ISSN: 2637-6067 DOI: 10.6918/IJOSSER.202203_5(3).0007

Image Classification Based on TensorFlow and Convolution Neural Networks

Zhuobao Tang, Hanyun Wen*

School of Computer Science, Yangtzeu University, Jingzhou, 434000, China

Abstract

It is no longer a fresh thing to use convolutional neural networks to classify images. The most common experiment is to use CNN for handwritten digit recognition. However, the pictures used for training in the recognition just have two colors of black and white while which mostly are colored in the real world. As a result, more complex models need to be designed for training and classification. This article mainly does the following work: Study the principle of CNN and its application on graph classification; Use TensorFlow to quickly build a classification model; Apply the classification model to cifar10 for graph classification and evaluated the classification model. The accuracy of the classifier designed is 78% on the training set and 70% on the verification set.

Keywords

TensorFlow; Convolutional neural network; Graph classification.

1. Introduction

With the substantial improvement of computer capabilities and the scale of image data has gradually grown, and the performance requirements of image classification algorithms have continued to increase. In recent years, image classification methods based on deep learning have made breakthroughs and are widely used in fields such as face recognition, satellite remote sensing, medical diagnosis, autonomous navigation, and human-computer interaction. After 2010, deep learning has gradually become a hot research topic in the field of image classification, and foreign researchers have carried out a lot of research work.

Ratle et al. proposed a semi-supervised image classification framework based on neural networks, and embedded regularization to generate an operational classifier to solve the remote sensing image classification problem [1]. Vincent et al. proposed an unsupervised classification algorithm for stacked denoising autoencoders, which used unsupervised training to improve the performance of subsequent support vector machine classifiers to denoise input damaged samples [2]. Krizhevsky et al. proposed a large-scale deep convolutional neural network model AlexNet, including 5 convolutional layers, 3 fully connected layers, and Softmax classification layer. It is one of the most popular image classification models in recent years [3]. Szegedy et al. proposed a 22-layer convolutional neural network model GoogleNet, which increases the network depth while reducing the dimensionality, and uses an average pooling layer instead of a fully connected layer to connect to the Softmx classifier [4]. Srivastava N et al. proposed a dropout technique to solve over-fitting in a deep neural network with a large number of parameters, whose key idea is to randomly delete units from the neural network during training. Can prevent the unit from over-adapting. And other regularization methods have been significantly improved. Greatly improve the performance of neural networks in supervised learning [5]. Use extremely deep network for image classification.

Classifiers, a key technology for image classification, have a wide variety of types, such as support vector machines (SVM), k-nearest neighbors [6], random forests [7], and softmax classifiers [8]. Although SVM has unique advantages in dealing with small sample, nonlinear

ISSN: 2637-6067 DOI: 10.6918/IJOSSER.202203_5(3).0007

and high-dimensional pattern recognition problems, its classification accuracy is not high for more complex classification problems; The process of k-nearest neighbors algorithm is simple and easy to understand, but it is a lazy learning method. When the data is distributed, the error rate of classification will increase. Random forests will produce over-fitting on some relatively noisy classification problems. This results in poor recognition, which limits its application in complex image classification problems. These shortcomings cause the above three classifiers to have certain limitations in application. The softmax classifier has advantages on simple application, high accuracy and easy training etc. The algorithm of image classification combined with the depth model has gradually occupied the mainstream of image classification algorithms, and the classification accuracy of the depth model has been continuously improved. At present, the classification accuracy of simple image data sets such as Handwritten Digit Database (MNIST) has reached more than 99%, and the classification accuracy of most images is also more than 90%. Softmax occupies an important position in the field of image classification, and its research and improvement are of great significance to improve the classification effect of images.

2. Related Basics

2.1. Introduction to Convolutional Neural Networks

Convolutional Neural Network,CNN, is a feed-forward neural network. Its artificial neurons can respond to a part of the surrounding units in the coverage area and have excellent performance for large-scale image processing. The convolutional neural network consists of one or more convolutional layers and a fully connected layer at the top, and also includes associated weights and a pooling layer. This structure enables CNN to use the two-dimensional structure of the input data. Compared with other deep learning structures, convolutional neural networks can give better results in image and speech recognition. This model can also be trained using backpropagation algorithms. Compared with other deep, feed-forward neural networks, convolutional neural networks need to consider fewer parameters, making it an attractive deep learning structure. The classic convolutional neural network is usually composed of three parts: convolutional layer, pooling layer, and fully connected layer.

