FINAL REPORT ASSIGNMENT 3:

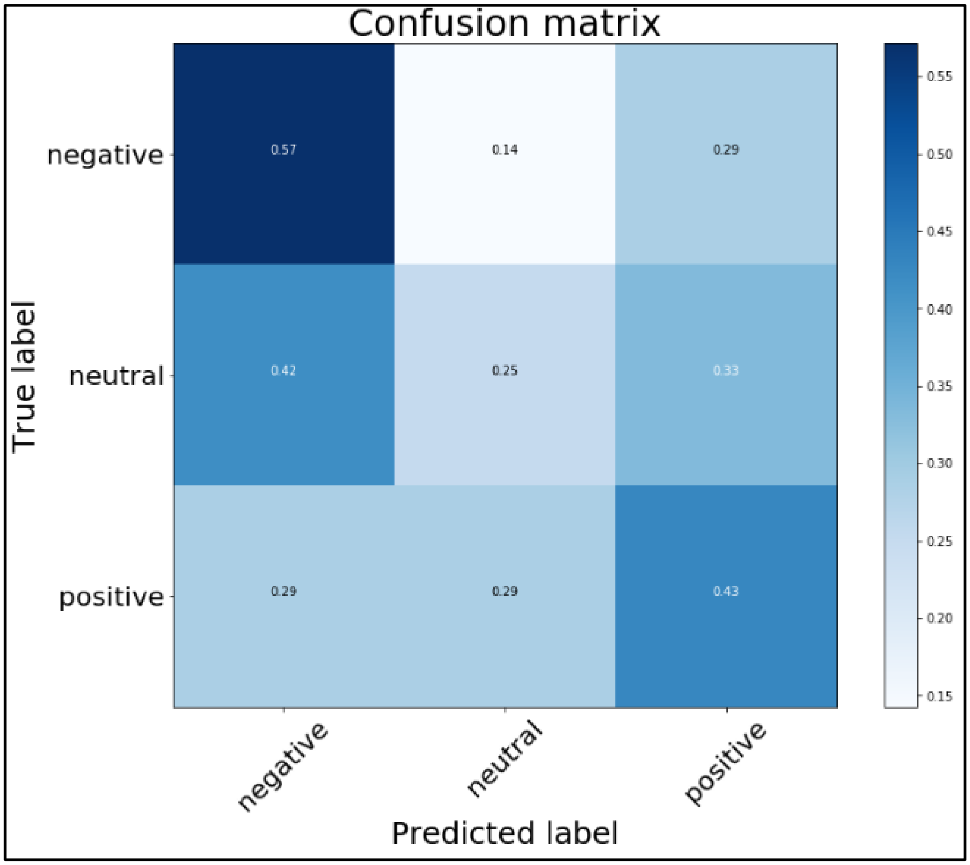
EXPERIMENT 1:

BAG OF WORDS (BOW MODEL)

batch_size = 30
epochs = 15
loss='binary_crossentropy',
optimizer='rmsprop'

Train Accuracy: 0.65
Test Accuracy: 0.60

Confusion Matrix:

