Assignment 2 Report

Arun Kodnani(ak6384) & Rahul Keswani(ryk248)

Hyperparameter Tuning

Vocab Size

From our understanding of vocab size = n, will chose the n most frequent words from the corpus and will only keep these words in the review. Therefore, we restricted vocab size in the range of 3000-5000 words.

Batch Size

We tried with different batch size combinations such as 32, 64, 128 and 256. We wanted to keep the batch size smaller so that there are a greater number of weight updates per epoch. To compensate for the computational time, we reduced the epochs because it was giving us good results.

Max Length

Since most of the review were either of 500 length of more, so we kept the max length of 500.

Padding

After trying the below architecture configurations, we had following observations:

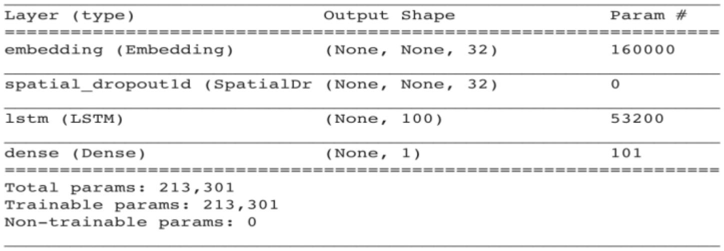
- If post padding is kept then Bidirectional models worked better as compared to simple sequence models.

Our reason for this observation is that since simple recurrent network compute across time, the padding at the end of the review would dilute the output of the actual reviews which is being passed as the h(t-1) component.

Architecture Experimented

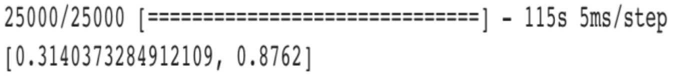
LSTM:

Configuration



```
Layer (type)                 Output Shape              Param #
=================================================================
embedding (Embedding)        (None, None, 32)          160000
_________________________________________________________________
spatial_dropout1d (SpatialDr (None, None, 32)          0
_________________________________________________________________
lstm (LSTM)                  (None, 100)               53200
_________________________________________________________________
dense (Dense)                (None, 1)                 101
=================================================================
Total params: 213,301
Trainable params: 213,301
Non-trainable params: 0
_________________________________________________________________
```

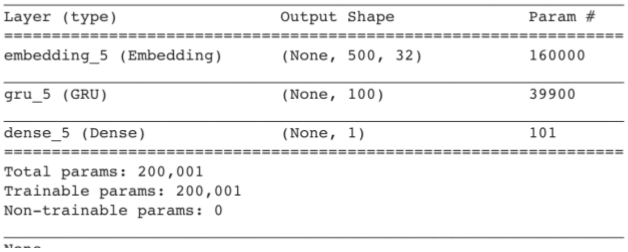
Accuracy – 87.62%



```
25000/25000 [==============================] - 115s 5ms/step
[0.3140373284912109, 0.8762]
```

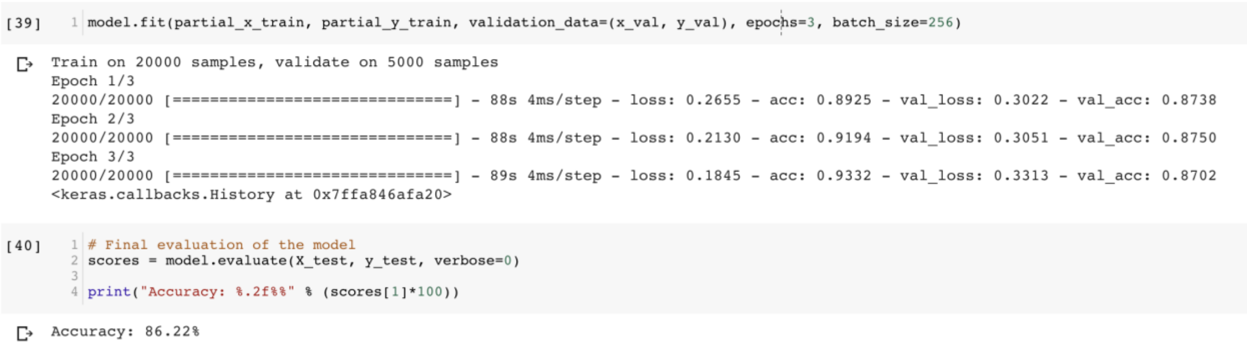
GRU

Configuration



```
Layer (type)                 Output Shape              Param #   
=================================================================
embedding_5 (Embedding)      (None, 500, 32)           160000    
_________________________________________________________________
gru_5 (GRU)                  (None, 100)               39900     
_________________________________________________________________
dense_5 (Dense)              (None, 1)                 101       
=================================================================
Total params: 200,001
Trainable params: 200,001
Non-trainable params: 0
_________________________________________________________________
```

