a plane, and the weights of all the neurons in the plane are equal.

The activation function [3] adds the nonlinear factors to remove redundant data while preserving features, it retains "active neuron feature" and maps out these features by nonlinear functions, which is the essential of the neural network to solve the complex nonlinear problem. At present, a variety of activation functions have been applied to construct convolution neural networks such as Sigmoid [4], Tanh [5], Softplus [6], ReLu [7] and so on. Because the saturated non-linear activation functions Signmoid and Tanh are prone to defects of slow convergence speed, gradient dispersion problem, so the trend of the activation function in the neural network model is the unsaturated nonlinear, as ReLu, Softplus, Softsign [8]. Among them, ReLu is the most widely used and has multiple improvements, such as Relu6 [9], Elu [10], Leaky_Relu [11], PRelu [12], RRelu [13] and so on, which greatly contributes to the improvements of neural network performance. ReLu is easy to calculate, simple to achieve and has fast convergence speed, so it can effectively alleviate the gradient disappearance problems, and provide a certain sparse characteristics for the neural networks after training， more in line with the nature of biological neuron activation. The Softplus function is an approximate smooth representation of the ReLu function, with unilateral suppression properties, and wider excitation boundary, but it does not have better sparsity. Although the Softsign function is similar to the hyperbolic tangent Tanh, the synchronization is more robust due to its smoother asymptotic line, the relatively slow and soft saturation. The activation value using the Softsign function is uniformly distributed in a large number of nonlinear but the gradient flow of good area, has better fault tolerant ability.

On the basis of traditional convolution neural network, this paper does data enhancement, adds local response normalization, uses overlapping maximum pooling and and other improvements. As the ReLu activation function can effectively alleviate the disappearance of the gradient and sparseness, combining with Softsign activation function which has the characteristics of high degree of non - linearization and good fault - tolerant ability, we propose an improved ReLu segmentation correction activation function. Based on the Google depth learning platform TensorFlow [14], the activation function is used to construct the modified convolution neural network. The CIFAR-10 data set is used as the neural network input to train and evaluate the model. Through experiments, we compare and analyse the effect of different neuron activation functions on the convergence rate of network and the image recognition accuracy.

2. Convolution Neural Network Model and Improvement

Convolutional neural network (CNN) is a high-efficiency identification method which has been developed in recent years and has attracted wide attention from society. At present, convolution neural network has become one of the hotspots in many scientific fields. Convolution neural network has a unique superiority in speech recognition and image processing with its special structure shared by local weights, especially the image of multi-dimensional input vector can be directly input to the network for parallel learning, avoiding the complexity of feature extraction and classification process of data reconstruction, thus has been more widely used. Convolution neural networks are mainly used to identify two-dimensional images of displacement, scaling and other forms of twist invariance.

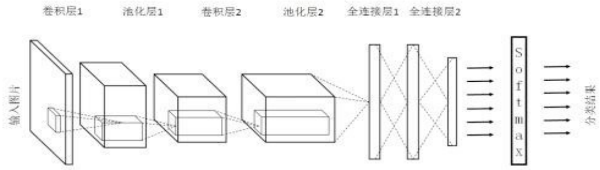

Fig .1 A convolutional neural network architecture for image classification problems

Figure 1 shows a concrete convolution neural network architecture. According to the figure, a convolutional neural network is mainly composed of five layers: input layer, convolution layer, pool layer, all-layer layer and Sortmax layer. The input layer is the input of the whole neural network. In the image processing of the CNN model, it represents a pixel matrix of a picture. The convolution layer is the most important part of a convolutional neural network. The input of each node in the convolution layer is only a small part of the upper layer of the neural network. The convolution layer analyses every smaller part of the neural network in depth, and as far as possible to get a higher degree of feature abstraction. The pooling layer does not change the depth of the three-dimensional matrix in the neural network, but it will reduce the size of the matrix, that is, reduce the number of nodes in the next layer, so as to reduce the parameters of the whole neural network and decrease the training time. After multiple rounds of convolutions and pooled layers, the information in the image has been abstracted into a higher information content, and the full connection layer is used to complete the classification task. The fully connected layer performs the combination matching and classification by modifying the nonlinear activation function, mainly used for classification problems, through the Sortmax layer, you can get the sample belongs to different types of probability distribution.