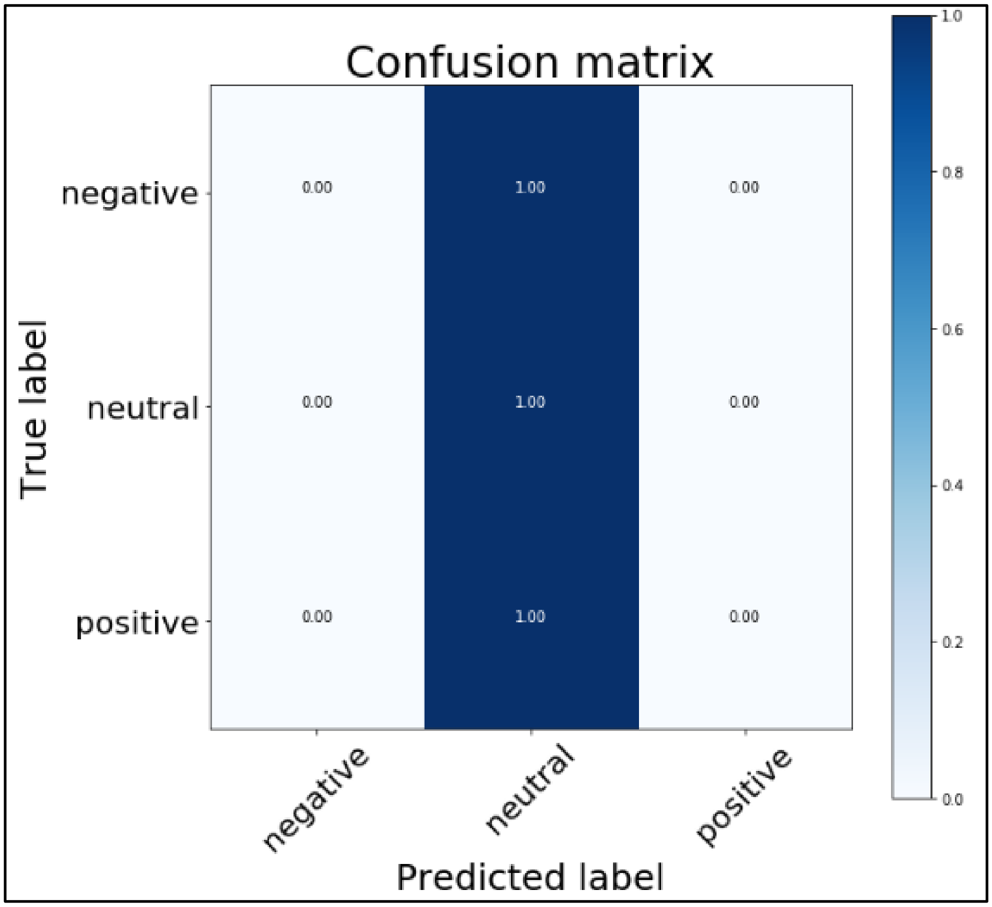


RNN MODEL:

epochs=10,
batch_size=128
optimizer='rmsprop',
loss='binary_crossentropy'

Train Accuracy: 0.67
Test Accuracy: 0.66

Confusion Matrix:

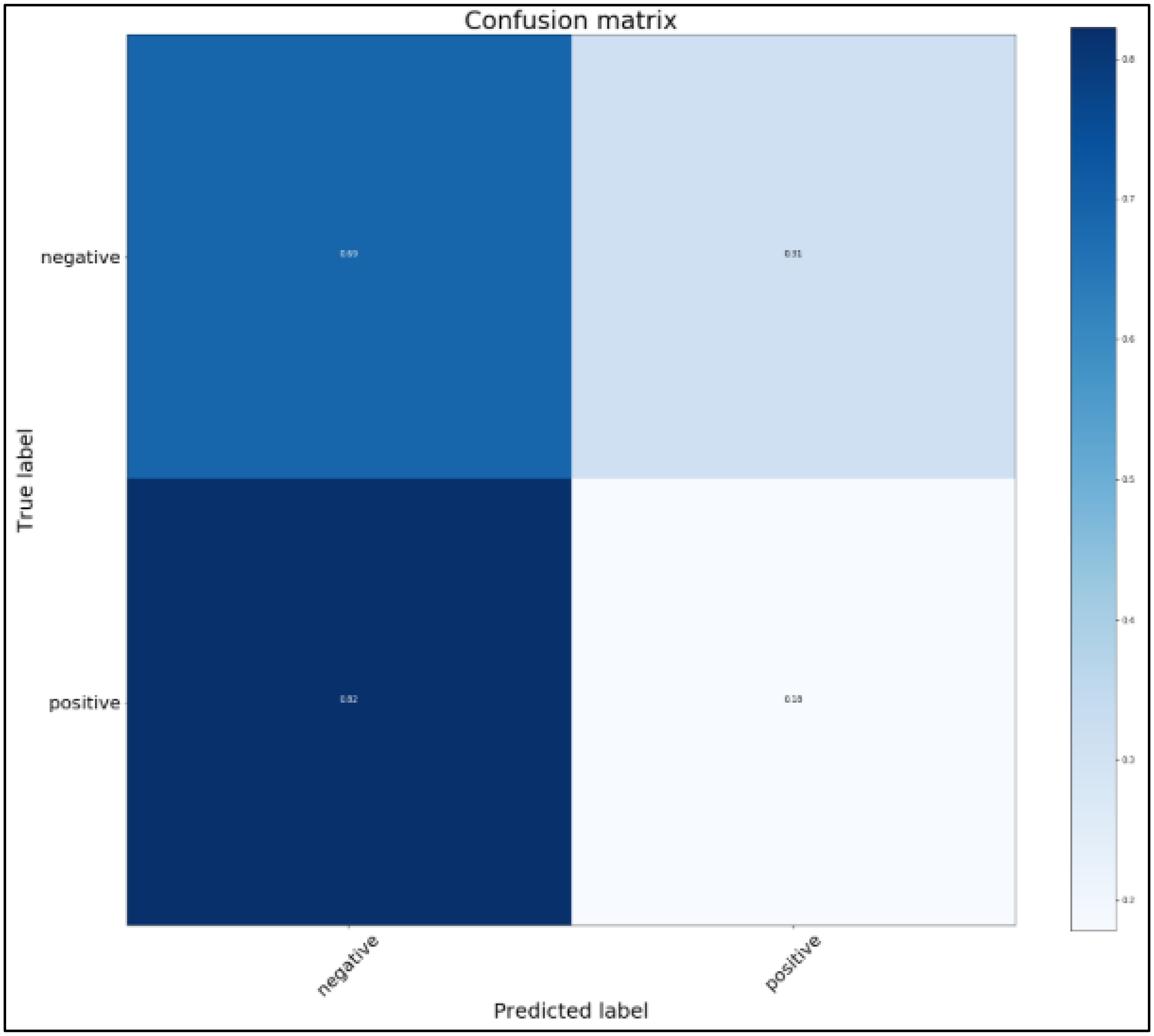


<u>**GLOVE MODEL:**</u>

epochs = 5,
batch_size = 10
optimizer = "rmsprop",
loss = "binary_crossentropy"

Train Accuracy: 0.99
Test Accuracy: 0.37

Confusion Matrix:



Inference: After running the following models we obtained the highest accuracy for the RNN model of 66% which performed better than both the bag of words and the Glove model

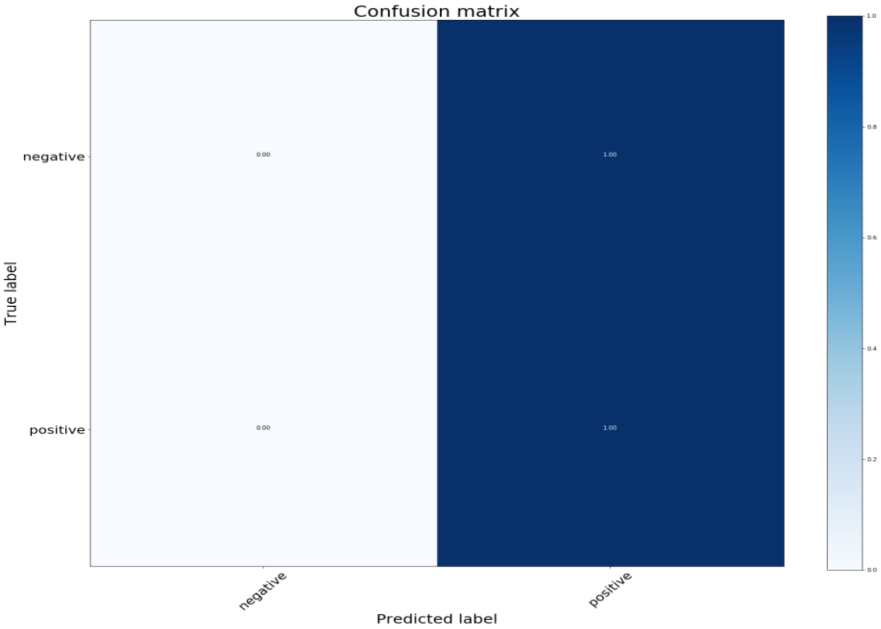
<u>EXPERIMENT 2:</u> TRANSFER LEARNING

In this part, we trained the models on IMDB datasets and tested the models on call transcripts. Below are the results we got after testing call transcripts on the models trained with IMDB dataset.