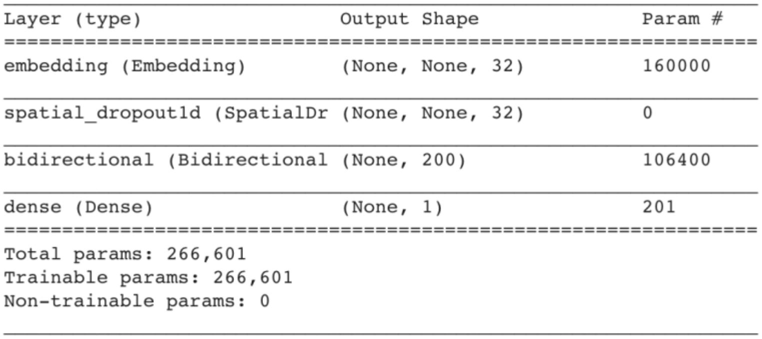
Accuracy – 86.22%



```
[39]  model.fit(partial_x_train, partial_y_train, validation_data=(x_val, y_val), epochs=3, batch_size=256)

      Train on 20000 samples, validate on 5000 samples
      Epoch 1/3
      20000/20000 [==============================] - 88s 4ms/step - loss: 0.2655 - acc: 0.8925 - val_loss: 0.3022 - val_acc: 0.8738
      Epoch 2/3
      20000/20000 [==============================] - 88s 4ms/step - loss: 0.2130 - acc: 0.9194 - val_loss: 0.3051 - val_acc: 0.8750
      Epoch 3/3
      20000/20000 [==============================] - 89s 4ms/step - loss: 0.1845 - acc: 0.9332 - val_loss: 0.3313 - val_acc: 0.8702
      <keras.callbacks.History at 0x7ffa846afa20>

[40]  # Final evaluation of the model
      scores = model.evaluate(X_test, y_test, verbose=0)

      print("Accuracy: %.2f%%" % (scores[1]*100))

      Accuracy: 86.22%
```

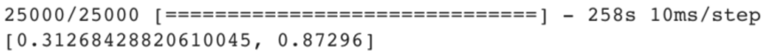
Bi-Directional LSTM

Configuration



```
Layer (type)                 Output Shape              Param #   
=================================================================
embedding (Embedding)        (None, None, 32)          160000    
_________________________________________________________________
spatial_dropout1d (SpatialDr (None, None, 32)          0         
_________________________________________________________________
bidirectional (Bidirectional (None, 200)               106400    
_________________________________________________________________
dense (Dense)                (None, 1)                 201       
=================================================================
Total params: 266,601
Trainable params: 266,601
Non-trainable params: 0
_________________________________________________________________
```

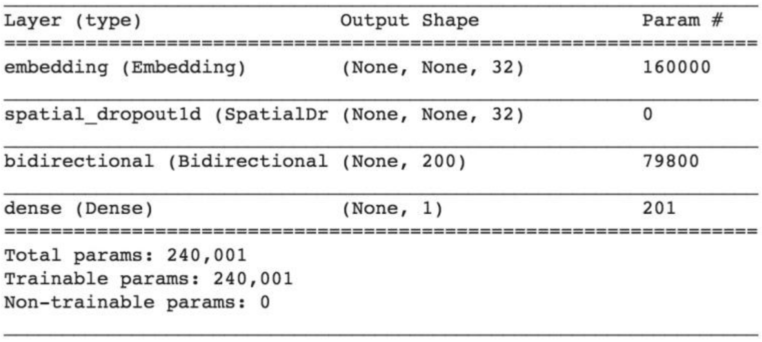
Accuracy – 87.29



```
25000/25000 [==============================] - 258s 10ms/step
[0.31268428820610045, 0.87296]
```

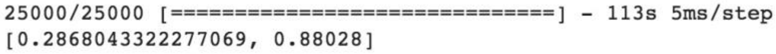
Bi-Directional GRU

Configuration



```
_________________________________________________________________
Layer (type)                 Output Shape              Param #   
=================================================================
embedding (Embedding)        (None, None, 32)          160000    
_________________________________________________________________
spatial_dropout1d (SpatialDr (None, None, 32)          0         
_________________________________________________________________
bidirectional (Bidirectional (None, 200)               79800     
_________________________________________________________________
dense (Dense)                (None, 1)                 201       
=================================================================
Total params: 240,001
Trainable params: 240,001
Non-trainable params: 0
_________________________________________________________________
```

