Research on Image Classification Algorithm Based on Pytorch

Lizheng Jiang¹, Zizhao Zhang²*

School of Control and Computer Engineering, North China Electric Power University, Beijing 102206, China

zhangzizhao@eplian.com

Abstract. With the development of artificial intelligence and deep learning, image classification technology has ushered in new opportunities and challenges. The so-called image classification problem is the problem that the user passes in the image, and then the computer sends out the classification and description of the content of the image. Feature description and detection as a traditional image classification method have many drawbacks, such as low accuracy and long time-consuming problems. Among the existing deep learning frameworks, Pytorch is relatively effective. Researchers can use the Pytorch framework to construct convolutional neural network models quickly and easily, and train them on massive data sets. This paper briefly introduces the image classification algorithm, neural network and Pytorch. The classification experiment is carried out on the Fashion MNIST data set, and the accuracy of the model reaches 88.39%. The future research direction is discussed.

1. Introduction

In recent years, with the rapid development of artificial intelligence technology, society has gradually entered the era of information and intelligence, which has a profound impact on people's production and lifestyle. The goal of artificial intelligence is to allow computers to have learning and perception capabilities like humans, to simulate human thinking, and to be used by humans. Deep learning is a creative requirement for artificial intelligence in the new era. Using deep learning algorithms, low-level features can be combined and calculated to construct a relatively more abstract and specific feature. We call this feature a high-level feature. Through deep learning, a large amount of data information to be expressed by distributed features was discovered.

The advancement of deep learning models has promoted the development of image classification technology[1]. In the 1970s, the deep learning training method began to emerge. Its concept came from the research of artificial neural networks by researchers. Its main motivation was to construct and simulate as well as to simulate the activities and behaviors of the artificial intelligence brain. This neural network method is analyzed and discussed. As an important part of deep learning, image classification has been discussed and studied by many researchers. For the research of image classification, deep learning can be used to build a framework and train a good neural network architecture to achieve significant classification results[2].

The Pytorch framework is a Python-first framework. Its design idea is linear, intuitive and easy to implement. Compared with the Tensorflow framework, it has the advantages of simplicity and intuitiveness. Because of its high flexibility, ease of use, and fast speed, it is favored in many frameworks of deep learning[3]. Based on the Pytorch deep learning framework, this paper constructs an effective image classification neural network architecture through the research of processing data, constructing a neural network, training verification and prediction, and achieves the expected classification goal.
2. Research Theory and Method

2.1. Image classification related algorithms
Since the 21st century, image classification algorithm has become one of the key points and difficulties in the field of deep learning and artificial intelligence. Its main task is feature extraction, which is also one of the key factors for the effectiveness of model classification. Vailaya et al. used a multi-level Bayesian classification method to classify 6 types of holiday pictures, including multi-class images of urban landscapes, and hierarchically classify them. Edge direction histograms and color moments are used as image classification features. The concept of support vector machine (SVM) was jointly proposed by two researchers, Cortes and Vapnik. For common nonlinear problems and identifying high-dimensional patterns, in addition, in the solution of small sample problems, all made outstanding contributions, greatly improving the accuracy of classification. Dalal first proposed the Histogram of Oriented Gradients (HOG), which has been widely used in the fields of target tracking and pedestrian detection. Although early researchers have made outstanding achievements in their respective fields, there are still certain limitations for deep-level image feature extraction. Until the introduction of the deep learning framework, the effect of image classification has been greatly reduced. Deep learning is separated from manual design. The key is to use computers for learning so that it can autonomously extract higher-dimensional features of images, thereby greatly improving the accuracy of classification.

