# Kanadech Jirapongtanavech z5176970

|  |  |
| --- | --- |
| Figure 1 Topic Frequency Chart | Figure 2 Sentiment Frequency Chart |

1. (1 mark) Give simple descriptive statistics showing the frequency distributions for the sentiment and topic classes across the full dataset. What do you notice about the distribution?

Topic statistics

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Variable | N | Mean | SE Mean | Standard Dev | Minimum | Q1 | Median | Q3 | Maximum |
| Frequency | 20 | 100.0 | 20.8 | 93.0 | 7.0 | 26.0 | 57.5 | 157.3 | 358.0 |

Sentiment Statistics

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Variable | N | Mean | SE Mean | Standard Dev | Minimum | Q1 | Median | Q3 | Maximum |
| Count | 3 | 667 | 334 | 579 | 153 | 153 | 553 | 1294 | 1294 |

From the figures and tables above, many tweets were talking about topic 10003, economic management and were negative tweets. The class distribution for both is skewed. More than half the tweets were negative tweets. The skewed topics distribution is, in my opinion, the expected distribution. Voters tend to care about things that are closed to them or things that could have significant impact on their livelihood such as economic management which accounted for 17.9% of the tweets.

1. (2 marks) Vary the number of words from the vocabulary used as training features for the standard methods (e.g. the top *N* words for *N* = 100, 200, etc.). Show metrics calculated on both the training set and the test set. Explain any difference in performance of the models between training and test set, and comment on metrics and runtimes in relation to the number of features.

Top N – Top number of feature words  
Dataset – Data used as test sets [train, test]  
P\_MI – Precision score micro average  
P\_MA – Precision score macro average  
P\_W – Precision score weighted average  
R\_MI – Recall score micro average  
R\_MA – Recall score macro average  
R\_W – Recall score weighted average  
F1\_MI – F1 score micro average  
F1\_MA – F1 score macro average  
F1\_W – F1 score weighted average  
R – Runtime in seconds (training time + prediction time)

