COMP551 Mini Project 3

Group 68 : Yuqi Liu, Anthony Ma, Rebecca Zhang

December 11, 2020

Abstract

In this project, we experimented with the Convolutional Neural Network (CNN) model on a multi-label classification task in this project. The CNN model used a ResNet(Residual Neural Network) structure as backbone and the Adam optimizer. We used a dataset consisting of numbers of images with different digits. We then manually picked the best hyper-parameters combination to optimize the model's performance by doing a simplified grid search. We found the accuracy to the highest with learning rate equals to 0.001, batch size equals 32, and discovered that the loss starts to converge after repeatedly training the model for seven rounds.

1 Introduction

In this project, we perform a multi-label classification on images with one to five digits, where an unknown/empty label is replaced by number ”10”. If fewer than 5 digits are detected during our prediction, the rest will be filled with ”10”s. (eg. label of “5” is “510101010”). The CNN model is one of the most popular feed-forward neural networks for such applications. In addition, the Deep Residual CNN structure is vital in this task since it addresses the issues of degradation and the initial vanishing/exploding gradient during training [1]. We chose Adam optimization algorithm because empirical results demonstrate that Adam works well in practice and compares favorably to other stochastic optimization methods [2]. Eventually, we selected the best hyper-parameter combination for our model by using a simple ”manual” grid search.

2 Convolutional Neural Network

CNN is a class of deep neural networks, which may take considerable time to train. Therefore we implemented ResNet for convolutional layers as it eases the traning of networks that are substantially deeper [1]. As shown in Figure 1, ResNet-18 and 34 contain a block structure similar to the left, where each block is a 56 x 56 feature maps, with 64-d refers to the number of feature maps(filters). The bottleneck architecture on the right has 256-d, designed for a deeper network, which takes higher resolution images. Hence, considering the time complexity, we chose the ResNet-18 network for this task. The classic ResNet-18 is consists of 17 convolutional neural network, plus one FC layer. However, for this dataset, whose label is of constant length, there could be upmost five digits in one image. So we used five parallel fully connected layers instead of only one. This method enables us to learn a non-linear combination of these five features(digits) without pre-slicing the images [3].

Each FC layer has a output of class 11 (0-9 digits and empty). We calculate the loss using the according label as target and feed the result back to the network to know how much of the loss each node is responsible for and update the weight to minimize the loss. The Adam optimizer uses the squared gradients to scale the learning rate, and it takes advantage of momentum by moving the average of the gradient instead of the gradient itself. Given parameters ”w” and a loss function ”L”, Adam's parameter update is given in Figure 2. The default beta parameters for Adam is usually ($\beta 1$=0.9, $\beta 2$=0.999). [4]

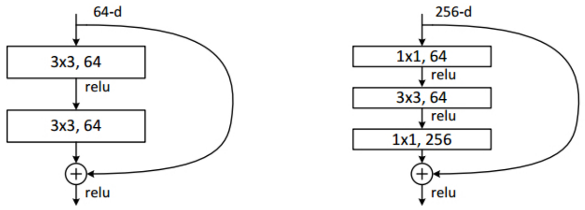

Figure 1: Left: for ResNet-18/34. Right: a ”bottleneck block” for ResNet-50

$$m_w^{(t+1)} \leftarrow \beta_1 m_w^{(t)} + (1-\beta_1)\nabla_w L^{(t)} \qquad w^{(t+1)} \leftarrow w^{(t)} - \eta \frac{\hat{m}_w}{\sqrt{\hat{v}_w}+\epsilon}$$

$$v_w^{(t+1)} \leftarrow \beta_2 v_w^{(t)} + (1-\beta_2)(\nabla_w L^{(t)})^2$$

Figure 2: ϵ is a smaller scalar used to prevent division by 0. $\beta 1$ and $\beta 2$ are the forgetting factors for gradients and second moments of gradients, respectively.