The key and outstanding contribution of image recognition research based on deep learning and artificial intelligence is to challenge the image classification problem on ImageNet ILSVRC and get a good training effect. The former researchers experienced multiple difficulties and challenges, and finally reduced the top5 error rate to 26.172% in this test summary. Fortunately, in 2012, hinton's research team worked hard to reduce the error rate from the average level of its predecessors to 15.315%. This successful case is mainly attributed to the convolutional network, which greatly reduces the error rate. Since then, in order to commemorate this network, researchers named it Alex Net. In 2014, there was a new breakthrough in image recognition and classification. In the ILSVRC competition, GoogLeNet was brave to innovate and put into action. He increased the depth of the convolutional network and significantly reduced the top5 error rate. To 6.656%, it greatly encouraged later researchers to conduct research and discussion in the field of deep learning. Among them, the more influential works include RCNN, VGG partial pyramid pooling in deep CNN, GoogLeNet and DeepID-Net.

2.2. Neural Networks
Neural networks are a very important link in the research and discussion of deep learning and computer vision. They use specific algorithms to process labeled data, and learn from the relationship between the results and the input to get the input and output corresponding formulas and laws to achieve the goal of reasonable prediction of new samples. In recent years, the neural networks that researchers have mainly studied are roughly divided into three categories: convolutional neural networks, fully connected neural networks, and recurrent neural networks, all of which have advantages in their respective fields. This experiment is based on the construction of a four-layer fully connected neural network.

In the development of deep learning and image recognition, many excellent neural network architectures have emerged. Among them, the fully connected neural network (FullyConnectedNetwork) is the most common in the history discussed by researchers[4]. The "neurons" in each layer are interconnected with the neurons in the next layer of the network to achieve fully connected mapping.

Fully-connected neural networks have achieved great success in many research fields. They have been deeply discussed in image processing. They have also been fully studied in the multi-classification of large-scale and massive data. The commercial image recognition of HKUST Xunfei The system is a good testimony. At present, the most effective way of feature extraction in the image domain is to build a neural network, which greatly improves the classification accuracy of data[5].
3. Experiment and Result Analysis

3.1. Data preprocessing

The experiment in this article is to build a four-layer fully connected neural network using the deep learning framework Pytorch under the Windows 7 operating system, an ordinary PC with 16G memory, IntelCore i7-8565U processor, and integrated graphics, and analyze the six in the Fashion-MNIST data set. Ten thousand training set pictures and ten thousand test set pictures, observe the changes of training error and verification error with the number of training.

This experiment uses the classic Fashion-MNIST dataset, which replaces the original MNIST handwritten digits dataset. Among them, the Fashion-MNIST dataset covers 10 categories of images, and Figure 1 shows some examples of each category.

![Fashion-MNIST data set example](image)

Download the Fashion-MNIST training set and test set separately from torchvisio, and then build the data loader trainloader. After the construction is completed, 64 images are randomly selected from the data set for input each time, and the order is shuffled. Adjust the grayscale data to maintain it at -1 to +1, which can facilitate processing and unify the standardized image data.

transform=transforms.Compose([transforms.ToTensor(), transforms.Normalize((0.5),(0.5))]).

3.2. Model building and training

Pytorch provides model building parameters and basic building blocks, including fully connected layers and nonlinear activation functions to complete the calculation of neural network models from input to output. This experiment uses the ReLU activation function for non-linearization. Secondly, there are various loss functions, auto-diff functions and model optimizers. Model training uses adam to automatically construct the backward pass operation of the neural network.

This article uses Pytorch1.8’s nn (Neural Networks) package and optim package. The nn package defines a series of modules. The function of each module is to calculate the received tensor and successfully output the tensor, and it can retain the tensor. A large number of learnable parameters; optim is a library that can implement various optimization algorithms. A neural network model is designed by calling torch.nn.functional, and the fully connected layer in the network is implemented using nn.Linear().

```python
self.fc1 = nn.Linear(784, 256)
self.fc2 = nn.Linear(256, 128)
self.fc3 = nn.Linear(128, 64)
self.fc4 = nn.Linear(64, 10)
```

Define the loss function as a negative log loss function.

