Tzvi Fisher 037580644

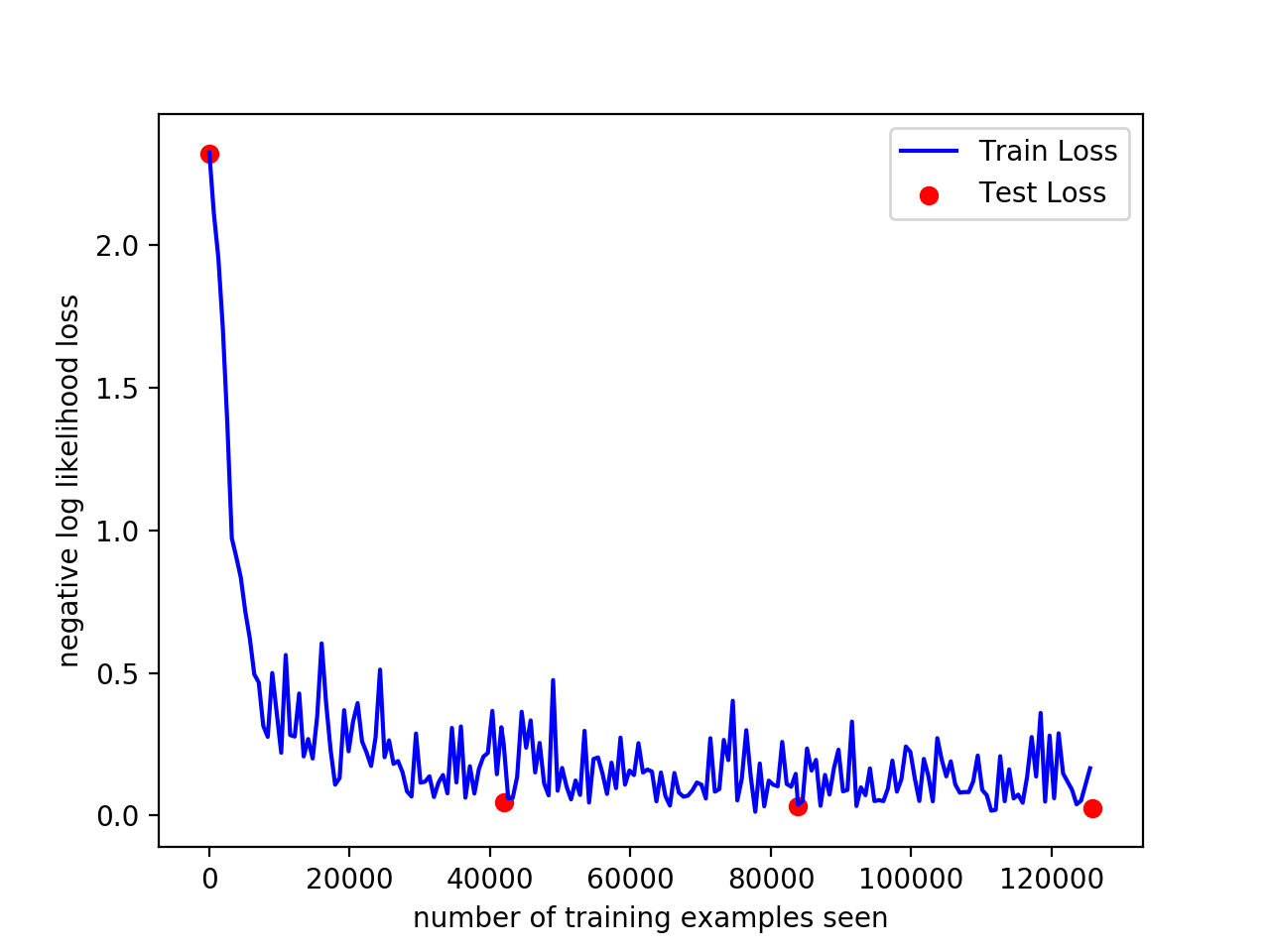
Micha Aga 039214820

Deep Leaning

Exercise 2

Assignment (Report):

