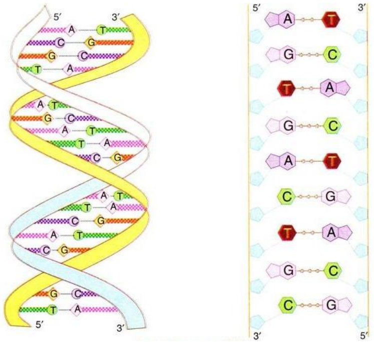
DNA Sequence Classification using LSTM Networks

Team:
Mehnaz Yunus 16CO124
Sharanya Kamath 16CO140

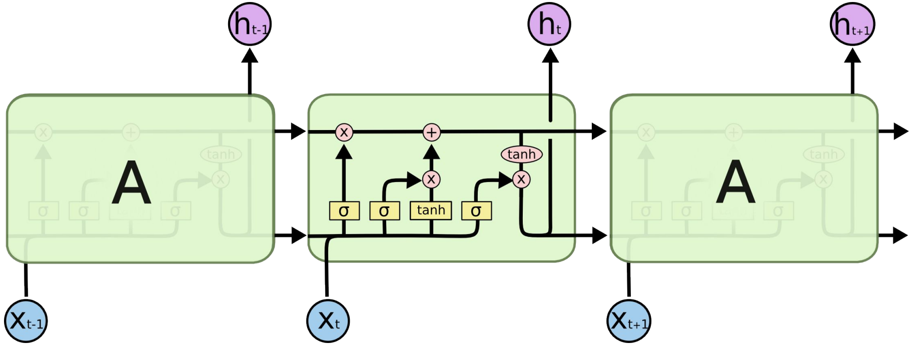
What is DNA Classification?

DNA classification is the problem of identifying the functionality of genes using only the sequence information (ATGTGT...) automatically.

<u>Methods already used:</u>
Artificial Neural Networks, Deep learning using CNN, RNN, SVM with k-means clustering.



What is LSTM?



Traditional neural networks require starting learning from scratch. RNNs address this issue. They are networks with loops in them, allowing information to persist. LSTMs are a special kind of RNNs explicitly designed to avoid long-term dependency problems. These networks can remember information for long periods of time.

Problem Statement and Dataset

Prediction problem:

Predicting next occuring character in genome sequence.

Classification Problem:

Detection of plasmid fragments in environmental samples - binary classification.

Dataset:

E. Coli complete genomes and plasmids from NCBI archives.

Literature Review

Analysis of DNA sequences is important in preventing the evolution of viruses, bacteria, and also used to diagnose disease during an early stage.From the existing work, the authors absorb some clustering algorithm and data analytics techniques like K-mean, k-mer, KNN, SVM, random forest correlation coefficient, and eigenvalue vector are used for predicting neurological disease.

References:

- Daniel Quang and Xiaohui Xie. Danq: a hybrid convolutional and recurrent deep neural network for quantifying the function of dna sequences. bioRxiv, page 032821, 2015.
- Qingda Zhou, Qingshan Jiang, Dan Wei, "A new method for classification in DNA sequence", Computer Science & Education (ICCSE) 2011 6th International Conference on, pp. 218-221, 2011.
- In Silico Detection and Typing of Plasmids, Alessandra Carattoli, a Ea Zankari

Character Level Genome Prediction

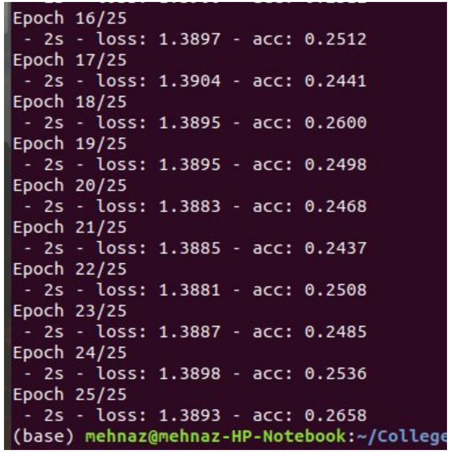
For this task, our baselines were the artificial genomes. Any model that does not perform as expected on these two genomes will not be used for future experiments.