criterion = nn.NLLLoss()
Adam gradient descent method is the optimization method of this experiment, which can greatly improve the performance of the experiment, and its learning rate is set to 0.003

```python
optimizer = optim.Adam(model.parameters(), lr=0.003)
```

Visualize the results of each training, such as the training error and the test error are stored separately, and then draw the error change graph to facilitate the user to view the change trend and law.

```python
train_losses, test_losses = [], []
```

Perform softmax operation on all elements in each row, and make the sum of all elements in each row be 1.

```python
x = F.log_softmax(self.fc4(x), dim=1).
```

The input test set needs to implement the following series of operations. First, forward inference, and then analyze and calculate the loss. Among them, the accuracy is specified as the accuracy rate, that is, the predicted probability of the model in the test set.

```python
accuracy+= torch.mean(equals.type(torch.FloatTensor))
```

The experiment has been carried out many times of learning and training, and the results of each training will change accordingly, and the results will be visualized, and the training error and the change rule of the test error will be drawn, as shown in Figure 2.

```python
plt.plot(train_losses, label='Training loss')
plt.plot(test_losses, label='Validation loss')
plt.legend()
```

![Figure 2](image_url) Training error and test error variation

### 3.3. Dropout regularization

Using the Dropout method, the researcher manually sets a probability of being deleted for a certain layer of nodes in the network, and then during the training process, the program randomly removes some nodes according to the previously set probability to achieve regularization, so that you can effectively prevent the problem of over-fitting. The training graph after dropout regularization is shown in Figure 3, and 20% of the neurons are randomly deleted.

```python
self.dropout = nn.Dropout(p=0.2)
```

![Figure 3](image_url) Dropout regularization
Analyzing the above figure, we can see that as the number of learning increases, the training error and test error are gradually reduced to the level we expect, and there is no over-fitting phenomenon in the previous experiment. Among them, Dropout regularization has played a big role.

After testing, the problem of over-fitting has been effectively solved, and the accuracy rate has also been significantly improved. Among them, the experimentally trained neural network guesses that the picture is a single shirt, with a probability of 88.39%. Figure 4 predicts the result.

![Figure 4 Forecast results](image)

### 4. Conclusion

This paper is based on Pytorch to study the image classification algorithm, and realizes the image classification and recognition of Fashion-MNIST by using nn to build a fully connected neural network model. It has practical significance. Although the model has over-fitting problems during the training process, it is regularized by Dropout. The method effectively prevents the phenomenon of over-fitting and greatly improves the classification effect. In addition, the construction and training of deep convolutional neural network models need to be further studied.

As an alternative to Numpy, Pytorch has advanced features and supports GPU acceleration, so it can quickly build neural networks and get effective training. The graph structure in Pytorch is easy to understand, and more importantly, it is easy to debug for researchers. This article uses the test set to test the accuracy of this network to reach 88.39%, and the image recognition has been able to have a higher accuracy. In summary, this neural network system can better realize the image classification and recognition of Fashion-MNIST.

### Acknowledgments

This work was supported by the Institute of Network and Information Systems of North China Electric Power University.

### References

[1] A. Vailaya, A Jain, and H. J Zb_ang. On Image Classification: City Images vs. Landscapes. Pattern Recognition, 1998, 3102): 1921-1935.

[2] Fang Xuejian. Research on image classification algorithm based on deep learning [D]. Liaoning University, 2016.

[3] Huang Yuping, Liang Weixuan, Xiao Zuhuan. Comparative analysis of deep learning framework based on TensorFlow and PyTorch[J]. Modern Information Technology, 2020, 4(04): 80-82+87.

[4] Zhou Xin. Implementation and optimization of fully connected neural network on FPGA [D]. University of Science and Technology of China, 2018.

[5] Wu Zhengwen. Application research of convolutional neural network in image classification[D]. University of Electronic Science and Technology of China, 2015.