1. 0-6 digits

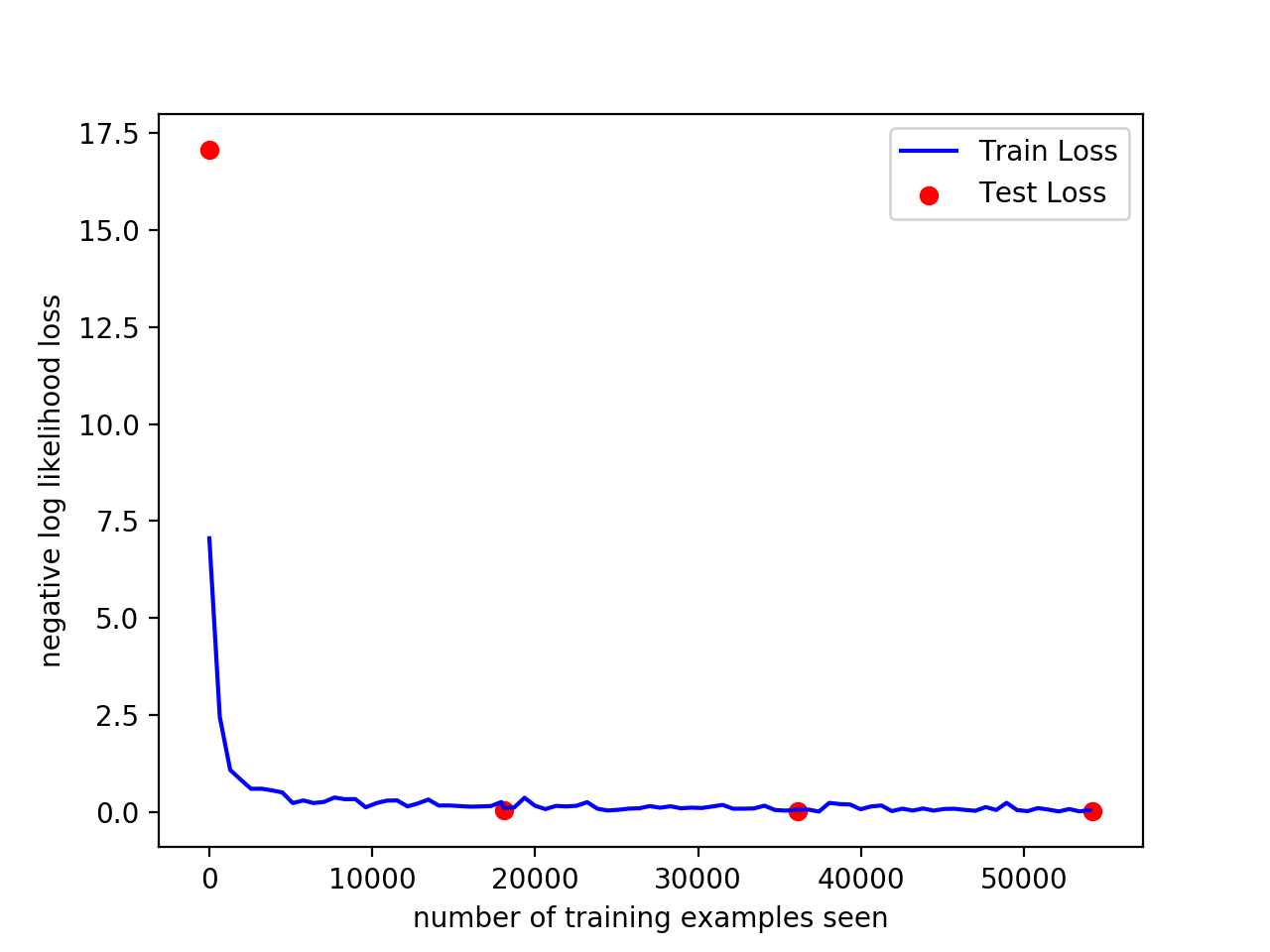


Terminal output:

*Test set: Avg. loss: 0.026353842550711427, Accuracy: tensor([6931])/6989 (tensor([99.1701])%*

*total training time:87.93*

1. 7-9 digits with freeze weights in the 1st layers and initialized weights in the last layers:

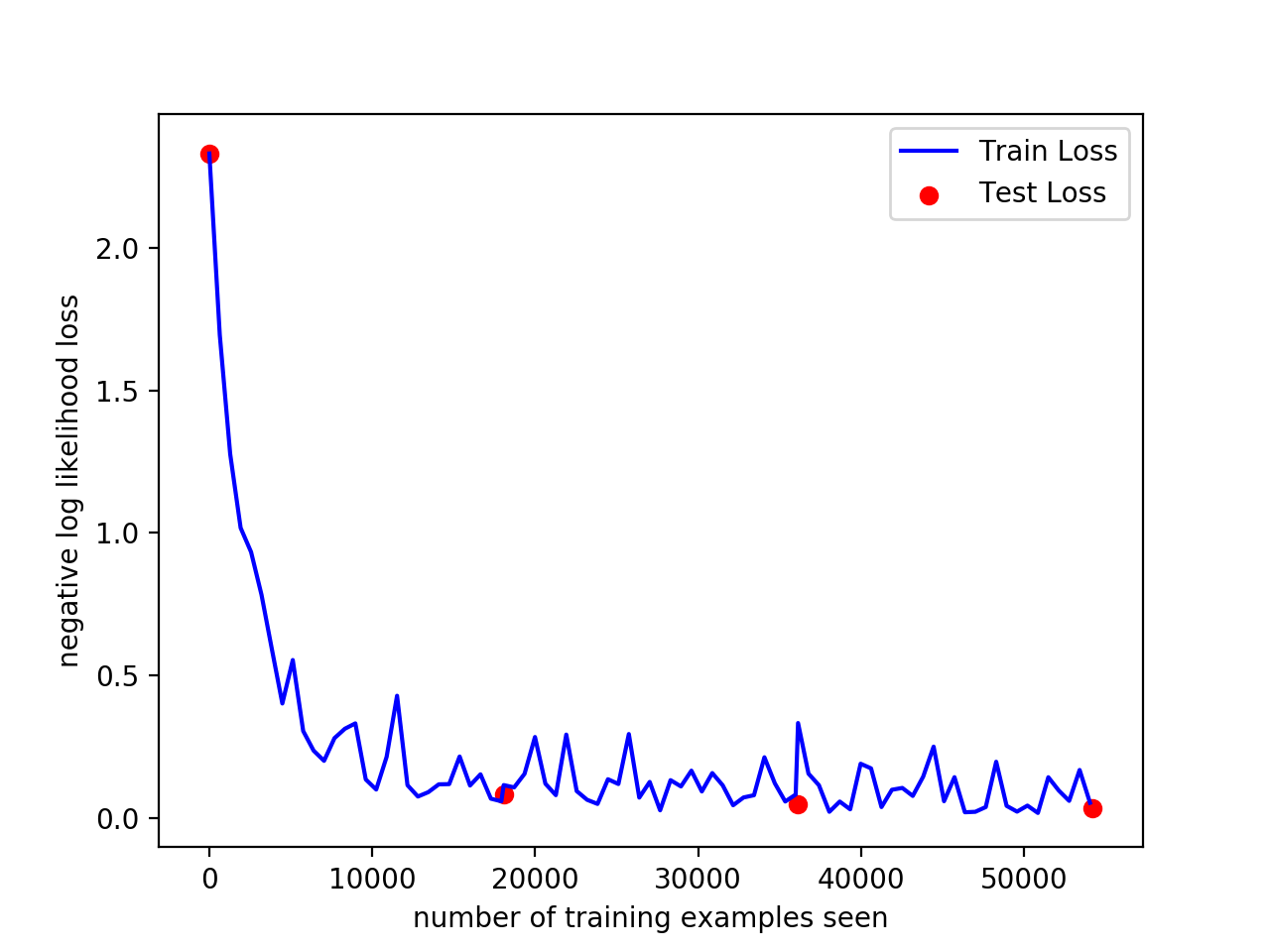


Terminal output:

*Test set: Avg. loss: 0.020874909084740215, Accuracy: tensor([2990])/3011 (tensor([99.3026])%*

*total training time:30.93*

3.



Terminal output:

*Test set: Avg. loss: 0.034320514934952175, Accuracy: tensor([2975])/3011 (tensor([98.8044])%*

*total training time:42.56*

4. The train loss in item 4 is much faster and smoother than in item 5.

I assume that the reason is the in item 4 the cnn has already been trained

Thus , it just needed fine tuning in the last layers to predict the new digits(7-9). The 1st layers are responsible for corners/edges which are in common to all digits(0-9) and the fact that the weights were frozen didn’t affect the cnn.

5. The training time in item 4 is faster because it didn’t need to propagate through all the layers(the 1st layers were with no\_grad), thus less computations.