Training Resnet on CIFAR10 and crowd dataset

Ju Yiang
Shanghaitech
Shanghai, China
juya@shanghaitech.edu.cn

Abstract

In the first part, This paper discussed the training process of Resnet on CIFAR10 dataset and Shanghaitech crowd counting dataset. I first trained the Resnet from scatch, then I trained it using the pretrained model on ImageNet. This paper will compare the performance of two models and discussed the possible reasons for the difference. In the second part, I trained a crowd counting network on Shanghaitech crowd counting dataset.

1. Training from scatch

1.1. structure of the network

The structure used in this paper is Resnet50.

1.2. hyperparameters

number of epochs = 50, batch size = 50, learning rate = 0.001

1.3. other training settings

loss function = cross entropy loss, optimizer = SGD(with momentum = 0.9), The network is trained on a Nvidia 3050ti laptop GPU.

1.4. performance

loss and accuracy are shown in figure 1 and figure 2 respectively. Final accuracy on test set is about 72%.

2. Training from pretrained model

2.1. stucture of the network

same as section 1.

2.2. hyperparameters

same as section 1.

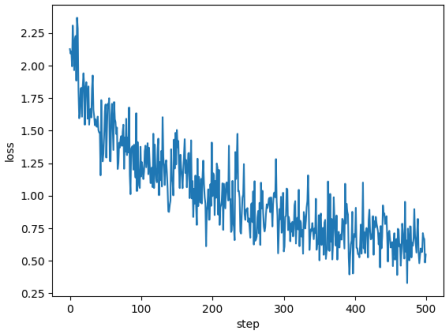

Figure 1. Training loss(from scatch)

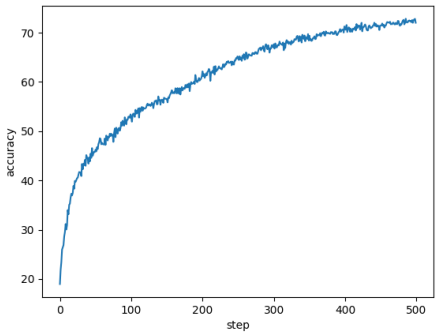

Figure 2. Training accuracy(from scatch)

2.3. other training settings

same as section 1. (pretrained model used is resnet50 pretrained on imagenet.)

2.4. performance

loss and accuracy are shown in figure 3 and figure 4(in the next page) respectively. Final accuracy on test set is about 87%.

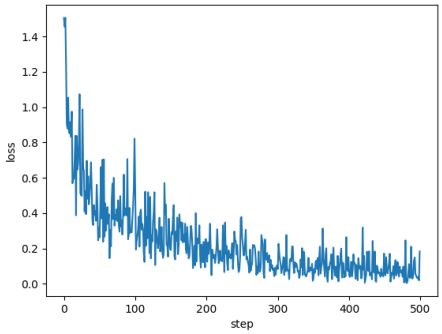

Figure 3. Training loss(from pretrained model)

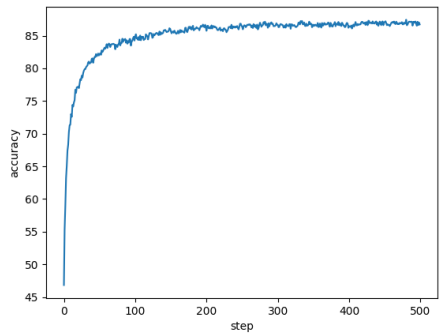

Figure 4. Training accuracy(from from pretrained model)

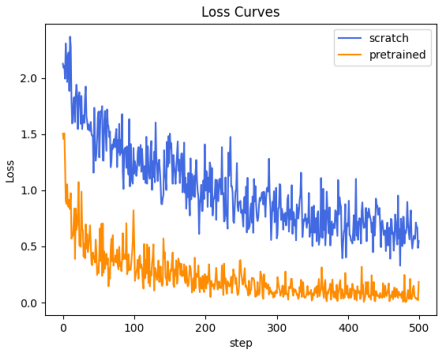

Figure 5. Training loss(from pretrained model)

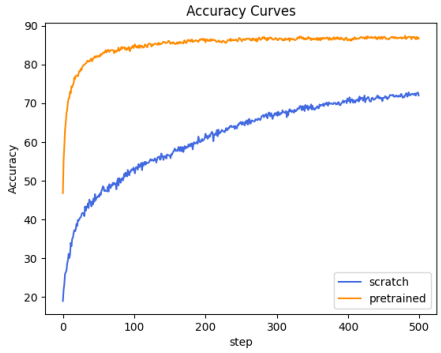

Figure 6. Training accuracy(from from pretrained model)

3. Comparison

3.1. performance comparison

Loss and accuracy of the two version are shown in figure 4 and figure 5 respectively. We can see that the pretrained model converges much faster and has a higher accuracy. What's more, the pretrained model has a much lower loss at the beginning of training.

