Natural Language Processing (CS60075)

Assignment 2: News Topic Classification

Name: Suthar Utsav Kaushikkumar

Roll No.: 23CS60R49

TASK 1(A): News Topic Classification using Word2Vec + NN model

Challanges Faced During Implementation:

1) The dataset contains HTML Tags. So, I have created regular expressions to remove the HTML Tags.
2) After Processing, I was getting irrelevant tokens like n't, n's, 's etc. So, I used simple_preprocess by gensim.utils.to remove these tokens.
3) Initially I created Custom Word2Vec Model to get Word Embeddings. As the train dataset size is very low, the word embeddings that I was getting were not able to capture very good contextual dependency. So, I used pretrained "word2vec-google-news-300" to get the word Embeddings. Now I am getting 85% test Accuracy.
4) Neural Networks are very sensitive to Hyperparameters. Initially I was getting around 30% Test Accuracy. But after trying various hyperparameters, I am getting good test Accuracy.
5) Model was getting overfitted as train accuracy was increasing but test accuracy was decreasing. So, I have added a dropout layer with dropout value 0.2.

Hyperparameters used for the Task:

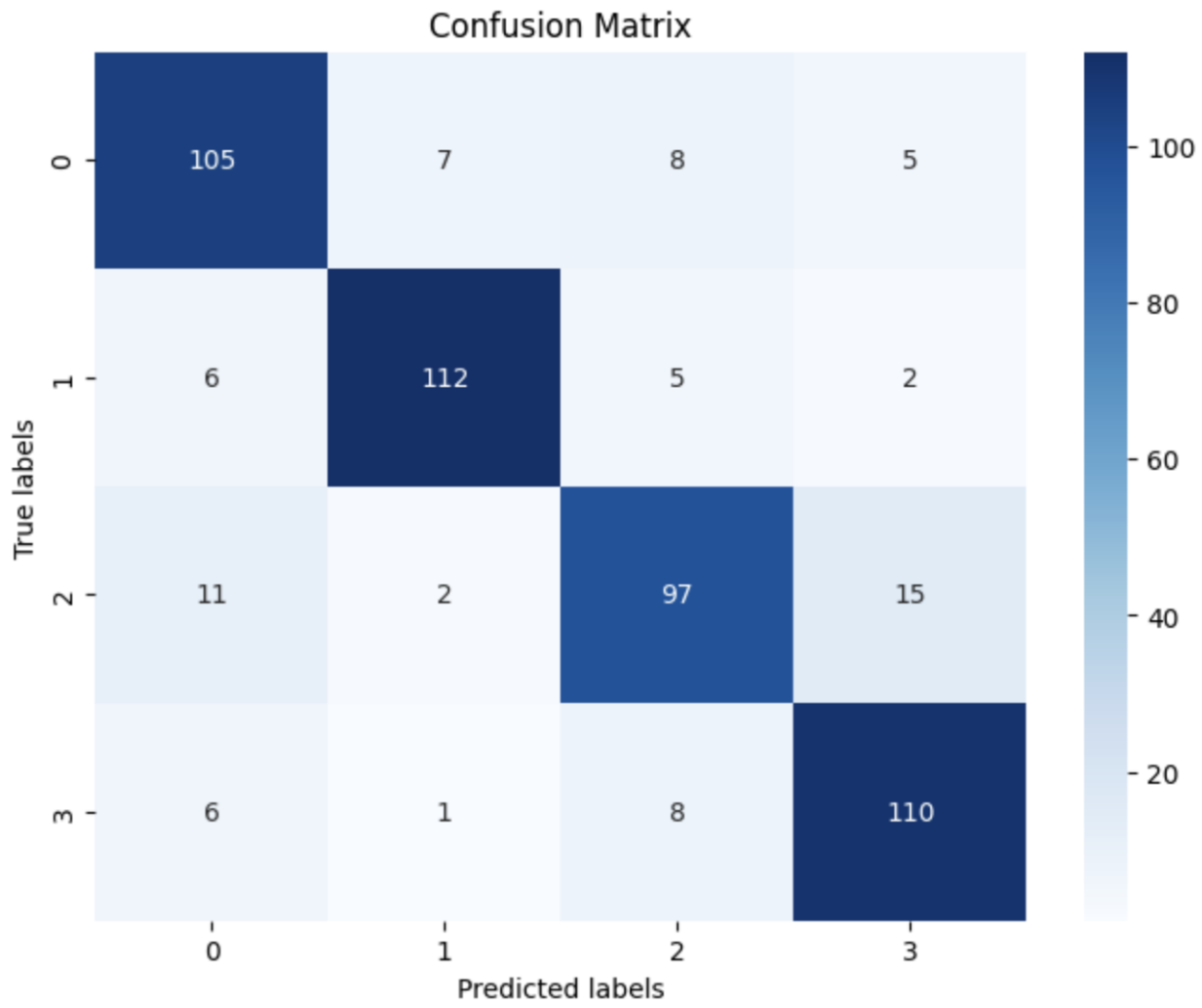
1) **Vocab Size:** I am filtering all the words which have frequency less than 3 in the document and more than 95% frequency in the document. So, my current vocab size is **3238.**
2) Hidden Size (for Layer 1): 256
3) Hidden Size (for Layer 2): 256
4) Learning Rate: 0.0001
5) Number of Epochs: 20
6) Loss Criteria: Cross Entropy Loss
7) Optimizer: Adam Optimizer
8) Number of Layers: 2
9) Activation Function: Relu