Two genomes were prepared:

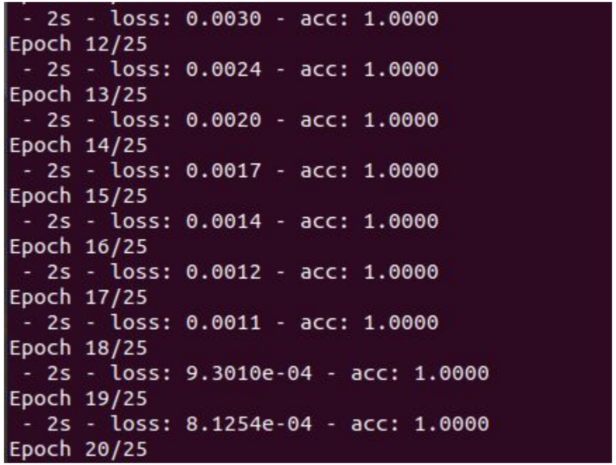
- A length-3000 repeating genome where the repeated unit is AGCTTGAGGC
- A length-3000 random genome

Character Level Genome Prediction

- LSTM was used for many to one sequence prediction
- Accuracy for repeating genome: 1.0 after 25 epochs
- Accuracy for random genome: 0.25 after 25 epochs

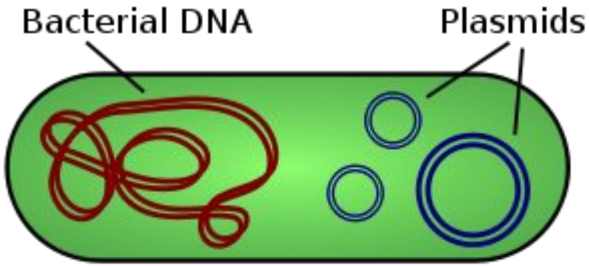


```
Epoch 16/25
 - 2s - loss: 1.3897 - acc: 0.2512
Epoch 17/25
 - 2s - loss: 1.3904 - acc: 0.2441
Epoch 18/25
 - 2s - loss: 1.3895 - acc: 0.2600
Epoch 19/25
 - 2s - loss: 1.3895 - acc: 0.2498
Epoch 20/25
 - 2s - loss: 1.3883 - acc: 0.2468
Epoch 21/25
 - 2s - loss: 1.3885 - acc: 0.2437
Epoch 22/25
 - 2s - loss: 1.3881 - acc: 0.2508
Epoch 23/25
 - 2s - loss: 1.3887 - acc: 0.2485
Epoch 24/25
 - 2s - loss: 1.3898 - acc: 0.2536
Epoch 25/25
 - 2s - loss: 1.3893 - acc: 0.2658
(base) mehnaz@mehnaz-HP-Notebook:~/College
```



```
 - 2s - loss: 0.0030 - acc: 1.0000
Epoch 12/25
 - 2s - loss: 0.0024 - acc: 1.0000
Epoch 13/25
 - 2s - loss: 0.0020 - acc: 1.0000
Epoch 14/25
 - 2s - loss: 0.0017 - acc: 1.0000
Epoch 15/25
 - 2s - loss: 0.0014 - acc: 1.0000
Epoch 16/25
 - 2s - loss: 0.0012 - acc: 1.0000
Epoch 17/25
 - 2s - loss: 0.0011 - acc: 1.0000
Epoch 18/25
 - 2s - loss: 9.3010e-04 - acc: 1.0000
Epoch 19/25
 - 2s - loss: 8.1254e-04 - acc: 1.0000
Epoch 20/25
```

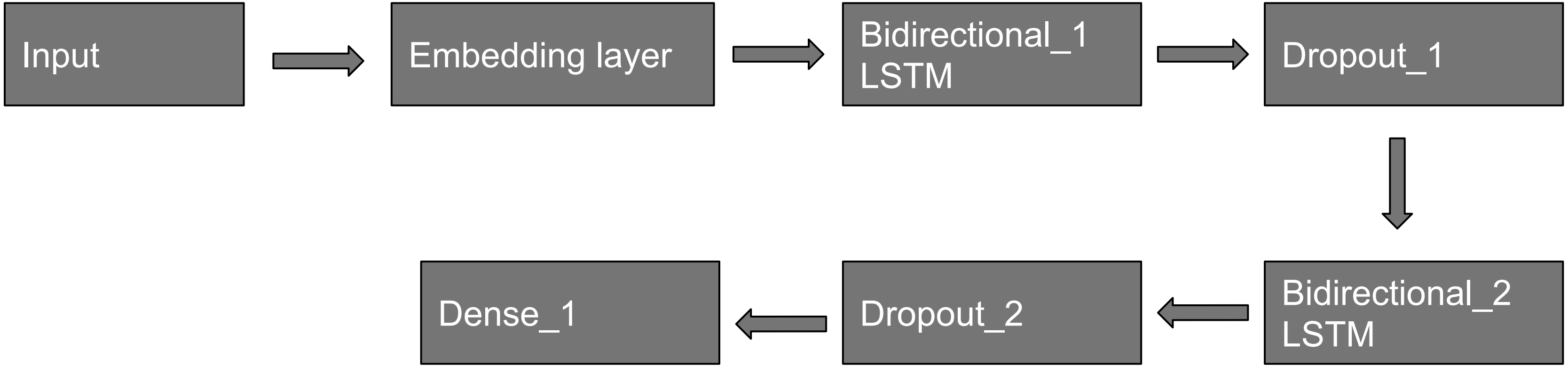
Structure of training data



Sequence = AGCTATGC....
Target label = {0:'chromosomal', 1:'plasmid'}

```
sequence,target
ACGTAGCT,1
ACCCTAAG,1
TCGTAACG,0
ACTGACCG,0
 ...
```