3 Hyper-parameter Selection

In this section, a detailed analysis of how different hyper-parameter choices affects the performance of the CNN model is presented.
As shown in Table 1, the test accuracy is not affected considerably by different learning rate and batch size choices. However, a larger batch size reduces the variance in the gradient estimation, leading to faster convergence. [0] This paper shows that training with a large batch usually suffers from performance degradation, the so-called ”generalization gap.” Therefore, smaller batch size is considered preferable here. Besides batch size choice, Initial learning rate of 0.001 leads to a higher test accuracy. Here we use the word ”initial” because the learning rate adapts to different parameters during train steps, meaning that every network parameter has a specific learning rate associated. Every single learning rate for the parameter is computed using lambda (the initial learning rate) as an upper limit. To make sure every update step does not exceed lambda, we applied step decay to the optimizer, meaning we multiplied the learning rate by 0.2 every ten epochs. It can help reduce loss during the latest training step when the computed loss with the previously associated lambda parameter has stopped to decrease. To ultimately minimize the loss, we trained the model back and forth through the network for many rounds, as indicated by number of epochs, and recorded its according accuracy and loss. As indicated in Figure 3, 4, and 5, accuracy increases as numbers of epoch increases. For our best performed hyper-parameter combination, loss function tends to converge when epoch reaches 7.

4 Discussion and Conclusion

In this project, we discovered that the CNN model using ResNet-18 and the Adam algorithm as the optimizer achieves a very high accuracy(above 0.990) in the multi-label classification on image data. A possible explanation for such an ideal result is that CNN can effectively use adjacent pixel information to effectively down-sample the image first by convolution and then uses a prediction layer at the end. Additionally, the reason that CNN model performance is not affected by learning rate and batch size might be that the hyperparameters already have relatively intuitive interpretations during Adam optimization (e.g., the learning rate is automatically adapted to different parameters) leaves little space for further tuning. Considering time complexity and prediction accuracy, we eventually chose ResNet-18 with (Learning Rate=0.001; Batch Size=32) combination to be the best model for this type of task. Also, given that the loss function starts to converge when the epoch reaches 7, we will choose the maximum epoch number to be around 7 - 10 for future tasks.

4.1 Team Work

The work load was distributed evenly amongst the team members. All three members contributed in each tasks and the report.

References

[1] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” 2015. 1

[2] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” 2014. 1

[3] S. S. Basha, S. R. Dubey, V. Pulabaigari, and S. Mukherjee, “Impact of fully connected layers on performance of convolutional neural networks for image classification,” *Neurocomputing*, vol. 378, p. 112–119, Feb 2020. [Online]. Available: http://dx.doi.org/10.1016/j.neucom.2019.10.008 1

[4] “adam$_{p}aram,''2017.[Online].Available:$ 1

E. Hoffer, I. Hubara, and D. Soudry, “Train longer, generalize better: closing the generalization gap in large batch training of neural networks,” 2018. 2

Appendix:

	Batch Size=32	Batch Size=64	Batch Size=128
Learning Rate=0.001	99.7	99.7667	99.6833
Learning Rate=0.0001	99.7667	99.7167	99.65

Table 1: Test Accuracy of Different Combinations of Learning Rate and Batch Size.

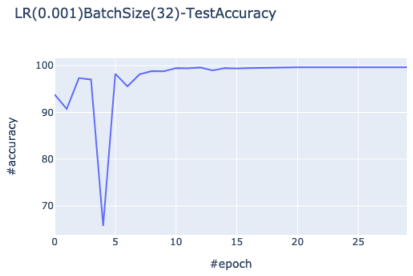

Figure 3: Best Model's Test Accuracy over different epochs.

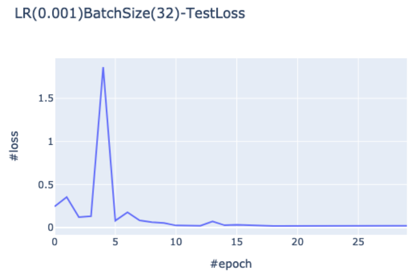

Figure 4: Best Model's Test Loss over different epochs.

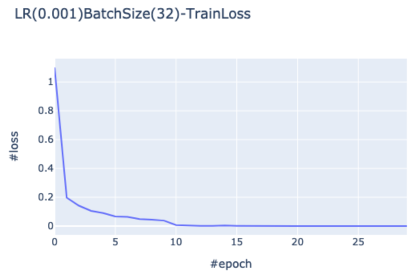

Figure 5: Best Model's Train Loss over different epochs.