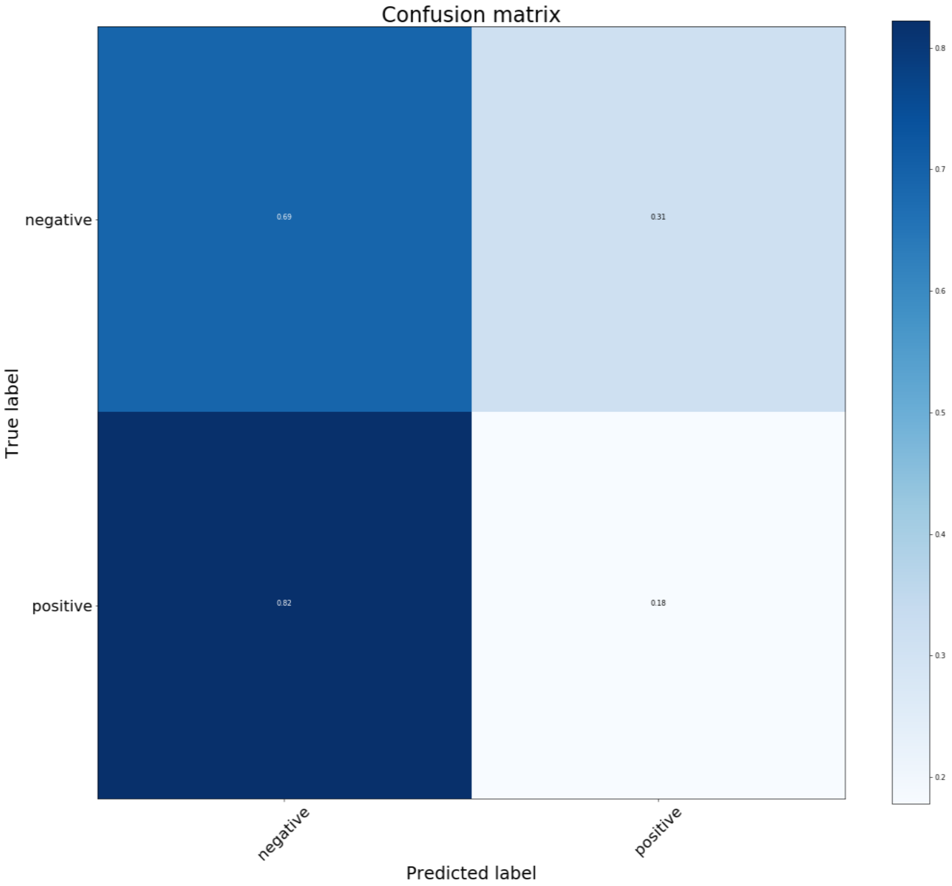
Model	IMDB Train Accuracy	IMDB Test Accuracy	Finance Data Train Accuracy	Finance Data Test Accuracy
BOW	0.501	0.50	0.50	0.4876
GLOVE	0.94	0.72	0.9562	0.3784
RNN	0.88	0.89	0.904	0.6216

