Twitter Sentiment Analysis

Tesla vs. Churchill Capital

IS8070 – Survey of Machine Learning
Aruna Singh
4/15/2021

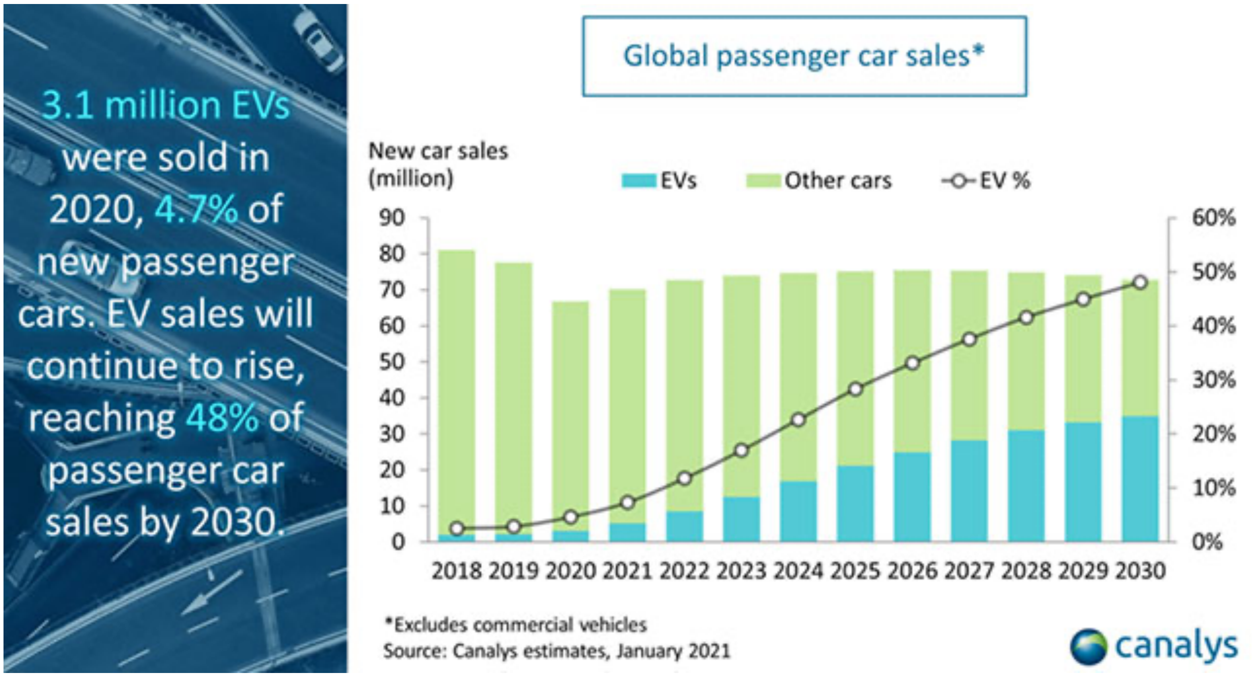
Introduction

Twitter is a social media platform where people are encouraged to share their opinions about their experiences and current events. According to Statista, there were 192 million monetizable daily active users on Twitter during the fourth quarter of 2020. Meaning, 192 million people were posting and interacting with Tweets daily. While doing so, these people were generating petabytes of data. This data can be mined and leveraged to make business decisions that are most likely to be favored by the public. Specifically, Tesla Motors has recently identified a new competitor, Churchill Capital, and will benefit from monitoring the public opinion of their brand on Twitter compared to that of the newly identified competitor. By doing so, Tesla can be the first to spot social trends and decide if a marketing campaign is necessary to help mitigate the threats presented by the new competitor.

In the type of analysis mentioned above, the syntax and semantics of the text within each Tweet are processed according to a selected algorithm. This algorithm is trained according to historical occurrences to produce a model capable of determining whether a Tweet carries a positive or negative connotation for the subject in question. Moreover, this type of analysis is referred to as Sentiment Analysis and the positive or negative connotation generated in the output of the model is the known as sentiment. After a model has been trained to identify sentiment, the model is applied to a specified sample of Tweets to determine the sentiment of each tweet. These outputs can be transformed and aggregated to derive sentiment insights. Natural Language Processing or, NLP, is the section of Machine Learning Artificial Intelligence that works to automate the human process of interpreting language. Sentiment Analysis is one of the many applications of Natural Language Processing and is used to interpret the feelings expressed by the speaker within their language.

