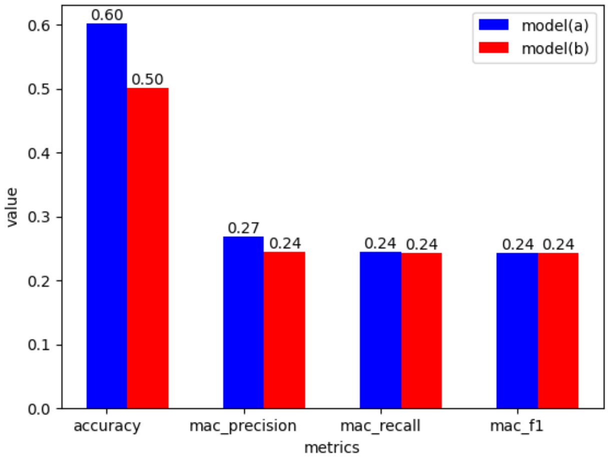
Name: Qiyao Zhou
zid: z5379852

COMP9414 Artificial Intelligence

Assignment 2: Rating Prediction

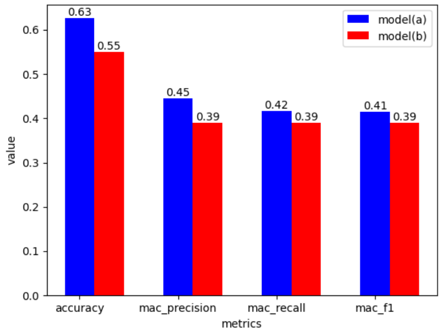
1. (3 marks) Develop Decision Tree models for training and testing: (a) with the 1% stopping criterion (the standard model), and (b) without the 1% stopping criterion.

(i) Show all metrics on the test set for scenario 1 comparing the two models (a) and (b), and explain any similarities and differences.



The two models are close in performance at the macro level, with model (a) even slightly outperforming model (b) in precision, while for accuracy model (a) performs significantly better than model (b). This is because the overfitting in model (b) without the stopping criterion affects the metrics of the model.

(ii) Show all metrics on the test set for scenario 2 comparing the two models (a) and (b), and explain any similarities and differences.



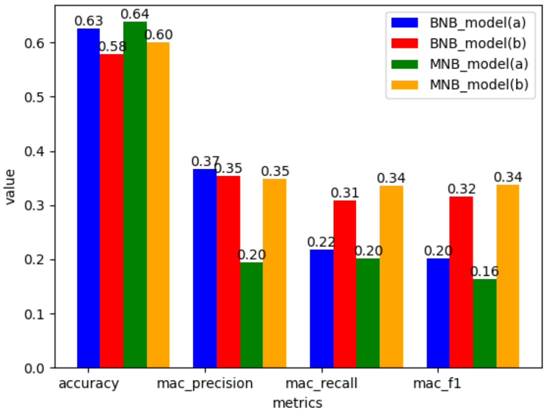
Unlike Scenario 1, all the metrics in Scenario 2 are significantly better for model (a) than for model (b).

(iii) Explain any differences in the results between scenarios 1 and 2.

The metrics of model (a) are significantly better than those of model (b) under scenario 2, while the metrics of models (a) and (b) under macroscopic scenario 1 are close to each other, and the metrics under scenario 2 are better than those of their metrics in scenario 1. This is mainly due to the fact that there are fewer categories under scenario 2.

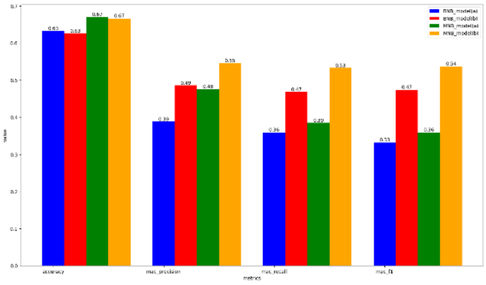
2. (3 marks) Develop BNB and MNB models from the training set using: (a) the whole vocabulary (standard models), and (b) the most frequent 1000 words from the vocabulary, as defined using sklearn's CountVectorizer, after preprocessing by removing "junk" characters.

(i) Show all metrics on the test set for scenario 1 comparing the corresponding models (a) and (b), and explain any similarities and differences.



For BNB model, all except macro recall and f1 show that the indicators in model (a) are significantly larger than their metrics in model (b). As for MNB model, all macro metrics show that the metrics in model (b) are significantly larger than the corresponding metrics in model (a) and the condition of accuracy is model(a) greater than model(b).

(ii) Show all metrics on the test set for scenario 2 comparing the corresponding models (a) and (b), and explain any similarities and differences.



For both BNB and MNB, both show that model (a) is close to (b) in accuracy, while model (a) has significantly smaller values than model (b) in terms of macro metrics.

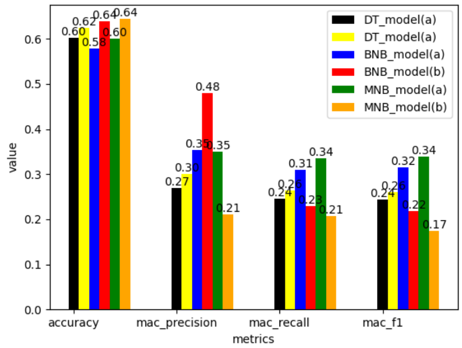
(iii) Explain any differences in the results between scenarios 1 and 2.

The metrics under scenario 2 show little change and a slight increase in accuracy compared to scenario 1, while the macro-metrics show a significant increase.

3. (3 marks) Evaluate the effect of preprocessing for the three standard models by comparing models developed with: (a) only the preprocessing described above

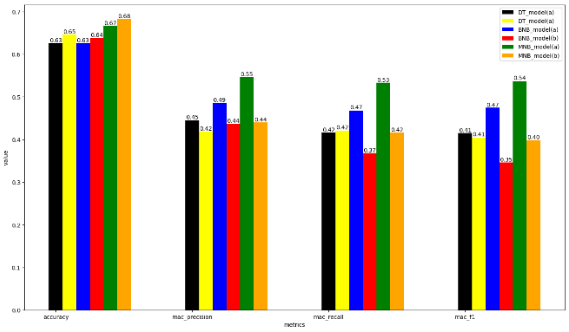
(standard models), and (b) applying, in addition, Porter stemming using NLTK then English stop word removal using sklearn's CountVectorizer.

(i) Show all metrics on the test set for scenario 1 comparing the corresponding models (a) and (b), and explain any similarities and differences.



For the DT, MNB and BNB models, the metrics in model (b) outperformed (a) and were more pronounced for the macro metrics. This is because that using NLTK Porter stemming and applying NLTK English stop word removal will decrease some low frequency words and unusual words effecting the accuracy and result.

(ii) Show all metrics on the test set for scenario 2 comparing the corresponding models (a) and (b), and explain any similarities and differences.



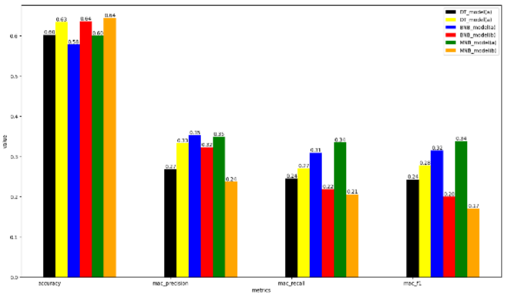
For the three models DT, MNB and BNB, accuracy in model (b) outperform (a) while model (a) has significantly bigger values than model (b) in terms of macro metrics.

(iii) Explain any differences in the results between scenarios 1 and 2.

The metrics in scenario 2 have larger values compared to those in the corresponding scenario 1, suggesting that streamlining the classification will appropriately improve the reliability of the model.

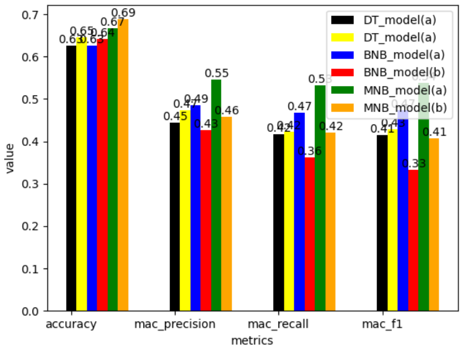
4. (3 marks) Evaluate the effect of converting all letters to lower case for the three standard models by comparing models with: (a) no conversion to lower case, and (b) all input text converted to lower case.

(i) Show all metrics on the test set for scenario 1 comparing the corresponding models (a) and (b), and explain any similarities and differences.



For the DT, MNB and BNB models, metrics in DT each accuracy in model (b) outperform (a), the macro metrics in BNB and MNB are the opposite.

(ii) Show all metrics on the test set for scenario 2 comparing the corresponding models (a) and (b), and explain any similarities and differences.