Convolution neural network model design is shown in Figure 2, which is divided into convolution layer Conv, pool layer pool, local response normalization (LRN) layer nom, full connection layer Local, the output layer Softmax. The first two layers are convolutions, and each convolution layer is followed by a maximum pooling layer and a localized normalized layer. The third and fourth layers are the full connection layer and the last layer is the output1 layer.

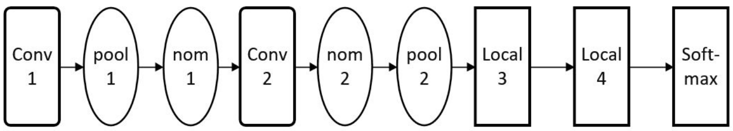

Fig. 2 Convolution neural network model

The activation function is an important part of the convolution neural network. In the three stages of convolution neural network, convolution, sub-sampling and full-connection, a nonlinear activation function is usually used to map the calculated features, thus to avoid insufficient expression problem caused by linear operation.

The expression of ReLu function is $f(x) = \max(0, x)$, the function and its derivative image are shown in Figure 3.

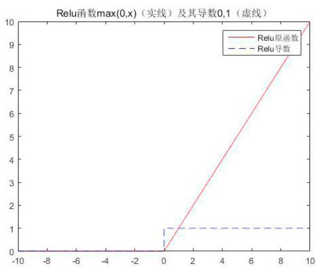

Fig. 3 ReLu function and its derivative image

As can be seen from Figure 3, Relu is hard saturated at $x < 0$. Since when $x > 0$, the derivative is 1, Relu can keep the gradient without attenuation when s, thus effectively alleviating the gradient disappearance problem. However, ReLu activation neurons are fragile, so in the training process, part of the input fall into the hard saturation area, which results in irreversible neuronal death, and the corresponding weight cannot be updated. Furthermore, Relu function sets part of the neuron output to zero, which causes the output with migration phenomenon. Such rude forced sparse processing may shield many useful features, resulting in poor effect of the model learning. Excessive sparseness may result in higher error rates and reduce the effective capacity of the model. Migration phenomenon and neuronal death can co-affect the convergence of the network.

The expression of Softsign function is $f(x) = x/(|x| + 1)$,the function and its derivative image are shown in Figure 4.

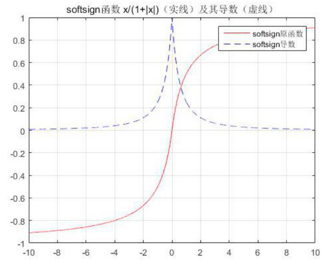

Fig. 4 Softsign function and its derivative image

Figure 4 shows that the Softsign function compresses the data into the interval of (-1,1), which is similar to the hyperbolic tangent Tanh. The output is centered on 0, but because its asymptote line is more smooth, both saturation slowly approaches to 0, the range is relatively wide, the initialization process is more robust. The middle part of the Softsign function is wide, and the area close to $x = 0$ is less, the degree of non-linearization is high, and it is easy to delineate the more complicated boundary. Softsign function is a polynomial non-linear activation function, which is relatively mid in the nonlinear part. It is simple in calculation with soft saturation, and can reduce the number of iterations, easy convergence.

