Natural Language Processing Bootcamp

Assignment 3

Santhosh BV

Preprocessing

Each review in the dataset is converted into lower case and unwanted expressions other than the letters and spaces are removed. The processed review is then lemmatized after removing the stop words. Each sentence is word tokenized and these words are used to create a dictionary which numbers them starting from index 1. Based on the indexing of words belonging to a review a sequence of vector representation is created. Based on the length of 90% of the sentences a max length of the sequence of vectors is determined to be 242 and sentences larger are post truncated and smaller ones are padded with zeros. This is done to ensure that all the input vectors are of the same length as required by the LSTM. The sentiments are converted into labels with negative to 0 and positive to 1. Train test split of 80-20 is performed after shuffling.

Model architecture

To create dense embedding vector for each input an embedding layer of the size of vocabulary and embedding dimension 128 is created. This is passed through a single LSTM of hidden dimension 128. The output of the LSTM is connected to a dense or fully connected neural network which produces the final activation. It then applies sigmoid to convert the final activation between 0 and 1. The drop out with probability 0.5 is applied to the neuron of the last layer to prevent over dependency on a single neuron.

Training

The training dataset is converted into batches of size 64 using a data loader and then fed into the LSTM model. Binary Cross entropy is used as the loss function, and the loss at the end of each batch is used to modify the parameters using Adam optimizer. The model is trained on the dataset for 5 times. The train accuracy, precision, recall, F1 score, train losses are recorded at the end of each epoch to get a gauge on the level of training done.

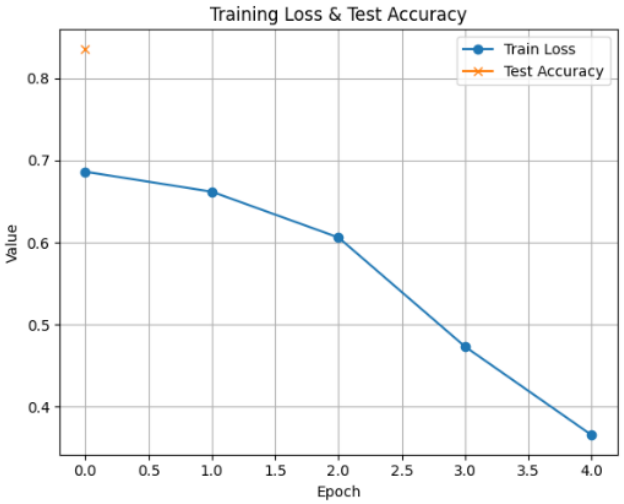
Testing

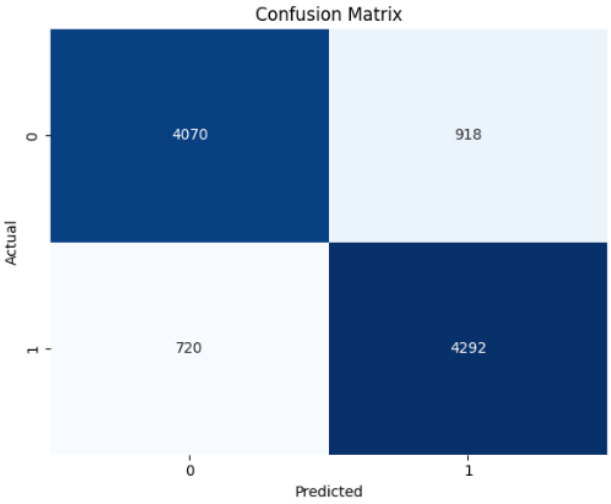
The test dataset is then used for evaluating the performance of the model on unseen data. Sigmoid output of greater than 0.5 are classified as class 1 or positive sentiments and accuracy, precision, recall, F1 score, confusion matrix is calculated.

Result Analysis

Epoch	Train Loss	Accuracy	Precision	Recall	F1 Score
1	0.6863	0.5216	0.5216	0.5216	0.5216
2	0.6619	0.5633	0.5633	0.5633	0.5633
3	0.6061	0.6826	0.6826	0.6826	0.6826
4	0.4730	0.8022	0.8022	0.8022	0.8022
5	0.3651	0.8604	0.8604	0.8604	0.8604

Clearly the training through multiple epochs have decreased the losses from 0.68 to 0.36. The ratio of correct positive predictions to the total positive predictions indicated by the precision, as well as the ration of correct positive predictions to the original number of positive predictions indicated by the recall continuously increase through iterations. This is due to updating of weights based on the errors generated.





Accuracy	Precision	Recall	F1 score
0.8362	0.8362	0.8362	0.8362

From the training loss, the losses have not settled to zero, so there is scope for further improvement through increased epochs. The test results are quite decent for a single layer LSTM with F1 score of 0.8362. The model learns both the classes equally as it can be inferred from the confusion matrix. This is mainly attributed to the balance in the dataset. The performance can be further increased by increasing the layers of LSTM , embedding dimension and hidden dimension. The increase in dimension helps the network to represent more information, but at the cost of higher computational expense and higher size of the training data.