10) Dropout Rate: 0.2

Performance of the Word2Vec + NN Model:

- Training Accuracy: 98.39%
- Validation Accuracy: 86.00%
- Testing Accuracy: 84.80%
- Test F1 Score: 0.848

Confusion Matrix for Test Data



Test Classification Report

Classification Report:

	precision	recall	f1-score	support
0	0.82	0.84	0.83	125
1	0.92	0.90	0.91	125
2	0.82	0.78	0.80	125
3	0.83	0.88	0.86	125
accuracy			0.85	500
macro avg	0.85	0.85	0.85	500
weighted avg	0.85	0.85	0.85	500

TASK 1(B): News Topic Classification using RNN and LSTM model

Challanges Faced During Implementation:

1) Initially I set the max length of the sequence as Max Length Sentence in the document. But the model was not performing well, and it was taking too much time to train. So, I changed Max Length Sequence to the average sequence length of all the sentences in the train.
2) RNN and LSTM models were not performing better than NN Model because of bad Hyper Parameters in the RNN and LSTM Model. I have used Python Octuna Library to find Hyper Parameters for reference. Then I tested few Parameters and got the best accuracy.
3) To Control the overfitting, I have added a Dropout layer in both RNN and LSTM.
4) I was getting different accuracies with respect to different vocab size. So instead, I used entire Vocabulary without filtering unlike I filtered Vocabulary.
5) While predicting, I was getting different accuracies with the same trained model because I missed to set model to evaluation mode.

Hyperparameters used for the Task:

1) Vocab Size: 10442
2) Max Sequence Length: 25

Bidirectional RNN Model Hyperparameters:

3) Loss Function: Cross Entropy Loss
4) Optimizer: Adam
5) Learning Rate: 0.001
6) Hidden dimension: 32
7) Number of RNN Layers: 1
8) Dropout: 0.5
9) Number of Epochs: 10

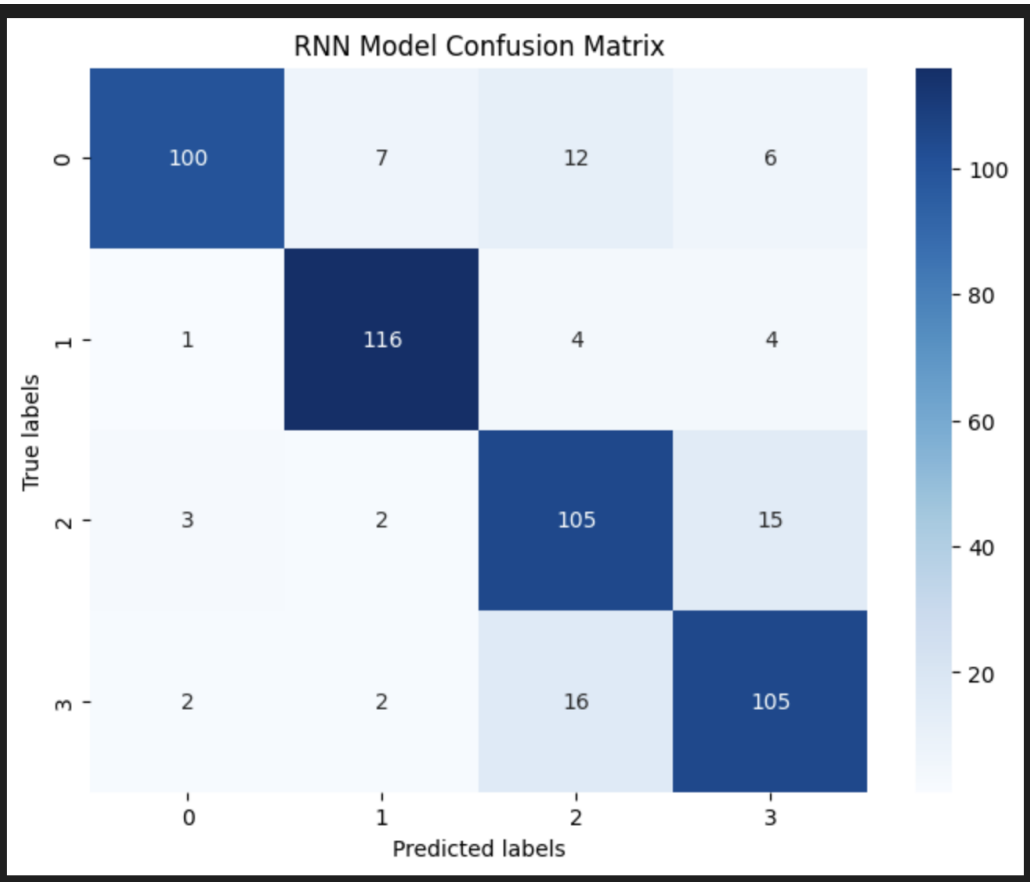
LSTM Model Hyperparameters:

10) Loss Function: Cross Entropy Loss
11) Optimizer: Adam
12) Hidden dimension: 32
13) Learning rate: 0.01
14) Number of LSTM Layers: 1
15) Number of Epochs: 10

16) Dropout: 0.5

Performance of the RNN Model:

- RNN Model Train, Test, Val Accuracies
- Training Accuracy: 88.67%
- Validation Accuracy: 89.00%
- Testing Accuracy: 85.20%
- Test F1 Score: 0.852

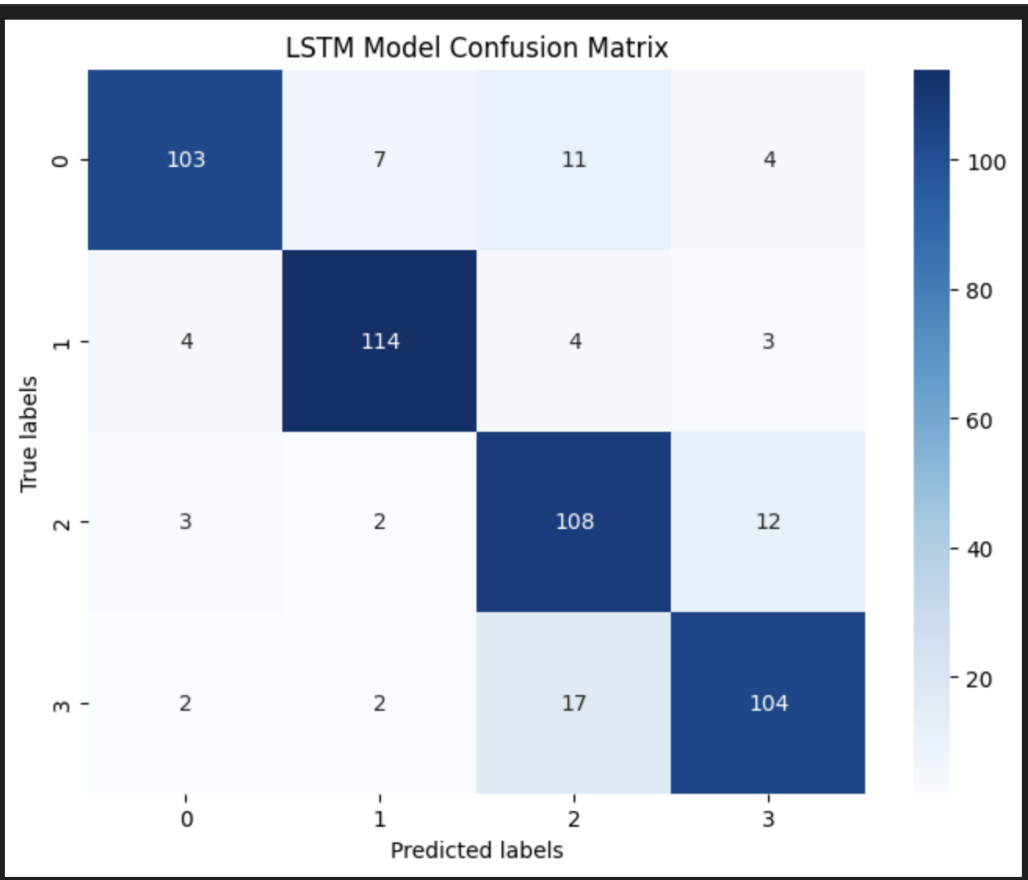


Classification Report for RNN Model:

	precision	recall	f1-score	support
0	0.94	0.80	0.87	125
1	0.91	0.93	0.92	125
2	0.77	0.84	0.80	125
3	0.81	0.84	0.82	125
accuracy			0.85	500
macro avg	0.86	0.85	0.85	500
weighted avg	0.86	0.85	0.85	500

Performance of the LSTM Model:

- LSTM Model Accuracies
- Training Accuracy: 90.17%
- Validation Accuracy: 87.50%
- Testing Accuracy: 85.80%
- Test F1 Score: 0.858



Classification Report for LSTM Model:

	precision	recall	f1-score	support
0	0.92	0.82	0.87	125
1	0.91	0.91	0.91	125
2	0.77	0.86	0.82	125
3	0.85	0.83	0.84	125
accuracy			0.86	500
macro avg	0.86	0.86	0.86	500
weighted avg	0.86	0.86	0.86	500