

The engine for sentiment analysis is developed using the dataset available for training. The three techniques namely the Naïve Bayes machine learning algorithm, convolutional neural network and the recurrent neural network using LSTM principle were utilised as a binary classifier individually to predict the sentiment for the user review data.

The models did not actually perform well when the text pre-processing step was not handled. For this reason, the technique of text pre-processing is made utilised before feeding the input to the constructed models. For the naïve bayes the optimization was not carried at a great extent. However, the CNN and LSTM were optimised under rigorous conditions and the respective sequential model were deployed after experimenting with the performance metrics. The epochs utilised in this case was only 3 (because the API demonstration takes a lot of time when it is otherwise), which must be usually in the range from 30-50 for this prediction engine to work very precisely.

Although CNN is popularly used for image processing tasks. I tried using the technique for this scenario to compare it with the usual recurrent neural network. It is working fine because of the considered layers and their associated concepts. A major priority is given for kernel size in convolutional layer and the choice of max-pool layer.

Below image populates the model performance metrics:

THE METRICS FOR NAIVE BAYES CLASSIFIER:

	precision	recall	f1-score	support
0	0.94	0.94	0.94	2771
1	0.74	0.75	0.75	692
accuracy			0.90	3463
macro avg	0.84	0.84	0.84	3463
weighted avg	0.90	0.90	0.90	3463

THE METRICS FOR CNN CLASSIFIER:

	loss	accuracy	val_loss	val_accuracy
0	0.397175	0.824462	0.272987	0.892001
1	0.187712	0.928943	0.227260	0.909616
2	0.102438	0.963481	0.246673	0.906151

THE METRICS FOR LSTM CLASSIFIER:

	loss	accuracy	val_loss	val_accuracy
0	0.403052	0.837336	0.230674	0.907017
1	0.168591	0.934390	0.206804	0.920878
2	0.096254	0.966081	0.244344	0.912792

Further enhancement could be to visualize the model behaviour at API deployment.