Background

The Electronic Vehicle market is becoming one of the most competitive markets within the global economy and is currently experiencing rapid growth (Global EV Sales Growth Leads Industry in 2020).


3.1 million EVs were sold in 2020, 4.7% of new passenger cars. EV sales will continue to rise, reaching 48% of passenger car sales by 2030.

Global passenger car sales*

*Excludes commercial vehicles
Source: Canalys estimates, January 2021

Figure 1: Global EV Sales Growth Leads Industry in 2020

As Tesla sits at the top of the market with a 795.8 Billion Dollar market cap, companies like Toyota ($207.5 Billion Market Cap) and Volkswagen ($96.7 Billion Market Cap) compete to identify and obtain new customers. But these Tesla competitors have been slow to achieve much success relative to the market leader (How Far Ahead Is Tesla Compared to Toyota and VW?). On February 22, 2021, Churchill Capital acquired Lucid Motors. Churchill Capital is a special purpose acquisition company and therefore, likely acquired Lucid Motors to take advantage of the massive amount of growth in the electronic vehicle sector. According to the article “Meet the EV Rival Gunning for Tesla's Market Share” from the Motley Fool, Churchill Capital’s newly acquired Lucid Motors has the potential to compete with Tesla because the performance and price of Lucid’s vehicles is said to be superior.

Furthermore, sentiment analysis can be conducted to determine if Tesla needs to take action to prevent Churchill Capital from obtaining a share of Tesla’s market. The results from this analysis can also be used by outside investors to determine which electronic vehicle manufacturer is the better investment. Like the competitor analysis, sentient of these manufacturers can be monitored to determine if the company is about to experience financial hardship caused by social factors.

Data Source: Twitter

The Twitter API was selected as the data source for the analysis. Twitter is a great platform for obtaining data for sentiment analysis because, it is widely accepted around the world as the platform for textual expression about current events. The chart below shows how the daily monetizable Twitter users has continued to rise since the beginning of 2017.

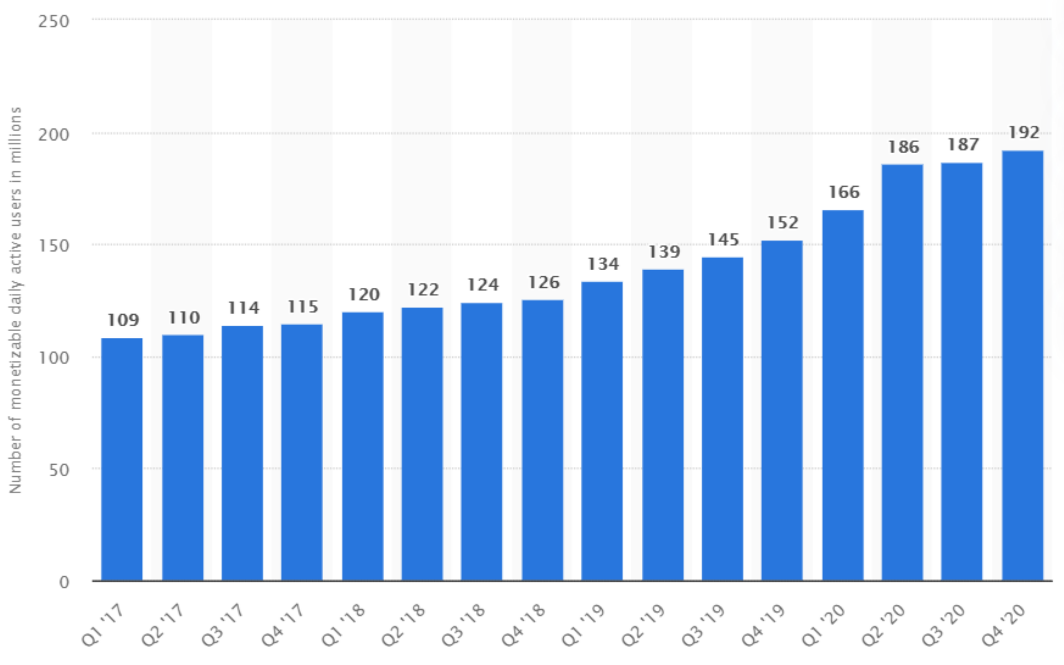


Figure 2: Statista Chart 1 - Daily Monetizable Twitter Users

Further, the field restrictions for a creating a Tweet adds a level of simplicity to the data preprocessing as this forces Twitter users to limit the number of words and types of characters within a Tweet. Additionally, the Twitter API is extremely well built out and offers extensive functionality. For example, in this analysis parameters were added to the API calls so that only important data was pulled.

