Piradon (Tien) Liengtiraphan
Assignment Number 2
Instructor: Dr. Pablo Rivas

CMPT 496L-111: Deep Learning w Tensor Flow
Due: 2017-10-23

Assignment 2

Exercise 1a.

Go to the labs and find the one for RNNs entitled "Applying RNNs/LSTMs to MNIST". You will be running a few experiments and will write a brief report about your results and the interpretation of your findings.

Here is what you need to do: you will vary the number of hidden units exponentially as follows, 16, 32, 64, 128, 256, 512, 1024, 2048, and 4096; you will also vary the learning rate exponentially as follows: 0.00001, 0.0001, 0.001, 0.01, 0.1, and 1.0.

Response.

Neurons / Learning Rate	0.00001	0.0001	0.001	0.01	0.1	1.0
16	0.1789	0.2042	0.7562	0.9319	0.8426	0.0974
32	0.0992	0.4726	0.895	0.952	0.7996	0.114
64	0.157	0.6923	0.9425	0.9663	0.9047	0.101
128	0.31	0.815	0.9461	0.9567	0.4604	0.098
256	0.4777	0.8691	0.9649	0.9631	0.1028	0.2281
512	0.6151	0.8814	0.9496	0.7684	0.0958	0.0958
1024	0.75	0.9195	0.96	0.1028	0.1135	0.101
2048	0.8126	0.9136	0.943	0.0892	0.1135	0.1009
4096	0.8228	0.8979	0.956	0.0958	0.0974	0.0974

Exercise 1b.

You will discuss your results and interpret them; what is this telling you about the training of an LSTM with respect to the number of neurons or learning rate or both? What can you learn from this?

Response.

As the number of neuron increases you will generally see a proportional increase in your Testing Accuracy; however, the larger factor in this particular case would seem to be finding the ideal learning rate.

Inferring from the results above, one can see that as the learning late increases, from 0.00001 to 0.001 there is a general increase of Testing Accuracy. But once one steps over that upper limit, what ensues

seems to be a downward slope, with the results getting worse and worse with every iteration and increase of both neurons and learning rate. The "downward slope" seems to increase with the number of neurons, and learning rates faster than 0.001. meaning that as the learning rate increases, if the number of neurons is not particularly high then the Testing Accuracy will not drop to a significant degree within a short amount of time, possibly due to under fitting. However should then number of neuron be high and the learning rate past the ideal point, one will see a significant drop in Testing Accuracy name the bottom left corner of 0.0974, which might be due to significant over fitting. This is most likely due to the fact that the problem scope is not large enough for the number of neuron present.

In regards to learning rate, as we increase learning rate we allow the the LSTM to minimize the function more, leading the more efficient "learning" by the neural network. However the problem arises that when you minimize too much, this could lead to NN skipping certin step in order to avoid local minimums; leading to inaccurate data.

What we can learn from this is that there is almost always an upper bound to things, at least in respects to Deep Learning, that by adding more neurons to a problem we may not be generating the best possible data. Deep Learning is a juggling act to find the best parameters that suite a certain problem and once those parameters are found, to extrapolate the best data from there.