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Classifier** | **Top N** | **Dataset** | **P\_MI** | **P\_MA** | **P\_W** | **R\_MI** | **R\_MA** | **R\_W** | **F1\_MI** | **F1\_MA** | **F1\_W** | **Runtime** |
| DT\_topics | 100 | test | 0.276 | 0.183 | 0.272 | 0.276 | 0.152 | 0.276 | 0.276 | 0.156 | 0.251 | 0.028 |
| DT\_topics | 100 | train | 0.357 | 0.215 | 0.333 | 0.357 | 0.199 | 0.357 | 0.357 | 0.201 | 0.328 | 0.028 |
| DT\_topics | 200 | test | 0.3 | 0.187 | 0.279 | 0.3 | 0.163 | 0.3 | 0.3 | 0.164 | 0.27 | 0.052 |
| DT\_topics | 200 | train | 0.385 | 0.242 | 0.367 | 0.385 | 0.219 | 0.385 | 0.385 | 0.221 | 0.357 | 0.056 |
| DT\_topics | 400 | test | 0.3 | 0.187 | 0.279 | 0.3 | 0.163 | 0.3 | 0.3 | 0.164 | 0.27 | 0.116 |
| DT\_topics | 400 | train | 0.385 | 0.242 | 0.367 | 0.385 | 0.219 | 0.385 | 0.385 | 0.221 | 0.357 | 0.12 |
| DT\_sentiment | 100 | test | 0.66 | 0.444 | 0.608 | 0.66 | 0.404 | 0.66 | 0.66 | 0.399 | 0.625 | 0.024 |
| DT\_sentiment | 100 | train | 0.695 | 0.575 | 0.665 | 0.695 | 0.469 | 0.695 | 0.695 | 0.482 | 0.665 | 0.028 |
| DT\_sentiment | 200 | test | 0.672 | 0.452 | 0.617 | 0.672 | 0.41 | 0.672 | 0.672 | 0.405 | 0.634 | 0.036 |
| DT\_sentiment | 200 | train | 0.699 | 0.577 | 0.667 | 0.699 | 0.469 | 0.699 | 0.699 | 0.482 | 0.667 | 0.037 |
| DT\_sentiment | 500 | test | 0.672 | 0.452 | 0.617 | 0.672 | 0.41 | 0.672 | 0.672 | 0.405 | 0.634 | 0.0899 |
| DT\_sentiment | 500 | train | 0.699 | 0.577 | 0.667 | 0.699 | 0.469 | 0.699 | 0.699 | 0.482 | 0.667 | 0.0909 |
| BNB\_topics | 100 | test | 0.27 | 0.166 | 0.26 | 0.27 | 0.144 | 0.27 | 0.27 | 0.145 | 0.249 | 0.014 |
| BNB\_topics | 100 | train | 0.407 | 0.38 | 0.424 | 0.407 | 0.262 | 0.407 | 0.407 | 0.286 | 0.394 | 0.021 |
| BNB\_topics | 200 | test | 0.322 | 0.205 | 0.305 | 0.322 | 0.176 | 0.322 | 0.322 | 0.176 | 0.297 | 0.033 |
| BNB\_topics | 200 | train | 0.509 | 0.472 | 0.522 | 0.509 | 0.33 | 0.509 | 0.509 | 0.356 | 0.492 | 0.038 |
| BNB\_topics | 600 | test | 0.344 | 0.202 | 0.343 | 0.344 | 0.172 | 0.344 | 0.344 | 0.168 | 0.309 | 0.049 |
| BNB\_topics | 600 | train | 0.584 | 0.485 | 0.598 | 0.584 | 0.329 | 0.584 | 0.584 | 0.337 | 0.545 | 0.045 |
| BNB\_topics | 900 | test | 0.33 | 0.193 | 0.332 | 0.33 | 0.148 | 0.33 | 0.33 | 0.143 | 0.276 | 0.055 |
| BNB\_topics | 900 | train | 0.575 | 0.378 | 0.548 | 0.575 | 0.296 | 0.575 | 0.575 | 0.294 | 0.523 | 0.067 |
| BNB\_sentiment | 100 | test | 0.712 | 0.562 | 0.68 | 0.712 | 0.485 | 0.712 | 0.712 | 0.504 | 0.686 | 0.015 |
| BNB\_sentiment | 100 | train | 0.719 | 0.646 | 0.702 | 0.719 | 0.551 | 0.719 | 0.719 | 0.581 | 0.7 | 0.018 |
| BNB\_sentiment | 200 | test | 0.712 | 0.565 | 0.689 | 0.712 | 0.505 | 0.712 | 0.712 | 0.519 | 0.696 | 0.024 |
| BNB\_sentiment | 200 | train | 0.751 | 0.687 | 0.744 | 0.751 | 0.63 | 0.751 | 0.751 | 0.653 | 0.746 | 0.027 |
| BNB\_sentiment | 400 | test | 0.74 | 0.678 | 0.731 | 0.74 | 0.555 | 0.74 | 0.74 | 0.584 | 0.727 | 0.038 |
| BNB\_sentiment | 400 | train | 0.79 | 0.755 | 0.786 | 0.79 | 0.673 | 0.79 | 0.79 | 0.704 | 0.785 | 0.044 |
| BNB\_sentiment | 1000 | test | 0.734 | 0.78 | 0.745 | 0.734 | 0.518 | 0.734 | 0.734 | 0.543 | 0.712 | 0.051 |
| BNB\_sentiment | 1000 | train | 0.847 | 0.856 | 0.849 | 0.847 | 0.705 | 0.847 | 0.847 | 0.747 | 0.84 | 0.065 |
| MNB\_topics | 100 | test | 0.256 | 0.164 | 0.245 | 0.256 | 0.141 | 0.256 | 0.256 | 0.144 | 0.237 | 0.008 |
| MNB\_topics | 100 | train | 0.408 | 0.404 | 0.434 | 0.408 | 0.26 | 0.408 | 0.408 | 0.286 | 0.394 | 0.012 |
| MNB\_topics | 200 | test | 0.32 | 0.198 | 0.3 | 0.32 | 0.18 | 0.32 | 0.32 | 0.182 | 0.3 | 0.012 |
| MNB\_topics | 200 | train | 0.533 | 0.525 | 0.549 | 0.533 | 0.393 | 0.533 | 0.533 | 0.43 | 0.525 | 0.016 |
| MNB\_topics | 400 | test | 0.352 | 0.217 | 0.339 | 0.352 | 0.202 | 0.352 | 0.352 | 0.203 | 0.334 | 0.016 |
| MNB\_topics | 400 | train | 0.614 | 0.644 | 0.636 | 0.614 | 0.455 | 0.614 | 0.614 | 0.495 | 0.603 | 0.02 |
| MNB\_topics | 1000 | test | 0.356 | 0.204 | 0.344 | 0.356 | 0.187 | 0.356 | 0.356 | 0.186 | 0.331 | 0.024 |
| MNB\_topics | 1000 | train | 0.719 | 0.712 | 0.734 | 0.719 | 0.522 | 0.719 | 0.719 | 0.56 | 0.703 | 0.032 |
| MNB\_topics | 1500 | test | 0.34 | 0.198 | 0.342 | 0.34 | 0.17 | 0.34 | 0.34 | 0.17 | 0.312 | 0.04 |
| MNB\_topics | 1500 | train | 0.723 | 0.731 | 0.743 | 0.723 | 0.498 | 0.723 | 0.723 | 0.54 | 0.705 | 0.048 |
| MNB\_sentiment | 100 | test | 0.72 | 0.582 | 0.687 | 0.72 | 0.475 | 0.72 | 0.72 | 0.495 | 0.686 | 0.024 |
| MNB\_sentiment | 100 | train | 0.725 | 0.679 | 0.71 | 0.725 | 0.557 | 0.725 | 0.725 | 0.593 | 0.705 | 0.052 |
| MNB\_sentiment | 200 | test | 0.736 | 0.646 | 0.719 | 0.736 | 0.534 | 0.736 | 0.736 | 0.559 | 0.719 | 0.0065 |
| MNB\_sentiment | 200 | train | 0.757 | 0.702 | 0.748 | 0.757 | 0.629 | 0.757 | 0.757 | 0.657 | 0.749 | 0.0085 |
| MNB\_sentiment | 400 | test | 0.746 | 0.73 | 0.743 | 0.746 | 0.579 | 0.746 | 0.746 | 0.618 | 0.734 | 0.036 |
| MNB\_sentiment | 400 | train | 0.792 | 0.753 | 0.788 | 0.792 | 0.687 | 0.792 | 0.792 | 0.713 | 0.788 | 0.064 |
| MNB\_sentiment | 1000 | test | 0.74 | 0.727 | 0.738 | 0.74 | 0.546 | 0.74 | 0.74 | 0.579 | 0.724 | 0.048 |
| MNB\_sentiment | 1000 | train | 0.859 | 0.843 | 0.857 | 0.859 | 0.779 | 0.859 | 0.859 | 0.806 | 0.857 | 0.068 |
| MNB\_sentiment | 1500 | test | 0.742 | 0.691 | 0.734 | 0.742 | 0.536 | 0.742 | 0.742 | 0.56 | 0.724 | 0.056 |
| MNB\_sentiment | 1500 | train | 0.882 | 0.882 | 0.882 | 0.882 | 0.808 | 0.882 | 0.882 | 0.839 | 0.88 | 0.084 |

Table 1

From the table above, all classifiers seem perform better when predicting the classes of training sets because these data is used to train the models. When N word is 100, all sentiment classifiers have similar performance. When the word is doubled, the performance for decision tree sentiment classification is remain rather similar but is marginally increased for the two Naïve Bayes methods. Overall, for sentiment classification, the three models perform equally well. This is likely because, there are only three predictable classes for sentiment, namely *negative, neutral and positive.*

For DT classifiers, increasing the top N words beyond 200 did not improve the performance any further. BNB\_topics and BNB\_sentiment performance leveled off after 900 and 1000 top words respectively. As for MNB classifiers, increasing the word beyond 1000, had contradicting effects – the test set accuracy decreased instead of increasing together with the training set accuracy. Regarding runtimes, decision tree classifiers are the slowest of all classifiers. This is as expected because DT models are known to be more complex for certain domains than NB models.

In a multiclass problem such as voting topics classification, each class is not equally important because I believe that voters did not concern themselves with every topic being discussed. Thus, in my opinion, micro setting is more suited because it gives each observation an equal weight rather than macro setting which gives each class an equal weight which may not necessarily be true in Federal Election because some matters were more pressing than others. Therefore, F1\_MI (micro) metric is considered, comparing the performance of topics classifiers, because it produces a high result when precision and recalled is balanced. The F1 scores show that in general NB models are better at classifying topics that DT models.

1. (2 marks) Evaluate the standard models with respect to baseline predictors (**VADER** for sentiment analysis, majority class for both classifiers). Comment on the performance of the baselines and of the methods relative to the baselines.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Baseline** | **Accuracy** | **F1 Micro Avg** | **F1 Macro Avg** | **F1 Weighted** | **Runtime (sec)** |
| Majority class topics | 0.174 | 0.17 | 0.01 | 0.05 | 0.0130 |
| Majority class sentiment | 0.670 | 0.67 | 0.27 | 0.54 | 0.0110 |
| VADER sentiment | 0.430 | 0.43 | 0.37 | 0.48 | 0.2200 |

Table 2 performance of the baseline classifiers

Table 2 above shows the performance of the baseline classifiers. Both majority class classifiers were trained using first 1500 tweets and tests against last 500 tweets. VADER was also used to predict the last 500 tweets. The results show that the performance majority class topics classifier was very poor, and that majority class sentiment classifier performed better than VADER.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Standard model** | **Vocab size** | **Accuracy** | **F1 Micro** | **F1 Macro** | **F1 Weighted** |
| DT\_topics | 200 | 0.296 | 0.30 | 0.17 | 0.27 |
| BNB\_topics | All (6907) | 0.178 | 0.18 | 0.02 | 0.06 |
| MNB\_topics | All (6907) | 0.290 | 0.29 | 0.12 | 0.25 |
| DT\_sentiment | 200 | 0.672 | 0.67 | 0.40 | 0.62 |
| BNB\_sentiment | All (6907) | 0.716 | 0.72 | 0.40 | 0.65 |
| MNB\_sentiment | All (6907) | 0.73 | 0.73 | 0.52 | 0.71 |

Table 3 performance of six standard models

Table 3 shows the performance of six standard models. These six models clearly outperformed the baseline models in all metrics. BNB\_topics and MNB\_topics would perform marginally better than majority class baseline when the vocab size is smaller (refer to table 1).

1. (2 marks) Evaluate the effect that preprocessing the input features, in particular stop word removal plus Porter stemming as implemented in **NLTK**, has on classifier performance, for the three standard methods for both sentiment and topic classification. Compare results with and without preprocessing on training and test sets and comment on any similarities and differences.



Figure Preprocessing performance comparison on testing set

The figure above was plotted using the data in Appendix A with top 200 words. The metric chosen was F1 micro average because the cost of false positives and false negatives are similar, and the class distribution is imbalance. Mirco metrics was chosen because not all topics deserved equal attentions during federal election. The party who wanted to get elected should rather focus on topics that were most concerned.

The figure shows every topic classifier benefitted from stop word removal and Porter stemming especially BNB\_topics which benefited the most from preprocessing. Without preprocessing, BNB\_topics performed slightly worse than MNB\_topics and slightly better. But with preprocessing, BNB\_topics became the best performer among the three topics classifiers. Sentiment classifiers seem to suffer from preprocessing although not by much. Preprocessing had no significant effect on MNB\_sentiment.



Figure Preprocessing performance comparison on training set

Like figure 3, the figure above is plotted using data in Appendix A with top 200 words but instead the model was used to predict the training set. The figure shows that with preprocessing, all topics classifiers performed marginally better than without preprocessing. However, for sentiment classifiers the results are mixed. DT\_sentiment performance was worse with preprocessing but BNB\_sentiment was generally better with preprocessing while MNB\_sentiment remained roughly the same.

1. (2 marks) Sentiment classification of neutral tweets is notoriously difficult. Repeat the experiments of items 2 (with N = 200), 3 and 4 for sentiment analysis with the standard models using only the positive and negative tweets (i.e. removing neutral tweets from both training and test sets). Compare these results to the previous results. Is there any difference in the metrics for either of the classes (i.e. consider positive and negative classes individually)?

From the table in Appendix B, all metrics precision, highlighted in yellow, recall and f1-score increased after the neutral tweets were removed from both the training set and the testing set. For negative metrics, all the three models achieved over 90% however the effects were mixed for positive metrics. MNB positive metrics, highlighted in blue, saw a rise in recall but a decrease in precision, thus saw the overall increase in f1-score. Positive metrics in other classifiers all increased.

|  |
| --- |
| precision recall f1-score support precision recall f1-score support  negative 0.89 1.00 0.94 335 negative 0.67 1.00 0.80 335  positive 0.00 0.00 0.00 40 neutral 0.00 0.00 0.00 125  positive 0.00 0.00 0.00 40 |

Table Majority classifier metrics for sentiment with and without neutral tweets

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| VADER on test set | precision | recall | f1-score | support |
| negative | 0.95 | 0.46 | 0.62 | 335 |
| neutral | 0 | 0 | 0 | 0 |
| positive | 0.2 | 0.62 | 0.3 | 40 |
| micro avg | 0.47 | 0.47 | 0.47 | 375 |
| macro avg | 0.38 | 0.36 | 0.31 | 375 |
| weighted avg | 0.87 | 0.47 | 0.58 | 375 |

Table VADER results on the test sets

The table above shows the result of majority classifiers which is as expected performed better now that most of the tweets were predominantly negative tweets. VADER still classified some tweets as neutral even though all neutral tweets were removed. Comparing both tables with the table in Appendix B, shows that without neutral tweets in the dataset, all standard classifiers performed better than the baselines.

As for stemming and stop words removal results, by comparing table in Appendix C with Appendix B, it can be concluded that their performances were worse after preprocessing. This agrees with previous results, ones with neutral tweets intact. Thus, stemming and stop words removal were not helpful for sentiment classification.

1. (6 marks) Describe your best method for sentiment analysis and your best method for topic classification. Give some experimental results showing how you arrived at your methods. Now provide a brief comparison of your methods in relation to the standard methods and the baselines.

For my topic model, I decided to use MNB model with NLTK stop words and porter stemming because it was clear from question 4 that these preprocessing had positive effects on topic classification. In addition to standard model’s text preprocessing, more were performed:

* all characters to lowercase
* remove words that started with ***#aus****,* ***au***, and ***#ele*** as all tweets are regarding Australian Federal Election and every tweet has #auspol

To experiment various parameters that MNB classifier and CountVectorizer offer, hyperparameter tuning and cross-validation were performed. Sklearn has GridSearchCv class that does parameter searching and cross validation, so I used that with the following parameters.

* 'clf\_\_alpha': (0.25, 0.5, 0.6, 0.75, 1.0) – classifier smoothing parameter
* 'vect\_\_max\_df': (0.2, 0.4, 0.6, 0.8, 1.0) – vectorizer max document frequency
* 'vect\_\_max\_features': (600, 700, 800, 1000, 1200, 1400) – vectorizer top vocabulary
* 'vect\_\_min\_df': (1, 2, 3, 4, 5) – vectorizer min document frequency
* 'vect\_\_ngram\_range': ((1, 1), (1, 2)) – unigram or bigram model

For experiment, first 1500 tweets were used to build vocabulary dictionary via CountVectorizer and the last 500 were used as test sets. I had a rough guess of the lower bound and upper bound of top N words, 400 and 1200 respectively. *Ngram\_range (min,max)* parameter of CountVectorizer was set to (1,2) and *min\_df* was set to 0.

|  |  |  |  |
| --- | --- | --- | --- |
| Dataset | Top N | F1-score micro average | Cross Validation F1-score 6 folds |
| Testing set | 400 | 0.44 | 0.44 (+/- 0.06) |
| Training set | 400 | 0.68 |
| Testing set | 1200 | 0.45 | 0.46 (+/- 0.05) |
| Training set | 1200 | 0.76 |
| Testing set | 818 | 0.45 | 0.46 (+/- 0.04) |
| Training set | 818 | 0.75 |

Table Own topic model results

Particle Swarm Optimization was used to optimize the following parameters [lower bound, upper bound]:

1. Top N words [400,1200]
2. ngram\_range (max) [2,3]
3. min\_df [0,5]

PSO results were 818 top N words, max ngram\_range of 2 (unigram and bigram) and lastly 0 for min\_df. The scores are highlighted in yellow on the table above. The model’s f1-score (0.45) was higher than the highest test f1-score micro average, 0.356, (table 1) and higher than baseline (table 2).

As for my sentiment model, I chose MNB to improve on.

# Appendix A

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Classifier | Stem & Stop | Accuracy | F1\_Micro | F1\_Macro | F1\_Weighted |
| DT\_topics | yes | 0.348 | 0.35 | 0.22 | 0.33 |
| DT\_sentiment | yes | 0.654 | 0.65 | 0.39 | 0.62 |
| BNB\_topics | yes | 0.458 | 0.46 | 0.27 | 0.43 |
| BNB\_sentiment | yes | 0.708 | 0.71 | 0.42 | 0.66 |
| MNB\_topics | yes | 0.44 | 0.44 | 0.27 | 0.42 |
| MNB\_sentiment | yes | 0.728 | 0.73 | 0.57 | 0.72 |
| DT\_topics | no | 0.3 | 0.3 | 0.164 | 0.27 |
| DT\_sentiment | no | 0.672 | 0.672 | 0.405 | 0.634 |
| BNB\_topics | no | 0.322 | 0.322 | 0.176 | 0.297 |
| BNB\_sentiment | no | 0.712 | 0.712 | 0.519 | 0.696 |
| MNB\_topics | no | 0.32 | 0.32 | 0.182 | 0.3 |
| MNB\_sentiment | no | 0.736 | 0.736 | 0.559 | 0.719 |

Table preprocessing performance metrics testing set

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Classifier | Stem & Stop | Accuracy | F1\_Micro | F1\_Macro | F1\_Weighted |
| DT\_topics | yes | 0.404 | 0.4 | 0.21 | 0.37 |
| DT\_sentiment | yes | 0.687 | 0.69 | 0.39 | 0.62 |
| BNB\_topics | yes | 0.546 | 0.55 | 0.37 | 0.53 |
| BNB\_sentiment | yes | 0.853 | 0.84 | 0.56 | 0.8 |
| MNB\_topics | yes | 0.556 | 0.56 | 0.44 | 0.54 |
| MNB\_sentiment | yes | 0.761 | 0.76 | 0.66 | 0.75 |
| DT\_topics | no | 0.385 | 0.385 | 0.221 | 0.357 |
| DT\_sentiment | no | 0.699 | 0.699 | 0.482 | 0.667 |
| BNB\_topics | no | 0.509 | 0.509 | 0.356 | 0.492 |
| BNB\_sentiment | no | 0.751 | 0.751 | 0.653 | 0.746 |
| MNB\_topics | no | 0.533 | 0.533 | 0.43 | 0.525 |
| MNB\_sentiment | no | 0.757 | 0.757 | 0.657 | 0.749 |

Table preprocessing performance metrics training set

# Appendix B

|  |  |
| --- | --- |
| Without neutral | With neutral |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | DT | precision | recall | f1-score | support | | negative | 0.91 | 0.99 | 0.95 | 335 | | positive | 0.62 | 0.2 | 0.3 | 40 | | micro avg | 0.9 | 0.9 | 0.9 | 375 | | macro avg | 0.76 | 0.59 | 0.62 | 375 | | weighted avg | 0.88 | 0.9 | 0.88 | 375 | |  |  |  |  |  | |  | precision | recall | f1-score | support | | negative | 0.92 | 0.98 | 0.95 | 959 | | positive | 0.65 | 0.25 | 0.36 | 113 | | micro avg | 0.91 | 0.91 | 0.91 | 1072 | | macro avg | 0.78 | 0.62 | 0.65 | 1072 | | weighted avg | 0.89 | 0.91 | 0.89 | 1072 | | |  |  |  |  |  | | --- | --- | --- | --- | --- | | DT | precision | recall | f1-score | support | | negative | 0.74 | 0.88 | 0.8 | 335 | | neutral | 0.41 | 0.33 | 0.37 | 125 | | positive | 0.2 | 0.03 | 0.04 | 40 | | micro avg | 0.67 | 0.67 | 0.67 | 500 | | macro avg | 0.45 | 0.41 | 0.4 | 500 | | weighted avg | 0.62 | 0.67 | 0.63 | 500 | |  | precision | recall | f1-score | support | | negative | 0.74 | 0.9 | 0.81 | 959 | | neutral | 0.57 | 0.41 | 0.48 | 428 | | positive | 0.42 | 0.1 | 0.16 | 113 | | micro avg | 0.7 | 0.7 | 0.7 | 1500 | | macro avg | 0.58 | 0.47 | 0.48 | 1500 | | weighted avg | 0.67 | 0.7 | 0.67 | 1500 | |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | BNB | precision | recall | f1-score | support | | negative | 0.93 | 0.97 | 0.95 | 335 | | positive | 0.56 | 0.35 | 0.43 | 40 | | micro avg | 0.9 | 0.9 | 0.9 | 375 | | macro avg | 0.74 | 0.66 | 0.69 | 375 | | weighted avg | 0.89 | 0.9 | 0.89 | 375 | |  |  |  |  |  | |  | precision | recall | f1-score | support | | negative | 0.94 | 0.96 | 0.95 | 959 | | positive | 0.61 | 0.51 | 0.56 | 113 | | micro avg | 0.91 | 0.91 | 0.91 | 1072 | | macro avg | 0.78 | 0.74 | 0.76 | 1072 | | weighted avg | 0.91 | 0.91 | 0.91 | 1072 | | |  |  |  |  |  | | --- | --- | --- | --- | --- | | BNB | precision | recall | f1-score | support | | negative | 0.79 | 0.86 | 0.82 | 335 | | neutral | 0.55 | 0.53 | 0.54 | 125 | | positive | 0.5 | 0.23 | 0.31 | 40 | | micro avg | 0.72 | 0.72 | 0.72 | 500 | | macro avg | 0.62 | 0.54 | 0.56 | 500 | | weighted avg | 0.71 | 0.72 | 0.71 | 500 | |  | precision | recall | f1-score | support | | negative | 0.8 | 0.86 | 0.83 | 959 | | neutral | 0.66 | 0.59 | 0.62 | 428 | | positive | 0.65 | 0.49 | 0.56 | 113 | | micro avg | 0.76 | 0.76 | 0.76 | 1500 | | macro avg | 0.7 | 0.65 | 0.67 | 1500 | | weighted avg | 0.75 | 0.76 | 0.75 | 1500 | |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | MNB | precision | recall | f1-score | support | | negative | 0.92 | 0.97 | 0.94 | 335 | | positive | 0.52 | 0.3 | 0.38 | 40 | | micro avg | 0.9 | 0.9 | 0.9 | 375 | | macro avg | 0.72 | 0.63 | 0.66 | 375 | | weighted avg | 0.88 | 0.9 | 0.88 | 375 | |  |  |  |  |  | |  | precision | recall | f1-score | support | | negative | 0.94 | 0.97 | 0.95 | 959 | | positive | 0.64 | 0.5 | 0.56 | 113 | | micro avg | 0.92 | 0.92 | 0.92 | 1072 | | macro avg | 0.79 | 0.74 | 0.76 | 1072 | | weighted avg | 0.91 | 0.92 | 0.91 | 1072 | | |  |  |  |  |  | | --- | --- | --- | --- | --- | | MNB | precision | recall | f1-score | support | | negative | 0.79 | 0.88 | 0.83 | 335 | | neutral | 0.58 | 0.52 | 0.55 | 125 | | positive | 0.57 | 0.2 | 0.3 | 40 | | micro avg | 0.74 | 0.74 | 0.74 | 500 | | macro avg | 0.65 | 0.53 | 0.56 | 500 | | weighted avg | 0.72 | 0.74 | 0.72 | 500 | |  | precision | recall | f1-score | support | | negative | 0.8 | 0.87 | 0.83 | 959 | | neutral | 0.66 | 0.58 | 0.62 | 428 | | positive | 0.64 | 0.43 | 0.52 | 113 | | micro avg | 0.76 | 0.76 | 0.76 | 1500 | | macro avg | 0.7 | 0.63 | 0.66 | 1500 | | weighted avg | 0.75 | 0.76 | 0.75 | 1500 | |