2.1.1. Convolutional Layer

The matrix transformation of image elements is a method of extracting image features, and multiple convolution kernels can extract multiple features. The range of the original image covered by a convolution kernel is called the receptive field. The features extracted by a convolution operation are often local, and it is difficult to extract more global features. Therefore, it is necessary to continue the convolution calculation on the basis of a layer of convolution, which is also called a multi-layer convolution.

2.1.2. Pooling Layer

The method of dimensionality reduction, the dimensionality of the feature vector calculated according to the convolution is surprisingly large, which will not only bring a very large amount of calculation, but also prone to overfitting. The solution to overfitting is to make the model as "generalized" as possible. , That is, to "fuzzy" a little more, then one method is to perform a smooth compression process on the features of the local area in the image, which stems from the similarity of some features of the local image.

2.1.3. Fully Connected Layer

The factor in the convolution kernel is actually the parameter that needs to be learned, that is, the value of the convolution kernel matrix element is the parameter value. If a feature has 9 values, 1000 features will have 900 values, plus multiple layers, there are still more parameters to learn.

ISSN: 2637-6067 DOI: 10.6918/IJOSSER.202203_5(3).0007

2.1.4. The advantages of CNN

The scope of use of CNN is data with local spatial correlation, such as images, natural language, and speech. Local connection: local features can be extracted. Weight sharing: reduce the number of parameters, thus reducing the difficulty of training. It can be fully shared or partially shared. Dimensionality reduction: achieved by pooling or convolution stride. Multi-level structure: Combine low-level local features into higher-level features. Features at different levels can correspond to different tasks.

2.2. Introduction to TensorFlow

TensorFlow is the second-generation artificial intelligence learning system developed by Google based on DistBelief. It uses data flow graphs, an open source software library for numerical calculations. Nodes represent mathematical operations in the graph, edges in the graph represent multi-dimensional data arrays that are interconnected between nodes, namely Tensors, and Flow means calculations based on the data flow graph , TensorFlow is the calculation process of tensor flowing from one end of the flow graph to the other end. TensorFlow is not limited to neural networks. Its data flow graph supports very free algorithm expression. Of course, it can also easily implement machine learning algorithms other than deep learning.

In fact, as long as the calculation can be expressed in the form of a calculation graph, TensorFlow can be used. TensorFlow can be used in multiple machine deep learning fields such as speech recognition or image recognition. One of the highlights of TensorFlow is that it supports distributed computing on heterogeneous devices. It can automatically run models on various platforms, from mobile phones, single CPU/GPU to full-scale computing. A distributed system composed of hundreds of GPU cards.

3. Experiments

3.1. Experimental Environment and Data Set

3.1.1. Experimental Environment

The main hardware and software used in this experiment are as follows:

Operating system: Ubuntu 18.04.3 LTS

Graphics card: Nvida Tesla P100 6G

Framework: Google Colab, TensorFlow 2.2.0

3.1.2. Data Set

The CIFAR-10 data set consists of 10 types of 32x32 color pictures, containing a total of 60,000 pictures, each of which contains 6,000 pictures. Among them, 50,000 pictures are used as the training set and 10,000 pictures are used as the test set. The CIFAR-10 data set is divided into 5 training batches and 1 test batch, and each batch contains 10,000 images. The pictures of the test set batch are composed of 1000 pictures randomly selected from each category, and the training set batch contains the remaining 50,000 pictures in a random order. The training set batch contains 5000 images from each category, a total of 50000 training images. Figure 1 shows the categories of the data set and 10 randomly selected images in each category.