Based on the characteristics of ReLu function and Softsign function, this paper proposes a novel type of unsaturated segment neuron activation function. When the data is greater than zero, the ReLU function is used to calculate the sparse ability. When the data is less than zero, the Softsign function is used to calculate in order to retain its negative axis information to correct the data distribution and ensure that it has a better fault tolerance. In this paper, the improved activation function is denoted as SignReLu function, the expression is set as follows:

$$f(x) = \begin{cases} x, & x \geq 0 \\ a\dfrac{x}{|x|+1}, & x < 0 \end{cases} \tag{1}$$

Where x represents the input of the nonlinear activation function f ; a represents the variable superparameters. When $a = 0$, the function is the ReLU function. The function image is shown in Figure 5.

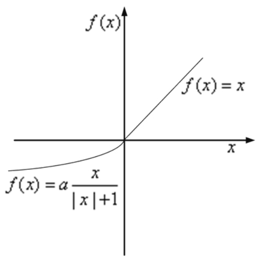

Fig 5 SignReLu function image

3. Experiment and result analysis

The convolution neural network model which is designed and implemented in this paper is relatively small and simple, and follow the principle of single variable, and controls the influence of other unrelated factors, as the main purpose of this paper is to verify the effectiveness of the proposed method.

The convolution kernel size of the two convolution layers is 5 × 5, the step is 1, the padding mode is SAME, and the convolution layer is filled with zero. The number of output feature graphs of conv1 is 64, the output feature graph size is 24 × 24, which is the same as the input feature graph, and the number of input channels is 3. The number of input channels for conv2 is 64, the number of output feature graphs is 64, and the size of input pattern is 12 × 12. The size of two Pooling layers, pool1 and pool2 are same, both are 3 × 3 and step length are 2 × 2. The localized normalized layer does not affect the number and size of the input and output feature graphs of the convolution layer. The input of the first fully connected layer is 64 feature graphs, their size are 6 × 6, the output is 384; the input of the second fully connected pool is 384, the output is 192, the output layer is 192, and the output is 10.

Table 1 Convolution neural network model architecture

Layer	Input	Convolution 5-64	Pooling3-2	Normalized	Convolution 5-64
Layer shape	24×24×3	24×24×64	12×12×64	12×12×64	12×12×64
Layer	Normalized	Pooling33-2	Full connection	Full connection	output
Layer shape	12×12×64	6×6×64	1×1×384	1×1×192	1×1×10

In the table 1, the convolution [m] - [n] indicates that the convolution kernel size is m × m, the number of output feature graphs is n, and pooling [m] - [n] indicates that the pool size is m × m, the step is n × n. In the implementation, the input image data are randomly flipped, random cut (size is 24 × 24), random brightness conversion, random contrast conversion, data standardization and other data enhancement operations. The input image size is 24 × 24, RGB three channels are not compressed, directly as a neural network input. In the process of training, the input data is first divided by 255, and the value of the input data is reduced to [0,1]. The initial learning rate is 0.1, the learning rate attenuation factor sets as 0.1, and the learning rate is attenuated exponentially based on the number of training rounds [15]. The truncated normal distribution is used to initialize the weights, set the standard deviation size, and regularize the weight by L2. Both two convolution layers and two fully connected layers are non-linearized using an activation function. For each tested activated function, this paper tests 60K steps, using a fixed batch size (batchsize = 128), data input in batches.

The experiment used the CIFAR-10 dataset for training and evaluation. The CIFAR-10 dataset covers a total of 60,000 color images with a size of 32 × 32 in 10 categories, which are divided into five training batches and one test batch. Each batch has 10,000 images and each type of picture is 6000, there is no overlap. The test batch contains 1000 images randomly selected from each category, the training batches contain the remaining images in random order, but some training batches may contain many images from the same category, and the training batches contain 5000 images of each category.