Confusion Matrix

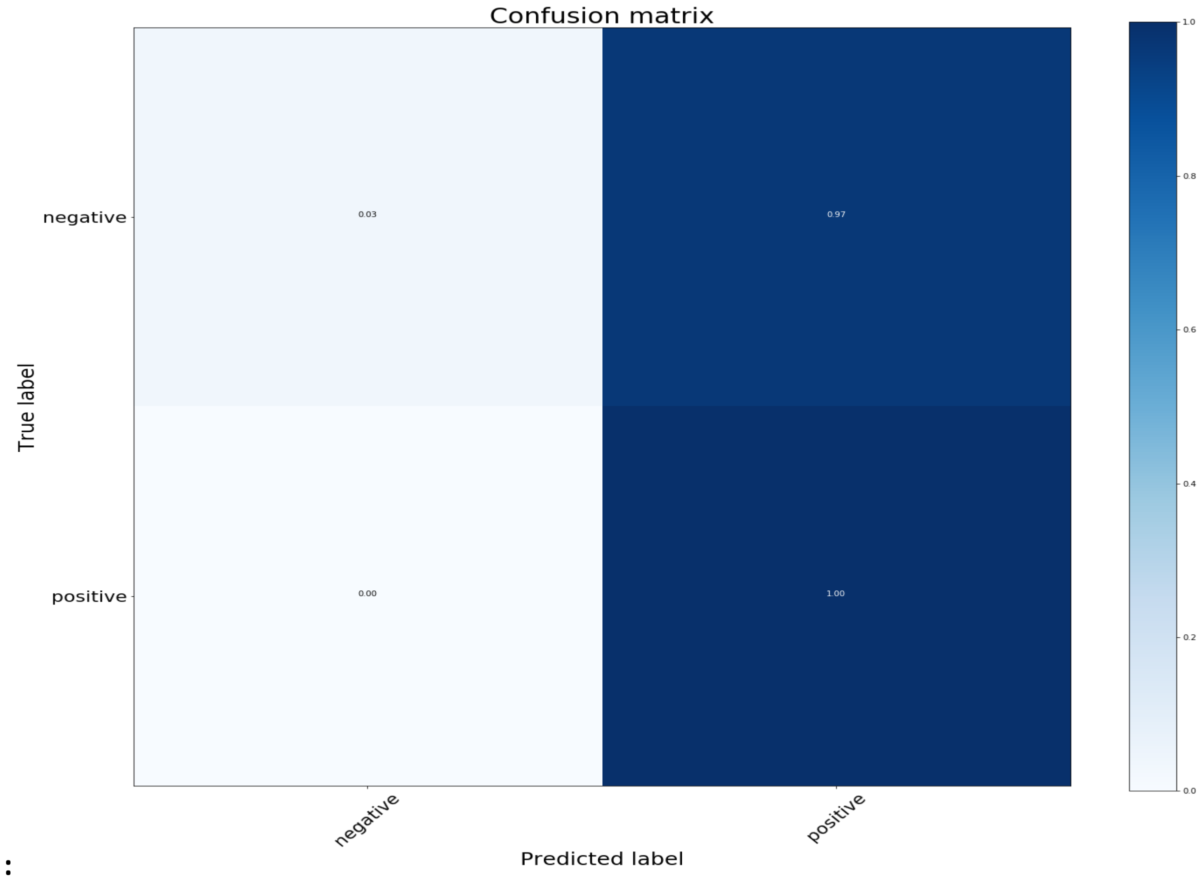
Bag of words



Glove:


Confusion matrix
True label
negative
positive
0.69
0.31
0.82
0.18
Predicted label
negative
positive
0.8
0.7
0.6
0.5
0.4
0.3
0.2

RNN:



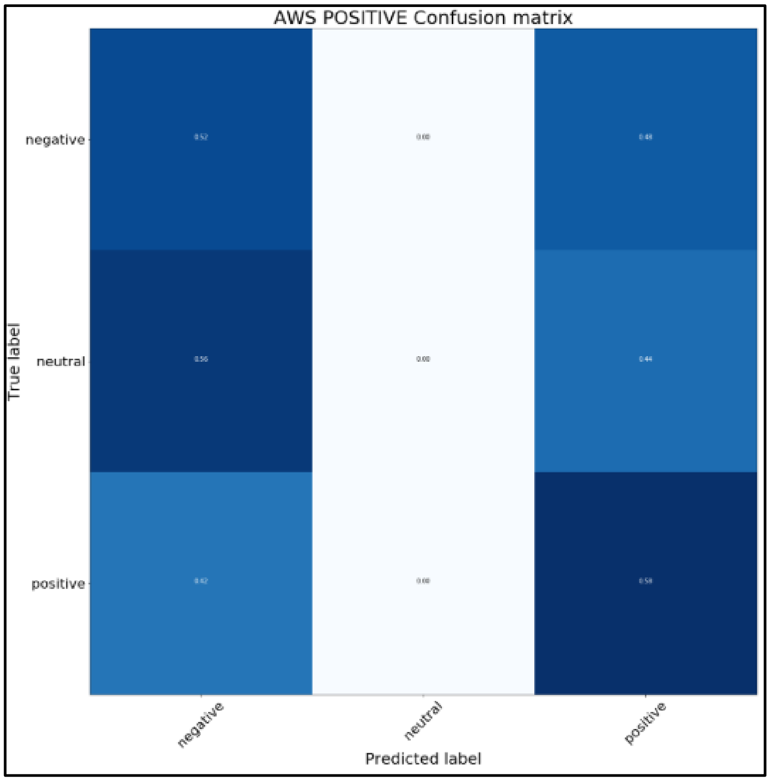
:

Inference: The bow model gave a test accuracy of 49% for the finance dataset while the IMDB test accuracy was 50%. The Glove model gave a test accuracy of 85% and performed much better than the IMDB dataset which had a test accuracy of 72%. Lastly, the RNN gave a test accuracy of 62.16% for the finance dataset which is same as the test accuracy obtained for the IMDB dataset for the RNN

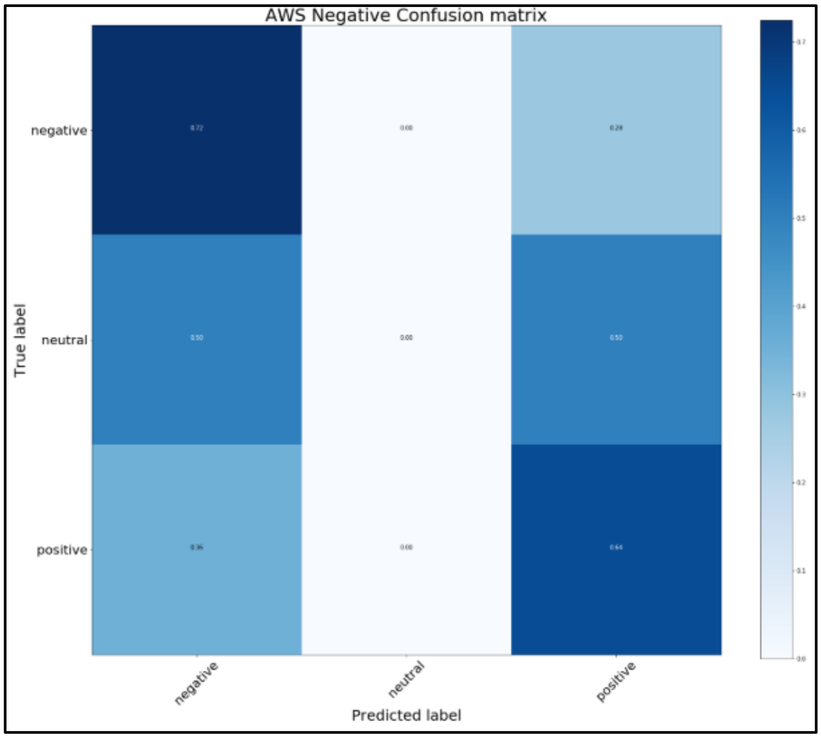
EXPERIMENT 3: USING API'S

In this part, we calculated the sentiment scores using different APIs such as IBM Watson, AWS, GCP and Azure. Below are the results for these:

AWS:

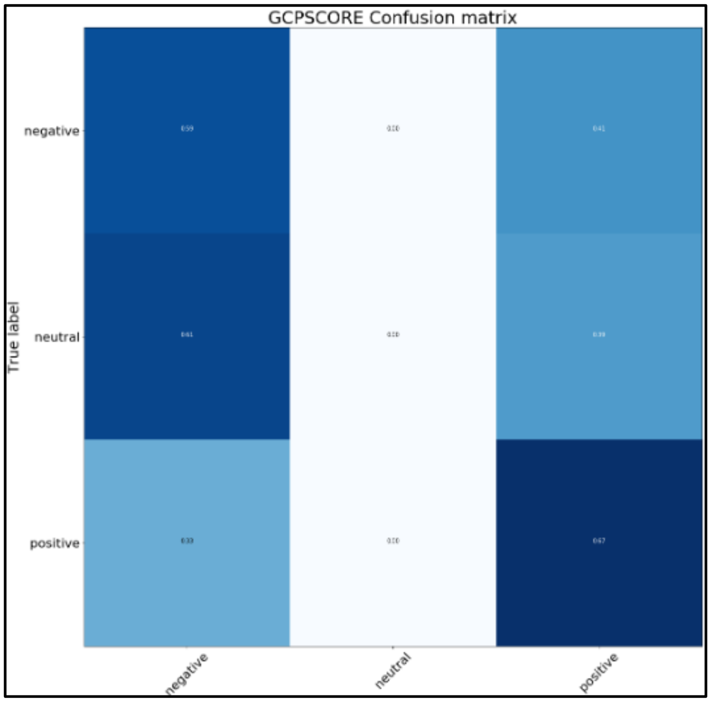


Accuracy: 0.32



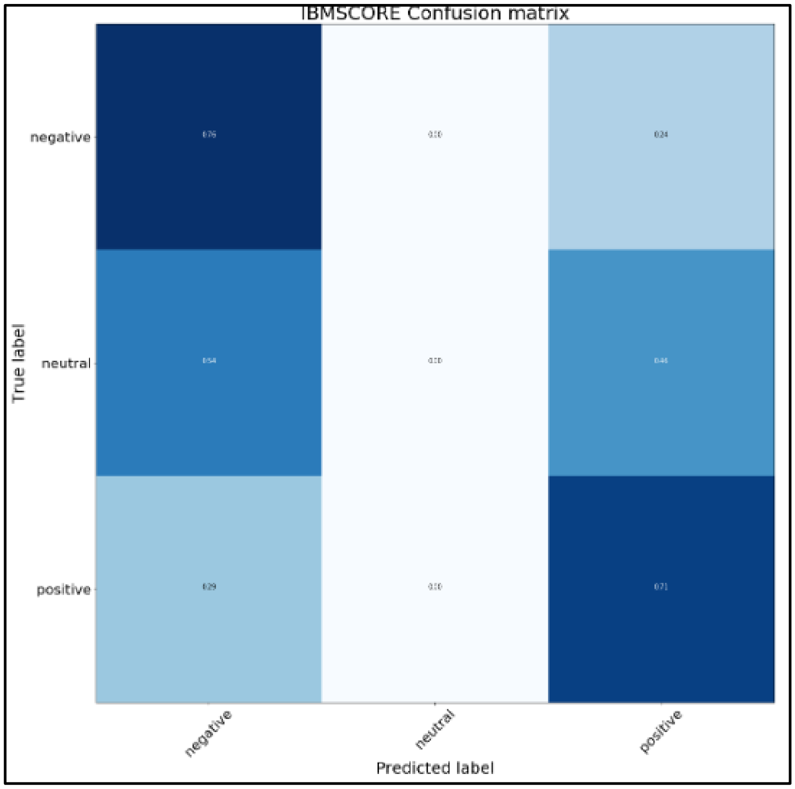
Accuracy: 0.39

<u>GCP:</u>



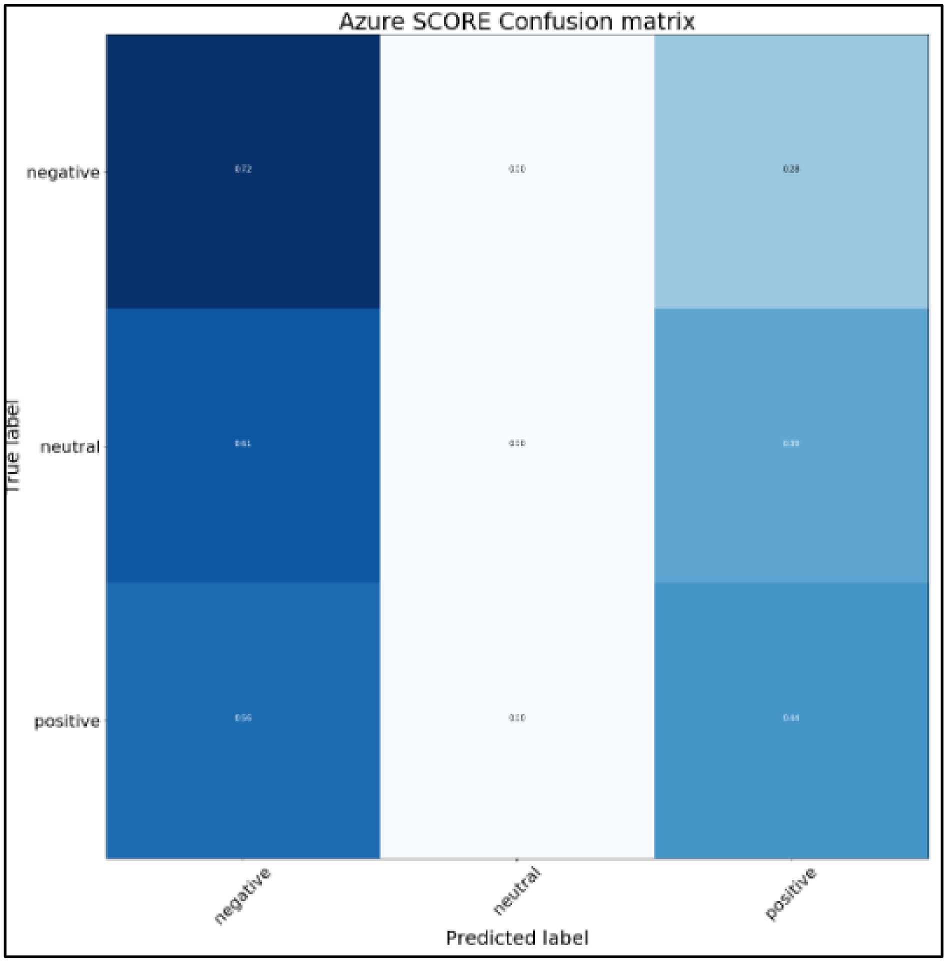
Accuracy: 0.36

IBM Watson:



Accuracy: 0.42

<u>Azure:</u>



Accuracy: 0.32

Inference: After running the following API's we observed that the IBM Watson performed the best among all the API's and gave an accuracy of 42% which was the best among the rest.

<u>EXPERIMENT 4:</u> ENSEMBLE LEARNING USING AUTOML

In this part, we ran Autosklearn, h2o and TPOT using different libraries and calculated the following confusion matrix.

<u>AutoSklearn:</u>

	Predicted Neg	Predicted Neutral	Predicted Positive
Actual Neg	14	3	3
Actual Neutral	34	15	23
Actual Positive	52	22	34

Accuracy: 31.5%

H2O:

Time: 500 seconds

negative	neutral	positive	Error	Rate
0.0	6.0	17.0	1.0	23 / 23
0.0	32.0	46.0	0.5897436	46 / 78
0.0	5.0	102.0	0.0467290	5 / 107
0.0	43.0	165.0	0.3557692	74 / 208

Accuracy: 64.43%

Time: 1200 seconds

Confusion Matrix:

negative	neutral	positive	Error	Rate
0.0	9.0	14.0	1.0	23 / 23
0.0	43.0	35.0	0.4487179	35/ 78
0.0	5.0	102.0	0.0467290	5 / 107
0.0	57.0	151.0	0.3028846	63 / 208

Accuracy: 67.9%

TPOT:

	Predicted Neg	Predicted Neutral	Predicted Positive
Actual Neg	2	3	3
Actual Neutral	0	4	4
Actual Positive	0	4	6

Accuracy: 46%

Inference: Although, h2o was our best model in the experiment 4 with accuracy of 67.9 % but even the h2o leaderboard suggested Random Forest as the best model and when we used Random Forest in the experiment 3 we were able to get an accuracy of 82.79% and hence that turned out to be our best model.

FINAL MODEL FOR PRODUCTION: Since, the Random Forest model performed the best with an accuracy score of 82.79% we used that model to get an accuracy for the GE transcripts.

MODEL USING GE TRANSCRIPTS:

We used Random forest on our GE Transcripts dataset and obtained an accuracy of 69.23% which was little lesser than the combined dataset accuracy with same Random Forest which was 82.79%