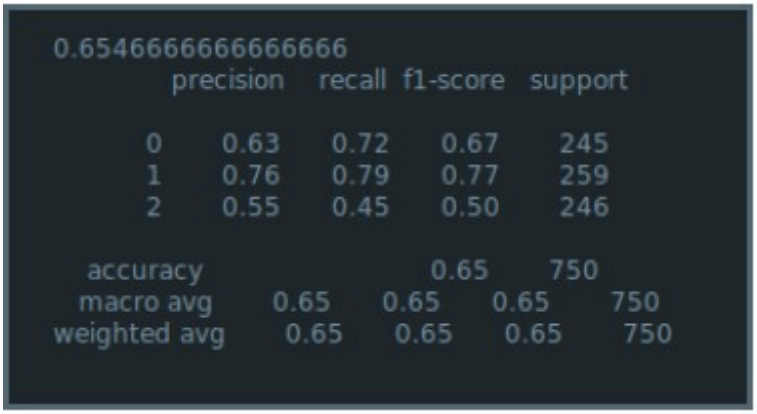
CMPE 462 Project Step 2

For the step 2 we did all of the coding together, sharing screens.

To classify the reviews, we have tried 3 methods.

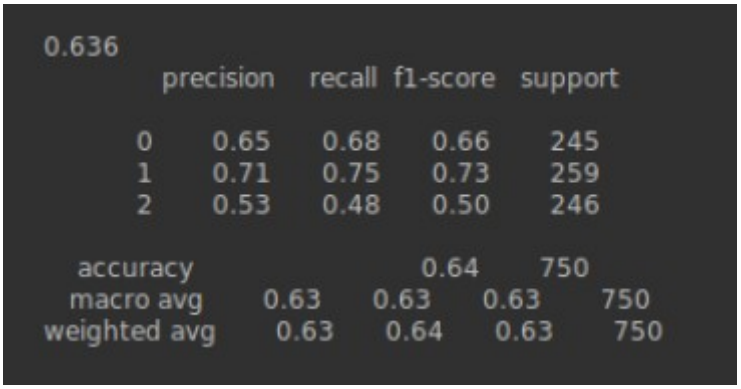
Multinomial Naive Bayes (MNB):



0.6546666666666666

	precision	recall	f1-score	support
0	0.63	0.72	0.67	245
1	0.76	0.79	0.77	259
2	0.55	0.45	0.50	246
accuracy			0.65	750
macro avg	0.65	0.65	0.65	750
weighted avg	0.65	0.65	0.65	750

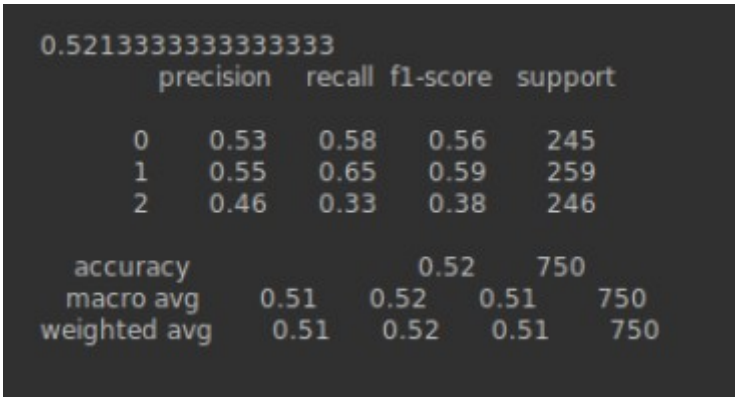
Support Vector Classifier (SVC):



0.636

	precision	recall	f1-score	support
0	0.65	0.68	0.66	245
1	0.71	0.75	0.73	259
2	0.53	0.48	0.50	246
accuracy			0.64	750
macro avg	0.63	0.63	0.63	750
weighted avg	0.63	0.64	0.63	750

K Nearest Neighbors (KNN):



0.5213333333333333

	precision	recall	f1-score	support
0	0.53	0.58	0.56	245
1	0.55	0.65	0.59	259
2	0.46	0.33	0.38	246
accuracy			0.52	750
macro avg	0.51	0.52	0.51	750
weighted avg	0.51	0.52	0.51	750

Among those 3, the best performing one was the multinomial naive bayes, as can be seen from the metrics. We think that is because dictionary of words used in different classes of reviews is pretty specific to each class.

While we were training the machine, we saved the best model as “step2_model_PudraSekerim.pkl” file, we also saved the vectorizer field, which is the regulator and which we fit in the code we trained the machine, in a “vectorizer.pkl” file and used it in the main code.
In the main code, we use the “vectorizer.pkl” file for us words and their editing and transformation.

Our result on val data:

As an output, first we print the desired output from us.

Secondly, we print the accuracy score of the model.

The last, we print the acc_{macro} score of the our calculations.