COMP9414: Artificial Intelligence

Assignment 2: Rating Prediction

Chengxuan Han
Z5334537

1.

(i) .under the scenario 1, with 1% stopping criterion, the metrics of model (a)::

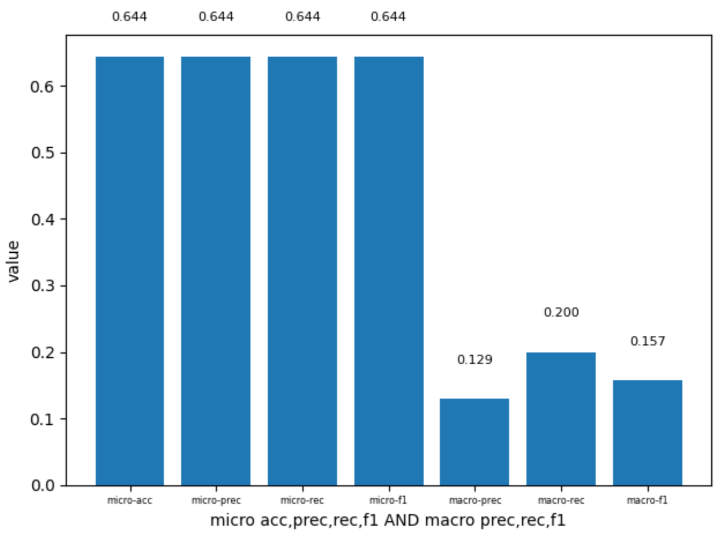

Figure1.1.1. metrics of model(a)

.under the scenario 1, without 1% stopping criterion, the metrics of model (b):

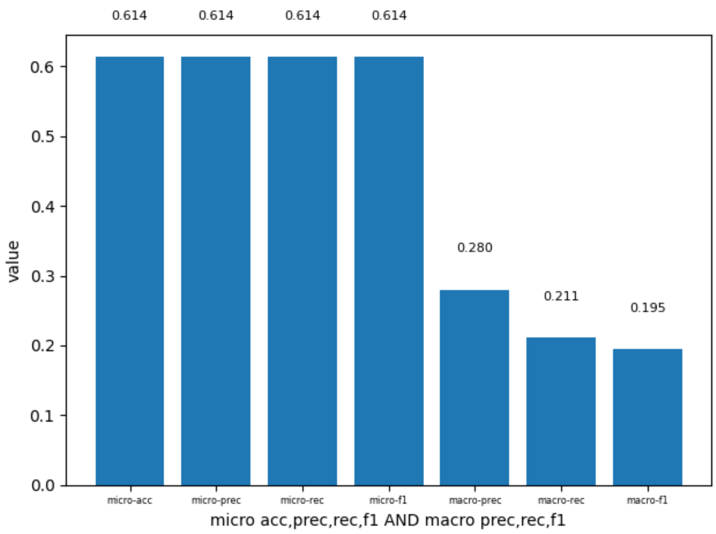

Figure1.1.2. metrics of model(b)

The similarities is that the micro precision, recall and f1 is greater than the macro precision, recall and f1 in both model(a) and model(b). The differences is that the macro precision, recall and f1 of model(b) is greater than model(a) because having the 1% stopping criterion leads to less nodes for splitting in each class, result in the lower macro precision, recall and f1. And micro precision, recall and f1 of model(a) is greater than model(b) because for all class, it can avoid low variance and overfitting.

(ii).under the scenario 2, with 1% stopping criterion, the metrics of model (a):

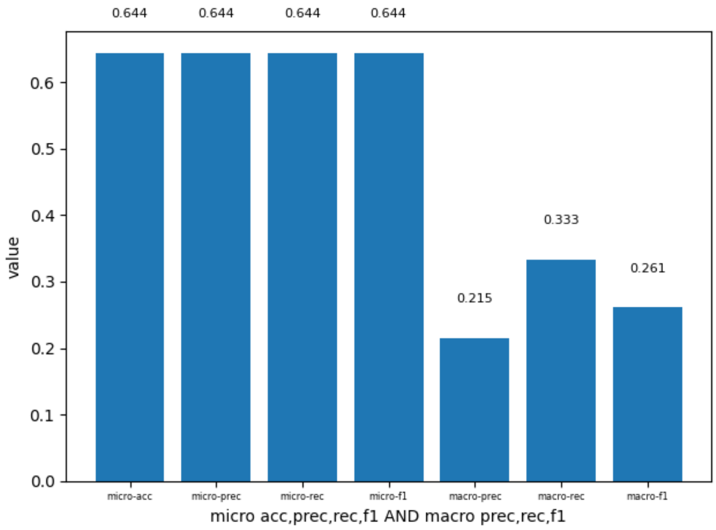

Figure1.2.1. metrics of model(a)

under the scenario 2, without 1% stopping criterion, the metrics of model (b):

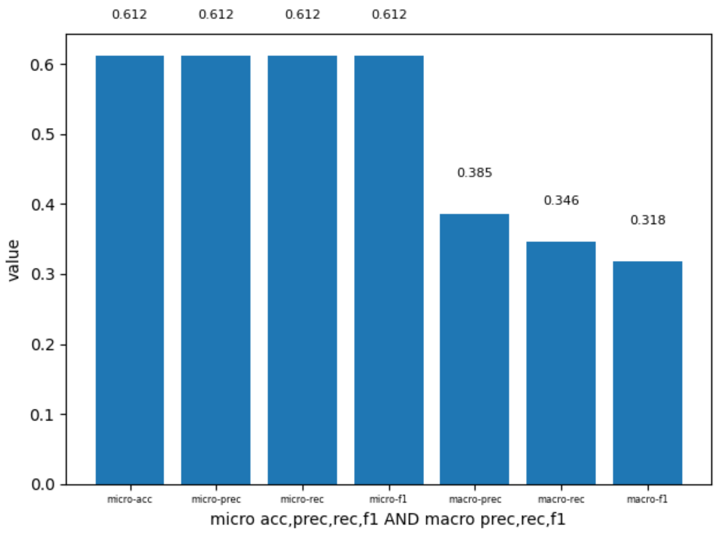

Figure1.2.2. metrics of model(b)

The similarities and differences of model(a) and model(b) in scenario 2 can reference from scenario 1 that the micro precision, recall and f1 is greater than the macro precision, recall and f1 in both model(a) and model(b) and the micro precision, recall and f1 of model(a) is greater than model(b), because having the 1% stopping criterion avoid low variance and overfitting instead of using the default value 1.

(iii).The similarities of scenario 1 and 2 is that the micro precision, recall and f1 is greater than the macro precision, recall and f1 because Macro-average is average of per-class measures that leads to the lower accuracy of models in macro precision, recall and f1. The difference is that the macro accuracy of scenario 1 is slightly higher than scenario 2 because and there are 2 more classes in scenario 1 than scenario 2 as the rating of 1, 2 and 3 are combined as negative, for example, some predicting errors like actual value is 1 and predict value is 2 in scenario 1, is considered to be correct prediction in scenario 2 as both rating of 1 and 2 are considered to be negative.

2.

(i).under the scenario 1, the metrics of models(a) (the whole vocabulary):

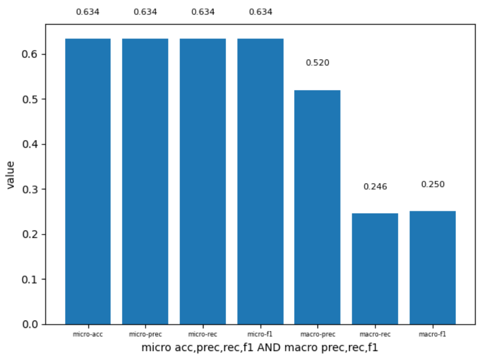

Figure2.1.1 BNB model(a)

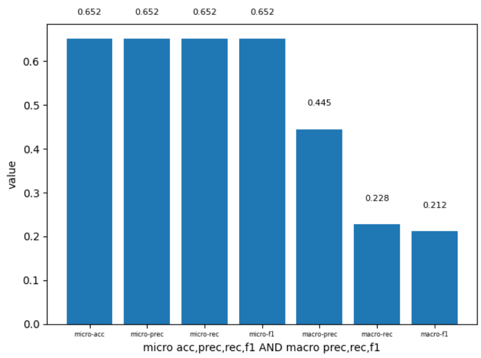

Figure2.1.1 MNB model(a)

under the scenario 1, the metrics of models(b) (the most frequent 1000 words)

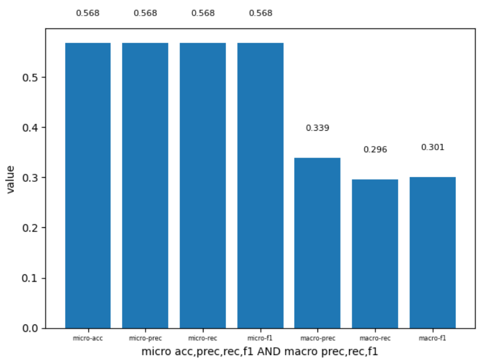

Figure2.1.3 BNB model(b)

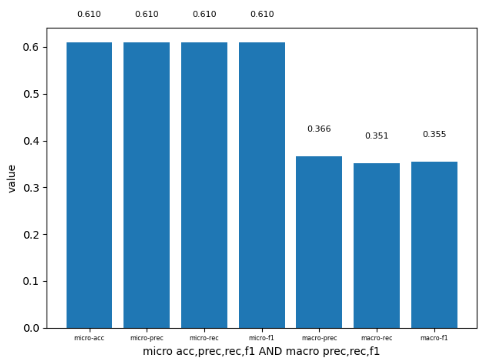

Figure2.1.4 MNB model(b)

The micro precision, recall and f1 and macro prediction of model(a) is higher than model(b), because using the 1000 most frequent words from the test set instead of using the whole vocabulary lost too much information and leads to the lower accuracy. When I change the max_features to a bigger number in experiments of question5, the accuracy grows higher.

(ii). under the scenario 2, the metrics of models(a) (the whole vocabulary):

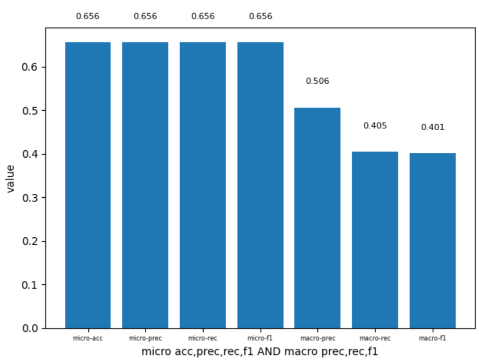

Figure2.2.1 BNB model(a)

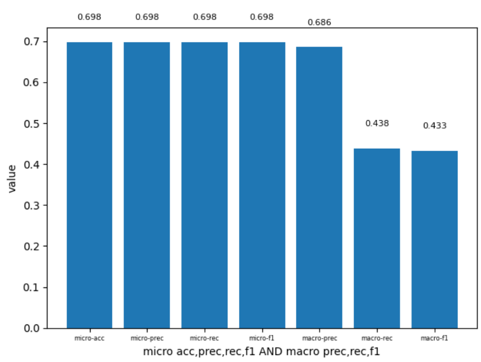

Figure2.2.1 MNB model(a)

under the scenario 2, the metrics of models(b) (the most frequent 1000 words)

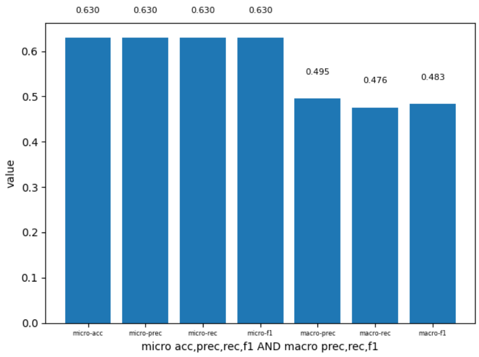

Figure2.2.3 BNB model(b)

