Group22: Optimization and Text Classification

Wenwen Xu [260786231], NianZhen Gu [261044523], Yinan Zhang [260764274]

November 28, 2021

1 Introduction

Convolutional Neural Networks (CNNs) are widely used in image classification tasks.[1] CNN is good at extracting the features with convolution layers, hence can give better classification results. In this project, we are given the combo MNIST dataset which has 60,000 training images, and are required to identify an image with a size of 56*56 where each of them contains one digit and one letter. The ultimate target is to train the model to predict the combination of a number and a letter from the image.

A CNN model can be composed of different types of layers, such as the convolutional layers, batch normalization layers, dropout layers and Gaussian-noise layers. Typically, batch normalization layers speed up the learning and pooling layers reduce the dimension of the data. Both dropout layers and Gaussian-noise layers help with mitigating overfitting. To get a better result, in our model, we pick the best combination of these layers after doing many experiments.

2 Experiment

Since we have 10 possibilities for the number and 26 possibilities for the letter, we have 260 possibilities in total for the image classification (26 * 10) and translated the original label into the one-hot labeling with a size of 260. With such a premise, the network is able to learn the probability distribution of images in low dimensions and learn the mapping relationship between the image and the labels from the dataset. More specifically, we need to calculate the probability of the image being classified as each of the 260 categories and get the most probable one. Therefore, we first use 30000 labeled data to train our first model. Then, we applied semi-supervised learning to label the 30,000 unlabelled images in the dataset and add them to the training dataset. Finally, re-train our model based on all these 60000 data.

We construct a network that is composed of 5 blocks. The following is its schematic diagram:

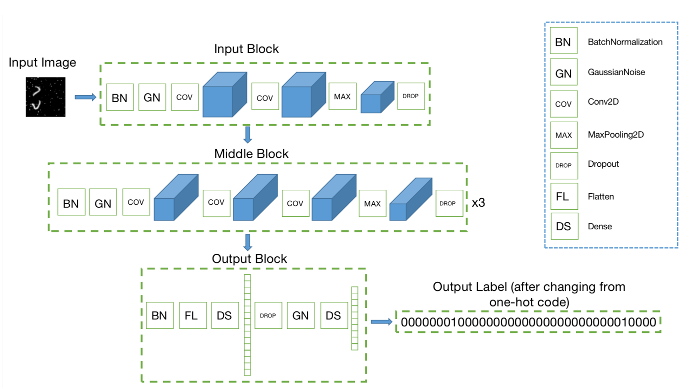

Figure 1: the Main Network Structure Constructed for this Project

The below figures are the hyperparameters of the network. The remaining hyperparameters are: batch size is 128, epoch is 60, the Adam algorithm is used for the optimizer, and cross-entropy is used for loss function.

Linear 1 - 2		
	Dimension	Activation
Linear 1	512	ReLU
Linear 2	260	SoftMax

Conv 1 - 4			
	Channel	Conv Kernels	Activation
conv 1	32	3 * 3	ReLU
conv 2	64		
conv 3	128		
conv 4	256		

Figure 2: Hyperparameters Used in the CNN Model

3 Results

We used the dataset described above, and training them spend about 12 minutes, the final result of the test datasets was above 93%. The decrease curves of the train loss and verification loss are as follows, the x-label is the epoch times, and the y-label is the information of entropy lost:

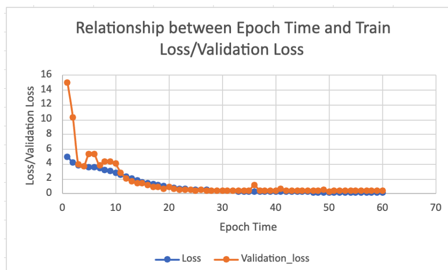

Figure 3: Relationship between Epoch Time and Loss/Validation Loss

The increase curves of of the training accuracy and verification accuracy are as follows, the x-label is the epoch times, and the y-label is the information of accuracy:

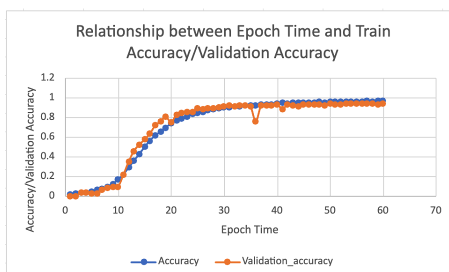

Figure 4: Relationship between Epoch Time and Accuracy/Validation Accuracy

4 Discussion and Conclusion

According to the figures from Dataset and Results, it can be seen that the model converges rapidly before epoch time = 20, and tends to be smooth after that. Therefore the model should be relatively stable. In addition, it can be seen that the curve of Loss and curve of validation loss are approximate, indicating that the model is not over-fitting.

In this project, we chose the convolutional neural network to train the dataset. We used one-hot encoding to embed the labels into 260-dimensional vectors for the training, and added Gaussian Noise and Dropout to prevent over-fitting. In later studies, we will consider embedding the labels into 36-dimensional vectors, where the first 10-bits represent the probability distribution of numbers and the last 26-bits represent the probability distribution of letters. Due to the fact that probability distributions with the same number or letter are more similar than those with different numbers or letters, such a method of encoding with a higher information density may lead to better results.

5 Statement of Contributions

Wenwen Xu is responsible for training the CNN and tuning the hyperparamters, Nianzhen Gu is in charge of data processing and network design, and Yinan Zhang is responsible for data organizing and the final report.

References

(1) Sinha, T.; Verma, B.; Haidar, A., *Optimization of convolutional neural network parameters for image classification*, 2017, pp 1–7.