Sentiment Analysis

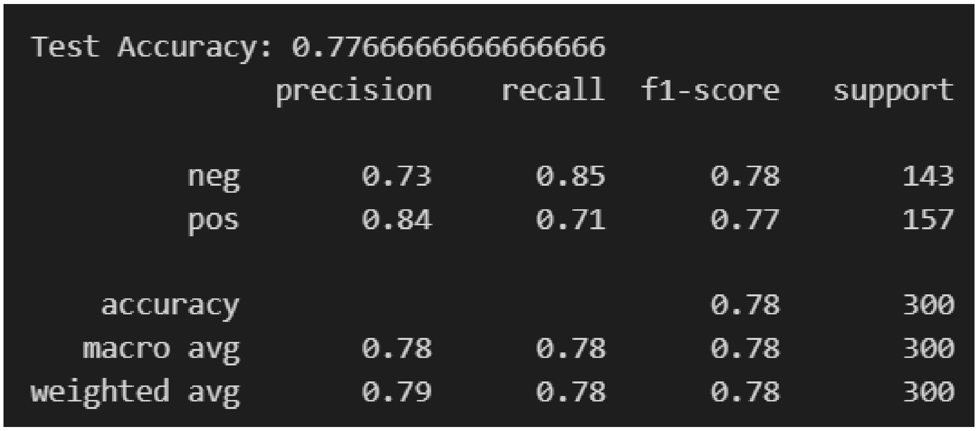
POS Tagger

Implemented Viterbi Algorithm, which takes input a tokenized sentence, and outputs their best part of speech tags (state where the product of emission probability and transmission probability is maximized for each word)

Vanilla Sentiment Analyzer (Baseline)

Loaded movie_reviews corpus from nltk. Extracted the text and labels from the corpus and split the data into train-validation-test in 70:15:15 ratio. Generated the sentence embeddings using Tf-idf vectorizer from sklearn and trained the classifier using Naive Bayes using the features.

Classification Report:



```
Test Accuracy: 0.7766666666666666
              precision    recall  f1-score   support

         neg       0.73      0.85      0.78       143
         pos       0.84      0.71      0.77       157

    accuracy                           0.78       300
   macro avg       0.78      0.78      0.78       300
weighted avg       0.79      0.78      0.78       300
```

Improved Sentiment Analyzer

Loaded movie_reviews corpus from nltk. Extracted the text and labels from the corpus and split the data into train-validation-test in 70:15:15 ratio.
Extracted the POS features for the data using POS tagger implemented in first part, sentence embeddings using tf-idf vectorizer similar way as in baseline model.
Integrated these features by concatenating the tf-idf vector for each sentence with total count of each POS tag in the sentence and the mean tf-idf for each tag. And then trained the classifier on these new features.
Since the dataset is too large, Viterbi algorithm takes long time to get the tags of the data. So, I couldn’t get the classification report for the combined Analyzer.

Sudeeksha Reddy Pala
20CS30053