LSTM Model



Input

Embedding layer

Bidirectional_1 LSTM

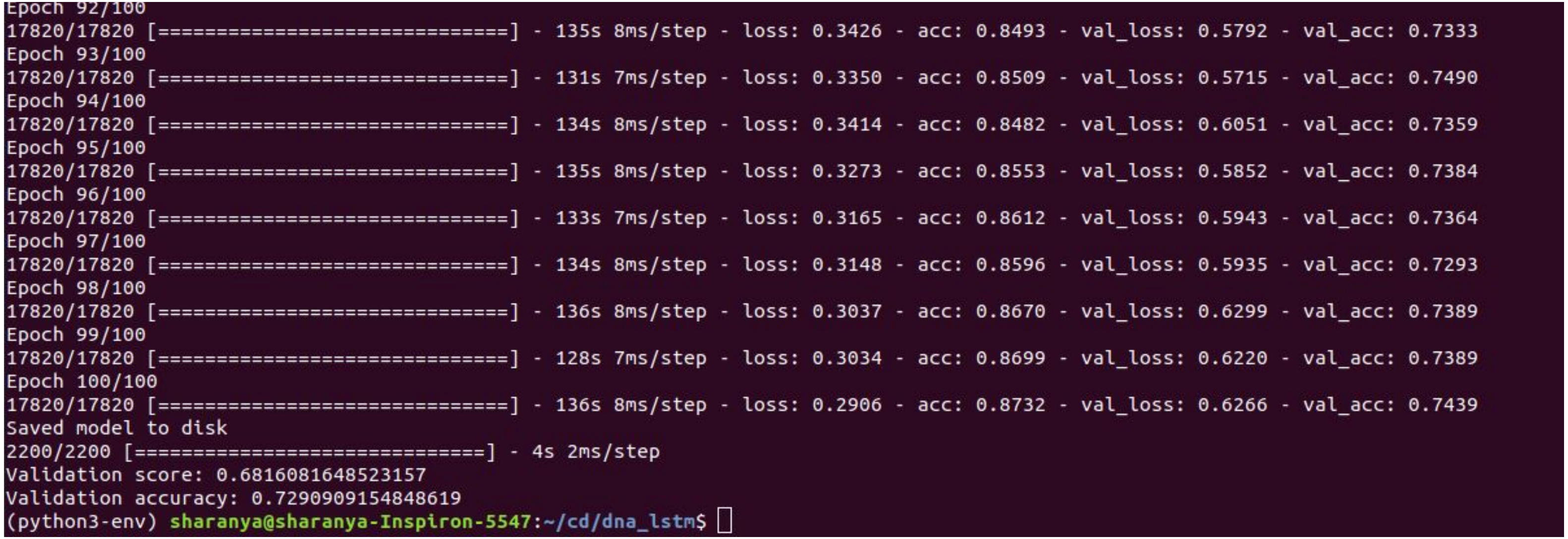
Dropout_1

Bidirectional_2 LSTM

Dropout_2

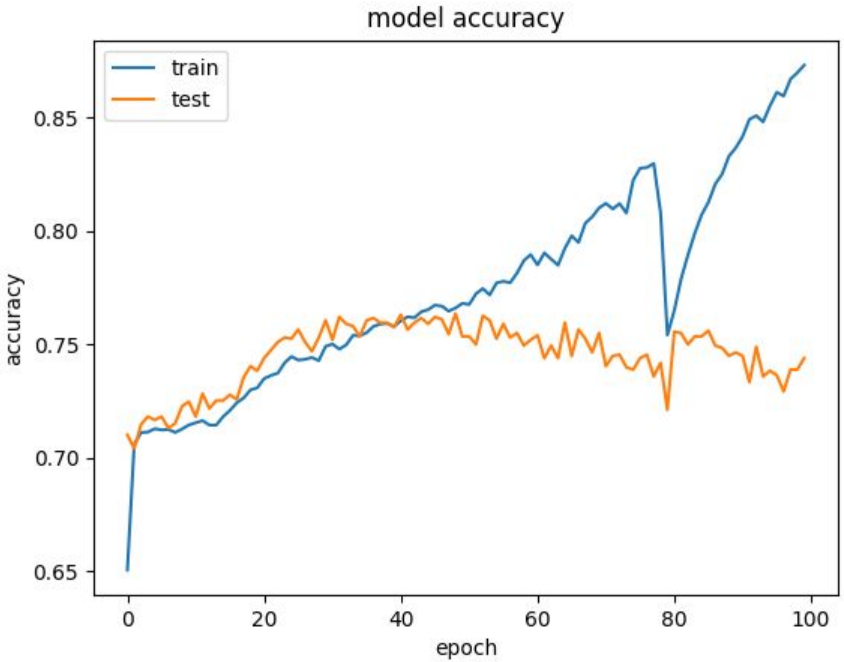
Dense_1

Binary Classification Problem results



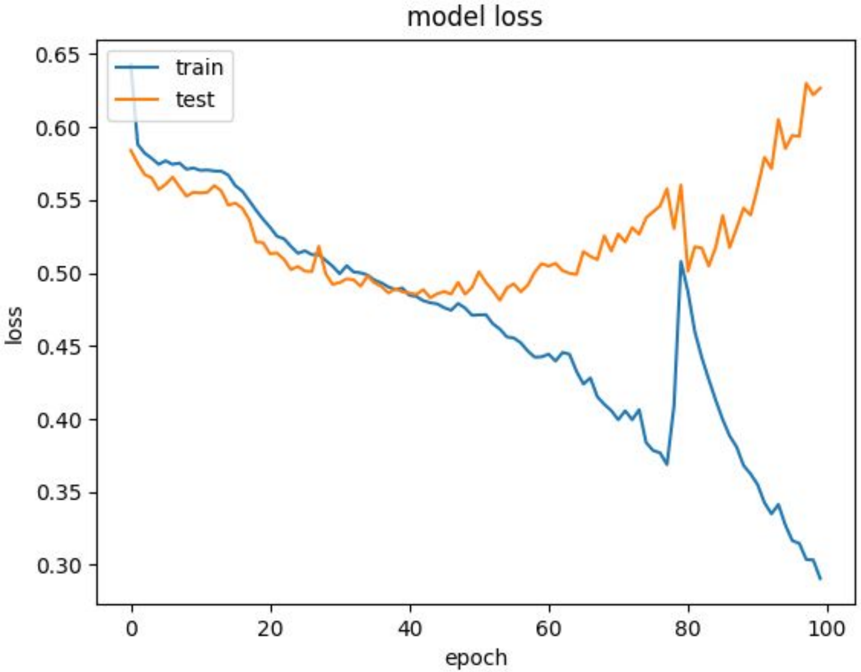
```
Epoch 92/100
17820/17820 [==============================] - 135s 8ms/step - loss: 0.3426 - acc: 0.8493 - val_loss: 0.5792 - val_acc: 0.7333
Epoch 93/100
17820/17820 [==============================] - 131s 7ms/step - loss: 0.3350 - acc: 0.8509 - val_loss: 0.5715 - val_acc: 0.7490
Epoch 94/100
17820/17820 [==============================] - 134s 8ms/step - loss: 0.3414 - acc: 0.8482 - val_loss: 0.6051 - val_acc: 0.7359
Epoch 95/100
17820/17820 [==============================] - 135s 8ms/step - loss: 0.3273 - acc: 0.8553 - val_loss: 0.5852 - val_acc: 0.7384
Epoch 96/100
17820/17820 [==============================] - 133s 7ms/step - loss: 0.3165 - acc: 0.8612 - val_loss: 0.5943 - val_acc: 0.7364
Epoch 97/100
