

PMDS603P Deep Learning Lab Experiment 11

October 2025

1 Work to do today

Note: Make a single PDF file of the work you are doing in a Jupyter notebook. Upload with the proper format. Please mention your name and roll no properly with the Experiment number on the first page of your submission.

Today we will look at how we can use Bi-directional LSTM (BI-LSTM) and Bi-GRU model to perform sentiment analysis using imdb movie dataset.

Question 1: Try to load the imdb dataset and import necessary modules and classes like LSTM and GRU, Also from tensorflow.keras.layers import Bidirectional wrapper as well.

```
from tensorflow.keras.layers import Embedding, LSTM, Dense, Bidirectional
```

Next find the maximum and minimum length of the reviews and fix the length of each review we are going to process as 500 (for good results, we need to take an appropriate one). Generally, the reviews vary from 70 to 2600 words.

Pad the sequences as done on the previous day.

Prepare the validation, training, and testing sets. There are 25000 reviews in training. Maybe you can use some 1000 reviews in validation set.

Fix the embedding length as 100 for each word and create two bi-directional LSTM layers.

```
BiLSTM_model = Sequential()
BiLSTM_model.add(Embedding(input_dim=vocab_size, output_dim=embd_len, input_length=max_words))
BiLSTM_model.add(Bidirectional(LSTM(128, return_sequences=True)))
BiLSTM_model.add(Bidirectional(LSTM(64, return_sequences=False)))
BiLSTM_model.add(Dense(1, activation='sigmoid'))
```

Compile and fit the Bi-LSTM model and see the results obtained. Print the performance metrics also for the same. Use yesterdays work for reference for today

Question 2: Next, build an Bi-GRU model and try the same problem and compare the outputs of the models.

Questions 3: Think how you can create a simple model with RNN to predict the next word once you give a sentence to the model. Try to create one such model that can do this task. Use the same imdb dataset for the task. (Hint: Try to first prepare the sequences for training just like we did in gold price prediction, Sequences in which we have say 10 words

as inputs and the next word as output. And we can plan like we can in our model the final layer with vocal size you have fixed that many number of neurons so that you can run with a softmax function to predict the probability of next word and train the model accordingly)