ISSN: 2637-6067 DOI: 10.6918/IJOSSER.202203_5(3).0007

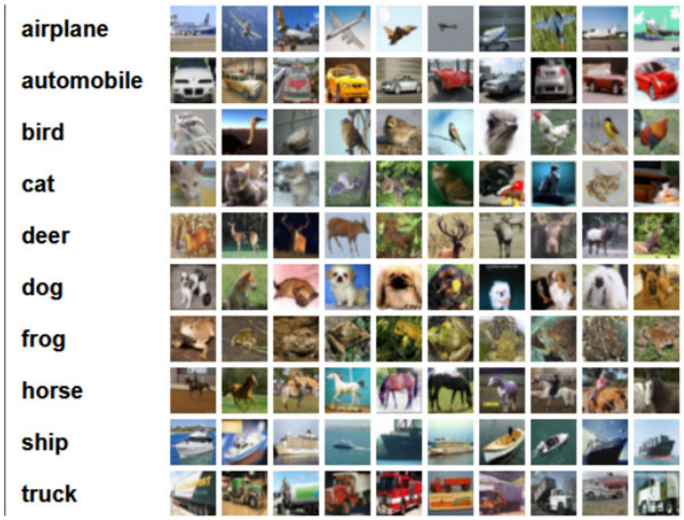

Figure 1. Samples of the CIFAR-10 data set

3.2. Data Preprocessing

First, import the TensorFlow library, load the cifar10 data set, and extract the training set and test set from the data set. Then, divide the data in the data set by 255 to scale them from the interval 0-255 to 0-1. It is convenient for processing later.

3.3. Build A Classification Model

The format of each picture in the data set is (32x32x3). The first two 32 means that the pixels of the picture are in the format of 32x32. Since the picture is in color, there are three channels to store RGB values. Therefore, the TensorFlow interface is called to process the input of shape (32, 32, 3). In order to fully extract the features of the data, set the shape of the neuron in the convolutional layer to 32x32, the step size to 1, and use the RELU activation function. The number of the first convolutional layer is set to 32, and the number of the last two is 64. In order to prevent overfitting, the dimensionality of the data is reduced in the pooling layer. The shape of the neuron is 2x2, the number and step length of each layer are 1, and there are 2 layers in total.

Finally, 3 fully connected layers are designed. In the first fully connected layer, the previously obtained multi-dimensional data is expanded into one dimension to facilitate the connection with the following neurons. The number of the second fully connected layer is set to 64, and the activation function uses the RULE function. In order to get the probability of 10 classification results, 10 neurons are used in the output layer, and the activation function is set to the SOFTMAX function. The specific classification model is shown in Figure 2.

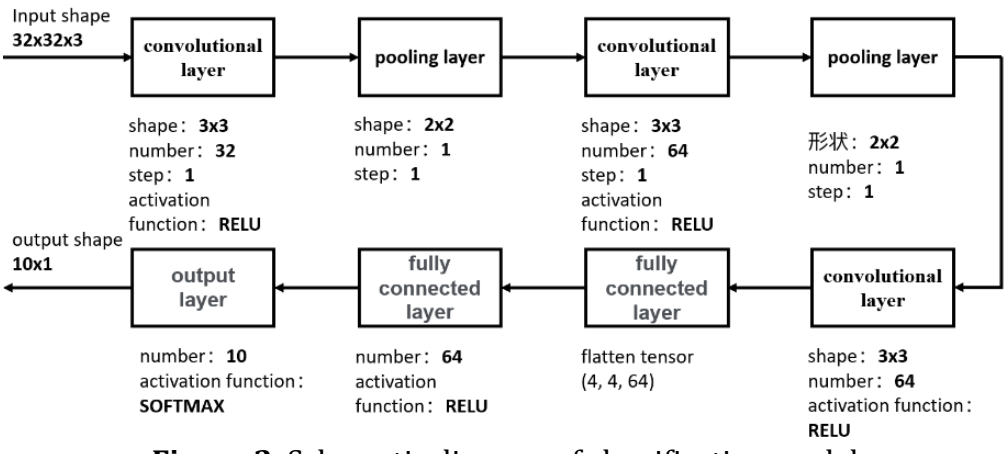

Figure 2. Schematic diagram of classification model

ISSN: 2637-6067 DOI: 10.6918/IJOSSER.202203_5(3).0007

In addition, some parameters need to be specified to guide model training. The adaptive moment estimation optimizer Adam is used to calculate the adaptive learning rate of each parameter. The initial learning rate is 0.001 by default. Use the Sparse Categorical Crossentrop function as loss function. It is a sparse multi-class cross-entropy loss function. It is suitable for multi-classification models and the classified tagets are in digital encoding format. Evaluation function is an indicator that evaluates the performance of the model during training and testing. When the value is accuracy, the ratio of correct predictions is displayed.