3.2. possible reasons for the outstanding performance of pretrained model

The pretrained model has a much higher accuracy because it has already mastered the ability to extract basic features, which can be directly used in the classification task on CIFAR10. That's why it has a better loss performance at the beginning. The pretrained model converges faster because it has learned proper weight to classify Imagenet, it only needs to do some moderate adjustments to fit CIFAR10.

4. Problems and solutions in training process

When I was training resnet50 at the begining, I use a relatively large learning rate, which is 0.1 and it turned out that although the loss drop quickly, the accuracy stuck at a bottleneck of about 60%, later I use a smaller learning rate with scehduler, which changes learning rate as the training process goes on, and the accuracy increased to about 74%.

5. Training a crowd counting network

This part correspond to task2 in this assignment, which requires us to use ResNet for crowd counting task on Shanghaitech Crowd Counting Dataset.

5.1. structure of the network

In this task, I use resnet18 because its size is smaller and can be trained faster.

5.2. hyperparameters

number of epochs = 500, batch size = 4, learning rate = 0.001

5.3. other training settings

loss function = MSE loss, optimizer = Adam, scheduler = StepLR(step size = 20, gamma = 0.9) The output

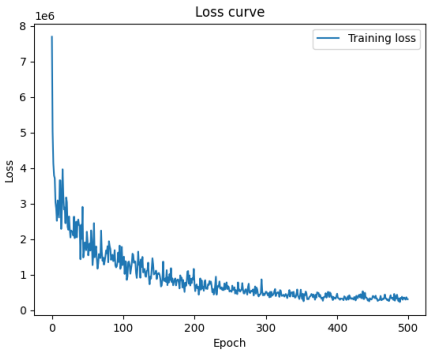

Figure 7. Training loss over epochs on dataset A

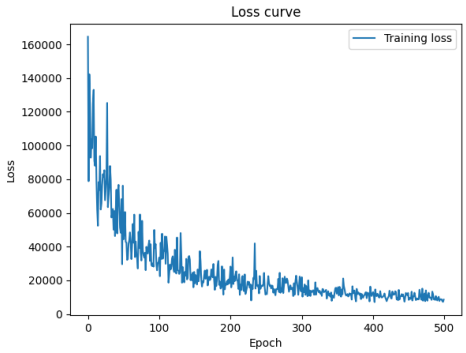

Figure 8. Training loss over epochs on dataset B

of the network is a single number, which is the predicted number of people in the image, this is achieved by changing the output size of the last fully connected layer of the network to 1.

5.4. performance

I trained the network on dataset A and dataset B seperately, call them ModelA and ModelB and the results are shown in figure 7 and figure 8. After training, I tested their MSE and MAE on test set, the results are shown in table 1.

Table 1. ModelA's performance on two test sets

	test set A	test set B
MSE	40638.03	2860.20
MAE	128.33	38.49

Table 2. ModelB's performance on two test sets

	test set A	test set B
MSE	85183.41	1761.31
MAE	185.10	25.01

5.5. Problems and solutions

When I was training the network, I found that the loss on each step will oscillate rapidly, so I employ a "loss sum" to add up all the losses among one epoch, so that I can see the trend of the loss. After that we can see a claeaer trend of the decline of loss.

5.6. possible improvements

Due to limitation of time, I didn't try transfer learning on this task, but I think it will improve the performance of the network. My reasons are as follows: datasetA provides figures with no fixed size, which add difficulty to learning how to count people from a transformed image, while datasetB provides figures with fixed size, which is easier to learn. So if we train the network on datasetB first, then train it on datasetA, we might get a more generalized counting network. Or it might work better if we train on datasetB first then train on datasetA. I guess I would try do this experiment in my leisure time.