Handwritten Digit Recognition with Convolutional Neural Network

Jingyi Zhou

Research School of Computer Science, Australian National University
U6015352@anu.edu.au

Abstract. With the development of deep learning, handwritten digit recognition becomes true and has little cost, which liberates the workforce of human. The key issue is to train a model for recognition. The work I did in this paper is designing and training such a Convolutional Neural Network (CNN) model with 4 layers including 2 convolutional layers, 1 fully-connected layer and 1 output layer. Additionally, I implemented dropout [1] and compared the performance with and without dropout layers. The average accuracy for testing is 99.11% with dropout and 95.21% without it, which is 0.66% lower than Dan Ciregan's multi-column Deep Neural Network in 2012 [2]. And Tom's training method was used to avoid overfittings [3].

Keywords: Convolutional Neural Network, Deep Learning, Handwritten Digit Recognition, Dropout

1 Introduction:

Handwritten digit recognition is an ability for computers to identify the digit from a given source image. In real life, this ability can be used to identify handwritten postal code from a scan image of an envelope.

1.1 Dataset Selection and Motivation

The dataset I used is MNIST [4]. There are 70000 labeled handwritten digit images with labels 0-9. I used 60000 for training and 10000 for testing.

To achieve handwritten digit recognition automatically, clear handwritten digit images with labels are required for training and test. MNIST is such a dataset and what is better is the images have been size-normalized and centered in a fixed-size image. So, I do not need to do the same preprocessing by myself. Also, the dataset was proven to be very high-quality by many scientists like Dan Ciregan mentioned before. Here is a figure example:

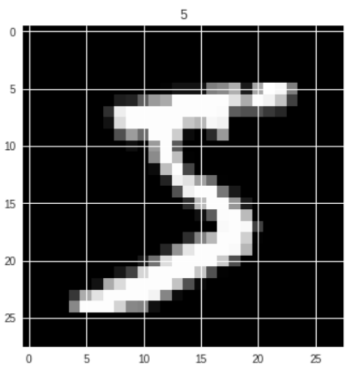

Fig. 1. Handwritten “5” in MNIST dataset

2 Method

2.1 Algorithm, Parameters and Topology of CNN

To reach the highest accuracy of the CNN model in my hardware condition, I kept adjusting the algorithm, parameters and topology of the model and finally found the best combination. My CNN has 2 convolutional layers with maximum pooling and ReLU activation function and 2 fully-connected layers with 32 and 10 neurons each. The Loss function I used is Cross Entropy Loss because it is the primary choice for classification and the optimizer is Adam which is more powerful than traditional SGD optimizer.

2.2 Overfitting and Dropout Method

The dropout method is used to avoid overfitting, but what is overfitting? Overfitting occurs when excessive parameters are used during training. This leads to excessive training data being fully fitted by model, so model will predict an incorrect function as figure 2 shows. Overfitting means although the accuracy of training set is high, testing accuracy is low.

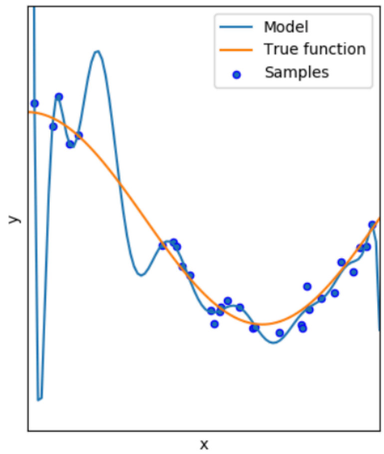

Fig. 2. Overfitting

(figure from online source

https://www.google.com.hk/search?q=overfitting&safe=strict&source=lnms&tbm=isch&sa=X&ved=0ahUKEwilod6L7a_bAhUH6LwKHWUpAWAQ_AUICigB&biw=1440&bih=736#imgrc=mbyhafni2AUYMM:)

Dropout is a method of *torch.nn*. It is an effective technique for regularization and preventing the co-adaptation of neurons, and its primary idea is to drop out outputs of hidden neurons randomly after each dropout layer [1]. Compared with fully connected layers, CNN with dropout layers will not rely on some local features too much, so it can relieve overfitting.