Accuracy – 88.02

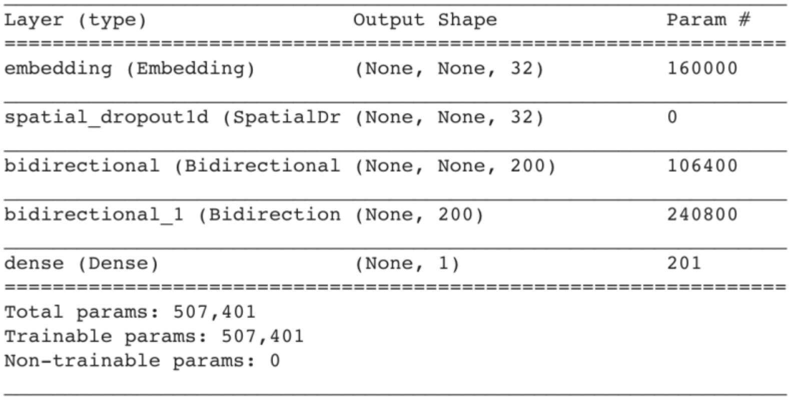


```
25000/25000 [==============================] - 113s 5ms/step
[0.2868043322277069, 0.88028]
```

NOTE: We got our best accuracy with this model (Bi-Directional GRU), but, unfortunately our Tensorboard didn't connect to the python notebook which had the code for this. Hence, we are not selecting this as our final architecture.

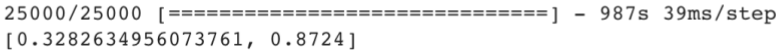
Stacked Bi-Directional LSTM

Configuration



```
_________________________________________________________________
Layer (type)                 Output Shape              Param #   
=================================================================
embedding (Embedding)        (None, None, 32)          160000    
_________________________________________________________________
spatial_dropout1d (SpatialDr (None, None, 32)          0         
_________________________________________________________________
bidirectional (Bidirectional (None, None, 200)         106400    
_________________________________________________________________
bidirectional_1 (Bidirection (None, 200)               240800    
_________________________________________________________________
dense (Dense)                (None, 1)                 201       
=================================================================
Total params: 507,401
Trainable params: 507,401
Non-trainable params: 0
_________________________________________________________________
```

Accuracy – 87.24



```
25000/25000 [==============================] - 987s 39ms/step
[0.3282634956073761, 0.8724]
```

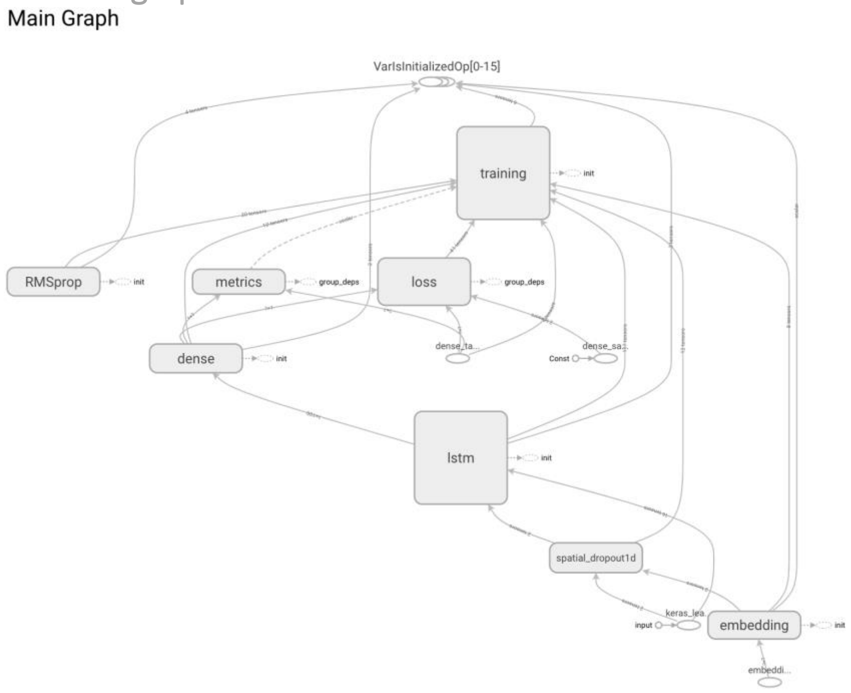
Final Architecture

Amongst all these architectures the base version of LSTM is giving us the best performance with an accuracy of 87.62%.

