JSC270 A4

Amane Takeuchi, Jessica Wang, Luqiong Xie

September 8, 2023

Colab Notebook Link:

- Part I: `https://colab.research.google.com/drive/1whgP8d46VWvolAXvP0CzME7p3oQnU-DQ?usp=sharing`
- Part II: `https://colab.research.google.com/drive/1R-5MVrYxoK2_a0jgVRHesc5Kz4SIwGqv?usp=sharing`

Contributions:

Amane Takeuchi: Part 1: question I, J, K, L(Bonus). Part 2: Report: model description, Results and conclusion. Review and make amendments to other sections of the report. Part 3: Make slides.

Jessica Wang: Part 1: question A, B, C, D, E, F, G, H (work together with Luqiong Xie). Part 2: Create and test models (Work toghether with Luqiong Xie). Report: Problem Description and Motivation, Description of the Data. Review and make amendments to other sections of the report. Part 3: Making slides.

Luqiong Xie: Part 1: question A, B, C, D, E, F, G, H (work together with Jessica Wang). Part 2: Create and test models (Work toghether with Jessica Wang). Report: Exploratory Data Analysis. Review and make amendments to other sections of the report. Part 3: Making slides.

1 Part I

For preprocessing, we first checked the number of observations in each sentiment category (shown below). We can see that the number of "Extremely Positive" and "Extremely Negative" tweets is relatively lower than the number of "Positive", "Negative" and "Neutral" tweets. Therefore, we decided to group "Extremely Positive" and "Positive" tweets together for simplicity. Similarly, we grouped "Extremely Negative" and "Negative" tweets together. Specifically, we created a new "label" column. In this column, we assigned both "Negative" and "Extremely Negative" a label 0, assigned both "Positive" and "Extremely Positive" a label 2 and assigned "Neutral" a label 1.

- Number of Positive tweets 11422
- Number of Negative tweets 9917
- Number of Neutral tweets 7713
- Number of Extremely Positive tweets 6624
- Number of Extremely Negative tweets 5481

A. Based on the results obtained below, we can see that there is a greater proportion of negative and positive tweets compared to neutral tweets. Specifically, approximately 37.4% of the tweets in the dataset are negative, 43.8% are positive while only 18.7% are neutral. One possible justification for this class imbalance is that people tend to express more extreme opinions or emotions about the pandemic on social media. Neutral posts that do not express a strong opinion might be considered as not worth posting. Another factor that we should consider is how the tweets are collected and labeled. For example, the labeling process might be biased towards more extreme labels than neutral labels, resulting in a class imbalance.

   – proportion for label 0 is: 0.374

- proportion for label 1 is: 0.187
- proportion for label 2 is: 0.438

B. We chose to tokenize the tweets using the WhitespaceTokenizer from the NLTK Tokenizer Package. This tokenizer works by splitting a string of text into tokens based on whitespace characters such as space, tab, and newline character. We opted for this approach because it is straightforward and perfectly achieves the required task. Additionally, we believe that since the NLTK Tokenizer Package has been extensively tested and refined, it is less error-prone compared to writing our own function from scratch.

C. To identify the URLs in the tweets, we used the regular expression "∧ http.*". This regular expression matches any token that begins with "http" and is followed by zero or more characters, which should correctly identify all the URL tokens. We then checked each token and removed any token that matched the regular expression.

D. While we removed all forms of punctuation from the tokens in our analysis, there are some scenarios where keeping certain punctuation marks may be important. For example, the dollar sign $ can indicate that a number represents an amount of money, which could be relevant in financial or business-related analyses. Similarly, the hashtag symbol # and the at symbol @ have special meanings in Twitter, representing a hashtag or a mention of a user respectively. These symbols can impact the meaning of the words that follow them and potentially influence the results of sentiment analysis. Thus, in some cases, it may be important to preserve certain forms of punctuation to ensure accurate analysis of the text.

E. We used the Porter Stemmer to stem the tokens. We selected this stemmer because it is a well-established and widely used stemming algorithm. Also, the Porter Stemmer has been shown to be effective in many NLP tasks including sentiment analysis.

F. As stopwords typically do not carry much meaning, we removed the first hundred

stopwords as required. By removing stopwords, we were able to reduce the noise in the data and focus our analysis on the most meaningful information. This was especially important for question H, where we were asked to identify the top 5 most probable words in each category. If we had not removed the stopwords, it is very likely that certain stopwords would have been included in the top 5 most probable words.

G. Using the method get_feature_names_out(), we identified the number of features to be 52,573.

H. The training accuracy of the Naive Bayes model is approximately 79.53% and the test accuracy of the Naive Bayes model is approximately 67.46%.

The top 5 words in the negative class are:

- coronaviru (prob: 0.0174, count: 6703.0)
- covid19 (prob: 0.0126, count: 4862.0)
- price (prob: 0.0113, count: 4332.0)
- food (prob: 0.0094, count: 3623.0)
- thi (prob: 0.0083, count: 3206.0)

The top 5 words in neutral class is:

- coronaviru (prob: 0.0218, count: 3792.0)
- covid19 (prob: 0.0158, count: 2752.0)
- store (prob: 0.0091, count: 1581.0)
- supermarket (prob: 0.0082, count: 1436.0)
- price (prob: 0.0078, count: 1361.0)

The top 5 words in positive class is:

- coronaviru (prob: 0.0168, count: 7467.0)

- covid19 (prob: 0.0135, count: 6003.0)
- store (prob: 0.0088, count: 3896.0)
- thi (prob: 0.0085, count: 3772.0)
- price (prob: 0.0075, count: 3323.0)

I. It is not appropriate to fit a ROC curve in this particular situation. This is because the ROC curve is usually used for binary classification models but we have a multi-class classification problem with three classes. Instead, we can perform other visualizations such as a confusion matrix to examine how well our model is performing for each class. For example, scikit-learn provides tools for computing and visualizing confusion matrix. In addition, the ROC curve is usually used for models that assign a threshold probability for classification. However, the Naive Bayes model is not a threshold-based classifier. Thus, we should not fit a ROC curve as it would not give any informative result.

J. The training accuracy of the Naive Bayes model with TF-IDF vectorization is approximately 72.01% and the test accuracy of the Naive Bayes model is approximately 62.72%. The number of features is 52,573.

The top 5 words in the negative class are:

- coronaviru (prob: 0.0041, count: 473.08679032512487)
- price (prob: 0.0035, count: 403.4152452476971)
- covid19 (prob: 0.0033, count: 387.8076170090054)
- food (prob: 0.0033, count: 379.5421983550111)
- thi (prob: 0.0027, count: 308.5946231571708)

The top 5 words in neutral class are:

- coronaviru (prob: 0.0040, count: 317.9205746138815)

- covid19 (prob: 0.0034, count: 266.69514503571935)
- store (prob: 0.0025, count: 200.98410778385693)
- supermarket (prob: 0.0022, count: 177.73960977384968)
- groceri (prob: 0.0022, count: 174.8987923653582)

The top 5 words in positive class are:

- coronaviru (prob: 0.0041, count: 521.0028742886448)
- covid19 (prob: 0.0037, count: 470.93729284906846)
- store (prob: 0.0031, count: 395.5931199113576)
- thi (prob: 0.0029, count: 366.4229132449829)
- groceri (prob: 0.0028, count: 356.0795608274181)

Notice that TF-IDF vectors calculate the TF-IDF score by dividing the number of times a word appears in a document by the number of documents the word appears. From the above result, it's clear that the accuracy has dropped. However, the top 5 vocabularies are very similar. This is because the TF-IDF score is high when the numerator which is the count of a word is high. Thus, there is no significant difference in results compared to part H other than the drop in accuracy. The drop in accuracy is caused by the nature of the TF-IDF score it gives a high score to a word that appears less frequently. Therefore, when the model notices a rare word in a tweet, it interprets it significantly. However, this is not true as the rare wording does not affect the sentiment analysis.

K. The training accuracy of the Naive Bayes model with TF-IDF vectorization and lemmatization is approximately 72.01% and the test accuracy of the Naive Bayes model is approximately 62.77%. The number of features is 52,483.
The top 5 words in the negative class are:

- coronaviru (prob: 0.0041, count: 473.301447132673)

- price (prob: 0.0035, count: 403.6696793106982)
- covid19 (prob: 0.0033, count: 387.97610758331126)
- food (prob: 0.0033, count: 379.79527312066034)
- thi (prob: 0.0027, count: 308.7540596134214)

The top 5 words in neutral class is:

- coronaviru (prob: 0.0040, count: 318.01763644113964)
- covid19 (prob: 0.0034, count: 266.7754350460448)
- store (prob: 0.0025, count: 201.11326158002367)
- supermarket (prob: 0.0023, count: 177.80683212887513)
- groceri (prob: 0.0022, count: 174.99592444847855)

The top 5 words in positive class is:

- coronaviru (prob: 0.0041, count: 521.2457104340449)
- covid19 (prob: 0.0037, count: 471.1488783586833)
- store (prob: 0.0031, count: 395.81415084943205)
- thi (prob: 0.0029, count: 366.6206995481537)
- groceri groceri (prob: 0.0028, count: 356.28765158923255)

Notice from the above result that there is no big difference compared to the result in part J. Although there is a subtle increase in the test accuracy compared to part J, the stemming and lemmatization make no big difference in the result. This is because though lemmatization preserves the context of the word, the stemming method also does not change the meaning of words that are in the top 5 of each class. Thus, we can barely observe a difference between the two methods of conversion.

L. (Bonus) Notice that the Naive Bayes model is a generative model where it tries to make a prediction by using the Bayes rule to calculate the probability of a label, y was given x using the joint probability distribution $p(x, y)$ assuming all the features are conditionally independent. Meanwhile, discriminative models try to discover posterior probability distribution directly from the features x. An example of a discriminative model would be logistic regression as this model makes no assumption on inputs.

2 Part II

Problem Description and Motivation

The main objective of this project is to predict the level of popularity a post might achieve after being posted on social media platforms such as Twitter. According to a report, global social media ad spending is expected to reach $223 billion in 2023 (Dimitrievski, 2023). Naturally, organizations aim to maximize their returns on investment by focusing on posts with the potential to become popular and reach a wider audience. Similarly, political candidates can benefit from accurate predictions of content popularity to gain broader support through social media outreach. However, solving this problem can be challenging given the ever-changing landscape of social media trends. To tackle this question, we extracted recent political tweets from Twitter and used them to train Naive Bayes models to identify popular political topics. Additionally, our models can be generalized to predict the popularity of a tweet given its content. In our literature review, we discovered, existing studies that use retweet count as a measure of popularity (Mahdikhani, 2021). Our analysis aims to differentiate itself by considering retweets, favourites and follower counts together to develop a new metric for measuring popularity.

Description of the Data

Ten thousands political tweets were extracted using Twitter's API with the search term '#cdnpoli'. This search term is a hashtag commonly used by CBC. To minimize duplicated posts, retweets were removed from the results. For each tweet, we extracted the content, favourite count, retweet count, user statuses count, user follower count and username. However, we focused our analysis on retweet count, favourite count, and user follower count as we believe these factors are most relevant to post popularity. One limitation of our data is the use of a single search term. This might result in tweets covering similar topics. Consequently, topics popular among these tweets might not represent the broader Twitter com-

munity's preferences. Furthermore, 10,000 tweets might not be sufficient to produce highly accurate models. In a related study, more than 1.25 million English tweets were employed (Mahdikhani, 2021). However, given our limited computational resources, we decided that 10,000 observations strike a balance between meaningful results and efficiency. A strength of our dataset is that it was extracted using Twitter's official API. This ensures that the data is accurate, up-to-date, and collected with ethical considerations in mind.

Exploratory Data Analysis

We used retweet count (mean 7.45, std 46.87, min 0.00, max 1562.00), favourite count(mean 25.59, std 163.86, min 0.00, max 4889.00), and user followers count (mean 7752.49, std 27802.22, min 1.00, max 755203.00) in the project (Appendix, Figure 5). Given the severe skewness in the data, we applied the log transformation to better observe the distribution. Figure 1 shows that after the log transformation, the retweet count and the favourite count are still right skewed while the user followers count is now normally distributed. Due to the severe skewness, we observed 1926 outliers for the retweet count, 1582 outliers for the favourite count, and 1267 outliers for the user followers count. All the outliers are kept because we want our model to study these extremely popular tweets in order to make predictions. There is also no missing data so no data is removed from the dataset.

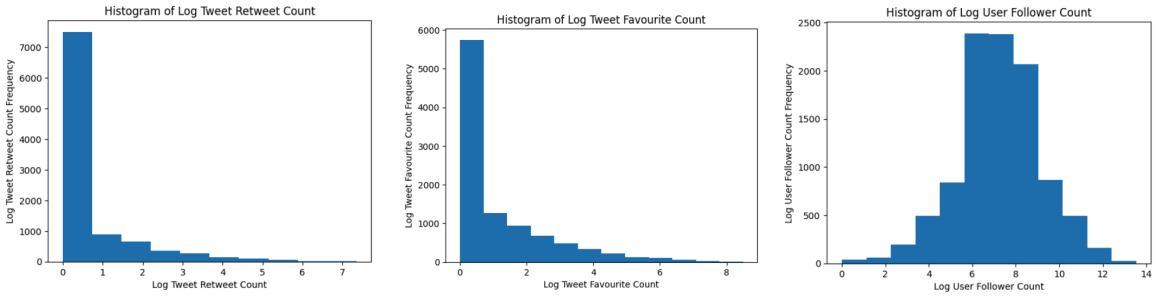

Figure 1: Distribution of log tweet retweet count, log tweet favourite count and log user followers count.

Since the objective is to identify the type of tweets that would get more attention, we for-

mulated the following function to mathematically represent popularity:

$$\text{Popularity score} = 10 \times \frac{(\text{Retweet count} + \text{Favourite count})}{\text{Number of followers}}$$

Intuitively, popularity level of a tweet can be represented by retweet count and favourite count. However, these values can be extraordinarily high for users with large number of followers. Therefore, the score is normalized by dividing it by the number of followers the user has to prevent tweets from popular accounts to have higher scores. We then preprocessed the tweet content by removing all punctuation and special characters, converting all tokens to lowercase characters, stemming the tokens, and removing stopwords as well as blank tokens.

Description of the Machine Learning Model

In this analysis, we employed the supervised machine learning model, Naive Bayes, for multi-class classification. The Naive Bayes model is a simple probabilistic model that is easy to train, performs effectively with large number of features and handles class imbalance well. However, it is essential to note that the model assumes the features to be conditionally independent of each other, which might not hold true with text classification. Then, we set the threshold value to classify the tweets into three categories. Since the majority of the data points have zero popularity, we classified them as "Not popular". Any tweets with a popularity score between 0 to the threshold would be classified as "Popular" and any tweets with a popularity score over the threshold would be classified as "Very Popular". This model classifies the tweets based on the words that are frequently used in each category. This method of analysis is consistent with the research proposal since this model predicts whether tweets could potentially gain attention in social media by their contents. This would allow us to analyze the potential effectiveness of social media marketing based on the classification.

We selected three threshold candidates, 0.2, 0.5, and 1.0. Each candidate threshold gen-

erates a distinct distribution of the label classes. Figure 2 shows the distribution of the labels for each candidate threshold. As the threshold increases, the proportion of "Very Popular" tweets decreases, and the proportion of "Popular" tweets increases.

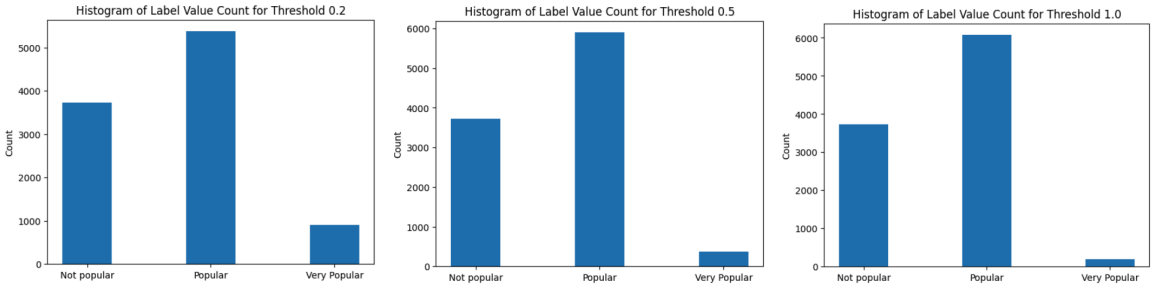

Figure 2: Distribution of different popularity classes.

The proportions of the classes are used as the baselines to evaluate the candidate thresholds. These proportions represent the precision levels that we would achieve by randomly guessing the labels based on the frequency of each class. Precision reflects the probability that a post predicted to be popular will actually become popular. This is crucial because we want to avoid situations where companies invest in posts that we predicted to have the potential to be popular, but which end up not gaining significant traction. We run our model on each of the threshold candidates and evaluate the model using proportion as the baseline. Figure 3. shows the precision vs. candidate thresholds for each of the three classes. Threshold 1.0 has the best precision for all three classes, and thus we decided to choose threshold = 1.0 to be the hyper-parameter for our model.

Results and conclusion

When running the model on the test dataset, it achieved 52.6% precision in the "Not popular" class, 66.7% precision in the "Popular" class, and 50% in the "Very Popular" class, which are higher than the proportion baselines 36.7%, 61.4%, and 1.9% (Figure 4). Particularly, the precision on the "Very Popular" class outperforms the baseline by 48.1%, which demonstrated the value of our model in predicting "Very Popular" tweets.

The model's performance on the training dataset is 90.4% precision in the "Not popular"

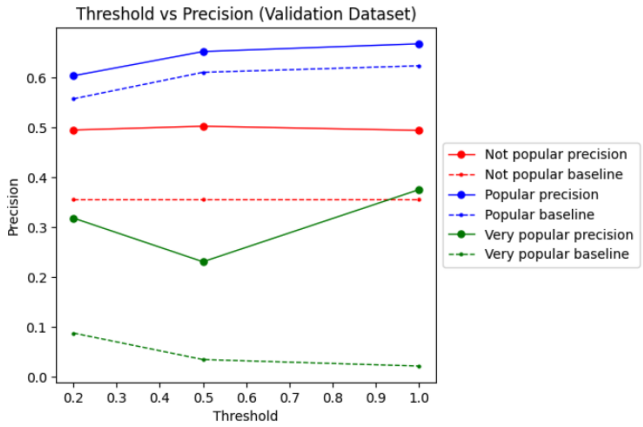

Figure 3: Precision vs. candidate thresholds on each class.

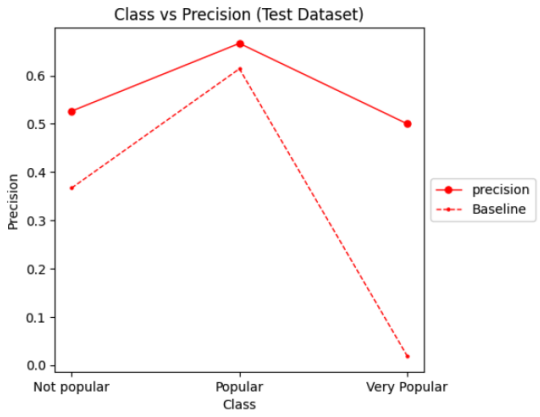

Figure 4: Precision for each class when threshold = 1.0

class, 81.8% precision in the "Popular" class, and 71.9% in the "Very Popular" class. Notice there is some overfitting to the model as the training precisions are much higher than the precision on the test dataset. One of the issues we ran into was the low precision in the "Very Popular" class before tuning the label threshold. We overcame this issue by testing multiple values for the threshold and finding the best one that produces the model with the highest precision (Figure 3). Furthermore, the top 10 words in the three categories are very similar (Appendix, Figure 6). This also indicates that the model could not discover a huge difference in the three categories. Thus, the significance of the model is that the popularity of a tweet does not solely rely on the contents of the text, but potentially on other factors as well.