The team obtained multiple data sets from April 8th 2021 to April 12th 2021 to determine the sentiments for Tesla and Churchil Capital. Training Set was obtained on April 3rd 2021 and April 4th 2021 and consists of 2360 tweets. Although almost 70% of the tweets are positive and only 30% are negative.

Challenges & Limitations

This analysis included many challenges. There were only four weeks allotted for the analysis so, project decisions were made to fit this time constraint. Further, an extensive analysis would require several months of work. Taking this into consideration, a simple technology stack was selected for the project. This stack included, Jupiter Notebook, scikit Learn, and Python. Moreover, it was difficult to find or create training data for the project within this short amount of time; so, the training data was labeled using a TextBlob categorization model. Ideally, the SVM algorithm would have been trained with a larger dataset that has been manually tagged. Overall, it was determined that the generated model categorized positive tweets better than negative tweets. This could be a result of the limited available training data for negative sentiment.

Qualities of the Twitter data also made it difficult to analyze. Twitter data is high velocity, volume, and variety so, it can be hard to obtain a valid sample of all the tweets out there. To resolve this, further analysis would be conducted only using tweets from specified user accounts. Additionally, sarcasm and other jargon can be difficult process. This type of language uses positive words negatively and can confuse categorization. Additionally, some tweets contained speech about both ‘cciv’ and ‘tesla’. If people Tweet about both Tesla and Churchill Capital at the same time, there is not a way to determine which manufacturer the Tweet is referring to when defining sentiment. A Tweet containing a good review for Tesla and a bad review for Churchill Capital may be classified as a positive sentiment for both manufacturers. Preprocessing also caused challenges for the analysis. Moreover, keywords in links and hashtags were removed and therefore could not add value to the analysis. Finally, API limitations made it difficult to acquire large sets of data. Only 3200 requests could be made every 15 minutes so, acquiring large sets of data was very time consuming.

Data Preparation

Data Preparation is one of the most important steps in any analytic project. We took good time to understanding and planning this process throughout the project.

DATA ACQUISITION

To fetch the data from the twitter we used Tweepy. An Open-Source Python package, Tweepy helps developers access Twitter API in a convenient way. Tweepy offers several functions, classes and methods describing Twitter API endpoints and template. It handles diverse design specifics clearly, such as:

- Data encoding and decoding
- HTTP requests
- Results pagination
- OAuth authentication
- Rate limits
- Streams

All the Twitter API functionalities can be used through Tweepy as it provides the convenient Cursor interface to iterate via various types of objects. Most Twitter elements are open to developers in the Twitter API. Developers have the option to use the API for reading and writing Twitter details including tweets, users, and patterns. Furthermore, the API has functionality to query across tweets and indicate data parameters such as language, date, and location.

Technically, the API provides hundreds of HTTP endpoints related to:

- Tweets
- Retweets
- Likes
- Direct messages
- Favorites
- Trends
- Media

The Twitter API utilizes an open authentication mechanism known as OAuth. It is a commonly used mechanism to authenticate a request. Developers must build and customize the authentication keys before making a request to the Twitter API. The Twitter API often restricts the number of occasions the Application can be called. Developers need to wait between 5 and 15 minutes to use the API again if developers go over the set thresholds.

Creating Twitter API Authentication Credentials

The requirement of Twitter API is that all requests use OAuth to authenticate any call to the API. Hence, there is a necessity to create the required authentication credentials to be able to use the API. The API class in Tweepy has several methods that provide access to the endpoints of the Twitter API.

Methods for searches

In our project we searched for tweets using text, language, and other filters leveraging over the api.Search() method.