3.1. Hyperparameter experiment

In this paper, the super-parameter a was selected 14 parameters, 0,0.01,0.05,0.1,0.2,0.3,0.35,0.4,0.5,0.6,0.7, 0.8,0.9,1 for the 60K steps training round of the experiment test. In this experiment, the softsign activation function hyperparameter a is the only variable, and the rest of the convolution neural network structure remains the same, and the interference of other factors is eliminated to verify the effect of the hyperparameters a . When a = 0, the SignRelu function is the same as the ReLu function. The experimental results are shown in Table2. It can be seen from Table 2 that the performance of the SignReLu activation function is superior than the ReLu activation function (i.e. when a = 0). At the same time, the fastest convergence rate is obtained and the best Performance can be get when a = 0.1, image recognition accuracy rate is 86.96%;

Table 2 SignReLu functions results of different parameters

parameter a	0	0.01	0.05	0.1	0.2	0.3	0.35
Accuracy/%	85.888	86.590	86.630	86.956	86.679	86.699	86.531
parameter a	0.4	0.5	0.6	0.7	0.8	0.9	1.0
Accuracy/%	86.719	86.897	86.778	86.679	86.709	86.630	86.758

3.2. Activation function experiment

In this paper, we use different activation functions (where SignReLu function hypera-parameter value was set 0.1) for training rounds are 60K step experimental test. In this experiment, the activation function is used as the only variable, and the other parts of the convolution neural network are kept the same, and the interference of other factors is eliminated to ensure the reliability of the experimental data and verify the effect of the activation function on the recognition accuracy and convergence speed. The experimental results are shown in Figure 6 to Figure 10.

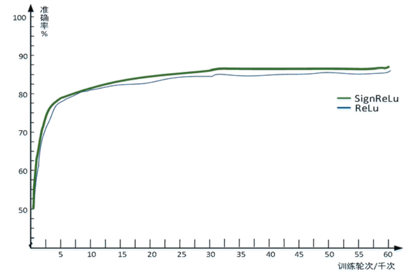

Fig. 6 Experimental comparison of SignReLu and ReLu (a = 0.1)

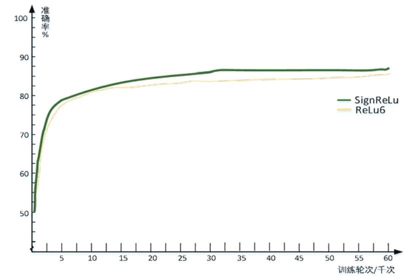

Fig. 7 Experimental comparison of SignReLu and ReLu6 (a = 0.1)

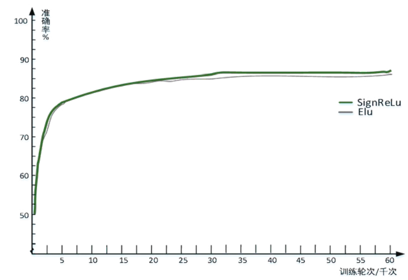

Fig. 8 Experimental Comparison of SignReLu and Elu (a = 0.1)

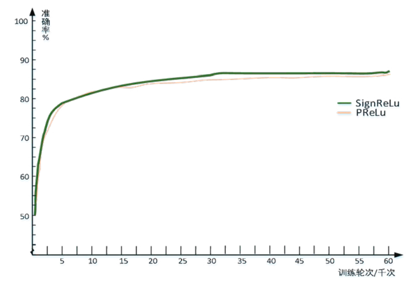

Fig. 9 Experimental comparison of SignReLu and PReLu (a = 0.1)

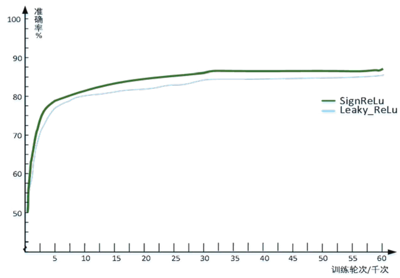

Fig. 10 Experimental comparison of SignReLu and Leaky_ReLu (a = 0.1)

Comparison of Fig. 6, Fig. 7, Fig. 8, Fig. 9 and Fig. 10 shows that the image recognition rate of the network using the ReLu6 function as the activation function on the CIFAR-10 data set is the lowest, only 85.83%; using the Leaky_ReLu function as the activation function, the convergence rate of the network is 85.85%. The convergence rate is 85.89% with the ReLu function as the activation function. The convergence rate of the network using the PReLu function and the Elu function as the activation function is higher than that of the ReLu function. And the recognition rate of images is 86.33% and 86.34% respectively. The SignReLu function is used as the activation function to converge quickly and the image recognition is the highest and the maximum recognition rate is 86.96%. The experimental results show that the SignReLu activation function is superior than other similar activation functions, the performance is good, convolution neural network convergence speed is fast, and image recognition accuracy has improved significantly.