Table Class metrics for sentiment with and without neutral tweets

# Appendix C

|  |
| --- |
| Neutral tweets removal + stem & stop words removal |
| |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | DT | precision | | recall | | f1-score | | support | | | negative | 0.89 | 1 | | 0.94 | | 335 | | | positive | 0 | 0 | | 0 | | 40 | | |  |  |  | |  | |  | | | micro avg | 0.89 | 0.89 | | 0.89 | | 375 | | | macro avg | 0.45 | 0.5 | | 0.47 | | 375 | | | weighted avg | 0.8 | 0.89 | | 0.84 | | 375 | | |  | precision | recall | | f1-score | | support | | | negative | 0.89 | 1 | | 0.94 | | 959 | | | positive | 0 | 0 | | 0 | | 113 | | |  |  |  | |  | |  | | | micro avg | 0.89 | 0.89 | | 0.89 | | 1072 | | | macro avg | 0.45 | 0.5 | | 0.47 | | 1072 | | | weighted avg | 0.8 | 0.89 | | 0.84 | | 1072 | | |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | BNB | precision | recall | f1-score | support | | negative | 0.92 | 0.97 | 0.95 | 335 | | positive | 0.57 | 0.3 | 0.39 | 40 | |  |  |  |  |  | | micro avg | 0.9 | 0.9 | 0.9 | 375 | | macro avg | 0.75 | 0.64 | 0.67 | 375 | | weighted avg | 0.88 | 0.9 | 0.89 | 375 | |  | precision | recall | f1-score | support | | negative | 0.95 | 0.97 | 0.96 | 959 | | positive | 0.69 | 0.53 | 0.6 | 113 | |  |  |  |  |  | | micro avg | 0.93 | 0.93 | 0.93 | 1072 | | macro avg | 0.82 | 0.75 | 0.78 | 1072 | | weighted avg | 0.92 | 0.93 | 0.92 | 1072 | |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | MNB | precision | recall | f1-score | support | | negative | 0.92 | 0.96 | 0.94 | 335 | | positive | 0.5 | 0.3 | 0.37 | 40 | |  |  |  |  |  | | micro avg | 0.89 | 0.89 | 0.89 | 375 | | macro avg | 0.71 | 0.63 | 0.66 | 375 | | weighted avg | 0.88 | 0.89 | 0.88 | 375 | |  | precision | recall | f1-score | support | | negative | 0.95 | 0.98 | 0.96 | 959 | | positive | 0.73 | 0.54 | 0.62 | 113 | |  |  |  |  |  | | micro avg | 0.93 | 0.93 | 0.93 | 1072 | | macro avg | 0.84 | 0.76 | 0.79 | 1072 | | weighted avg | 0.93 | 0.93 | 0.93 | 1072 | |

Table Class preprocessed metrics for sentiment with no neutral tweets