- api.search returns a search object which is like list of dictionary like objects containing a lots of elements for a tweet such as the short-text, long-text, date of creation, name of the author, number of followers the user has etc.
- The search parameter q='\"{}\" -filter:retweets'.format(search_term): ensures that the tweets sent contain the specific search phrase and that retweets are not returned in the results as it can result in identical data.
- The extended parameter for the tweet_mode ensures that we get back full text of each tweet and not just the preview along with a url to the original tweet. Omitting this gets the developer a short version of the tweet followed by ellipses and a url to the tweet.

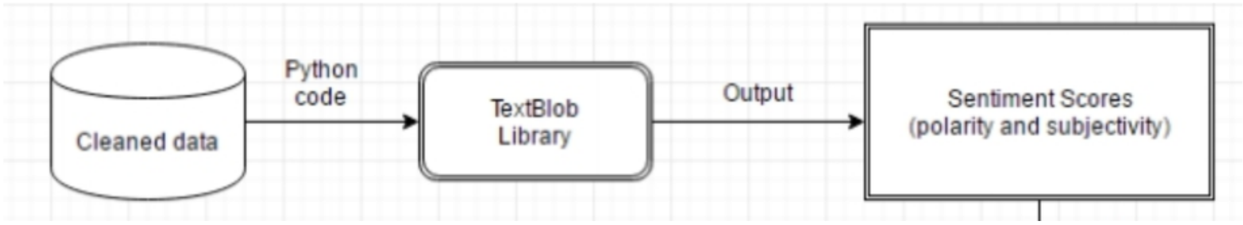
Data Processing

The initial step of 'cleaning' the data was to convert all letters to lowercase, then remove punctuation, numbers and URLs and usernames from the Tweets using Regular Expression. A simple regular expression in Python was used to remove any punctuation, HTML characters, hash marks, URLs and whitespaces in the Tweets. The cleaned tweets were then saved in a file in Comma-Separated-Values format.

Duplicate tweets are then omitted. We chose to remove duplicate tweets after the other pre-processing steps due to the nature of Twitter which consists of 'retweets', replies to other users or remarks, they may have the exact same content. The cleaned data was then stored in a CSV file.

To label our training dataset, TextBlob was used in the analysis to label the training data. TextBlob is a Python library for processing textual data. It provides a consistent API for diving into common natural language processing (NLP) tasks such as part-of-speech tagging, noun phrase extraction, sentiment analysis, and more. Ideally, this task should have been done by a human. It is essential that the training set is labeled correctly so the correctness, accuracy, performance, robustness, and reliability of the model can be determined.

However, due to lack of time, we used TextBlob instead for the task. TextBlob applies a rule-based approach to analyze the sentiments of a tweet. It therefore requires a pre-defined set of categorized words. These words can be uploaded from the NLTK database and the textblob determined the sentiment scores in the form of polarity. We can see how this process works in the below mentioned diagram:



There was an issue in the spread of the Tweets when looking at the figures in the breakdown of the dataset. Positive Tweets are ubiquitous and are over thrice the proportion of negative tweets. This had consequences on how well the classifier worked in practice. Trained like this, the classifier has great understanding of positive tweets but has not much practice identifying negative tweets or anything else.

Support Vector Machine (SVM)

As our classifier we used, Support Vector Machine or, SVM. SVM is a supervised machine learning algorithm that can be used for classification and regression as well. Classification is predicting a label/group and Regression is predicting a continuous value. In our case we have used SVM to classify our sentiment labels. SVM performs classification by finding the hyper-plane that differentiates the classes that are plotted in an n-dimensional space.

SVM derives this hyperplane by transforming the data with the help of mathematical functions called “Kernels”. In our project we have used the Kernel strategy as “linear”, which is for linear separable problems. Since this analysis only determines sentiment between positive and negative, it is linear. So, linear SVM was used for this analysis.

Furthermore, we useTF-IDF Method transform text data into vectors:

TF-IDF defines importance or relevance of a term by taking into consideration the importance of that term in a single document, and scaling it by its importance across all documents. TF-IDF is a popular method which is actually "Term Frequency and Inverse Document Frequency". TF-IDF is a word frequency score that tries to emphasize terms/words that are more relevent, e.g. frequent in a document but not across documents.

In implementation of this analysis, we first use the TfIdfVectorizer of Scikit-learn to extract features from our training set. Using the vectorizer and features from the training set we acquire the predicted labels of the test set by classifying the test set using SVM function of scikit-learn package.

The classification results in the predicted labels which we can then compare to determine the accuracy of the model.

Evaluation of SVM

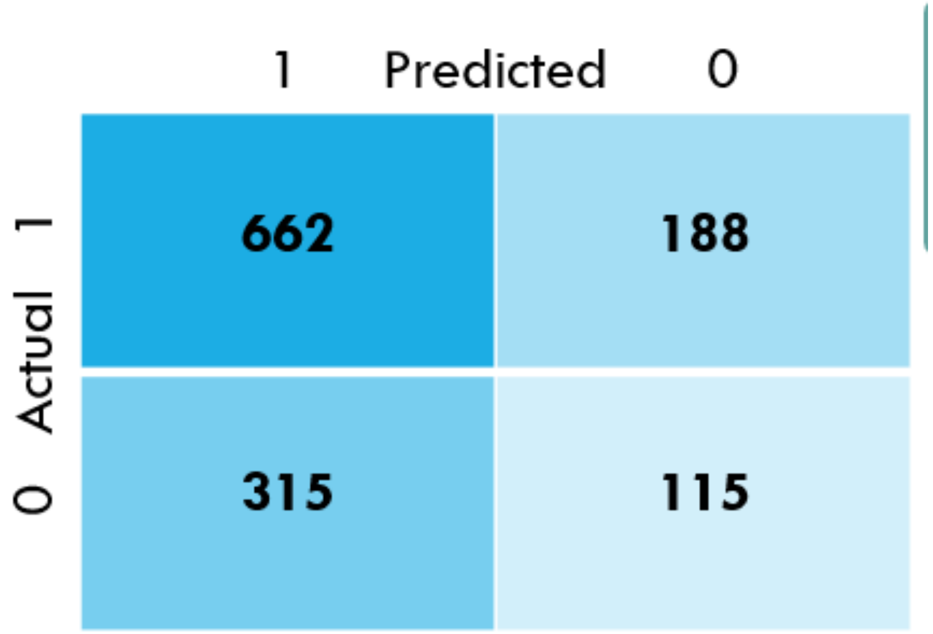
Experimental evaluation metrics vary and are usually dependent on the nature of the task being conducted. Some of the typically used evaluation metrics for analytics for analytical procedures include, but are not limited to; accuracy, precision, recall, Mean Squared Error, analysis of the Loss Function, Area Under the Curve, F1-Score. Different models in different domains will result in different results for each metric and suitable one must be decided upon and must meet evaluation criteria necessary.

Accuracy

Accuracy is one of the most frequently measured evaluation metrics in classification tasks and is most often defined as the number of correctly classified labels in proportion to the number of predictions in total.

Confusion Matrix

Confusion matrix is one of the most popular metrics for binary classification problems. It is used to evaluate the quality of the output of a classifier on the twitter data set. The diagonal elements represent the number of points for which the predicted label is equal to the true label, while off-diagonal elements are those that are mislabeled by the classifier. The higher the diagonal values of the confusion matrix the better, indicating many correct predictions.



		Predicted	
		1	0
Actual	1	662	188
	0	315	115

F1-Score

The ‘F1 Score’, ‘F-Score’ or the ‘F-Measure’, is a common metric used for the evaluation of Natural Language based tasks. It is often said to be ‘The Harmonic Mean of Precision and Recall’ or conveys the balance between precision and recall.

The F-Measure expresses the balance between the precision and the recall. As accuracy only gives the percentage of correct results of the model but does not show how adept the model is at finding true positive results, both measures have merit, depending on the need.

The Support Vector Machine found it extremely difficult to make correct classifications when trained on imbalanced training data. Not using a parameter tuning technique and just a simple linear approach may have caused issues also.

SVM’s are sensitive to imbalanced data and work best with naturally balanced classes. That may have caused decreased performance. It also explains how the unbalanced experimentation yielded less useful results.

Results

SVM Outputs:

Accuracy determines the number of correctly classified labels of positive and negative sentiments in proportion to the number of predictions in total. We are using it as the base metric to quickly evaluate the models.

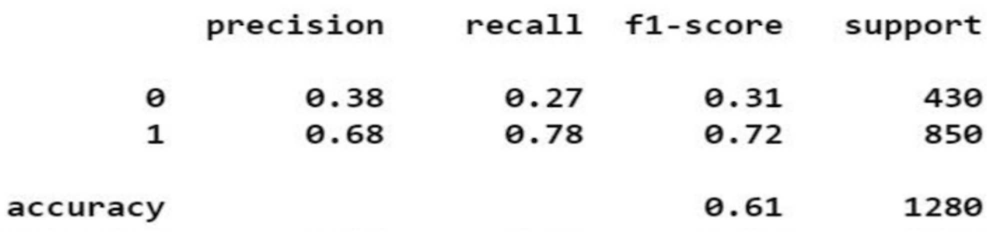
In addition, precision is a measure of how good the classifier is classifying reviews as positive sentiment.

Recall measures how good the classifier is at correctly classifying reviews as negative sentiment.

F1 score is a metric that combines the trade-offs of precision and recall. It decides the overall goodness of the classifiers.

In order to conclude the overall analysis, the cost of false positive is markedly higher than the false negative, that is why predictive accuracy is not enough to measure the performance of a model.

Therefore, the result shows that classifying negative reviews is more difficult than classifying positive reviews as our input dataset is imbalanced with 70% positive and 30% negative tweets. Henceforth, It is always good to keep the size of the input data is sufficiently large to get the performance of the model better.



```
              precision    recall  f1-score   support

           0       0.38      0.27      0.31       430
           1       0.68      0.78      0.72       850

    accuracy                           0.61      1280
```

WordCloud:

WordCloud is a technique to show which words are the most frequent among the given text. However, the only required argument for a WordCloud object is the text, while all others are optional.

Using the first observation description as the input for the wordcloud. The three major steps are:

1. Extract the tweets
2. Create and generate a wordcloud image
3. Display the cloud using matplotlib

Henceforth, the positive and negative wordcloud has been created out of the sentiment score after extracting from the textblob and added some optional arguments like max_font_size, max_word, colormap, collocations, and random state.

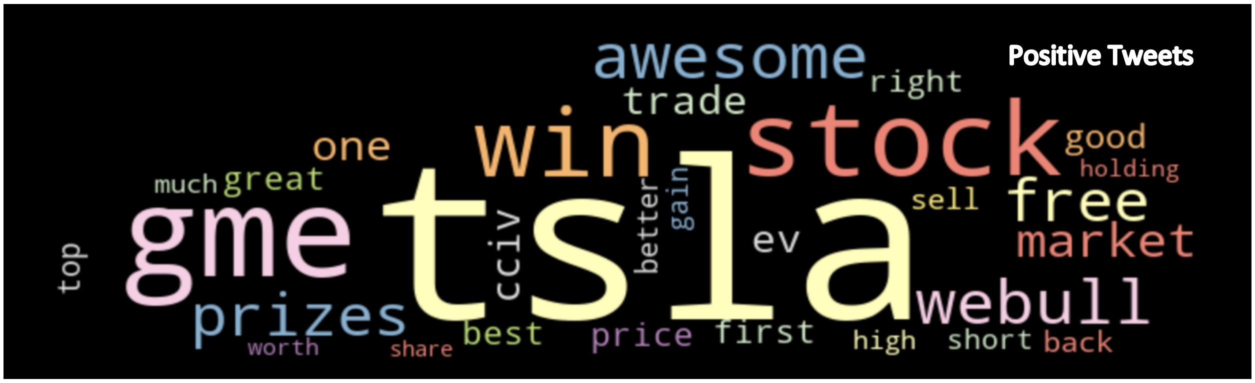
Now, let's fill these words into a shape of car!

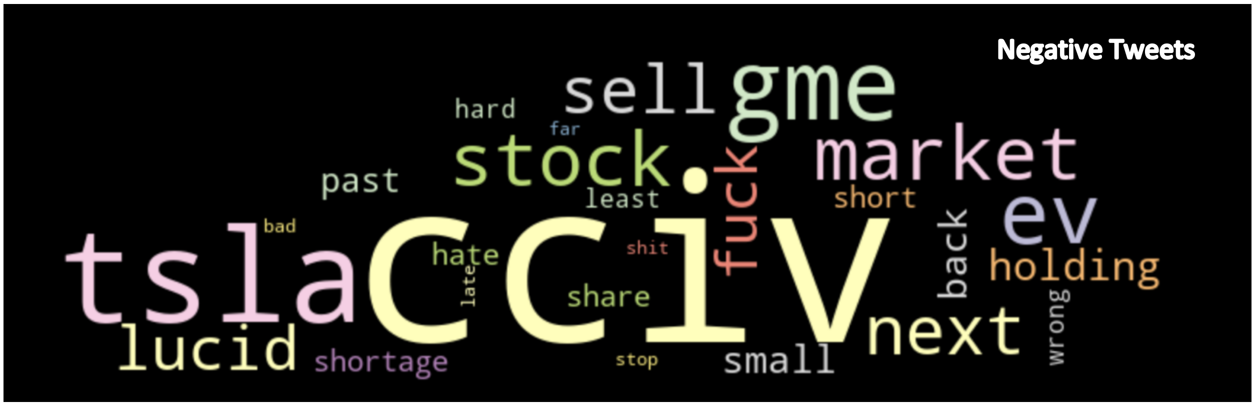
In order to create a shape for your wordcloud, first, you need to find a PNG file to become the mask and add one more optional argument as mask to the wordcloud function.

The way the masking functions works is that it requires all white part of the mask should be 255 not 0 (integer type). This value represents the "intensity" of the pixel. Values of 255 are pure white, whereas values of 1 are black. Here, you can use the provided function below to transform your mask if your mask has the same format as above. Notice if you have a mask that the background is not 0, but 1 or 2, adjust the function to match your mask.

Finally, the argument interpolation as bilinear used in the plt.show() to make the displayed image appear more smoothly.

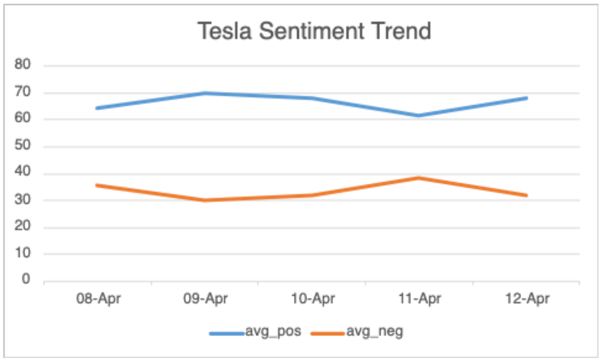
Voila! The wordcloud in the shape of a car is created for both positive and negative sentiments! It seems like tesla is emerging more in positive tweets and cciv is more in negative tweets which is clearly depicted in the below mentioned wordclouds:

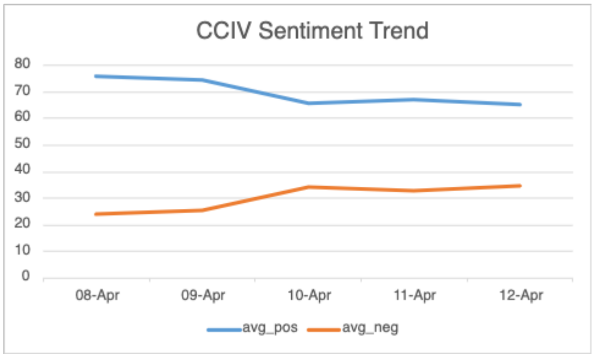


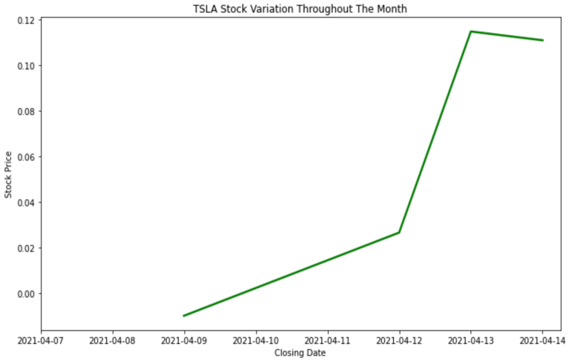


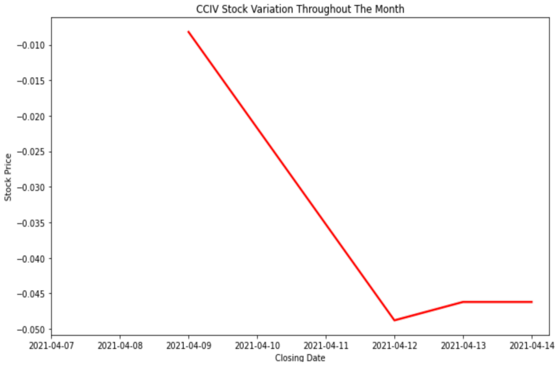
Stock Price Variation: Our SVM model seems to work well, but do the tweets' overall sentiment correlate with real stock price movements?