4. Conclusion

The activation function is an important part of the convolution neural network, which can map the nonlinear features of the data, so that the convolution neural network has enough ability to capture the complex pattern. On the basis of the traditional convolution neural network, this paper enhances data, adds the local response normalization layer, and using the maximum pooling and so on. Besides the problem of insufficient expression of the Relu function, And the softsign activation function is nonlinear and the improved fault tolerance, an improved ReLu segmentation correction activation function is proposed. Based on the Google deep learning platform TensorFlow, this paper uses the activation function to construct the modified convolution neural network structure model. The CIFAR-10 data set is used as the neural network input to train and evaluate the model. The effect of different neuron activation functions on network convergence speed and image recognition accuracy is compared and analyzed through experiments. The experimental results show that the proposed improved activation function in image classification results in excellent, faster convergence speed, effectively alleviate the problem of the gradient diffusion model, and improves the image recognition accuracy of neural network.

References

1. Krizhevsky A, Sutskever I, Hinton G E. ImageNet classification with deep convolutional neural networks[C]. International Conference on Neural Information Processing Systems. Curran Associates Inc. 2012:1097-1105.
2. Lecun Y, Boser B, Denker J S, et al. Backpropagation Applied to Handwritten Zip Code Recognition[J]. Neural Computation, 2014, 1(4):541-551.
3. Mass A L, Hannun A Y, Ng A Y. Rectifier Nonlinearities Imporove Neural Network Acoustic Models[C/OL].2016-05-01.https://web.stanford.edu/~awni/papers/relu/hybrid_icml2013_final.pdf.
4. Iliev A, Kyurkchiev N, Markov S. On the approximation of the step function by some sigmoid functions[J]. Mathematics & Computers in Simulation, 2017, 133:223-234.
5. Ali Hamidoğlu. On general form of the Tanh method and its application to nonlinear partial differential equations[J]. Numerical Algebra, Control\s&\soptimization:naco, 2017, 6(2):175-181.
6. Senior A, Lei X. Fine context, low-rank, softplus deep neural networks for mobile speech recognition[C]// IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, 2014:7644-7648.
7. Nair V, Hinton G E. Rectified linear units improve restricted boltzmann machines[C]// International Conference on International Conference on Machine Learning. Omnipress, 2010:807-814.

8. Glorot X, Bengio Y. Understanding the difficulty of training deep feedforward neural networks[J]. Journal of Machine Learning Research, 2010, 9:249-256.
9. Krizhevsky A. Convolutional Deep Belief Networks on CIFAR-10[J]. 2010.
10. Clevert, Djork-Arné, Unterthiner T, Hochreiter S. Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs)[J]. Computer Science, 2015.
11. Anthimopoulos M, Christodoulidis S, Ebner L, et al. Lung Pattern Classification for Interstitial Lung Diseases Using a Deep Convolutional Neural Network[J]. IEEE Transactions on Medical Imaging, 2016, 35(5):1207.
12. He K, Zhang X, Ren S, et al. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification[J]. 2015:1026-1034.
13. Xu B, Wang N, Chen T, et al. Empirical Evaluation of Rectified Activations in Convolutional Network[J]. Computer Science, 2015.
14. Mart Wang N, Chen T, et al, Paul Barham, et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. 2015.Software available from tensorflow.org.
15. Cai C H, Ke D, Xu Y, et al. Learning of Human-like Algebraic Reasoning Using Deep Feedforward Neural Networks[J]. 2017.