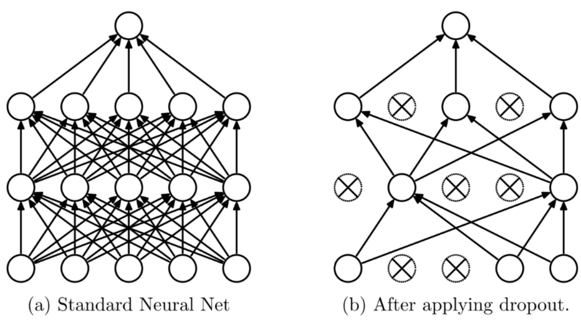

Fig. 3. Dropout

(figure from online resource https://blog.csdn.net/u013989576/article/details/70174411)

2.3 The Method of Tom Gedeon

In the paper, Classifying Dry Sclerophyll Forest from Augmented Satellite Data: Comparing Neural Network, Decision Tree & Maximum Likelihood [3], the Neural Network part, to avoid overtraining on the training set, “the neural network was trained on the training set, until the error on the test set was minimum”. I implemented it by setting 2 conditions for terminating of training - when the test set getting the target accuracy (say 99.11%) or experiencing the maximum of number of epoch, 50 epochs. The highest accuracy I got by this method is 99.11% with dropout. The method does work.

3 Result

3.1 Result of Dropout

I compared the CNN accuracy with and without dropout in same conditions for 5000 steps. Figure 4 shows the accuracy without dropout, and the red curve is test accuracy and the blue one is for training accuracy:

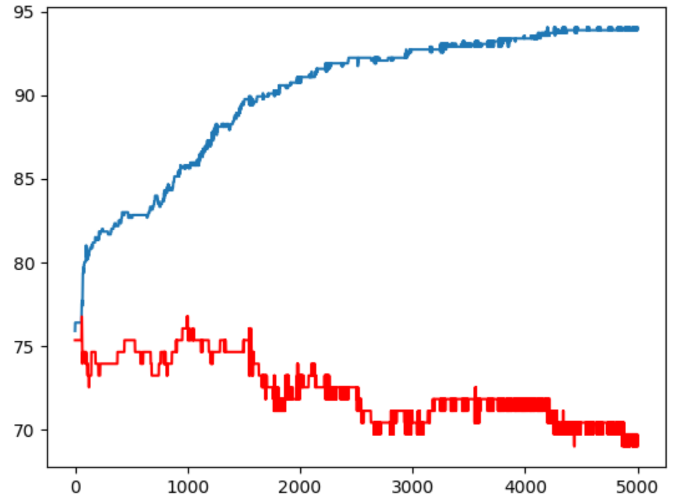

Fig. 4. Test accuracy (red curve) and training accuracy (blue curve) when without Dropout

Figure 5 shows the accuracy with dropout:

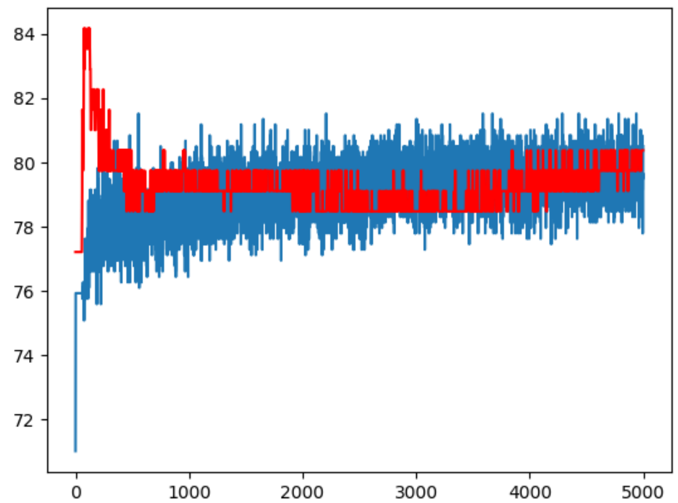

Fig. 5. Test accuracy (red curve) and training accuracy (blue curve) when with Dropout

From the 2 figures, dropout does relieve overfitting and improve the test accuracy. For many times test, the average accuracy for the CNN model with dropout layers is 99.11%.