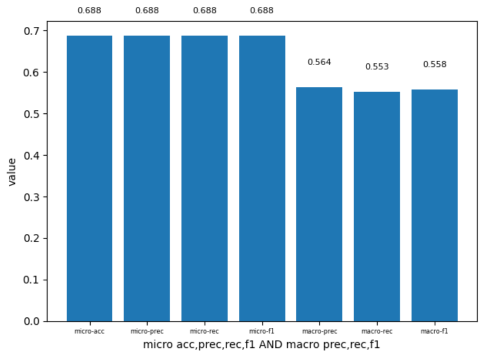

Figure2.2.4 MNB model(b)

The micro precision, recall and f1 and macro prediction of model(a) is higher than model(b), the reason is the same as question 2.1, using the 1000 most frequent words from the test set instead of using the whole vocabulary lost too much information and leads to the lower accuracy.

(iii). It shows that the macro precision, recall and f1 under scenario 2 is greater than scenario 1, because the rating classes 1, 2, 3 are combined as negative class, that leads to the less prediction errors, for example, some predicting errors like actual value is 1 and predict value is 2 in scenario 1, is considered to be correct prediction in scenario 2 as both rating of 1 and 2 are considered to be negative.

3.

(i).under the scenario 1, the metrics of models(a) (standard models):

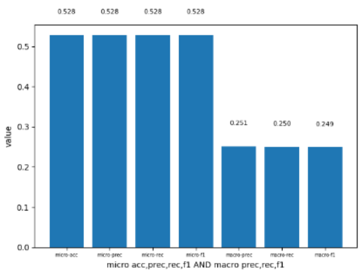

Figure3.1.1 DT model(a)

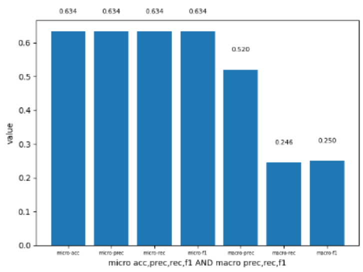

Figure3.1.2 BNB model(a)

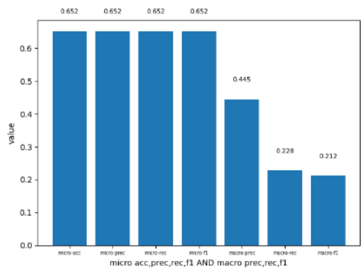

Figure3.1.3 MNB model(a)

under the scenario 1, the metrics of models(b) (Porter stemming using NLTK then English stop word removal):

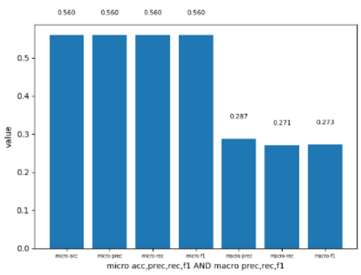

Figure3.1.4 DT model(a)

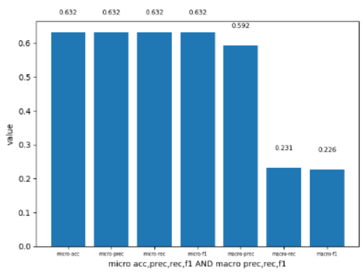

Figure3.1.5 BNB model(a)

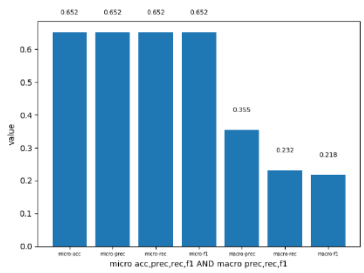

Figure3.1.6 MNB model(a)

It shows in the graphs that the micro and macro precision, recall and f1 are increase after applying the porter stemming using NLTK and English stop words, because with porter stemming using NLTK stop words, it removes the words with less important information in our test set, let the model identify words that are more rare and potentially more relevant to what we're interested in.

(ii) under the scenario 2, the metrics of models(a) (standard models):

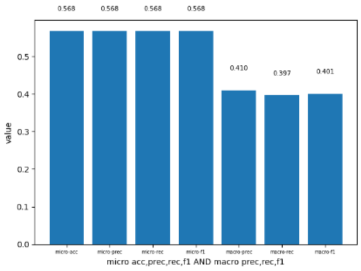

Figure3.2.1 DT model(a)

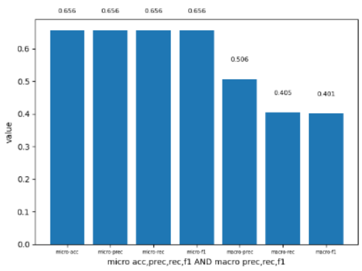

Figure3.2.2 BNB model(a)

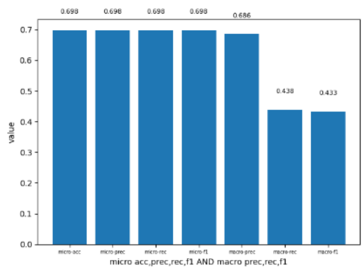

Figure3.2.3 MNB model(a)

under the scenario 2, the metrics of models(b) (Porter stemming using NLTK then English stop word removal):

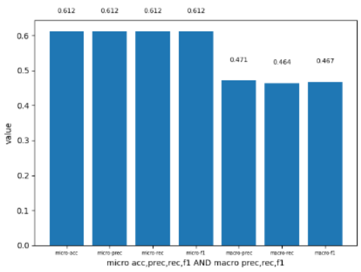

Figure3.2.4 DT model(a)

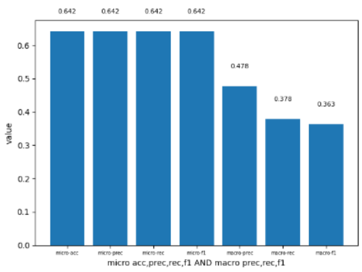

Figure3.2.5 BNB model(a)

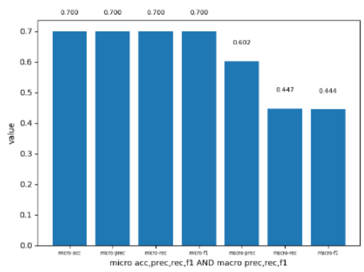

Figure3.2.6 MNB model(a)

It shows in the graphs that the micro and macro precision, recall and f1 are increase after applying the porter stemming using NLTK and English stop words, as I mentioned in question3.1, the prediction become more accurate because the with porter stemming using NLTK stop words, it removes the words with less important information in our test set, let the model identify words that are more rare and potentially more relevant to what we're interested in.

(iii) The micro and macro precision, recall and f1 are increasing in scenario 2, because the classes of rating 1, 2, 3 are combine into 1 class: negative, that leads to less error when it predicts rating of 1, 2 ,3 as negative considered as correct in scenario 2.

4.

(i).under the scenario 1, the metrics of models(a) (no conversion to lower case):

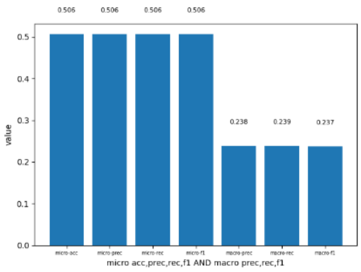

Figure4.1.1 DT model(a)

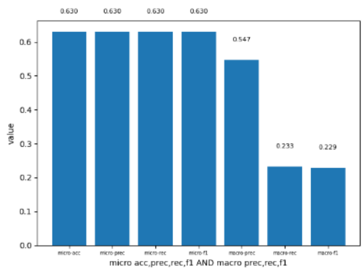

Figure4.1.2 BNB model(a)