Through conducting research on popularity analysis, we realized our current model does not perform well compared to the existing approach such as the Random Forest model or classifier using the Stochastic Gradient Descent(Mahdikhani, 2021). Notice these two models are discriminative models whereas the Naive Bayes model is a generative model and it assumes that features are independent of each other. This is not necessarily true with our dataset, thus generative model could have performed better for the prediction model. If we had more time to work on this project, we could have used a large language model such as chatGPT to conduct popularity analysis. This could have allowed us to make a better prediction model as GPT models are good at interpreting text data.

References

- Dimitrievski, M. (2023, February 25). 30 crucial social media marketing statistics 2023. TrueList. Retrieved April 7, 2023, from https://truelist.co/blog/social-media-marketing-statistics

- Mahdikhani, M. (2021, December 17). Predicting the popularity of tweets by analyzing public opinion and emotions in different stages of covid-19 pandemic. International Journal of Information Management Data Insights. Retrieved April 7, 2023, from https://www.sciencedirect.com/science/article/pii/S266709682100046X

Appendix

	tweet_retweet_count	tweet_favourite_count	user_followers_count	log_retweet_count	log_favourite_count	log_followers_count	popularity
count	10000.000000	10000.00000	10000.000000	10000.000000	10000.000000	10000.000000	10000.000000
mean	7.449800	25.58930	7752.491300	0.657881	1.195430	7.237223	0.129199
std	46.872183	163.85507	27802.219776	1.133834	1.489266	1.895981	1.082379
min	0.000000	0.00000	1.000000	0.000000	0.000000	0.000000	0.000000
25%	0.000000	0.00000	461.250000	0.000000	0.000000	6.136102	0.000000
50%	0.000000	1.00000	1348.000000	0.000000	0.693147	7.207119	0.006269
75%	1.000000	5.00000	4521.000000	0.693147	1.791759	8.416710	0.042143
max	1562.000000	4889.00000	755203.000000	7.354362	8.494948	13.534743	58.230088

Figure 5: Basic Summary Statistics of Tweet Retweet Count, Tweet Favourite Count, User Follower Count, log retweet count, log favourite count, log followers count, and popularity

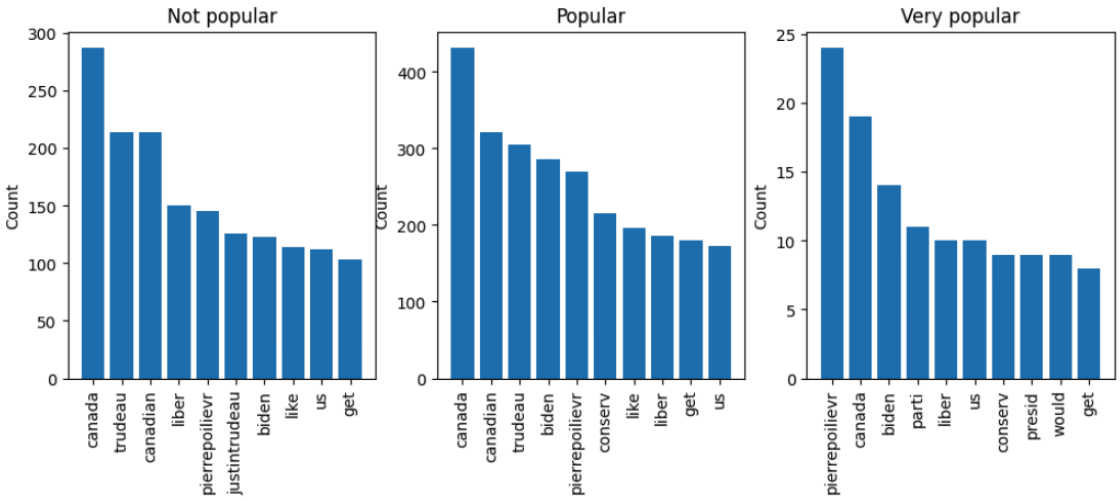

Figure 6: Frequent words used in all three classes.