For the DT, MNB and BNB models, metrics in DT each accuracy in model (b) outperform (a), the macro metrics in BNB and MNB are the opposite.

(iii) Explain any differences in the results between scenarios 1 and 2.

The metrics in scenario 2 have larger values compared to those in the corresponding scenario 1, suggesting that streamlining the classification will appropriately improve the reliability of the model.

5. (5 marks) Describe your chosen "best" method for rating prediction. Give new experimental results for your method trained on the training set of 2000 reviews and tested on the test set of 500 reviews. Explain how this experimental evaluation justifies your choice of model, including settings and parameters, against a range of alternatives. Provide new experiments and justifications: do not just refer to previous answers.

Combining the information obtained from the previous topics, I decided to choose MNB as the base model. As for the pre-processing, using porter stemming by NLTK then English stop word removal using sklearn's CountVectorizer and convert all input text to lower case are proved to be better compared with the standard model.

To ensure that the pre-processing combinations did not interfere with each other, I tested three scenarios:

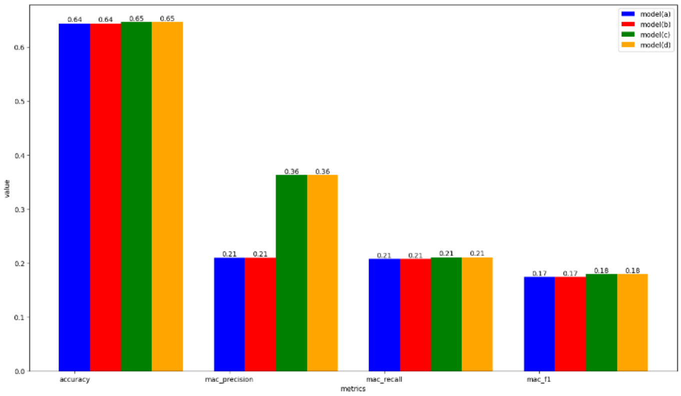
Model(a): using porter stemming by NLTK then English stop word removal using sklearn's CountVectorizer.

Model(b): using porter stemming by NLTK then English stop word removal using sklearn's CountVectorizer and convert all input text to lower case

Model(c): using porter stemming by NLTK only

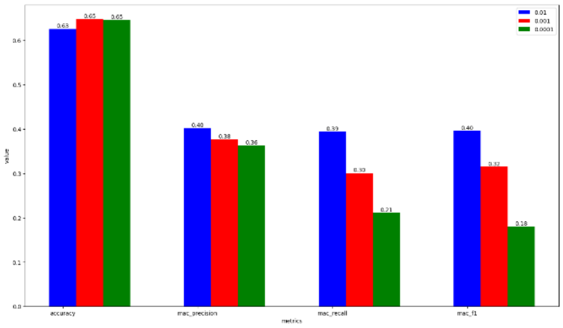
Model(d): using porter stemming by NLTK and convert all input text to lower case

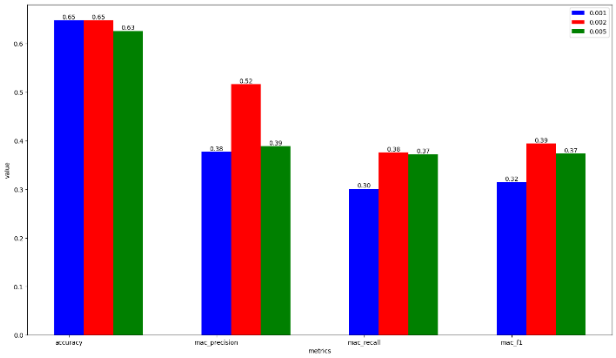
The results are as followed:



It's clear that model(c) and (d) are better, in the spirit of simple handling, I decided to use the model (c).

Then, I start to optimize the model based on the minimum keyword frequency, which was achieved by adjusting the min_df of the CountVectorizer, with the following parameter adjustments:





From this, I get the most suitable min_df parameter of 0.002.

	precision	recall	F1-score	support
1	0.57	0.37	0.45	43
2	0.67	0.08	0.14	25
3	0.26	0.22	0.24	41
4	0.31	0.36	0.33	69
5	0.77	0.84	0.80	322
accuracy			0.65	500
Macro avg	0.52	0.38	0.39	500
Weighted avg	0.64	0.65	0.63	500

In summary, the best rating predictor I have used based on the standard model using porter stemming by NLTK and has a min_df parameter of 0.002.