

# Twitter Sentiment Analysis

Paige Gonzales

January 10, 2024

## Contents

<b>Overview</b>	<b>2</b>
Initial Project . . . . .	2
Data Source and Integrity . . . . .	2
Initial Results . . . . .	3
Project Expansion . . . . .	5
Expanded Results . . . . .	6
Text Wrangling . . . . .	6
Naïve Bayes Model Improvement . . . . .	6
ChatGPT Tweets . . . . .	7
Challenges and Further Work . . . . .	11
<b>References</b>	<b>13</b>

# Overview

This project focuses on natural language processing, specifically sentiment analysis using twitter data. First, I will discuss the initial project assignment, the data source and integrity, and the initial results. Next, I will explain my goals for improving the original project, as well as an extension of these skills to a different set of tweets that are specifically focused on ChatGPT, a product developed and maintained by the private company OpenAI (O’Sullivan, 2023). Finally, I will discuss some of the challenges I faced in this project and further work I would like to do in the subjects of natural language processing and text wrangling.

## Initial Project

This project was completed in Spring 2023 in a course focused on data mining, including data pre-processing, supervised and unsupervised learning algorithms, and model evaluation techniques. The project was completed in a group with two other people. We all independently completed the task, then compared results, ultimately submitting one file. The purpose of this project was to complete text wrangling of twitter data and construct a naïve Bayesian model that would predict whether the tweets have a negative or positive sentiment.

After training a model, we were to predict sentiment on a pre-made test set that we submitted to a [Kaggle](#) competition. The text wrangling and mining were completed using RStudio. This project focused entirely on text wrangling and naïve Bayes modeling. There was no communication of findings, data acquisition, or exploratory data analysis, which are skills I will explore further when making project enhancements.

## Data Source and Integrity

The data set for this project came directly from the professor of Data Mining in Spring 2023. The data consists of 69,992 tweets, a tweet ID number, and the sentiment of the tweet, which has been determined as either positive (1) or negative (0). The tweets contain the twitter handle of the person who posted the tweet and mentions of other twitter accounts. I don’t know who originally determined the tweets are positive or negative in the training set. I also don’t know what time frame the tweets were created, nor the method of obtaining these tweets. The test set to be submitted to the Kaggle competition is in a similar format, except it is missing the sentiment column since that will be predicted by the analyst for submission. This test set contains 29,997 individuals’ tweets. Again, I don’t know where the data comes from or when the tweets were sourced.

In the second part of this project, I will expand my text wrangling and natural language processing skills to two different data sets, both containing tweets that involve ChatGPT:

1. The first data set spans from November 30th, 2022, to December 31, 2022, and contains 219,294 instances and 11 features, where each instance represents a single tweet. Some features include a time stamp of when the tweet was created, the tweet ID, and how many likes, quotes, re-tweets and replies the tweet received. There are some other sparse features, such as the geographical location the tweet was posted in and a photo URL. Neither of these will be used in this project. The most important features for my analysis will be the time stamp and the tweet text. The individual who submitted this data set on Kaggle says they extracted the tweets using the Twitter API where the tweet contains the words “chatgpt” or “ChatGPT.” More details can be found about this data set on Kaggle [here](#).
2. The second data set spans from January 4, 2023, to March 29, 2023, so nearly the entire first quarter of 2023. The data set is quite large with 500,036 instances and 6 features, where each instances represents an individual tweet. The features include the date the tweet was posted, a tweet ID, the actual tweet text, a username, a like count, and a re-tweet count. For my project I will only be using the time stamp and the tweet content. The person who created this data set on Kaggle used a Python package called *snsscrape* to extract any tweets with keywords, hashtags or mentions including “chatgpt” or “chat gpt.”

More information about their method of scraping the data can be found [here](#). The Kaggle data set is available for download [here](#).

## Initial Results

Before any text cleaning can be done on the original twitter data, the tweets themselves need to be converted into a corpus, which is a special type of structure that converts each tweet into its only individual text document (Feinerer, 2023). Once the conversion was done, we cleaned the tweet text to prepare for analysis by using the sequence shown in Figure 1, as recommended in our textbook (?). When we first completed this project, it seemed like the text wrangling sequence didn't account for all variation in the data. For example, in Figure 1, you can see that the twitter handle, which would be a meaningless consideration for sentiment determination, is included. In addition, any portion of a hyperlink is still included by the end of the sequence, creating words that are irrelevant, such as "com" or "twitpic." I was curious if this was the most efficient and logical way of text cleaning for this data but left it as-is to follow along with the textbook. The text wrangling sequence is something I will focus improving on in the project enhancements.

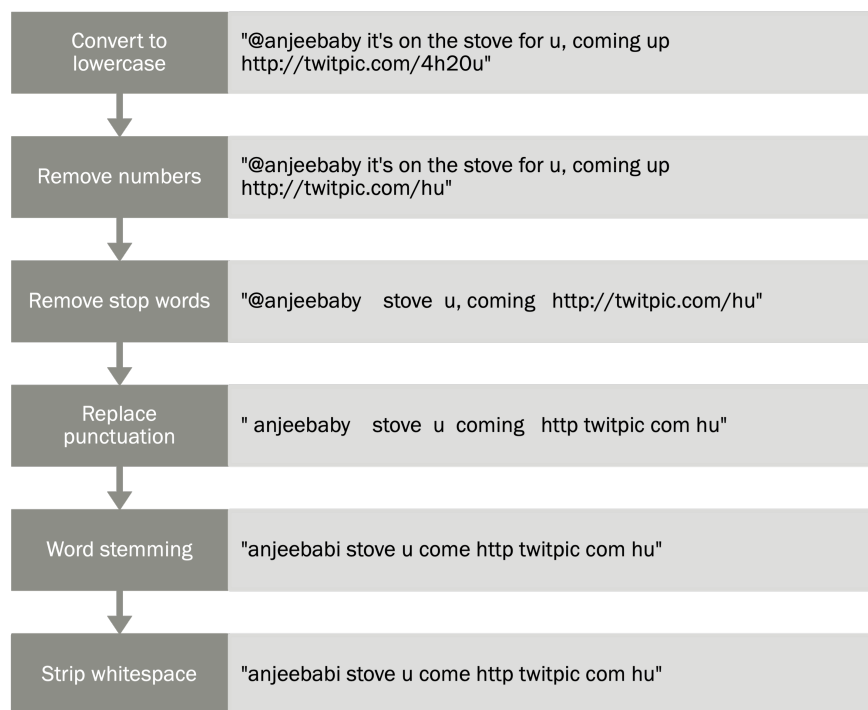


Figure 1: Original Text Wrangling Sequence With Example Tweet

When we finished text wrangling, the documented was converted from a corpus to a document term matrix. To be brief, the document term matrix will look like a very large sparse matrix, where the rows are documents, or tweets, and the columns are words within the tweets. Each cell will be filled in with the count of how many times that words appears in the tweet. Figure 2 shows a snapshot of the full data set represented as a document term matrix. The document term matrix has 69,992 documents, one for each tweet. There are 32,314 terms, so 32,314 words exist among all the tweets. This makes a huge matrix of over 2.3 billion cells, where only 0.01% of the cells have actual values. The **Docs** in the table are a tweet document number, while the 0s are the absence of the words in each column, so document 18690 contains none of the 10 words shown, while document 19537 has the word "quot" 10 times.

```

<<DocumentTermMatrix (documents: 69992, terms: 32314)>>
Non-/sparse entries: 447300/2261274188
Sparsity           : 100%
Maximal term length: 116
Weighting          : term frequency (tf)
Sample            :

```

Docs	dont	get	good	just	like	lol	love	now	quot	thank
18690	0	0	0	0	0	0	0	0	0	0
19537	0	0	0	0	0	0	0	0	10	0
32188	0	0	0	0	0	0	0	0	0	0
36559	0	0	0	0	0	0	0	0	6	0
39136	0	0	0	0	0	0	0	0	6	0
42825	0	0	1	0	1	0	0	0	8	0
59159	0	0	1	0	1	0	1	1	8	0
8636	0	0	0	0	0	0	0	0	0	0
9069	0	0	0	0	1	1	0	0	6	0
9780	0	0	0	0	0	0	0	0	6	0

Figure 2: Snapshot of Document Term Matrix Created From Full Twitter Data Set

The document term matrix is split into a training and testing set, separated by tweet, then combined back with the sentiment labels for each individual tweet. It is important to confirm that the proportion of negative and positive tweets are balanced between each of the two sets as to avoid any modeling inconsistency. To accomplish this, the tweets in each set were determined using random sampling, with 75% of the tweets being assigned to the training set, and the remainder to the test set. The table below shows the proportion of each of the sentiments for both the training and testing model. As you can see, the proportions are nearly equivalent, with both positive and negative split nearly in half:

Data Set	Negative	Positive
Training	43.4%	56.6%
Test	43.3%	56.7%

The most important consideration for training this model is a particular parameter, called the “lower frequency bound” found in the *findFreqTerms* function in R package *tm* (Feinerer, 2023). This value represents the lowest number of occurrences the word can have in the matrix to be considered for modeling. When we initially completed the project, we tried several different values for this parameter, eventually deciding on a lower frequency bound of 25. This means that if the word occurred at least 25 times in the entire word matrix, the word was added to a list that would be used to extract only this list of words for use in the model training and testing. As mentioned above, the document term matrix is extremely large, so it is necessary to use only frequent terms to avoid excess processing time and computational memory. Using only 25 words takes the number of words for training down to 1,650, a 95% reduction the number of words in the original document term matrix! The data is now ready for training the model, which was completed using the function *naiveBayes* in R package *e1071* (?). One special parameter, called the Laplace estimator, was used when training this model. I will quickly explain why this parameter is needed by briefly explaining the background of the algorithm. The naïve bayes algorithm is represented as:

$$P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{P(B|A)P(A)}{P(B)}$$

$P(A|B)$  represents the probability of event A occurring, given event B occurs. In the case of this project, we are calculating the probability that the tweet is positive or negative, given a certain set of words occur in

the tweet. As stated previously, the matrix is very sparse. Even with the reduction of words to those with frequency at least 25, there are still only 0.3% of cells that contain a value other than 0. If we are multiplying anything by zero, then the entire probability calculation becomes zero, which would result in many rows with zero probability. To avoid this from happening, the Laplace estimator is used to add a small number to each of the counts, so that none of the words that have zero as their count will result in a probability that equates to zero (?). In our case, we chose the Laplace estimator to be 1, so each cell in the document term matrix that has a 0 will be changed to 1. This helped to improve the model accuracy by over 1%.

After model training and evaluation, we applied the model with 25 as the lower frequency bound and Laplace estimator of 1 to the test data set. The trained model achieved a 74.6% accuracy, meaning 74.6% of the sentiments were correctly predicted. When applied to the test set, we achieved 73.3%, which seemed like a reasonable accuracy to achieve on such a complex data set. However, when we submitted our predictions of the Kaggle data set to the competition, our prediction accuracy was much lower than our peers. We placed 4th in the class, with the highest team reaching 74.1% accuracy. Figure 3 shows the placing of our team compared to the rest of the groups. This was a disappointment, and I was curious what we failed to do that the other teams did do.

#	△	Team	Score
1	—	Team 5	0.74058
2	—	Team 2	0.74051
3	▲ 1	Team 4	0.73444
4	▼ 1	Team 3	0.73284

Figure 3: Kaggle Results Compared to Peers

## Project Expansion

When we learned about sentiment analysis in class, I found it extremely interesting and highly useful. We didn't get too much exposure to sentiment analysis because the chapter we were studying was focused on probabilistic learning. The sentiment analysis assignment was more of a means of practicing training a supervised learning algorithm. I realized when researching this topic further that we had barely scratched the surface on what kind of techniques, analysis, and interpretations can be made from text mining. We didn't spend too much time in class with sentiment data, so I would like to get more practice with it, especially since my model did not perform well when I first did this homework. For this project, my objectives for enhancements are the following:

1. **Text Wrangling Improvement:** As mentioned in the initial project section, I thought that when we cleaned the data according to the textbook sequence, some of the words didn't make much sense or contribute to the sentiment analysis of the tweets. I will re-approach text wrangling of the data to achieve cleaner tweets by the end of the sequence, particularly by adding extra clean-up functions, changing the order of functions, and creating a single operation that will execute these functions so that this sequence can be easily applied to other data sets.
2. **Model Adjustments:** Much to my disappointment, we did not place first in the Kaggle competition. To redeem myself, I want to adjust the parameters of the naïve Bayes algorithm to increase the accuracy of the trained the model to reach as high of an accuracy as the winner.

3. **Application to Other Data Sets:** I will apply my improved text wrangling sequence to tweets about ChatGPT and create data visualizations that visualize tweet rate over time and word frequency. Additionally, I will compare sentiment/emotion of tweets about ChatGPT from December 2022 to March 2023 to investigate whether sentiment about ChatGPT has changed in the three months following its public release.

## Expanded Results

**Text Wrangling** The first step in my analysis was to approach the original twitter data set to improve the text wrangling sequence. The conversion to a corpus was treated the same as originally, but some new cleanup functions were added, and the order of application altered. After some trial and error, Figure 4 shows the final text wrangling sequence. The boxes shown in black are new functions that remove the twitter handles and hyperlinks. The result looks much cleaner and more intuitive than those in Figure ???. After finalizing the sequence, I combined the cleanup functions into one single parent function that will execute in one line of code to any data set. This will come in handy when the same text wrangling is applied to the ChatGPT tweets.

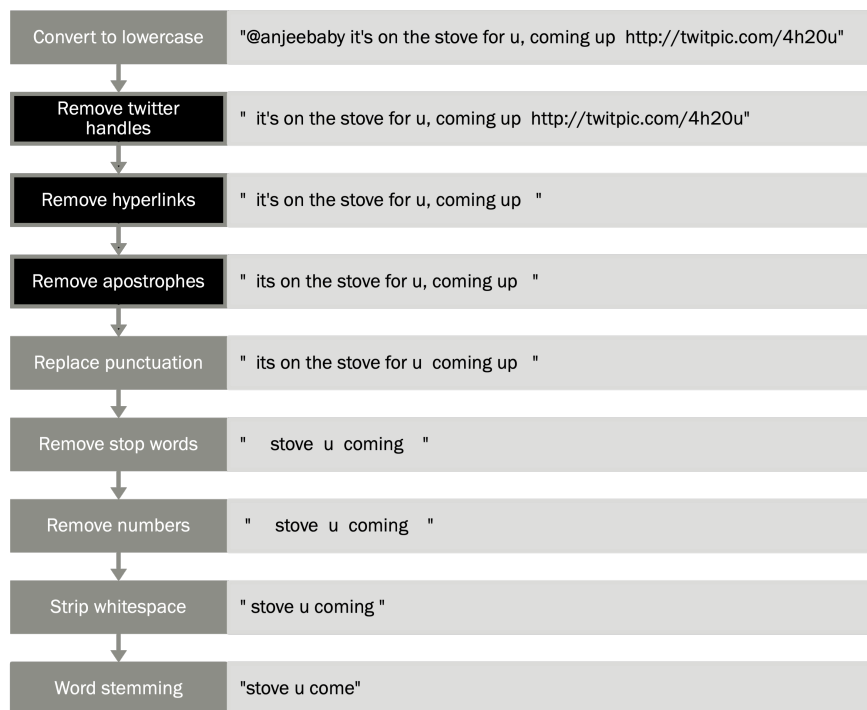


Figure 4: Text Wrangling Changes and Additions

**Naïve Bayes Model Improvement** After text cleanup, the document was converted into a document term matrix and split into a training and test set the same way as the original project. The biggest effect on the improvement of the model is the low frequency bound parameter. Figure 5 shows the accuracies using naïve Bayes algorithms both with and without the Laplace smoothing parameter. As the lower frequency bound decreases, the accuracy both with and without the Laplace parameter increases. The lowest model I was able to create had a lower frequency bound of four, meaning if a word occurred four times, then the word was included in the model. The drawback to lowering the frequency bound is that it includes more words, which increases the processing time and computational power needed to train the model. When I tried to expand the model even further and keep all words that occurred at least three times, I received an error that I had run out of memory.

Lower Frequency Bound	Accuracy Without Laplace	Accuracy With Laplace
25	0.742	0.745
20	0.744	0.747
15	0.744	0.748
10	0.746	0.752
5	0.746	0.756
4	0.746	0.758

Figure 5: Accuracy of Test Sets of Naïve Bayes Algorithm

There are a few concerns about this project’s enhancements that I noticed after spending some time working on it. One is that there are many unknowns to the data. As said earlier, I don’t know where the data came from or when the tweets were posted. I also don’t know the method that was used to determine the sentiment of each tweet. The biggest issue in this enhancement, however, is that I don’t know the “answers” to the test data set from the Kaggle competition, so there is nothing to confirm if the model improves in application. It is for these reasons that I decided to expand my plans using a different data set and focus more on exploratory analysis and sentiment analysis rather than supervised learning algorithms.

**ChatGPT Tweets** Before I go further in this portion of the project, I will give some background about ChatGPT and why I decided to choose tweets that related to this topic. ChatGPT is a chatbot that has gained immense popularity and controversy since its release in November 2022 (Wiggers, 2023). The chatbot has been in many news headlines, most of which depict the product as a dystopian-creating technology that will cause economic issues, get rid of needed jobs, and create disinformation among the public (Zinkula, 2023). ChatGPT is owned by a company called OpenAI, a private company founded in 2015 and funded by big names like Microsoft (O’Sullivan, 2023). ChatGPT has a free version that allows any user to ask questions about nearly anything, provided the question isn’t deemed inappropriate by the bot. Out of curiosity, when starting this project, I asked ChatGPT, “What are you?” to which is replied:

‘I am ChatGPT, an AI language model created by OpenAI. I am based on the GPT-3.5 architecture, which enables me to understand and generate human-like text based on the data I was trained on. My primary purpose is to assist users like you by providing information, answering questions, and engaging in conversations on a wide range of topics. However, please remember that I am just a program and do not have consciousness, emotions, or personal experiences. My responses are solely based on patterns in the data I was trained on, up until my knowledge cutoff in September 2021. How can I assist you today?’

Impressive! As for my own personal feelings, I cycled through several different opinions on the chatbot. When I first heard about ChatGPT, I rolled my eyes and thought it would be some silly technology that

over-promised and under-delivered. It wasn't until someone showed me its capabilities that I grew more interested. I found it very exciting to ask the chatbot questions and be amazed at how human-like the responses were. After some further use, however, I realized the bot isn't as smart as I thought it was and lost interest. It is for these reasons that I became curious how the public felt about this new technology so readily available to a normal internet user. I then decided I would apply sentiment analysis techniques to two separate sets of public tweets so I could compare the general feel of the product between December 2022 and March 2023.

There are two data sets used in this analysis. As a reminder, the first data set starts in November 2022, the month of ChatGPT's public release, and the other spans from January to March 2023. My first goal was to clean up the data sets so their columns match exactly, that way I can combine them into one larger data set to create visualizations. Once they were combined and cleaned, the final data frame consists of 500,036 tweets and two columns. The first column is a time stamp for the tweet, and the second is the text of the tweet itself. Figure 6 shows the first five rows of the combined data.

created_at	tweet
2023-03-29 22:58:21	Free AI marketing and automation tools, strategies, and coll
2023-03-29 22:58:18	@MecoleHardman4 Chat GPT says it's 15. 🤖
2023-03-29 22:57:53	<a href="https://t.co/FjJSprt0te">https://t.co/FjJSprt0te</a> - Chat with any PDF! Check out how
2023-03-29 22:57:52	AI muses: "In the court of life, we must all face the judge of
2023-03-29 22:57:26	Most people haven't heard of Chat GPT yet. First, elite factio

Figure 6: The First Five Rows of the Combined Data

Once the data was cleaned and combined, I wanted to first look at the tweet rate by day that contained the word ChatGPT. In Figure 7, notice there are three major peaks in tweet counts over the span of three months. Interestingly, these spikes all correspond to major news releases about ChatGPT. The first spike on 12/6/2022 is less than a week after ChatGPT was first released for public use (Wiggers, 2023). The second major spike is when Microsoft announced it would be incorporating ChatGPT into their technologies (Novet, 2023). Lastly, there is a final major spike on 3/15/23, the same day as when ChatGPT released a newer, updated version (Fried, 2023). Other than these three high points, the tweet count per day has stayed relatively constant within the three month time span, with a mean of about 6,200 tweets per day.



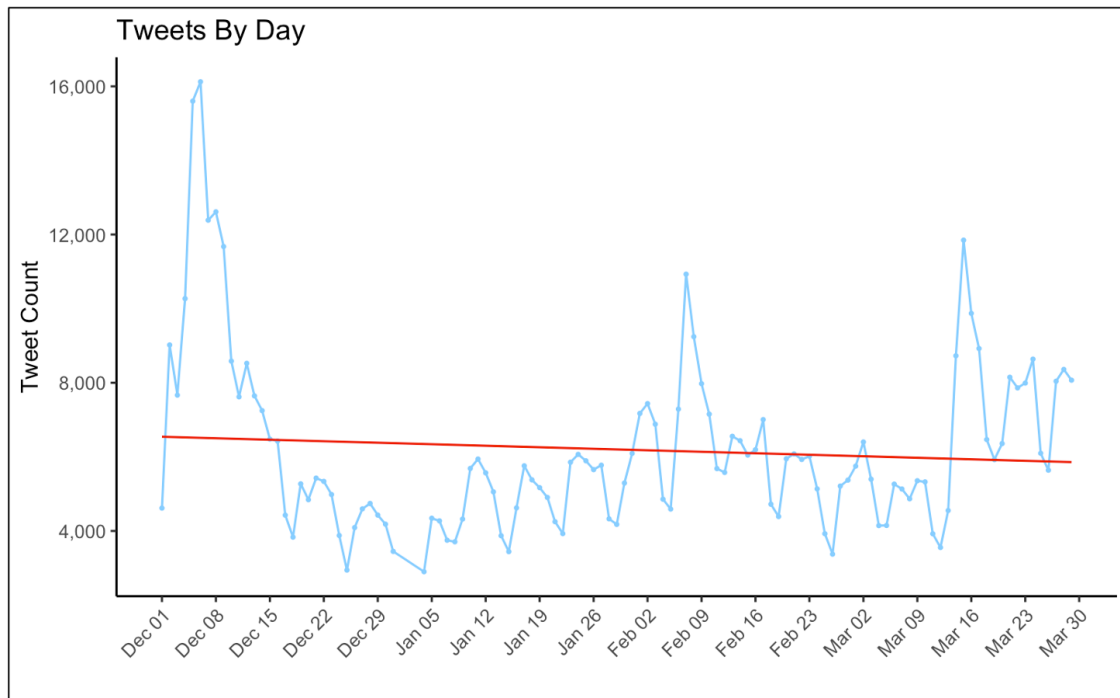


Figure 7: Time Series of Number of Tweets Per Day

Zooming out further, Figure 8 shows the number of tweets by month. In December there were 218,920 tweets, the most of any of the four months. This seems intuitive since the product was released shortly before the month of December. Then, there was a major decline of about 38% in the month of January. This could possibly be because the novelty of the product has worn off. After January, the number of tweets per month increases, but never reaches as high as December 2022.

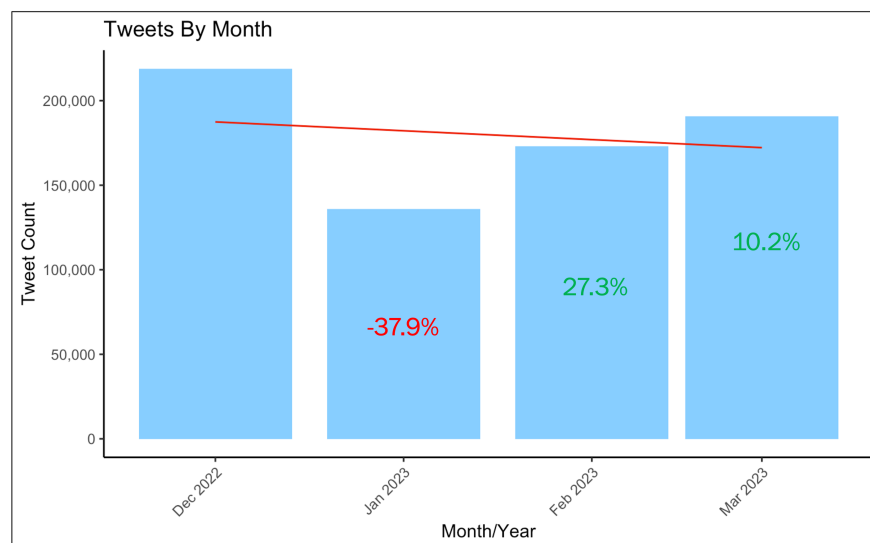


Figure 8: Number of Tweets Per Month

Next, I want to compare the months of December and March to see if the contents of the tweets change within these three months. Figure 9 shows the top ten words for each of the months, as well as their word

clouds. Surprisingly, there are no major differences in either the top ten words or word clouds. The top ten words for both months contain many common words for each of the months, like “use”, “can”, “write”, “ask”, “will”, “like”, “just”. Some more specific words are “openai” and “bot.” The most interesting point to make in the top-ten word comparison is that the word “new” doesn’t appear in the top ten until three months after the release of the product. It is possible that the mainstream user took this long to get exposure to ChatGPT and its uses. The word clouds are very similar to each other, with “use” again being the major word used word in tweets.

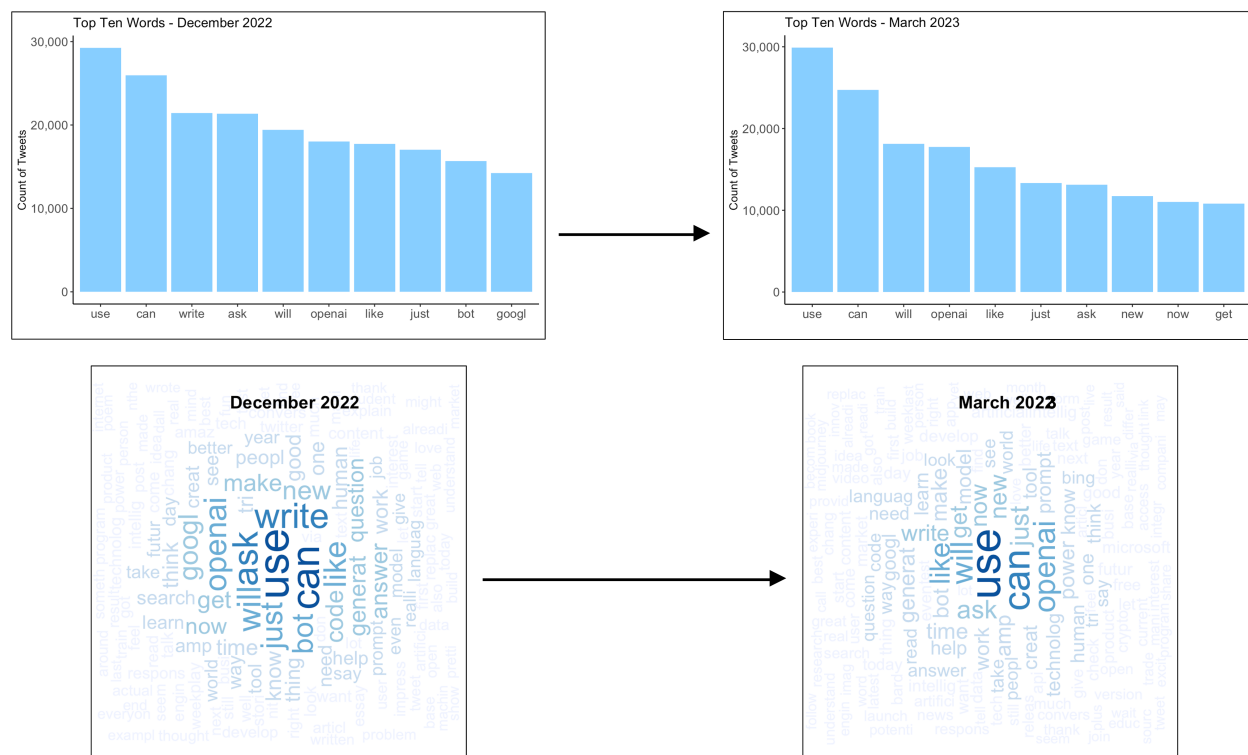


Figure 9: Top Ten Words and Word Clouds for December 2022 and March 2023

My next step was to use a package in R called *syuzhet* to compare the sentiment between the two months (Jockers, 2020). This package has a function called *get\_nrc\_sentiment*, which uses a lexicon created by Saif M. Mohammad at the National Research Council Canada, hence the name NRC (Mohammad, 2022). A lexicon in this sense is like a dictionary, where each word is assigned some meaning, like a definition in a dictionary (Merriam-Webster, 2023). This lexicon is a list of words and an assigned emotion (anger, fear, anticipation, trust, surprise, sadness, joy, and disgust) that has been determined through crowd sourcing (Mohammad, 2022). In addition to an emotion, the words are separately assigned a positive or negative sentiment. There are many available lexicons and methods of determining sentiment. The *syuzhet* package itself offers three additional lexicons in addition to the NRC lexicon. I chose NRC because it was the first to associate emotion and remains the largest available (Mohammad, 2022). In my analysis I want to compare both the sentiment and the emotions and interpret them as if I were someone interested in the success of ChatGPT.

First, I will look at the sentiment, or whether the words in the tweets are considered positive or negative according to this particular lexicon. Figure 10 gives an overview of the percent of words assigned to each emotion in December 2022 compared to March 2023. The green arrows represent a success for ChatGPT stakeholders, while the red represents a failure. Trust and anticipation have increased, while anger, disgust, fear, and sadness have decreased. However, joy and surprise have decreased, possibly because the novelty of the new technology has worn off after three months of exposure.

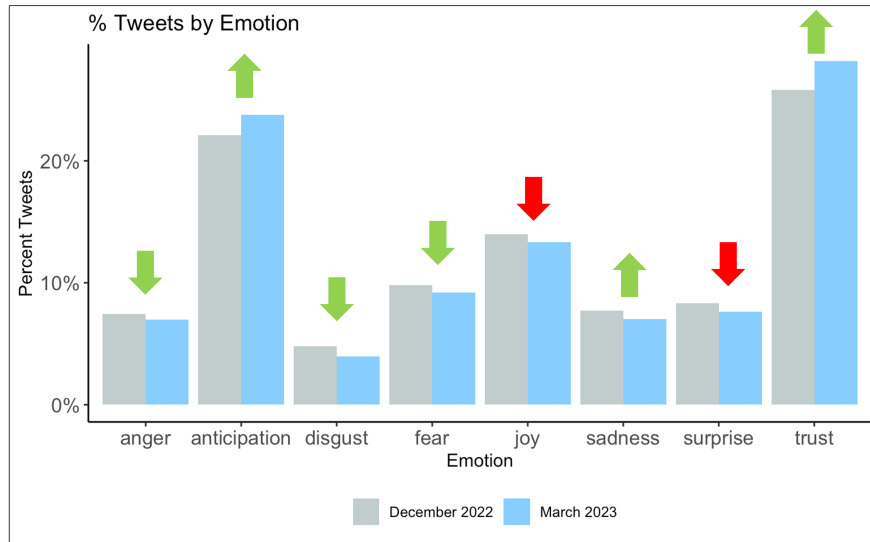


Figure 10: Percent of Tweets by Emotion

Next, I will look at the the overall distribution of positive and negative tweets, shown in Figure 11. The results are positive for someone in the interest of ChatGPT's success. The percent of negative words has decreased by nearly 13% in three months, while the percent positive have increased by about 6%. Also, it appears as if from its launch, ChatGPT sentiment has been positive, since over 60% of tweet words were positive in December 2022. This is an interesting point because in my online news environment, I am typically exposed to generally negative articles about the rise of artificial intelligence and technologies like ChatGPT.

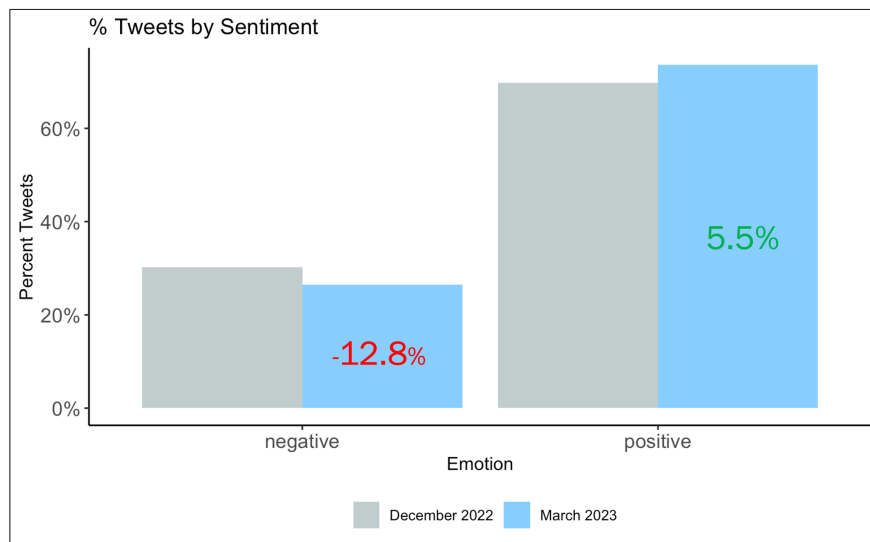


Figure 11: Percent of Positive and Negative Tweets

## Challenges and Further Work

For both aspects of this project, it is worth mentioning that sentiment analysis can be tricky and difficult to master. For one, the analysis is calculating sentiment based on words, not phrases. Even if phrases are used, the human language is complex, and an algorithm cannot distinguish between irony, sarcasm, tone, or the

like. While the analysis above gives some insight into the overall sentiment of these tweets, it is not unbiased since the lexicon of words used is crowdsourced from people who could have very different background and understanding of words and phrases ([Vodovatova, n.d.](#)).

One of the challenges to the original twitter data set is the amount of processing power required to train the naïve Bayes algorithm. As the number of frequent terms is reduced, the data set becomes larger, and the training requires more processing time and computational power. This data set has been a great practice in text cleaning and supervised learning training, testing, and evaluation using the naïve Bayes algorithm. However, I don't think I would continue any work on this data because I don't have any information about the data background. In the future, I would like to continue practicing machine learning algorithms to predict sentiment in social media, such as linear regression and support vector machines, as well as continuing to refine the natural language processing of the text.

The ChatGPT portion of the project has extended my ability to gather, explore, and analyze text data much further than what was originally assigned to me. I set out with the curiosity of whether the public feeling on ChatGPT has improved or declined since its initial launch, and to learn that the sentiment has improved since December is a very interesting finding that I would like to explore further. It would be interesting to analyze more recent months of twitter data. Unfortunately, the Twitter API is no longer available for free use as of February 9, 2023 ([Barnes, 2023](#)). There are alternative methods to gathering tweets, such as the Python package *sns scrape*, but this is a skill that I have not learned yet.

## References

- Ansari, K. (2023). *500k ChatGPT-related tweets jan-mar 2023*. <https://www.kaggle.com/datasets/khalidryder777/500k-chatgpt-tweets-jan-mar-2023>
- Barnes, J. (2023). *Twitter ends its free API: Here's who will be affected*. <https://www.forbes.com/sites/jenaebarnes/2023/02/03/twitter-ends-its-free-api-heres-who-will-be-affected/?sh=2fb5f6762664>
- Feinerer, I. (2023). *Introduction to the tm package text mining in r*. <https://cran.r-project.org/web/packages/tm/vignettes/tm.pdf>
- finnstats. (2021). *Sentiment analysis in r*. <https://www.r-bloggers.com/2021/05/sentiment-analysis-in-r-3/>
- Fried, I. (2023). *When "scary good" AI gets even better*. <https://www.axios.com/2023/03/15/gpt4-openai-chatgpt-new-version>
- Jockers, M. (2020). *Syuzhet*. <https://www.rdocumentation.org/packages/syuzhet/versions/1.0.6>
- Li, D. (2020). *Basic r guide for NSC statistics*. <https://bookdown.org/dli/rguide/bar-graph.html>
- Merriam-Webster. (2023). *Lexicon*. <https://www.merriam-webster.com/dictionary/lexicon>
- Mohammad, S. M. (2022). *NRC word-emotion association lexicon*. [https://saifmohammad.com/WebPages/NRC-Emotion-Lexicon.htm#:~:text=The%20NRC%20Emotion%20Lexicon%20is,sentiments%20\(negative%20and%20positive\).](https://saifmohammad.com/WebPages/NRC-Emotion-Lexicon.htm#:~:text=The%20NRC%20Emotion%20Lexicon%20is,sentiments%20(negative%20and%20positive).)
- Novet, V., Jordan. (2023). *Microsoft announces new a.i.-powered bing homepage that you can chat with*. <https://www.cnbc.com/2023/02/07/microsoft-open-ai-chatgpt-event-2023-live-updates.html>
- O'Sullivan, I. (2023). *Who owns ChatGPT and its creator, OpenAI?* <https://tech.co/news/who-owns-chatgpt-and-openai#:~:text=Sam%20Altman%20is%20the%20CEO,Ilya%20Sutskever%2C%20and%20John%20Schulman.>
- Pham, M. (2022). *ChatGPT tweets first month of launch*. <https://www.kaggle.com/datasets/pcminh0505/chatgpt-twitter>
- Verma, Y. (2021). *A guide to term-document matrix with its implementation in r and python*. <https://analyticsindiamag.com/a-guide-to-term-document-matrix-with-its-implementation-in-r-and-python/>
- Vodovatova, E. (n.d.). *WHAT IS SENTIMENT ANALYSIS: DEFINITION, KEY TYPES AND ALGORITHMS*. <https://theappsolutions.com/blog/development/sentiment-analysis/>
- Wiggers, S., Kyle. (2023). *ChatGPT: Everything you need to know about the AI-powered chatbot*. <https://techcrunch.com/2023/07/13/chatgpt-everything-you-need-to-know-about-the-open-ai-powered-chatbot/#:~:text=November%2030%2C%202022%20is%20when%20ChatGPT%20was%20released%20for%20public%20use>
- Zinkula, J. (2023). *ChatGPT creator says AI advocates are fooling themselves if they think the technology is only going to be good for workers: 'Jobs are definitely going to go away'*. <https://www.businessinsider.com/chatgpt-sam-altman-jobs-replaced-ai-openai-2023-7>