3.4. Training and Evaluating Classification Models

After defining the model using the TensorFlow interface, set the number of training periods, and then train the model. The main training process is shown in Figure 3.

```
Epoch 1/10
1563/1563 [ 5s 3ms/step - loss: 1.5405 - accuracy: 0.4377 - val_loss: 1.2870 - val_accuracy: 0.5370
Epoch 2/10
1563/1563 [ 5s 3ms/step - loss: 1.1698 - accuracy: 0.5856 - val_loss: 1.0811 - val_accuracy: 0.6151
Epoch 3/10
1563/1563 [ 5s 4ms/step - loss: 1.0147 - accuracy: 0.6430 - val_loss: 1.0074 - val_accuracy: 0.6414
Epoch 4/10
1563/1563 [ 6s 4ms/step - loss: 0.9146 - accuracy: 0.6776 - val_loss: 0.9434 - val_accuracy: 0.6709
Epoch 5/10
1563/1563 [ 5s 3ms/step - loss: 0.8460 - accuracy: 0.7015 - val_loss: 0.9195 - val_accuracy: 0.6787
Epoch 6/10
1563/1563 [ 5s 3ms/step - loss: 0.7838 - accuracy: 0.7249 - val_loss: 0.8752 - val_accuracy: 0.6939
Epoch 7/10
1563/1563 [ 5s 3ms/step - loss: 0.7367 - accuracy: 0.7410 - val_loss: 0.8486 - val_accuracy: 0.7075
Epoch 8/10
1563/1563 [ 5s 3ms/step - loss: 0.6910 - accuracy: 0.7561 - val_loss: 0.8548 - val_accuracy: 0.7064
Epoch 9/10
1563/1563 [ 5s 3ms/step - loss: 0.6550 - accuracy: 0.7703 - val_loss: 0.8704 - val_accuracy: 0.7053
Epoch 10/10
1563/1563 [ 5s 3ms/step - loss: 0.6143 - accuracy: 0.7829 - val_loss: 0.8896 - val_accuracy: 0.7006
```

Figure 3. Model training process diagram

The values after accuracy and val_accuracy are the accuracy of the training set and validation set obtained after each training period is completed. As can be seen from the figure, the accuracy of final training set and validation set are approximately 0.78 and 0.7. In order to see their changing trends intuitively, draw a straight line diagram between them and the training period, as shown in Figure 4.

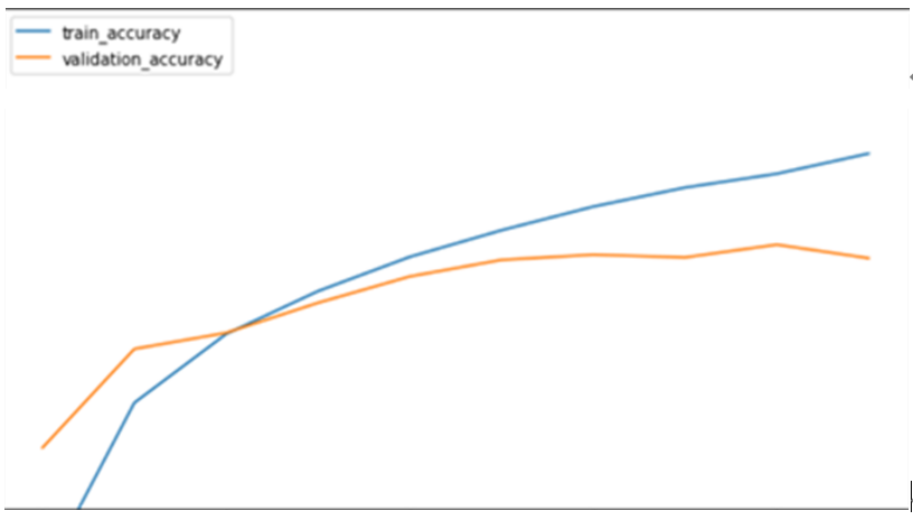

Figure 4. Accuracy varieties with the training period

ISSN: 2637-6067 DOI: 10.6918/IJOSSER.202203_5(3).0007

The abscissa is the training period, and the ordinate is the accuracy, from 0.5 to 1. It can be seen from the figure that as the number of training increases, the accuracy of the training set is getting higher and higher, and it is still increasing; but the accuracy of the validation set first increases, and then tends to level off. After period 6, around A slight fluctuation of 0.7 indicates that the number of training sessions at this time has been saturated.

Finally, in order to intuitively experience the classification effect of the classifier, 15 photos are selected from the verification set for testing. The test result is shown in Figure 5.