By plotting Tesla and CCIV tweets' sentiment alongside their historical stock price performance, we can assess our approach's potential viability. Therefore, it shows some pretty impressive results as the average positive sentiments are rising for Tesla on the daily basis while the average negative sentiments are overpowering in the case of CCIV. Similarly, the same variation can be visualized through the stock price fluctuations during the period of one week.









It’s clear that the Twitter sentiment and stock price are correlated during this week. Of course, a larger timespan would provide greater confidence, but this provides us with an initial positive outcome to investigate further.

Extensions & Takeaways

Further work could be done to increase the performance of the model. One major improvement that could be made is in the training data. A larger training set should be created with more specific text to CCIV, Tesla and the electronic vehicle sector. Also, data from additional social medias can be pulled to add to create a sentiment score for a business across multiple social media platforms.

In this analysis, the SVM was used to monitor the sentiment of Tesla vs Churchill capital and compared to the stock price to determine if Tesla needs to take competitive action against Churchill Capital to maintain Tesla’s market share. In this case, SVM is being used for brand management. Other applications also include monitoring a product release. For example, upon launch of a new product, a business can gauge the public’s interest before and after the product launch. By monitoring the public opinion of the product, a business can more accurately predict demand and resolve product issues before they come to light. Also, Sentiment analysis can be used for market research. Potential customers can be identified based off of what they like and a business can monitor key word trend to determine how to market their products.

Conclusion

In conclusion, it was determined that Tesla can use the sentiment model created in this analysis to determine if their stock price is suffering because a new competitor has entered the market. From the results of the analysis, it was not determined that Tesla needs to take any preventive action to protect its market share. This is because, sentiment of CCIV has not seemed to spike over the last week. Additionally, Tesla sentiment has increased over the week.

References

- Statista Chart 1: https://www.statista.com/statistics/970920/monetizable-daily-active-twitter-users-worldwide/#:~:text=Twitter%3A%20number%20of%20monetizable%20daily%20active%20users%20worldwide%202017%2D2020&text=This%20statistic%20shows%20a%20timeline,amounted%20to%20192%20million%20users
- Meet the EV Rival Gunning for Tesla's Market Share: https://www.fool.com/investing/2021/03/27/meet-the-ev-rival-gunning-for-teslas-market-share/
- How Far Ahead Is Tesla Compared to Toyota and VW?: https://www.motorbiscuit.com/how-far-ahead-is-tesla-compared-to-toyota-and-vw/
- Global EV Sales Growth Leads Industry in 2020: https://www.technewsworld.com/story/87013.html
- Yahoo Finance
- Lucid EV Picture: https://finance.yahoo.com/news/lucid-motors-deal-churchill-capital-224942799.html
- Tesla EV Picture
- https://scikit-learn.org/stable/modules/svm.html#classification
- https://towardsdatascience.com/a-three-level-sentiment-classification-task-using-svm-with-an-imbalanced-twitter-dataset-ab88dcd1fb13
- https://www.researchgate.net/publication/321084834_Sentiment_Analysis_of_Tweets_using_SVM
- https://medium.com/@vasista/preparing-the-text-data-with-scikit-learn-b31a3df567e
- https://medium.com/@vasista/sentiment-analysis-using-svm-338d418e3ff1
- https://aadaobi.medium.com/exploring-twitter-api-and-data-using-tweepy-pandas-and-matplotlib-part-1-2ac07fcc4717
- https://stackabuse.com/python-for-nlp-introduction-to-the-textblob-library/
- https://www.kdnuggets.com/2017/02/yhat-support-vector-machine.html
- https://www.datacamp.com/community/tutorials/wordcloud-python