IMDB REVIEW CLASSIFICATION REPORT

- **A report explaining all the preprocessing steps you made, the design choices for your network, the training and testing accuracies, and any comments you have on the output.**

PreProcessing:-
A preprocessing function called as text cleaner was written. The function lowers all the text. All the special characters are removed. All numbers are removed thus keeping only alphabets. The stop words are removed, thus unnecessary and most common words are removed thereby retaining only the most meaningful.

Design Choices :-
Initially, to convert the sentences into vector **tokenizer** from keras has been used. The vectors are formed by using texts to sequence. By calculating the mean length of the sentences for the entire corpus, it was found that the length was around 235 words. By trying different length, it was found that the length of 450 performed the best. Then the sentences were padded with zeros for the sentences less than 450 words.

The model consists of an input **embedding layer**, 2 hidden **LSTM layers** and an **output layer**.

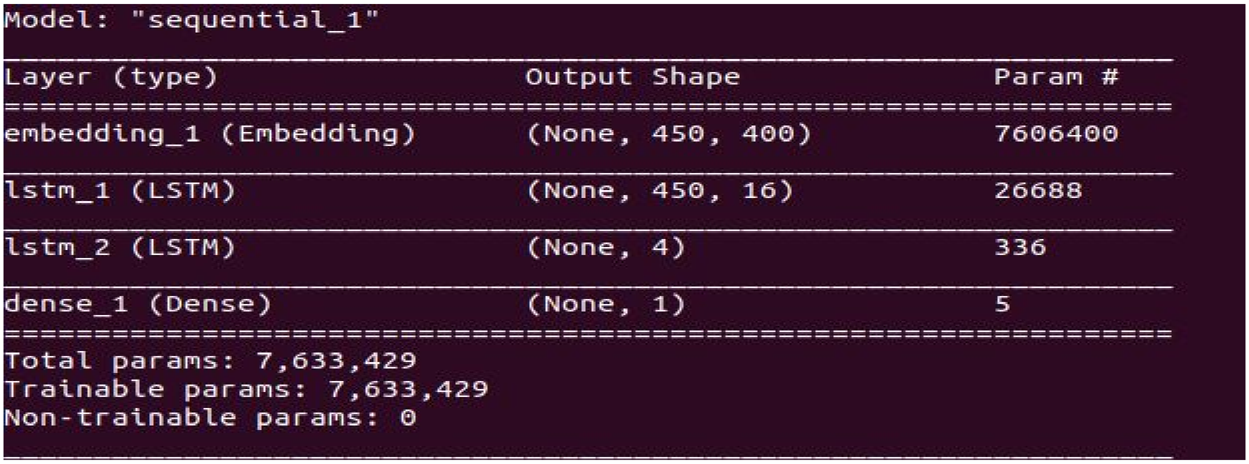
Inorder to get the embedding matrix, I have trained a word2vec model from gensim library using the words from my training. The dimension length for the w2v model was given as 400 and the min count value as 10. The embedding vectors obtained from the w2v model are given as the weights in the embedding layer.

LSTM layers with different hidden layers and hidden units were tested. It was seen that for this review dataset more hidden units and more hidden layers performed poorly. Finally by testing different hidden units and layers, 2 hidden layers with units as 16 and 4 a testing accuracy of **84.27%** was obtained.

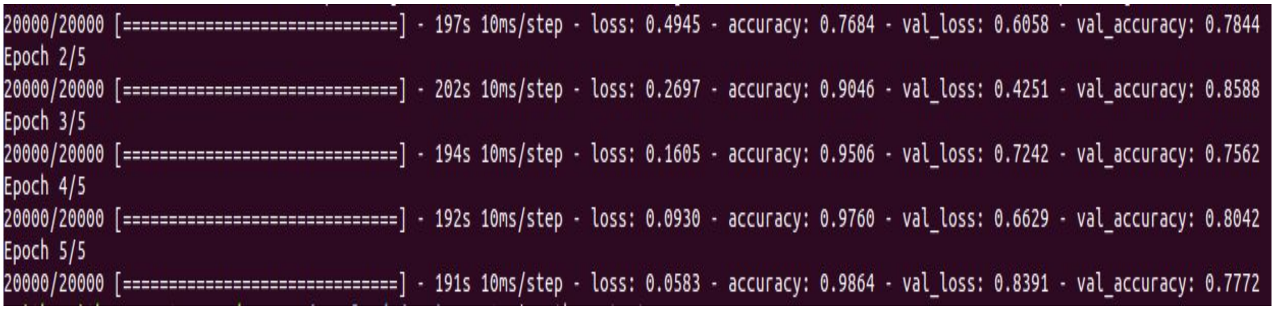
The output layer has 1 unit and the activation function used is sigmoid since it is a binary class classification. The loss used is binary cross entropy loss functions. The optimizer used was adam. I tried a different learning rate and found the default learning rate of Adam which is 0.001 trained faster and performed better. Callbacks was used to save the model when it performs better.

TRAINING AND TESTING ACCURACIES
From the obtained model it is noted that for 5 epochs, the model trained well and at the end of 5 epochs the training accuracy was around 98.64% and the validation accuracy is around **77%** .

Model summary

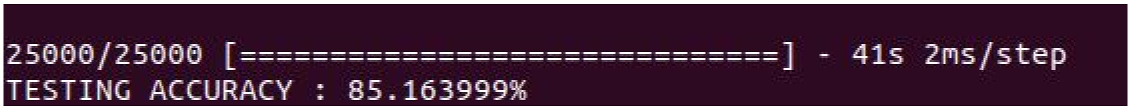


```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding_1 (Embedding)      (None, 450, 400)          7606400
_________________________________________________________________
lstm_1 (LSTM)                (None, 450, 16)           26688
_________________________________________________________________
lstm_2 (LSTM)                (None, 4)                 336
_________________________________________________________________
dense_1 (Dense)              (None, 1)                 5
=================================================================
Total params: 7,633,429
Trainable params: 7,633,429
Non-trainable params: 0
_________________________________________________________________
```



```
20000/20000 [==============================] - 197s 10ms/step - loss: 0.4945 - accuracy: 0.7684 - val_loss: 0.6058 - val_accuracy: 0.7844
Epoch 2/5
20000/20000 [==============================] - 202s 10ms/step - loss: 0.2697 - accuracy: 0.9046 - val_loss: 0.4251 - val_accuracy: 0.8588
Epoch 3/5
20000/20000 [==============================] - 194s 10ms/step - loss: 0.1605 - accuracy: 0.9506 - val_loss: 0.7242 - val_accuracy: 0.7562
Epoch 4/5
20000/20000 [==============================] - 192s 10ms/step - loss: 0.0930 - accuracy: 0.9760 - val_loss: 0.6629 - val_accuracy: 0.8042
Epoch 5/5
20000/20000 [==============================] - 191s 10ms/step - loss: 0.0583 - accuracy: 0.9864 - val_loss: 0.8391 - val_accuracy: 0.7772
```

TESTING ACCURACIES



```
25000/25000 [==============================] - 41s 2ms/step
TESTING ACCURACY : 85.163999%
```

It was noted that the testing accuracy at the end of 5 epochs was around **84%** but since call backs were used the best model was saved and an accuracy of **85.16%**.