Task 4 : Recurrent Neural Networks

Sanjeev Narayanan

3rd March 2018

Introduction

Traditional Neural Networks (NN) have a drawback in persistence of information in the system. This issue is solved by Recurrent Neural Networks (RNN) [1]. They are NN with a feedback loop that keep the information digested for future access.

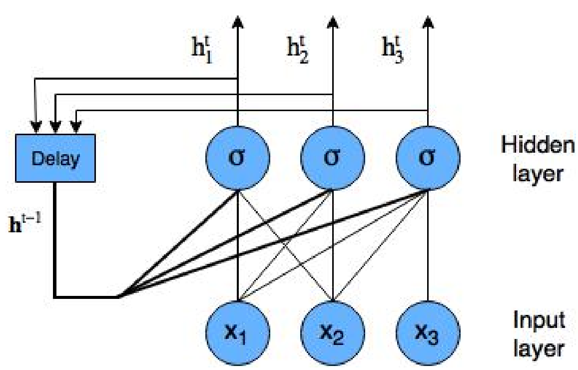

Figure 1: Example of RNN

In the figure above, we have a simple RNN with three hidden layers and three input nodes with an activation as per any NN. However, the output is fed back to the hidden layers. The main reason to involve the feedback mechanism is that we can model time/sequence independent data.

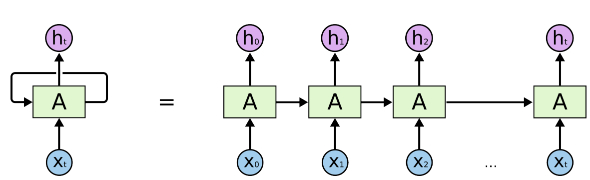

Figure 2: Feedback unrolled to show sequence interpretation of RNN

Long Short-Term Memory Networks

Long Short-Term Memory or LSTMs [2][3] are a special kind of RNN that handle long-term dependencies. These dependencies connect information supplied initially and discovered later through feedback mechanism. LSTMs have their main component as the cell state. They have multiple gates (input, forget and output) to regulate the flow of information. Gates are composed of sigmoid NN layer and an optimization function. The main steps are briefly listed as follows:

- Decide what to do with previous information in the cell - Forget gate
- Decide what new information to store in the cell - Input gate + *tanh* activation
- Filtered output - Output gate

Since the main idea is to let every step of RNN to discover larger collection of information it is linked to, LSTMs have many variations that can be customized for particular tasks.

Experiment

In this experiment, we create an LSTM model for recognizing handwritten digits from the MNIST database. The model uses a learning rate of 0.001 and 28 time steps. On training the network, we get the accuracy graph as shown in figure 3.

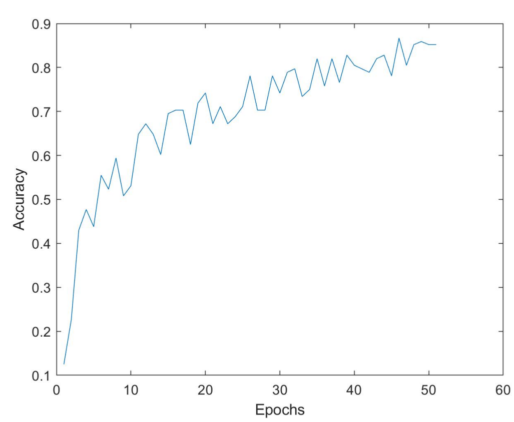

Figure 3: Plot of Accuracy vs Epoch for the trained RNN model

On testing with random images, the testing accuracy was found to be around 85% for the testbench.

References

[1] Wojciech Zaremba, Ilya Sutskever, and Oriol Vinyals. Recurrent neural network regularization. *arXiv preprint arXiv:1409.2329*, 2014.

[2] Andrej Karpathy. The unreasonable effectiveness of recurrent neural networks. *Andrej Karpathy blog*, 2015.

[3] Christopher Olah. Understanding lstm networks–colah's blog. *Colah. github. io*, 2015.