17820/17820 [==============================] - 134s 8ms/step - loss: 0.3148 - acc: 0.8596 - val_loss: 0.5935 - val_acc: 0.7293
Epoch 98/100
17820/17820 [==============================] - 136s 8ms/step - loss: 0.3037 - acc: 0.8670 - val_loss: 0.6299 - val_acc: 0.7389
Epoch 99/100
17820/17820 [==============================] - 128s 7ms/step - loss: 0.3034 - acc: 0.8699 - val_loss: 0.6220 - val_acc: 0.7389
Epoch 100/100
17820/17820 [==============================] - 136s 8ms/step - loss: 0.2906 - acc: 0.8732 - val_loss: 0.6266 - val_acc: 0.7439
Saved model to disk
2200/2200 [==============================] - 4s 2ms/step
Validation score: 0.6816081648523157
Validation accuracy: 0.7290909154848619
(python3-env) sharanya@sharanya-Inspiron-5547:~/cd/dna_lstm$ 
```

Model Accuracy vs Epochs



Maximum accuracy: 0.87
Average accuracy: 0.729

Cross Entropy Loss vs Epochs



Comparison with previous work done

- kmer based SVM: SVM score: 0.871625
- Random Forest approach: Random Forest score: 0.791423

Future Work to be done

Improving accuracy of binary classification

Extending prediction to multiple classes