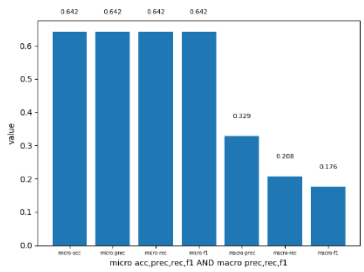

Figure4.1.3 MNB model(a)

under the scenario 1, the metrics of models(b) (all input text converted to lower case):

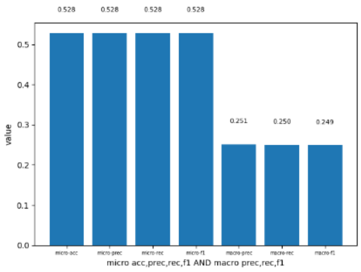

Figure4.1.4 DT model(a)

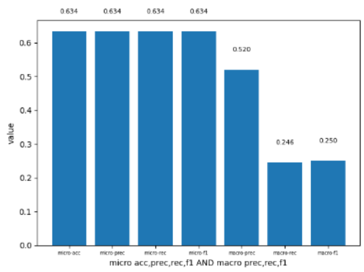

Figure4.1.5 BNB model(a)

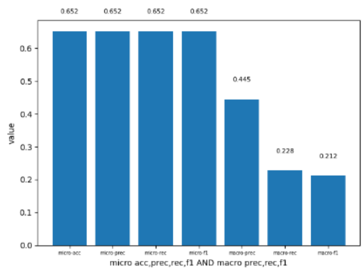

Figure4.1.6 MNB model(a)

It shows in the graphs that the micro and macro precision, recall and f1 are increase after all input text converted to lower case, because with all input are lower case, it can make the model consider the same word in either upper case or lower case as the same meaning, and avoiding ambiguity.

(ii).under the scenario 2, the metrics of models(a) (no conversion to lower case):

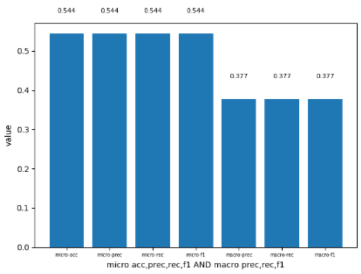

Figure4.2.1 DT model(a)

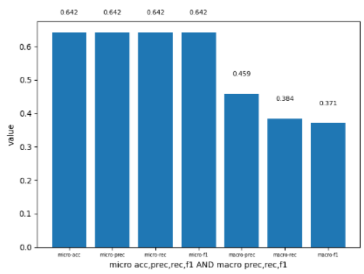

Figure4.2.2 BNB model(a)

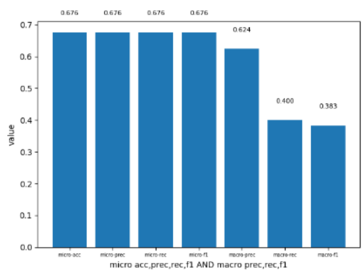

Figure4.2.3 MNB model(a)

under the scenario 2, the metrics of models(b) (all input text converted to lower case):

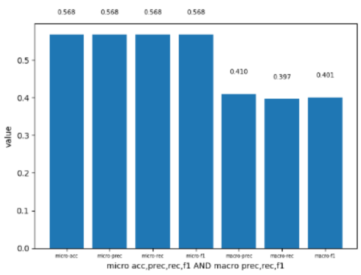

Figure4.2.4 DT model(a)

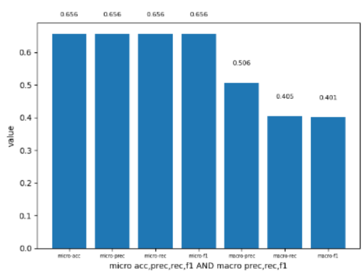

Figure4.2.5 BNB model(a)

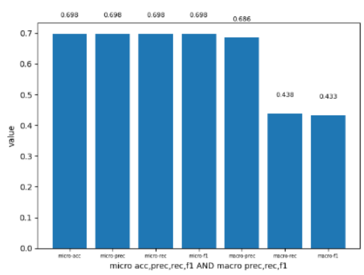

Figure4.2.6 MNB model(a)

It shows in the graphs that the micro and macro precision, recall and f1 are increase after all input text converted to lower case, as I mentioned in 4,1, because with all input are lower case, it can make the model consider the same word in either upper case or lower case as the same meaning, and avoiding ambiguity.

