EEL6935
Programming Assignment 3
Abhinav Reddy Podduturi 1599-2633

GitHub Link: https://github.com/Abhinav-Reddy/BigDataAssignment3

Google Doc Link:
https://docs.google.com/document/d/1r1lf5l1Cz3kktEDQF49AB7DxSJpONDY_zpqQEVwFa3w/edit?usp=sharing

Architecture Overview:
Architecture consists of three main layers embedding layer, LSTM and sigmoid layer.
DNA sequence is converted into a embeddings and fed to LSTM. Output from LSTM is fed to sigmoid layer which outputs the probability of the input sequence being a TFBS.
LSTM was chosen mainly for their good performance on sequences and their ability to remember long term dependencies.

Preprocessing:
Each nucleotide was assigned a unique number between 1 and 4. One hot vector representation of each word was used as embeddings in the embeddings layer.

LSTM:
Embeddings from the embedding layer are given as input to the LSTM layer. This layer outputs a vector of length 8.

Sigmoid Layer:
Output vector from LSTM is fed to sigmoid layer which outputs the final probability.

Training:
Adadelta optimizer was used with a batch size of 8 and gradient clipping norm of 1.25.
Model was trained for 10 epochs with a training set of 1900.

Results:
Secured 8th position and a score of 84.615 on Kaggle.

Tried following enhancements:
Different things were tried to improve the performance of the system.

1) Different optimizers such as SGD, adanorm were used to check their impact on the performance on the system. No significant improvements were observed.
2) A probability of 0.4 was used as a cutoff to classify the sequences. This improved the F1 score by 1 percent.

3) Deep LSTM's with different number of layers from 2 to 4 were tried. None of them could offer any significant improvements.
4) Instead of considering one letter in the sequence as an individual event to LSTM, two adjacent letters were clubbed together and given as input to LSTM. Correspondingly one hot vector size was increased to 16. This gave an improvement of 2 percent. Also clubbing 3 or 4 letters didn't provide any improvement.

Screenshot: