Final Report:
Twitter Sentiment Analysis

Problem Statement

a) Background

There is evidence to suggest that there can be a correlation between tweets and stock prices, particularly for publicly traded companies. The idea behind this correlation is that tweets can influence public perception, which in turn can impact stock prices.

For example, if a company releases positive news, such as a new product launch or strong financial results, this may lead to positive sentiment on social media and an increase in stock prices. On the other hand, if a company is involved in a scandal or there is negative news about it, this can lead to negative sentiment on social media and a decrease in stock prices.

Some studies have shown that sentiment analysis of tweets can predict stock prices with a moderate level of accuracy. However, it is important to note that correlation does not necessarily imply causation, and there are many factors that can influence stock prices.

b) Goal

By scraping live tweets for the month of December 2022. I created a model to help Apple Inc. quantify the sentiment of the tweets written regarding them. Measuring sentiments in tweets can provide valuable insights into the attitudes and opinions of people on their products. For example, Having a specific scores on Apple related tweets can help the company with:

1. Understanding customer feedback: Many businesses use social media platforms, including Twitter, to connect with their customers and receive feedback. Sentiment analysis can help these businesses to understand the opinions and attitudes of their customers towards their products or services.
2. Monitoring brand reputation: By monitoring sentiments on Twitter, businesses can also track the sentiment towards their brand and products. Negative sentiment could indicate a potential PR crisis or highlight areas that require improvement.
3. Tracking public opinion: Sentiment analysis can provide insight into the opinions of people on various topics, including politics, social issues, and current events. This can help organizations, including governments, to make informed decisions that are in line with public opinion.

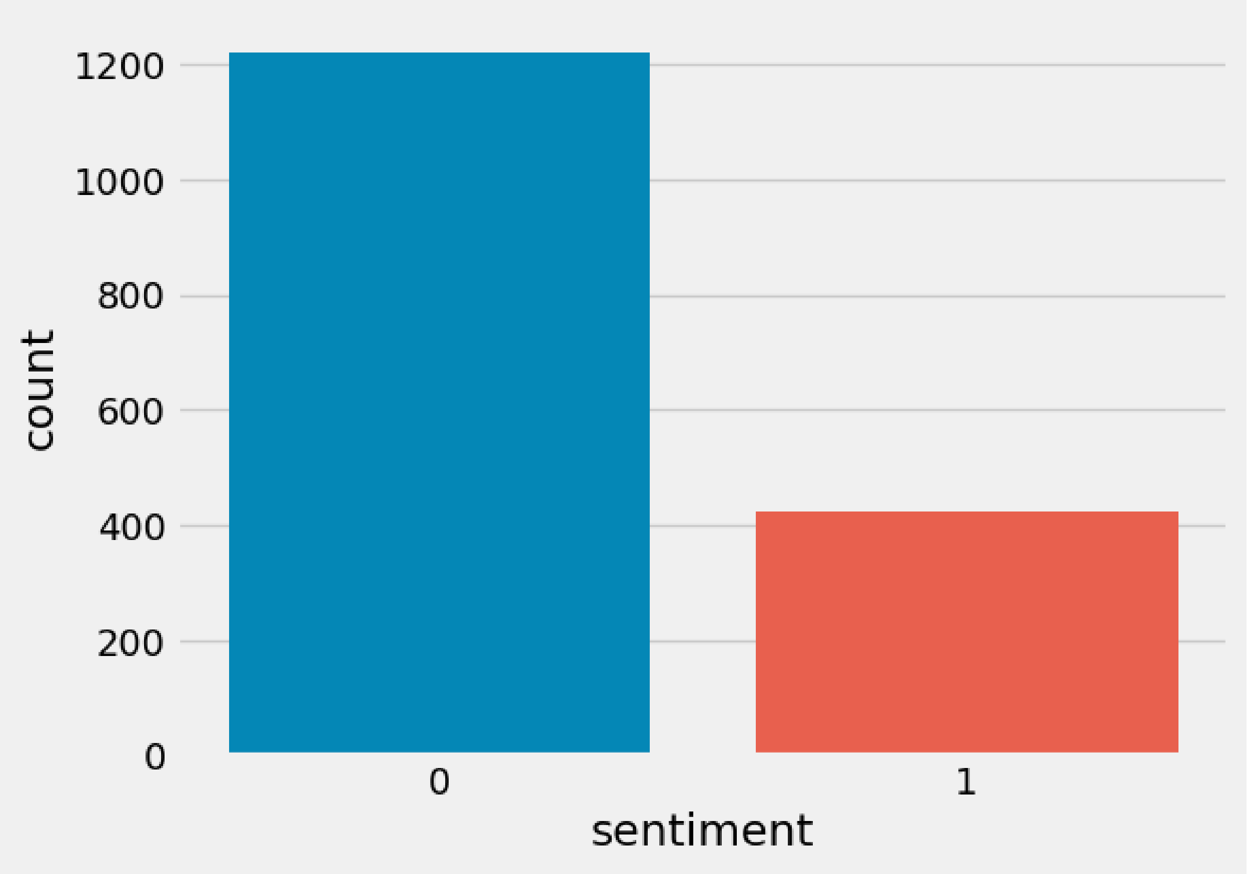
4. Improving social media engagement: By analyzing the sentiment of tweets, businesses can identify the topics and issues that resonate with their audience. This information can then be used to develop targeted content that increases engagement.
5. Research and analysis: Researchers and analysts can use sentiment analysis to gain insights into public opinion and attitudes towards various topics. This can be useful for social science research, marketing research, and political analysis.

Historical Data (*00_Historical_Data_2014.ipynb*)

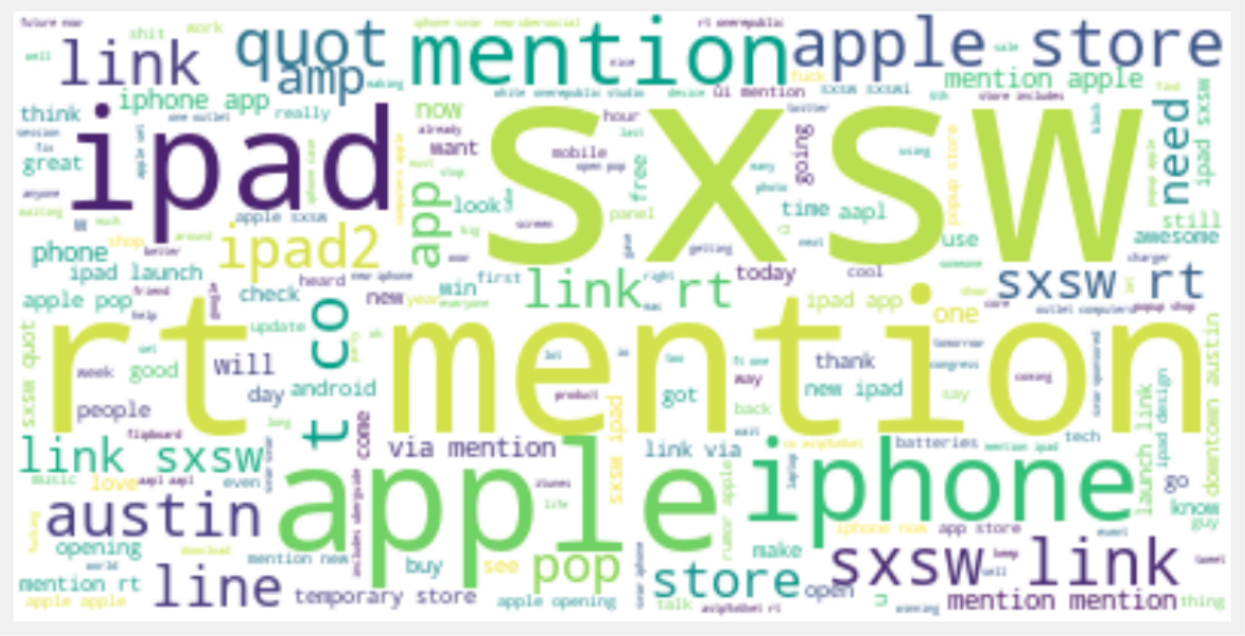
Before diving into live tweets scraping, I first explored a pre-scraped dataset (Dec. 5 to Dec. 10, 2014) that contains 3886 rows and 12 columns. Although processed, the dataset was still not perfect and has a lot of null values. Because we only care about the contents of the tweets, I eliminated columns that we do not need such as "_unit_id, "_golden", "_unit_state", etc. The author had also provided a column with the keyword they used to pull their tweets: #AAPL OR @Apple. Knowing this information, I applied it to my own queries in the Data Wrangling of this project.

After I've dropped all of the columns and null values that I believed were irrelevant , the sentiment column in the dataset was left with two representations: 5 (positive), and 1 (negative). To simplify the numbers, we created the function new_sentiment(x) to convert 5 to 1, signifying positive, and 1 to 0, signifying negative. By plotting the values count of these numbers, we can see that there were three times the amount of negative tweets to the amount of positive tweets. This result was not surprising since historical records show that there were a mix of positive and negative indicators in the global economy. For instances, some economic highlights at the time were:

1. The US economy continued to show signs of improvement, with the unemployment rate falling to 5.6%, its lowest level since 2008. This was seen as a positive sign for the US economy.
2. The European Central Bank (ECB) announced a new stimulus program to help boost the Eurozone economy, which had been struggling with low inflation and weak growth. This program included buying government bonds, known as quantitative easing.
3. Falling oil prices continued to put pressure on oil-producing countries, such as Russia and Venezuela, which experienced economic difficulties as a result.
4. The Chinese economy showed signs of slowing down, which led to concerns about the impact on global growth.



To help me get a better understanding of what words users were tweeting in these tweets, I used another dataset with a completely different set of tweets, but this time it also contains their mentioned apple products and the emotions associated with it. Looking at the data, we could see that the owner has done some natural language processing to get the positive and negative emotions associated with the tweets. I used this dataset, together with the first csv, to generate a word cloud.

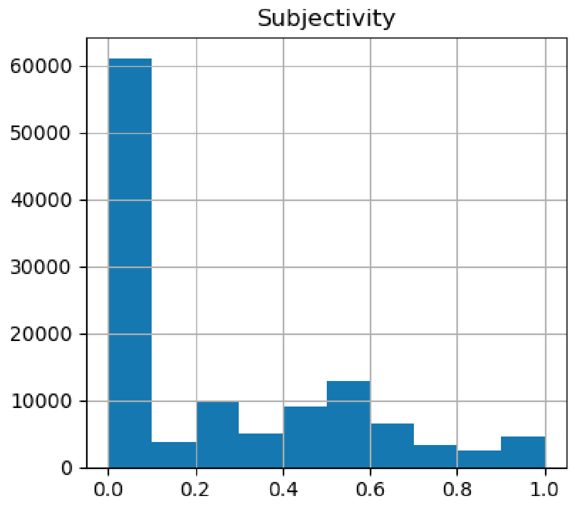


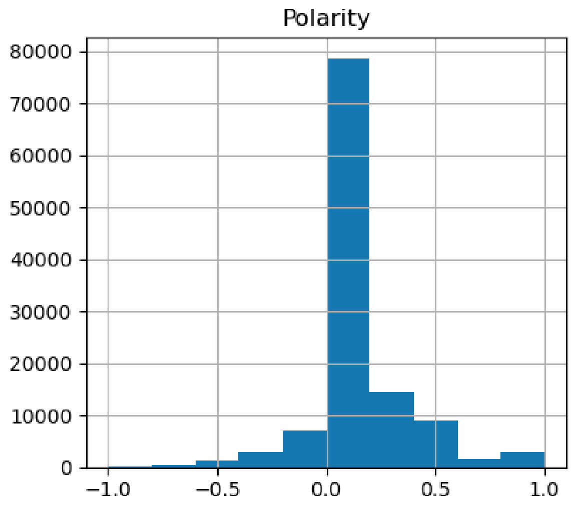
One common 'big' word that appeared in both of these dataset is **SXSW**. After doing a simple google search, SXSW is a music festival that is usually held in mid-March. It is surprising to see that both the first dataset (dates in December 2014) and second dataset (unknown dates) mentioned this festival. This implies that tweets might be 3 months ahead of real life events!

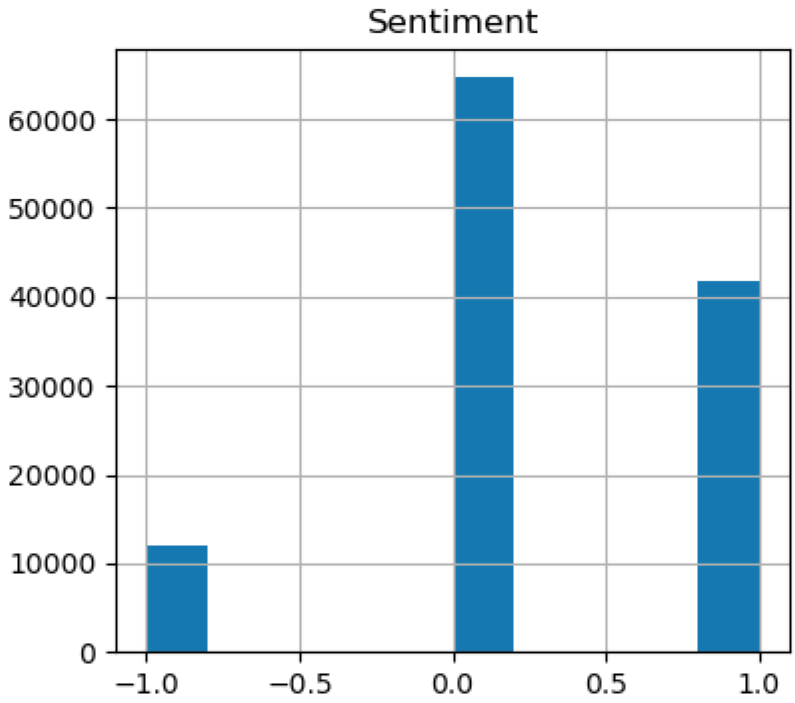
Data Wrangling (*01_Data_Wrangling.ipynb*)

After exploring historical datasets, I scraped live tweets for the month of December 2022. I chose this month because it was the current month that we were in while carrying out this project and I wanted to see whether the statement that tweets are 3 months ahead of real life events stays true in March 2023.

I scraped live Tweets with specific hashtags (#appl, #apple, #ipad, & #iphone) using the SNScrape method, and generated Apple stock data for the month of December 2022. The raw dataset first had 30,000 rows and 5 columns ('user', 'likes', 'source', 'text', 'Time'). I then extracted their subjectivity and polarity scores. Subjectivity detection and polarity detection are subtasks under sentiment analysis. Subjectivity detection aims to remove 'factual' or 'neutral' content, i.e., objective text that does not contain any opinion. Polarity detection aims to differentiate the opinion into 'positive' and 'negative'. After having their polarity scores, I simplified it into numbers, called Sentiment numbers, representing the following: -1 = Negative (if polarity <0); 0 = Neutral (if polarity = 0); 1 = Positive (if polarity > 0). The plots below represent the ranges of Subjectivity, Polarity, and Sentiment scores.





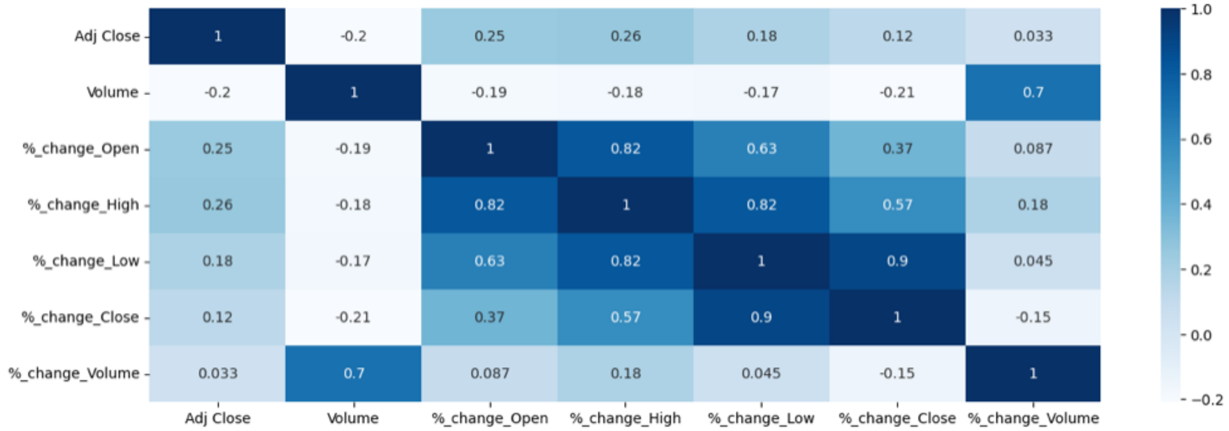


At the end of this notebook, I was able to produce two datasets: live tweets with quantified sentiments from Dec. 2022 (01_tweets_data) and historical stock price from Dec. 2022 (01_stock_data).

Exploratory Data Analysis (*02_Exploratory_Data_Analysis.ipynb*)

In this notebook, we extracted some features from stock data and visualized their correlation. We also looked at the tweets from the previous notebook (01_Data_Wrangling) to extract the most common words, likes, time of tweet, and frequent user.

From the stock data that we generated from Data Wrangling, we examined the change in price between a given date and its next day's pricing (one period into the future) using the function .shift(). For example, "shifted_Open" implies the next day's opening price for the current date. From there, we were able to find the percentage change of AAPL opening, high, low, closing prices, and daily trading volume.

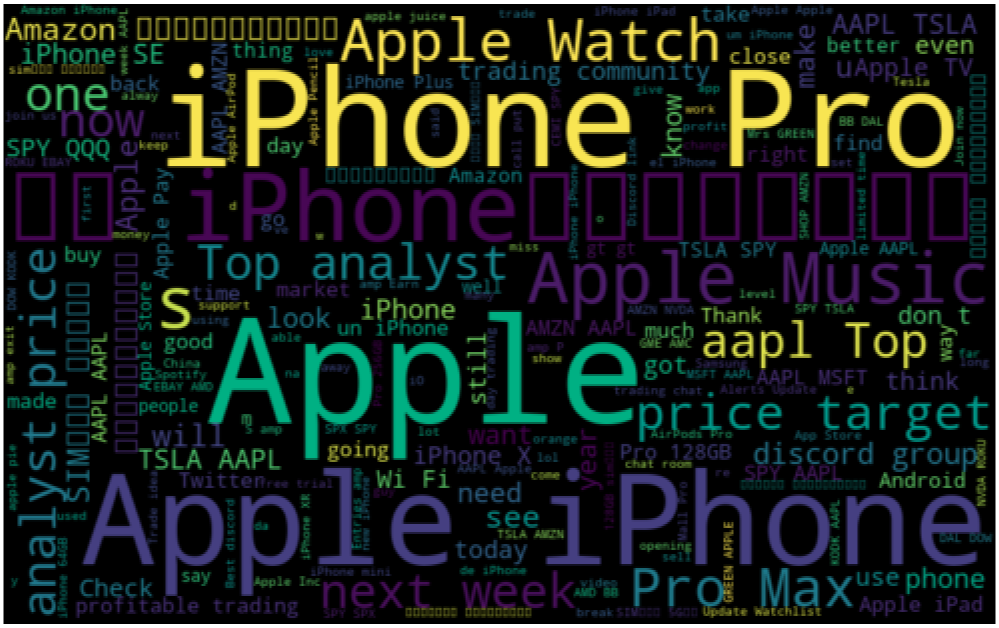


	Adj Close	Volume	%_change_Open	%_change_High	%_change_Low	%_change_Close	%_change_Volume
Adj Close	1	-0.2	0.25	0.26	0.18	0.12	0.033
Volume	-0.2	1	-0.19	-0.18	-0.17	-0.21	0.7
%_change_Open	0.25	-0.19	1	0.82	0.63	0.37	0.087
%_change_High	0.26	-0.18	0.82	1	0.82	0.57	0.18
%_change_Low	0.18	-0.17	0.63	0.82	1	0.9	0.045
%_change_Close	0.12	-0.21	0.37	0.57	0.9	1	-0.15
%_change_Volume	0.033	0.7	0.087	0.18	0.045	-0.15	1

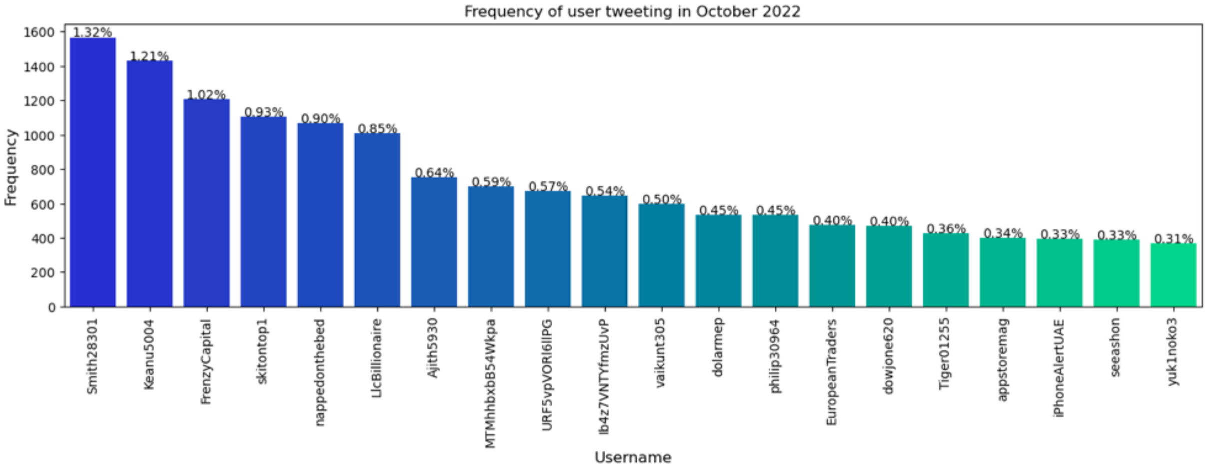
The main variable we are going to focus on in this dataset is the ‘Volume’. Variables having a large correlation value with volume represent that those numbers might have intrigued a large number of buyers and sellers. Correlation between those features and the volume feature will tell us how a change in that feature impacts the number of stocks traded that day.

The %_change_Volume shows the most positive correlation with the volume feature. This means that the greater the difference between the volume of the stock today and the volume of the stock price of yesterday, the greater will be the stocks traded that day. As there is no information about the number of buyers and sellers, we can only guess that a high difference in volume might attract more buyers. Whereas a fewer difference may attract more sellers. With this in mind, we know to focus on days with high percentage change in trading volume.

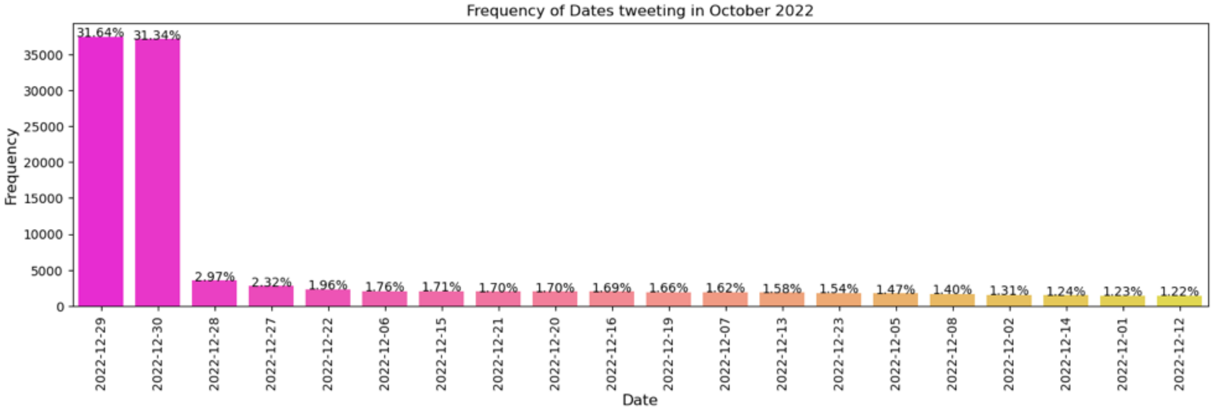
Now to actually understand the content of the tweets which people tweet, we created a Word Cloud of the top most words used in the tweets, and some plots representing frequency of these tweets.



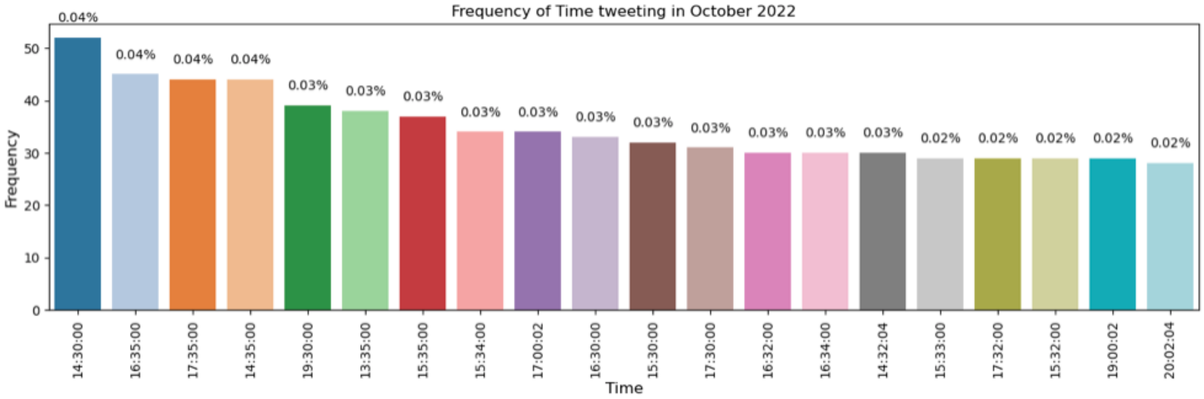
Note: The below illustrations have October as their title, but it was actually for December 2022.



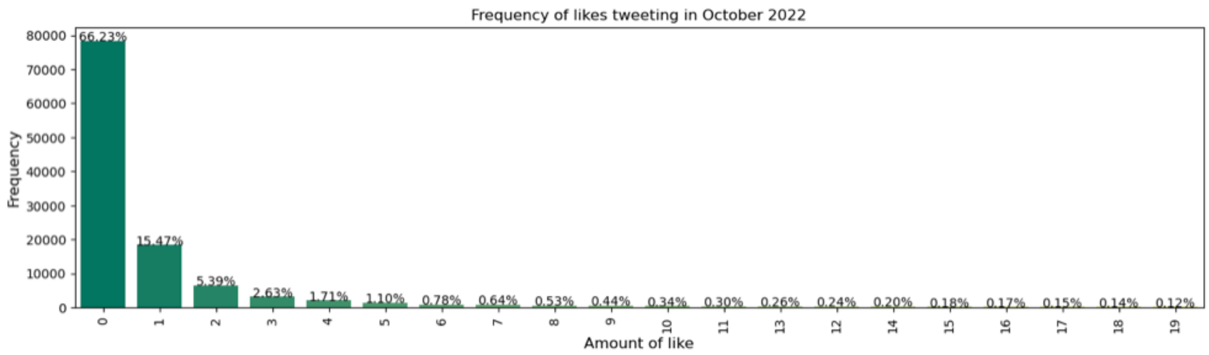
In December 2022, user Smith28301 posted most frequently (1.32%) using the following hashtags: apple, iphone, ipad.



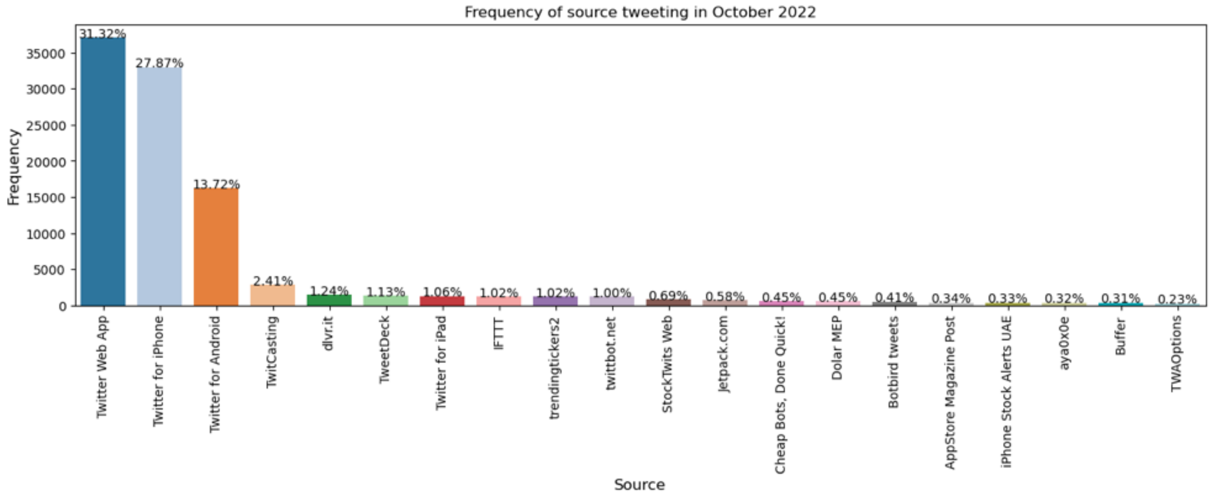
Over 60% of the tweets were posted on December 29th, and December 30th, 2022 as traders were anticipating a Santa Rally. A "Santa rally" is a term used to describe a phenomenon in which the stock market experiences a rise in stock prices in the final weeks of the calendar year, typically in the last week of December. The name "Santa rally" comes from the fact that it often coincides with the holiday season, and some traders believe that the surge in buying during this time is due to increased consumer spending and a generally positive mood.



Most tweets were published around 2:30 pm to 2:35 pm UTC (6:00 am to 6:30 am PST)



Over 65% of tweets received 0 likes and about 15% received 1 like



Over 30% of tweets were published using the Twitter app

Pre-processing & Training Data (*03_Preprocessing_and_Training_Data.ipynb*)

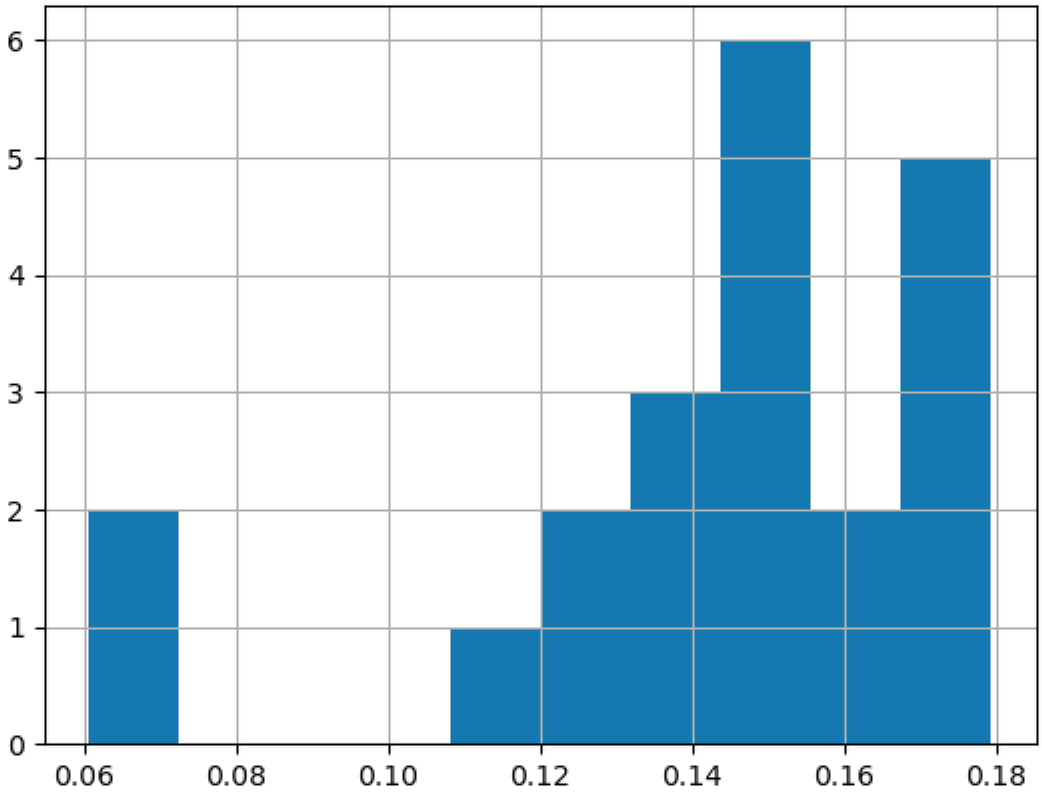
The objective of this notebook is to prepare data for fitting models. In notebook 02_Exploratory_Data_Analysis, we have already featured new columns for the stock data, measuring the changes each day. In this notebook, we will be using word embedding to convert words to vectors, then use that as the main input to our training set. Word Embedding is a language modeling technique used for mapping words to vectors of real numbers. It represents words or phrases in vector space with several dimensions.

I merged the tweets dataset and the stock dataset together (merged_dataframe), showcasing the overall sentiment of tweets of each trading day. The combined dataset now has Dates and the following features:

Adj Close	**stock_volume**	**twitter_volume**	**open_trend**	**high_trend**	**low_trend**	**close_trend**	**volume_trend**	**likes**	**Subjectivity**	**Polarity**	**Sentiment**

The features open_trend, high_trend, low_trend, close_trend, and volume_trend are binary encodings to measure stock trends in December 2022. If today's change in price is greater than yesterday's price, trend equals 1. If it is less than, stock trend equals 0. Although stock trends will not be the primary input for our modeling, these encodings are there for future research that we might have beyond natural language processing.

I plotted the polarity to see the ranges of the sentiments of these tweets based on its overall daily score. Looking at the range, we can see that the range of 0.14 to 0.16 seems to be the most common score for the month of December 2022. Due to this, we'll choose 0.15 as the "mean", or "neutral" zone for the tweets. Anything that is above 0.15 will be considered a "positive" day, and anything below 0.15 is "negative".

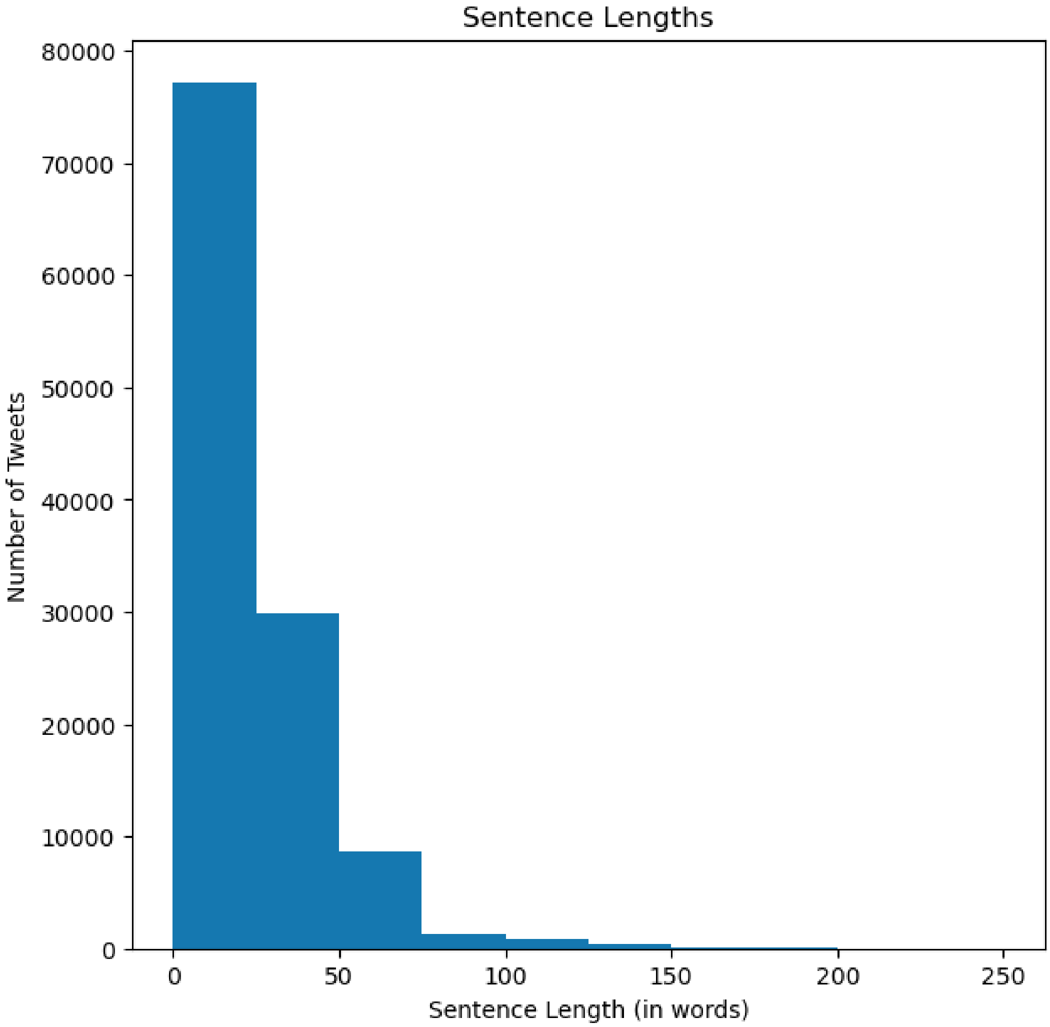


After appointing the daily polarity scores to its corresponding label, we now have 11 positive days, and 10 negative days.

Text Mining

Text Mining is the process of deriving meaningful information from natural language text. I first extracted number of characters present in a tweet, number of words present in each line of tweet, number of punctuation, number of words in quotation marks, number of sentences, number of unique words, count of hashtags, count of mentions, count of stopwords, count of average word length, count of average sentence length, ratio of unique words to total word count, and ratio of stopwords to total word count. Although these feature extractions do not go into our final model, this is to help me understand which features correlate with the tweets sentiment.

Here we also turned our twitter strings to lists of individual tokens (words, punctuations). It is the process of breaking strings into tokens which in turn are small structures or units. Tokenization involves three steps which are breaking a complex sentence into words, understanding the importance of each word with respect to the sentence and finally producing a structural description on an input sentence. The tokens generated produced the following plot for sentence lengths, representing a vocabulary size of 69391 unique words.



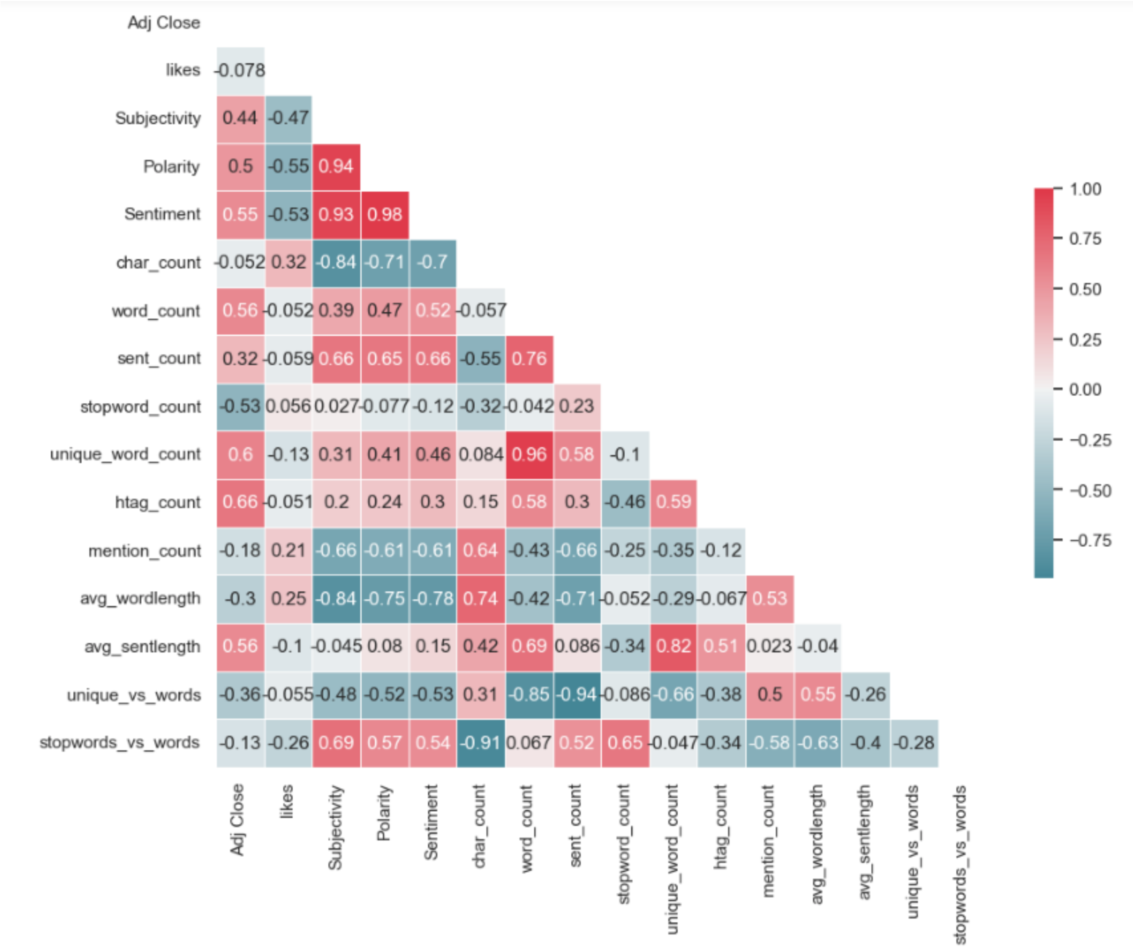
Next, I proceeded with removing stop-words tokens and lemmatization . “Stop words'' are the most common words in a language like "the ","a ","at ","for ","above ","on ","is ","all ". These words do not provide any meaning and are usually removed from texts. On the other hand, lemmatization is the process of converting a word to its base form. Lemmatization considers the context and converts the word to its meaningful base form. For example, lemmatization would correctly identify the base form of ‘caring’ to ‘care'. This is extremely valuable because we want to identify keywords that lead to negative, positive and neutral sentiments.

Feature Elimination

Like previously mentioned, the purpose of extracting additional features (number of characters present in a tweet, number of words present in each line of tweet, etc.) from the tweets is for feature "elimination" to help reduce chances of overfitting and running into the curse of dimensionality, if we decide to use these features for our model.

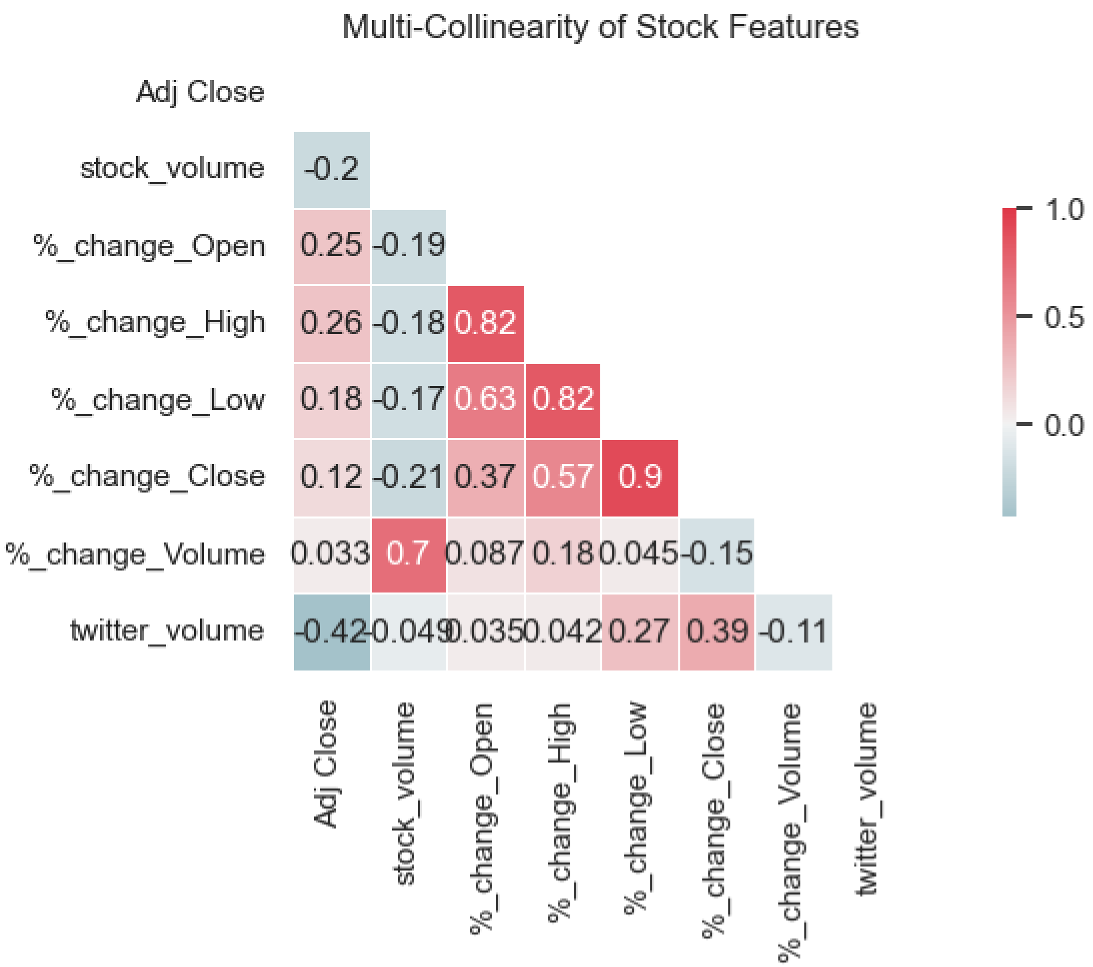
Understanding how the features in a dataset interact with each other is crucial when deciding which features to use in a model. There are many ways to construct a model that is effective and accurate. One of the fastest ways to strengthen a model is to identify and reduce the features in the dataset that are highly correlated. Correlated features will add noise and inaccuracy to a model, which in turn will make it harder to achieve the desired outcome.

Highly correlated variables should be avoided when creating models because they can skew the output. If there are two independent variables that are representing the same occurrence (i.e SqFt of a house vs bedrooms in a house) it can create “noise” or inaccuracy in the model. Models rely solely on outside information in order to create a useful output and having collinear (correlated) variables can create an inflated variance in at least one of the regression outputs.



	Adj Close	likes	Subjectivity	Polarity	Sentiment	char_count	word_count	sent_count	stopword_count	unique_word_count	htag_count	mention_count	avg_wordlength	avg_sentlength	unique_vs_words	stopwords_vs_words
Adj Close																
likes	-0.078															
Subjectivity	0.44	-0.47														
Polarity	0.5	-0.55	0.94													
Sentiment	0.55	-0.53	0.93	0.98												
char_count	-0.052	0.32	-0.84	-0.71	-0.7											
word_count	0.56	-0.052	0.39	0.47	0.52	-0.057										
sent_count	0.32	-0.059	0.66	0.65	0.66	-0.55	0.76									
stopword_count	-0.53	0.056	0.027	-0.077	-0.12	-0.32	-0.042	0.23								
unique_word_count	0.6	-0.13	0.31	0.41	0.46	0.084	0.96	0.58	-0.1							
htag_count	0.66	-0.051	0.2	0.24	0.3	0.15	0.58	0.3	-0.46	0.59						
mention_count	-0.18	0.21	-0.66	-0.61	-0.61	0.64	-0.43	-0.66	-0.25	-0.35	-0.12					
avg_wordlength	-0.3	0.25	-0.84	-0.75	-0.78	0.74	-0.42	-0.71	-0.052	-0.29	-0.067	0.53				
avg_sentlength	0.56	-0.1	-0.045	0.08	0.15	0.42	0.69	0.086	-0.34	0.82	0.51	0.023	-0.04			
unique_vs_words	-0.36	-0.055	-0.48	-0.52	-0.53	0.31	-0.85	-0.94	-0.086	-0.66	-0.38	0.5	0.55	-0.26		
stopwords_vs_words	-0.13	-0.26	0.69	0.57	0.54	-0.91	0.067	0.52	0.65	-0.047	-0.34	-0.58	-0.63	-0.4	-0.28	

According to the multicollinearity for the tweets dataset, as the color becomes darker in either direction (red or blue), meaning that those variables are more highly correlated and should not be paired together in the same model. With that said, since our y is Adj. Close, we need to eliminate any features that are highly correlated to Adj. Close.



No block has significant correlation for stock data.

Modeling

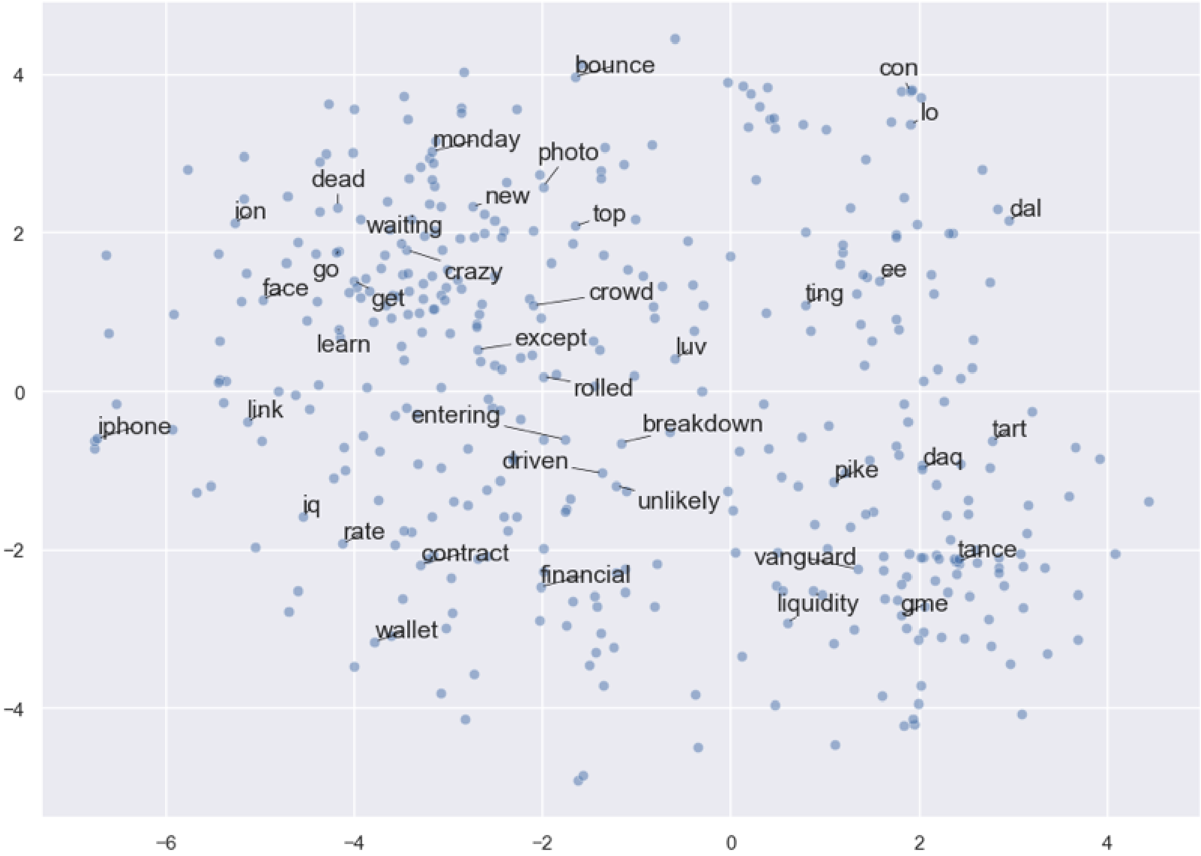
This is a classification problem, in unsupervised learning. In this project, I used the following classification models to measure the sentiment score for each tweet.

1. Logistic Regression
2. Random Forest
3. Gradient Boosting
4. Naive Bayes

Text Classification

The main feature that I used to predict the adjusted closing price in this project is the vectorized representation of the tweet. To get the numbers, we need to build a text classification model using word embedding.

After obtaining the pre-trained embedding from the previous notebook, we applied it to our tweets data and reduced its dimension using t-SNE. The graph below represents 400 words from our list of tokens. Notice how we see certain items clustering together. For example, we have movement terms in the upper left, we have corporate finance terms near the bottom.

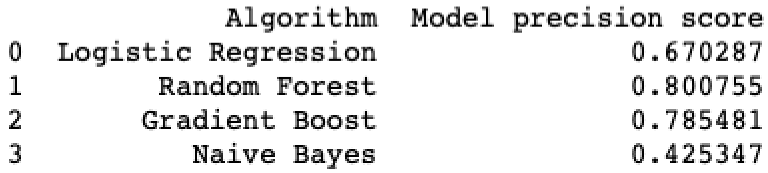


Evaluating the performance of a model by training and testing on the same dataset can lead to the overfitting. Hence the model evaluation was based on splitting the dataset into train and validation set. But the performance of the prediction result depends upon the random choice of the pair of (train,validation) set. Inorder to overcome that, the Cross-Validation procedure is used where under the k-fold CV approach, the training set is split into k smaller sets, where a model is trained using k-1 of the folds as training data and the model is validated on the remaining part.

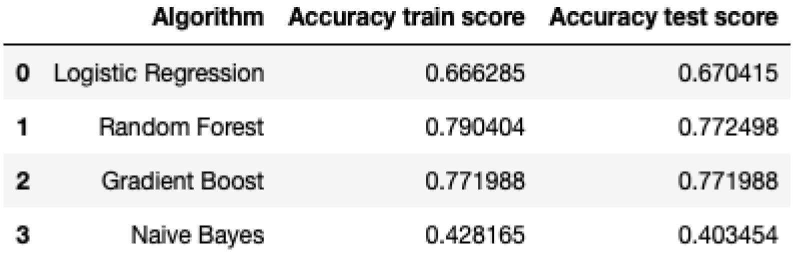
Performance Evaluation

Classification/ Confusion Matrix: This matrix summarizes the correct and incorrect classifications that a classifier produced for a certain dataset. Rows and columns of the classification matrix correspond to the true and predicted classes respectively. The two diagonal cells (upper left, lower right) give the number of correct classifications, where the predicted class coincides with the actual class of the observation. The off diagonal cells give the count of the misclassification. The classification matrix gives estimates of the true classification and misclassification rates.

Recall or Precision?
It depends on the context and the specific use case of the sentiment analysis model. In general, if the goal of the sentiment analysis model is to identify all instances of negative sentiment in a dataset, regardless of false positive predictions, then recall would be the priority. However, if the goal is to ensure a high level of accuracy in sentiment predictions, such that false positive predictions are minimized, then precision would be the priority. In our capstone, we value a high level of accuracy, and do not emphasize on whether the sentiment is negative/positive, hence precision is chosen.

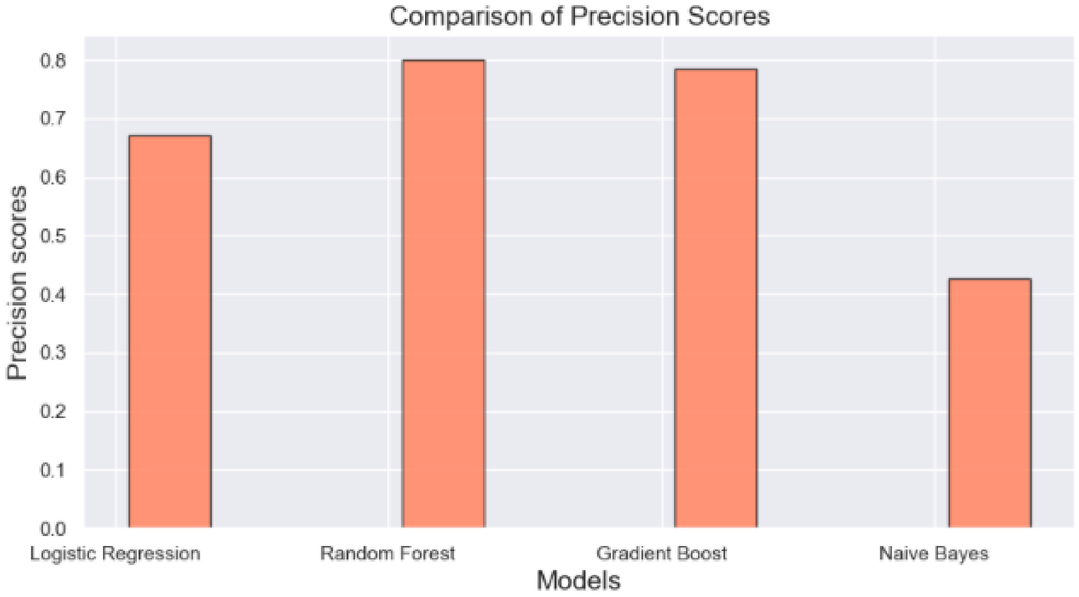


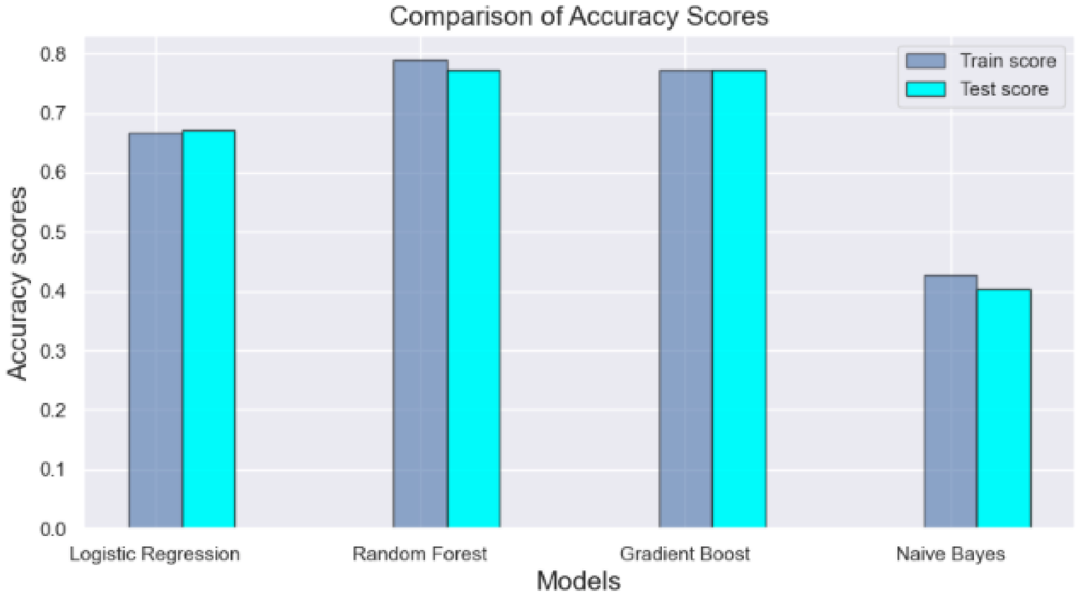
```
             Algorithm  Model precision score
0  Logistic Regression               0.670287
1        Random Forest               0.800755
2       Gradient Boost               0.785481
3          Naive Bayes               0.425347
```



	Algorithm	Accuracy train score	Accuracy test score
0	Logistic Regression	0.666285	0.670415
1	Random Forest	0.790404	0.772498
2	Gradient Boost	0.771988	0.771988
3	Naive Bayes	0.428165	0.403454

We could see that Random Forest and Gradient Boost models have the highest precision scores. I also plotted precision scores and accuracy scores for better visuals.





Conclusion

In order to predict the sentiment of each tweet, we have vectorized the tweets and applied classification machine learning models. Here we have used the following classification models:

- Logistic Regression
- Random Forest
- Gradient Boost

- Naive Bayes

Evaluating the performance of a model by training and testing on the same dataset can lead to the overfitting. Hence the model evaluation is based on splitting the dataset into train and validation set. But the performance of the prediction result depends upon the random choice of the pair of (train,validation) set. Inorder to overcome that, the Cross-Validation procedure is used where under the k-fold CV approach, the training set is split into k smaller sets, where a model is trained using k-1 of the folds as training data and the model is validated on the remaining part.

We have evaluated each model in terms of model accuracy score, and 'precision' score for both the training and test data, and plotted them. The two best performing models are the Random forest and the Gradient boost. Both are the ensemble model, based on decision trees.

Next, we have carried out the Randomized Search CV for the hyperparameter tuning for both the models separately. This step was the most time consuming one in terms of computation. (The RF model took much longer time). I originally attempted the exhaustive Grid Search CV, but with

With the result of the optimized hyperparameters, we have again fitted the two models, and got the predictions separately. The model precision did not improve much with the optimized parameters, increasing from 0.800 to 0.801 and staying the same as 0.785, for RF and GB, respectively.

Future Research

There are possible alternatives to this project that could lead to potential improvements:

- Include engineered features from the stock data obtained from the Exploratory Data Analysis notebook as the values for X in our models
- Include text mining features from the tweets data obtained from the Preprocessing and Training Data notebook as the values for X in our models
- Use a whole year worth of tweets instead of just the month of December 2022
- Use different hashtags