(iii). It can refer from the question 2 and 3, the differences between scenario 1 and 2 is that the micro and macro precision, recall and f1 are increasing in scenario 2, because the classes of rating 1, 2, 3 are combine into 1 class: negative, that leads to less error when it predicts rating of 1, 2 ,3 as negative considered as correct in scenario 2.

5.
From the experience of the previous questions, we can find out that there are some conditions that the model always has the better accuracy than other models :
①. Use the MNB model:

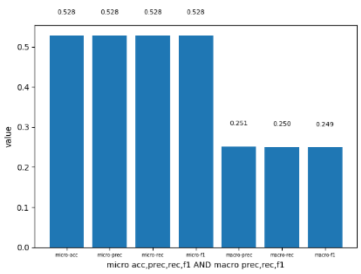

Figure5.1 DT model

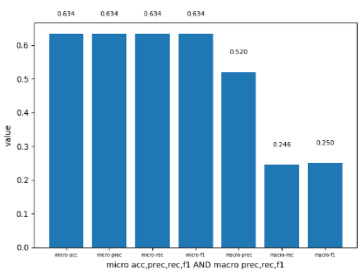

Figure5.2 BNB model

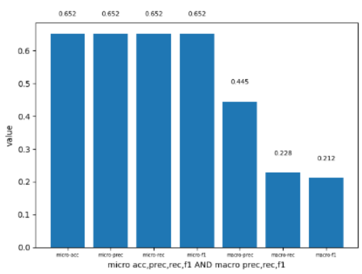

Figure5.3 MNB model

From the graph above, we can tell that MNB model has the better micro accuracy over all three models. Even the macro recall and f1 of DT and BNB model is slightly higher than the MNB model, the MNB model still has the overall advantages over other models.
②. Set the lowercase to True: we can tell set lower case to True(default value) is better from the experience from question 4.
③. Find the suitable value of max_features parameter:
So I choose MNB model and set lowercase to True(default value) and compared the accuracy under different max_features values from 1000:

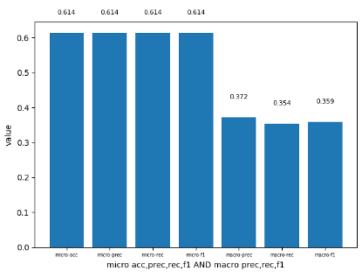

Figure5.4 max_features=1000

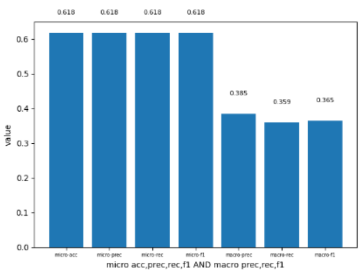

Figure5.5 max_features=1500

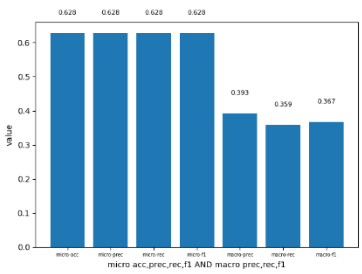

Figure5.6 max_features=2000

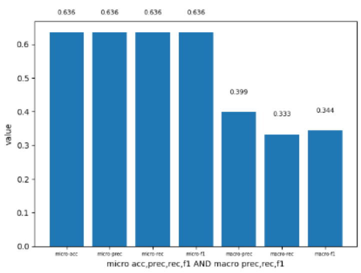

Figure5.7 max_features=2500

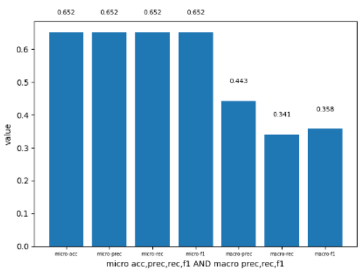

Figure5.8 max_features=3000

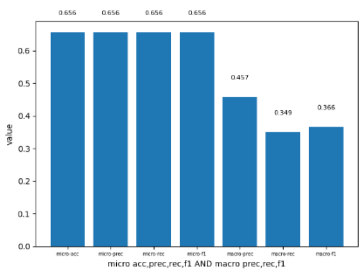

Figure5.9 max_features=3100

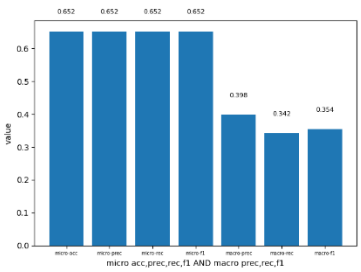

Figure5.10 max_features=3300

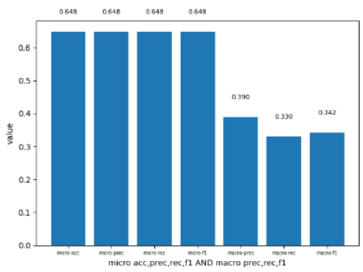

Figure5.11 max_features=3500

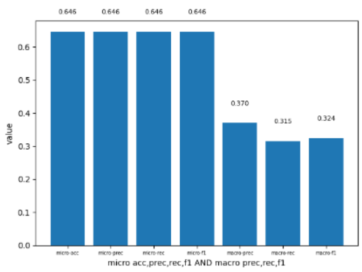

Figure5.12 max_features=4000

We can tell from the graph above that the accuracy is increasing and peaking at the max_features=3100, with micro acc, prec, rec, f1 at 0.656, and macro prc, rec, f1 at 0.457, 0.349, 0.366(all are highest among all of the experiments), and when max_features goes higher beyond 3100, the accuracy is starting to decrease. That means 3100 is the most suitable value for max_features.

With the same parameter setting in BNB model, the accuracy is showed in the graph below:

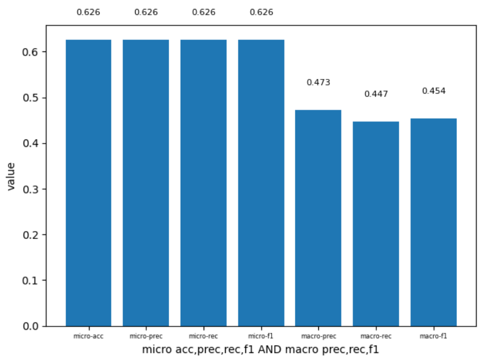

Figure5.13 BNB with max_features=3100

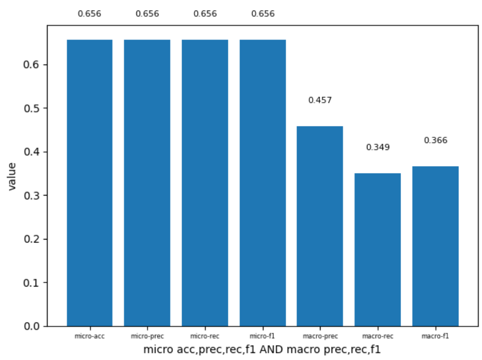

Figure5.14 MNB with max_features=3100

It is obvious that MNB model has a better performance than BNB model as the micro accuracy of MNB(0.656) model is higher than BNB model(0.626).

Overall, I choose MNB with max_features=3100 and all letters set to lowercase. Although the accuracy is 0.656, which is not absolutely accurate(might due to training set size is not big enough or having more classes than other scenario, for example, using sentiments), but it has the best performance over all the models I have experimented.