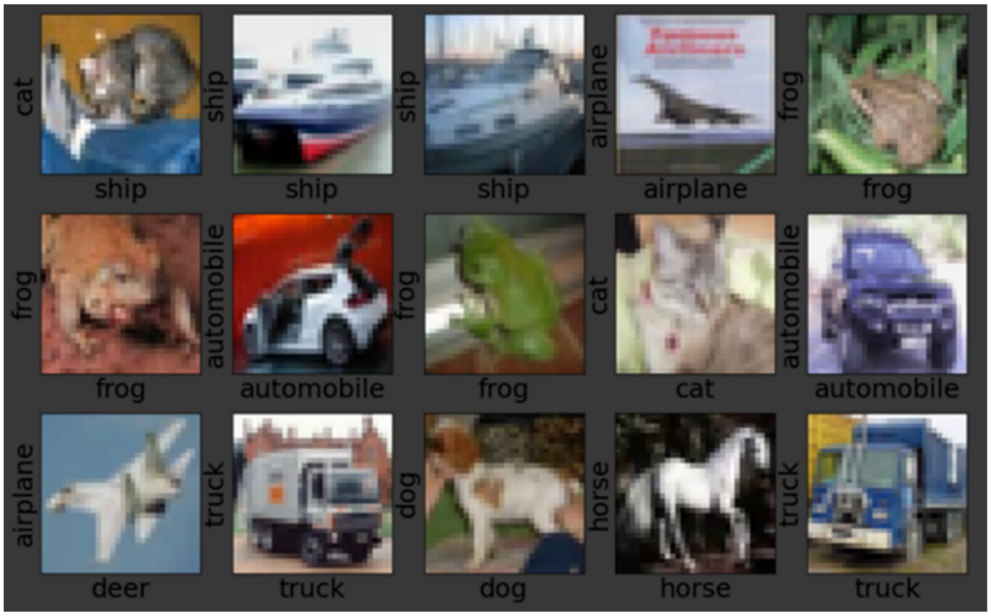

Figure 5. The test result of the classifier

In Figure 5, the X label represents the prediction result; the Y label represents the true category. As can be seen from the figure, in the selected 15 pictures, except for the two marked ones that were not predicted correctly, the rest of the pictures were predicted to be correct. At the first try, it is already very good to get such a classification effect.

4. Conclusion

In this experiment, I tried to use a convolutional neural network to classify cerfar10 images. The accuracy of the experimental model is only over 70%, and two of the 12 predicted photos are wrong. Such a model cannot be put into practice. Of course, these are all expected, and there are many reasons.

The main reason is that the classification model is still very simple, and the ability of simple models to express is limited. Secondly, I didn't do a lot of experiments to find a better initial value of weight and learning rate, just used the default initial parameters. What is gratifying is that I found a good learning platform Google Colabortory, and I also initially stepped into the door of artificial intelligence, and stepped into the bridge of machine learning to deep learning. Many APIs are called in the experiment, which can be very convenient and quick to build the model, but the details of some algorithms are still unclear. The next step will be to do a certain understanding of these algorithms. And explore other classification models and find suitable parameters to improve the accuracy of the classification model.

ISSN: 2637-6067 DOI: 10.6918/IJOSSER.202203_5(3).0007

References

[1] Weston J, Ratle F, Mobahi H: Neural networks: Tricks of the trade (Springer, Berlin, 2012), p.639-655.

[2] Vincent P, Larochelle H, Lajoie I: Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion, Journal of Machine Learning Research Vol. 11 (2010) No.12, p.3371-3408.

[3] Krizhevsky A, Sutskever I, Hinton G E: ImageNet Classification with Deep Convolutional Neural Networks, Advances in Neural Information Processing Systems, Vol. 25 (2012), p.1097-1105.

[4] C. Szegedy et al.: Going deeper with convolutions, Boston, 2015 IEEE Conference on Computer Vision and Pattern Recognition (Boston, MA, 2015), Vol. 1, p.1-9.

[5] Srivastava N, Hinton G, Krizhevsky A, et al: Dropout: a simple way to prevent neural networks from overfitting, The Journal of Machine Learning Research, Vol. 15 (2014) No.1, p.1929-1958.

[6] Fukunaga K, Narendra P M: A branch and bound algorithm for computing k-nearest neighbors. IEEE Transactions on Computers, Vol. 100 (1975) No.7, p.750-753.

[7] Breiman L: Random forests. Machine Learning, Vol. 45 (2001) No.1, p. 5-32.

[8] Wolfe J, Jin X, Bahr T, et al.: Application of softmax regression and its validation for spectral-based land cover mapping, The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, Vol. 42 (2017) No.1, p. 455.