Description of the implementation:

Since the assignment required the implementation of the whole bayesian classifier from scratch, I made sure that the different main and secondary processes that together define a bayesian classifier are implemented ,each one with its own method.

Firstly the map_5_to_3 method is implemented to map the sentiments of the data from 5 scale to 3 scale.A suitable method was then implemented for preprocessing ,which tokenised each sentence using regex and removed stopwords. Then a method for feature extraction was implemented ,using the appropriate selection of model(all words as features or just specific words) returned a data structure with all the extracted features.The getAllFeatures method was then added for returning a collection of all the unique features(which was later used for computing the likelihoods of each feature). Afterwards, the index method (one of the most important methods) was implemented to return a data structure including all features and the number of their occurrences in each class, this method was later used for the calculation of the likelihoods, for which we need these specific count metrics.Later on , the compute_likelihoods method was implemented to compute the actual likelihoods of each word in each of the classes.The compute_priors method was then added for returning the priors of each class, and then the compute_posterior method for computing and returning the posteriors for a given sentence. At this point, all the essential parts of a bayesian classifier were fused together to implement the classification method, which takes the data to be classified as input , uses the posterior computation method and returns the maximum(the prediction) for each sentence in the given data. At last , a baseline classifier which classified the same data using the majority class was added, alongside with the f1 method which computes the evaluation metrics (precision,recall,f1,confusion matrix).

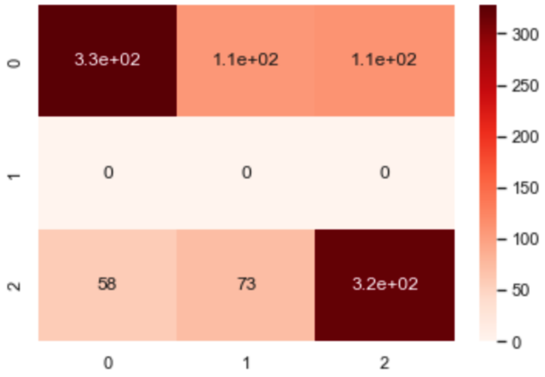
Note that , most of these methods are taking the choice of scaling and model as parameters ,which are used inside of the methods in order to use the right resources(three scale or five scale data etc,).The choices of model and scaling are specified by the user as input.

Performance of different systems

Baseline classifier(F1 scores): Scale 3: 0.20 , Scale 5: 0.09

Configuration 1: 3 Scale- Model 1(all words are features)

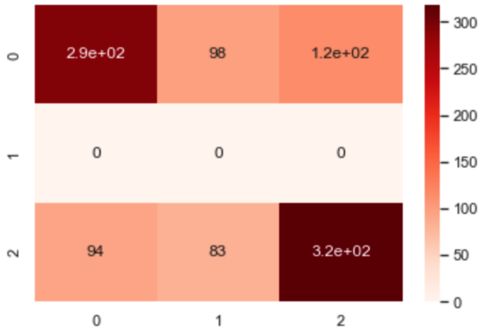
	Precision	Recall	F1
Class 0	0.60	0.85	0.70
Class 1	0	0	0
Class 2	0.71	0.74	0.73
Macro F1	-	-	0.48



As we can see, the classifier performs way better than the baseline for 3 scale sentiments and model 1.This is obviously expected. Comparing the results with those of the 5 scale classification , we can see that 3 scale ones are better, mainly because classification with just 3 classes is easier than another with 5 classes, as there's a bigger probability to get a prediction right.No predictions for class 1 have been made, which makes sense because of the severely imbalanced training data(priors for classes 0 and 1 are way larger than class 0 prior). We can see that the true 0s and true 2s are many(compared to other cells), which is a good sign.For the above reasons, this 3 scale model was used to generate the output for both development and test set in the submitted tsv files.