Configuration

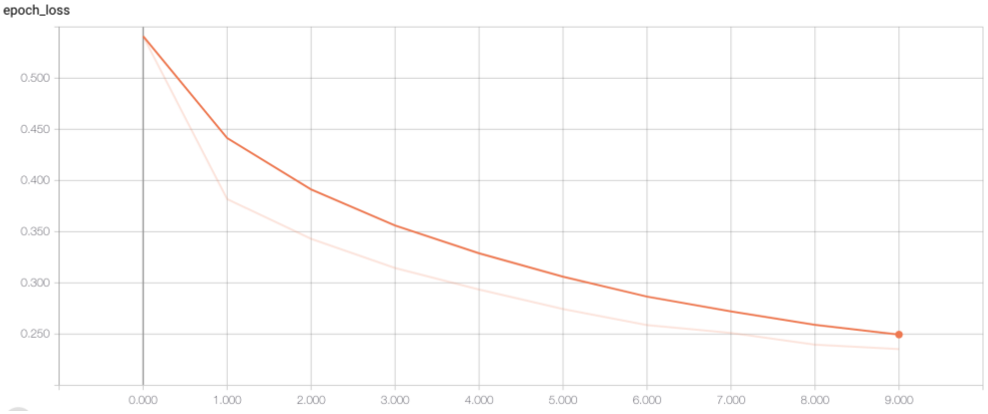
1. vocab_size = 5000
2. batch_size = 64
3. embedding_size = 32
4. max_len = 500
5. optimizer='rmsprop'
6. loss='binary_crossentropy'

Architecture graph

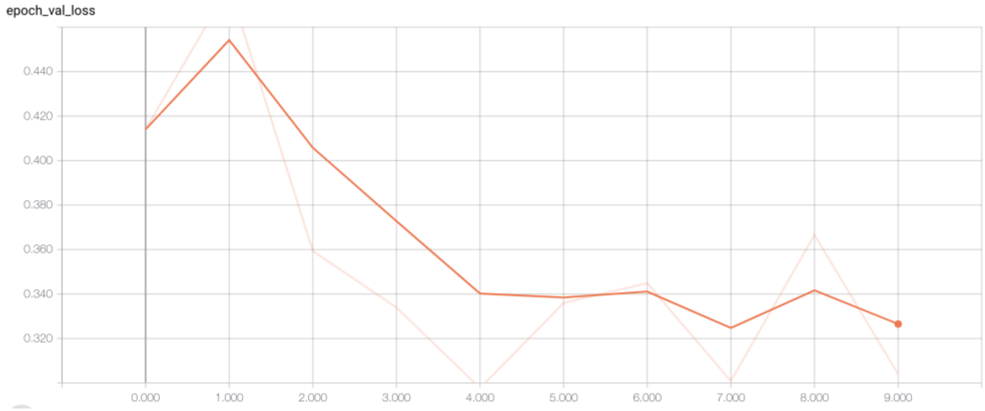


Loss plots

Epoch Loss

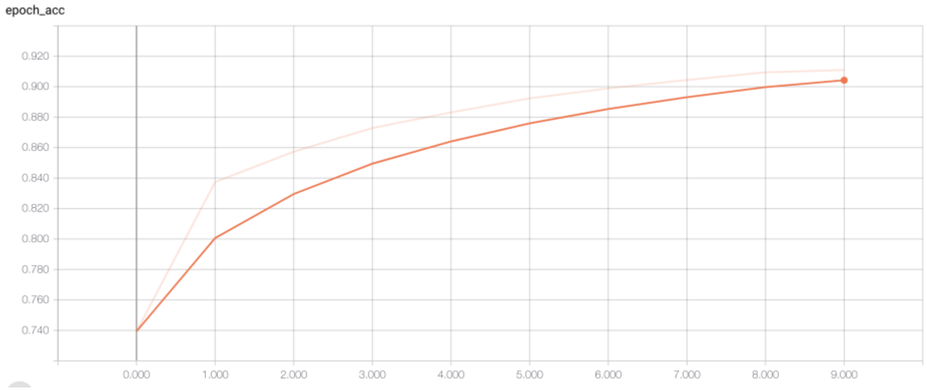


Validation Loss

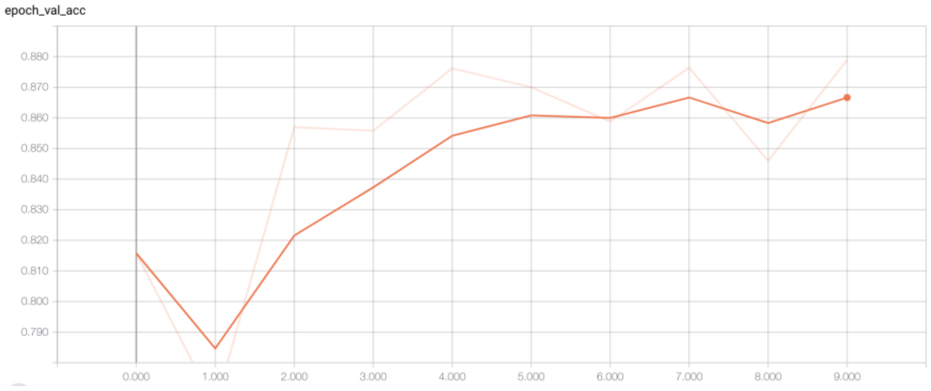


Accuracy Plots

Epoch Accuracy



Validation Accuracy


epoch_val_acc