3.2 Evaluation of Model

Besides using accuracy as evaluation, for visualization, I plot the model output in 2 dimensions and marked each class (0-9):

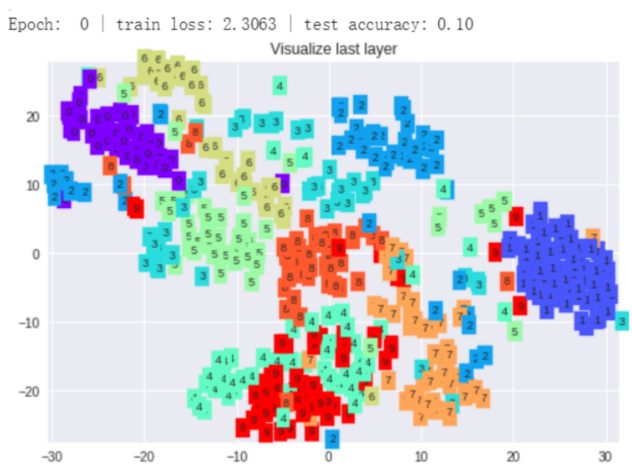

Fig. 6. Visualization when test accuracy is 10%

When the accuracy is only 10%, we can see from figure 6 that no class was identified correctly because nearly all labels are mixed up. Let us see when accuracy is 85%:

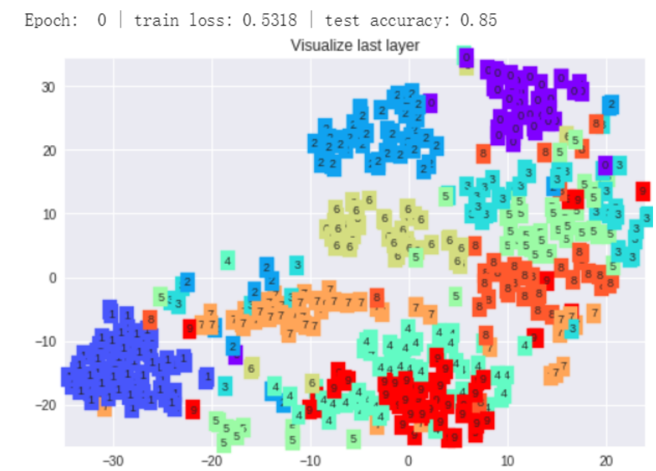

Fig. 7. Visualization when test accuracy is 85%

Most classes are gathered in their own areas and little is mixed up. So, the accuracy increased. Evaluating model in such way is more visualized than using accuracy.

4 Conclusion and Future Work

In conclusion, the CNN model with algorithms mentioned in section 2.1 can identify the handwritten figures with a satisfied accuracy, 99.11%, and both Dropout and Tom's methods work efficient against overfitting problem and improve the performance of the model.

In the future, to increase the accuracy, some outliers' detection techniques should be implemented, like Absolute Criterion Method, Least Median Squares (LMS), Least Trimmed Squares (LTS) and Bimodal distribution removal (BDR) [4]. And the model should be used to more complex dataset (e.g. digit images without fixed size) to test whether it is good enough to identify handwritten figures in real world. More challenging, the model can be improved to recognize other characters like English or Chinese characters.

Reference

1. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. The Journal of Machine Learning Research, 15(1), 1929-1958.

2. Ciregan, D., Meier, U., & Schmidhuber, J. (2012, June). Multi-column deep neural networks for image classification. In Computer vision and pattern recognition (CVPR), 2012 IEEE conference on (pp. 3642-3649). IEEE.

3. Milne, L. K., Gedeon, T. D., & Skidmore, A. K. (1995). Classifying Dry Sclerophyll Forest from Augmented Satellite Data: Comparing Neural Network, Decision Tree & Maximum Likelihood. In Proceedings Australian Conference on Neural Networks (pp. 160-163).

4. Slade, P., & Gedeon, T. D. (1993, June). Bimodal distribution removal. In International Workshop on Artificial Neural Networks (pp. 249-254). Springer, Berlin, Heidelberg.