Configuration 2: 3 Scale- Model 2(only adjectives,adverbs,verbs as features)

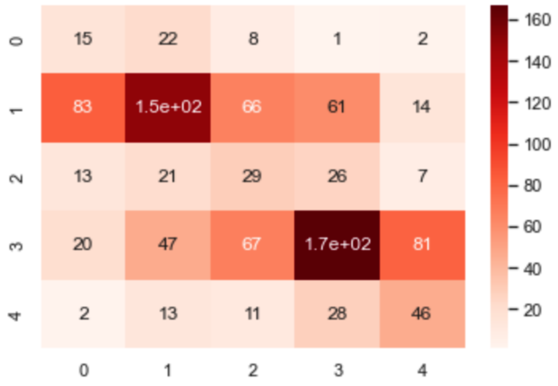
	Precision	Recall	F1
Class 0	0.58	0.76	0.66
Class 1	0	0	0
Class 2	0.64	0.74	0.69
Macro F1	-	-	0.45



For this model, where 3 scale sentiment is used and features with postag JJ,NN,RB (adjectives) are used , we would expect better overall results than previous one, since only seemingly important words such as adjectives were used as features.Even though true 0s and 2s are more than any other false predictions which is good, It turns out that this configuration produces slightly poorer results(still achieving 0 f1 score for class 1 because of imbalanced data).The imbalance of the training data might be one of the reasons(class 2 reviews in training data are way more than the other 2, resulting to a biased model).But most importantly, it just seems that just selecting words of certain postags is not generally a good selection approach, as it's not derived by study and analysis of the training data(in ML, features should be chosen according to their correlation to the classes, which is determined by careful analysis, that's also why poor training data lead to poor models).

Configuration 3: 5 Scale- Model 1(all words are features)

	Precision	Recall	F1
Class 0	0.33	0.12	0.17
Class 1	0.40	0.60	0.48
Class 2	0.28	0.15	0.20
Class 3	0.43	0.58	0.50
Class 4	0.44	0.30	0.35
Macro F1	-	-	0.34

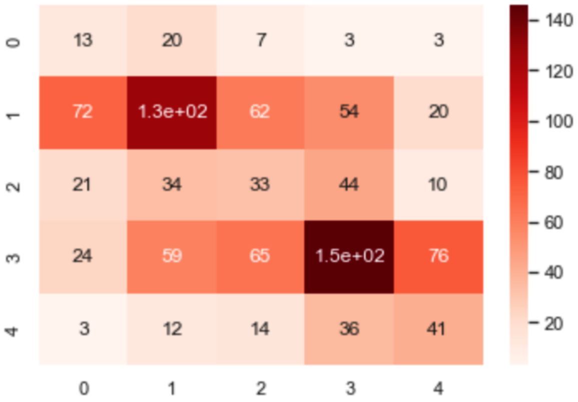


As expected , the 5 scale and model 1 configuration performs poorer than the 3 scale version, since now the classifier had to predict between 5 and not 3 classes.We can see that evaluation metrics are higher for class 3, since the model is biased towards that class(training data are imbalance-way more class 3 examples than other classes). A good sign is that again true 1s,3s,4s are among the highest values(which is good, since these are the correct classifications). Most importantly , this model performs way better than the baseline(0.09).But let's not forget that once again, the main reason why true 3s are more is because of the bias generated by the imbalanced training data. For the above reasons, this 3 scale model was used to generate the output for both development and test set in the submitted tsv files.

Configuration 4: 5 Scale- Model 2(only adjectives,adverbs,verbs as features)

As observed before with the 3 scale models, the 5 scale model using only certain words as features is again performing slightly poorer than the other version of the 5 scale model.True 1s and true 3s are again way more than true 0s and 2s. Once again this can be attributed to the fact that training data are imbalanced , or the fact that just picking certain parts of speech as features might not be enough for getting better results in classification problems with imbalanced and overall not so rich training data.(see comments on 3scale version of this model)

	Precision	Recall	F1
Class 0	0.30	0.10	0.15
Class 1	0.37	0.47	0.41
Class 2	0.23	0.19	0.21
Class 3	0.39	0.52	0.45
Class 4	0.35	0.24	0.30
Macro F1	-	-	0.30



What is the most important aspect to be taken into account to get the best results with a Naive Bayes approach?

When dealing with any kind of machine learning models, not just one thing can be vital in order to obtain the best possible results. One of these important aspects is having properly balanced , rich and descriptive training data. In this assignment the impact of imbalanced(and small amount) data leading to biases was evident and it probably makes it difficult for our classifier to get better results.

Another aspect that should always be considered and could help models produce better results , is the proper study between the correlations of all possible features.In ML projects, features which are highly correlated with specific classes are always guaranteed to produce better results.Therefore careful study of the features and proper feature selection is always needed. For this assignment, a chi squared score for the correlation of features and classes could be considered, which could help to only take into account the highest scoring features in each phrase(which might not necessarily be the adjectives, but certain words that , given the training data, could be closely associated with specific classes).