George Padavick, Matt Gilgo
CS 506, Spring ‘21
Final Project Report (Draft)

# War on Wall Street: Analysis of Today’s Stock Market vs. Social Media Influence

## Introduction

The year of 2020 was no doubt an unprecedented time. A pandemic struck the masses and caused a paradigm shift in society’s day-to-day interactions. Families were confined to their homes, left without the various methods and ways they formerly passed the time or spent their money. In an effort to entertain themselves and spend the extra money on hand, a primary avenue many people turned to was the stock market. With the perfect storm of a massive stock market crash offering discounted prices, mobile apps like Robinhood and Webull offering easy-to-use methods of opening an account, and ample time for investors to spend on their social media accounts, this combination turned a place for passive money growth into an all-out gambling venture.

This trend culminated in January 2021 with the pumping of then-dying stock Gamestop (GME) from $17.85 to $483.00 in less than a month. The growth in this stock was not based on improved revenue, a business model transition, or anything that would traditionally move the needle. This time, it was driven by a movement on various social media platforms. Users across the world on applications such as Twitter, Reddit, Facebook, etc. were seen tagging GME in posts and telling others to band together and get rich as they drove up the price to squeeze the “short” position hedge funds into bankruptcy. This mass craze garnered a ton of media attention and left many people blown away at how people could influence the market so vastly just by banning together on social media and presenting a call to action to a mass following.

What people had not realized though is that, at a smaller scale, movements like these had been going on for many months prior. Stocks in companies such as XspresSpa (XSPA), Nikola (NKLA), and Genius Brands (GNUS) had seen major price inflation due to a social media community driving in new investors through using similar rhetoric. Just as GME grew, however, these stocks also saw a similar plummet of the price soon after the popularity had hit its peak. In only 3 weeks after posting its All-Time High, Gamestop had crashed back down to the $40 range, leaving investors at the top of the hype with only 10% of their original investment.

Through popularized data science tools, our group sought to determine if there were obvious indicators between stock price changes and their respective rise in social media popularity. These findings will seek to answer the question of social media as a reliable source of information for a new investor. With time permitting, the stretch goal would be to develop a model that could help a new investor determine what stocks were being “pumped and dumped”, times to get in and out of trades, and avoid any potential massive losses from popularized risky investments.

**Data Collection**

In an effort to start working towards these goals, finding proper data sources was the first step. There is a vast existing landscape of social media platforms that could be used, but in respect to time, it was determined to choose one platform that would provide us with a large amount of data as well as a high ease-of-use for that data. Of the platforms considered, Twitter was determined to be the best platform for this study. With that in mind, a Twitter API and a stock market API were needed to pair the social media sentiment to the stock market performance. Tweepy is a reliable, direct-from-source API for pulling data from Twitter, so that was determined to be the best option for tweet characteristics. With a breadth of financial APIs though, research and tests were done on APIs for Alpha Vantage, Finnhub, and YahooFinance to find which API would be the most useful. Due to a high level of documentation, ease-of-use, and having all the features needed for the research, YahooFinance's API became the source of the financial data seen in this report. Tests on these API's can be found in the code/api_testing folder of the project GitHub repository.

**Data Progression/Cleaning**

With the data sources determined, the next step was to set up our data pipeline and clean the data for better results downstream. Below in Figure 1 is a data map of our proposed pipeline:

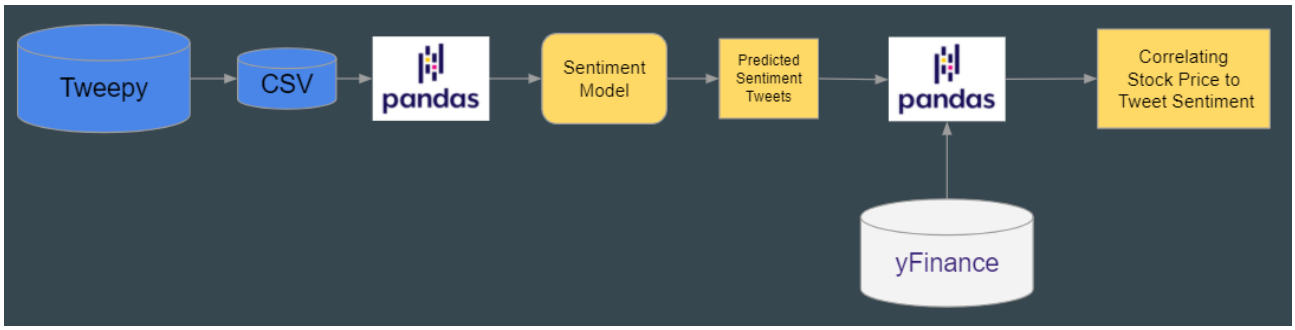


**Figure 1: Data Progress Map**

To give further description into Figure 1, Twitter data was extracted from the Tweepy API and stored in pickle files specific to each stock in consideration. Due to API limitations that restrict data pulls to only tweets in the last seven days, only stocks that had quick pump-and-dumps over the last two months could be considered. Stocks that were used for the training data set were tickers:

- EEENF
- GYST
- DLPN
- TTCM
- WNRS
- SEAC
- CERPQ

To clean the data for these tweets, “text” data from the API was retrieved, then stripped down to its bare syntax by removing uppercase letters, unnecessary punctuation, and hyperlinks to websites that did not provide useful information. Besides the text data, the provided timestamp needed to be adjusted to a usable format, so the times were merged into a datetime format and set to Universal Time. Lastly, the tweet ID was causing issues being extracted into csv format, so a character had to be added to the ID to keep it as a string and not reduced to scientific number format.

There were various decisions that had to be made on the Twitter data in efforts to present the most realistic dataset. For example, tweets such as retweets and quote tweets were included as additional tweets since it represented an expression or occurrence of somebody’s sharing or repeating one's information to a separate audience. Along with that, emojis were deemed as important to keep in the dataset since they are an effective way to express a users opinion on a stock with less than the 280 character limit Twitter enforces.

Besides Twitter data, the Yahoo Finance data had a few decisions that needed to be made as well. For example, the stock data could be exported in various buckets of intervals (5-Minute, 30-Minute, 1-Day, etc.), so choosing a proper one to mate well with the twitter data was important. With both of these datasets in a Pandas dataframe, the merge_on function provided the capability to round tweets to their closest time bucket of the stock data. It was found that stocks with high tweet volume had great success being joined with 5-Minute interval data, while lower tweet volume stocks had more success with 1-Hour intervals.

**Feature Extraction**

With the data retrieved and cleaned from the API sources, the next step was to identify important features that would be used in the model training portion of the project. These features would have to be ones that showed a correlation between tweets and price, or features that can be tied directly to the sentiment of a tweet.

Starting off, we looked at various features from the Twitter data and plotted them against the stock price over the time. This was achieved by visualizing pairwise plots and checking for correlations amongst the data, as well as positive vs negatives trends with price performance. Alongside reviewing the correlations, another path we pursued was parsing the text for strings that would give us an indication of what the tweet sentiment should be. Because Twitter language around stock prices is much different from the traditional Twitter sentiment, we chose to instead make our own training sets based off the stocks listed in the section above. During the scoring process, four additional features were added on to the original dataset provided by twitter. These features were Sentiment Score, Known Pumper, Price Region, and Inflection Point. Sentiment Score and Price Region were both rated on a -1, 0, 1 (Negative, Neutral, Positive) scale, as to give the model the ability to train off both sentiment of the tweets and points in time whether it was advantageous to sell, hold, or buy the stock. The Known Pumper and Inflection point features serve as a binary decision to show where the tweet came from a

popular pumper on Twitter, as well as whether or not an inflection point in the stock price occurred and if it was going up or down when it drastically changed direction.

Using those scored tweets, the text was passed through a word tokenizer to determine which words and emojis would be the most beneficial to use in determining the remaining tweets sentiment and price action. An example of this step can be seen below in Figure 2.

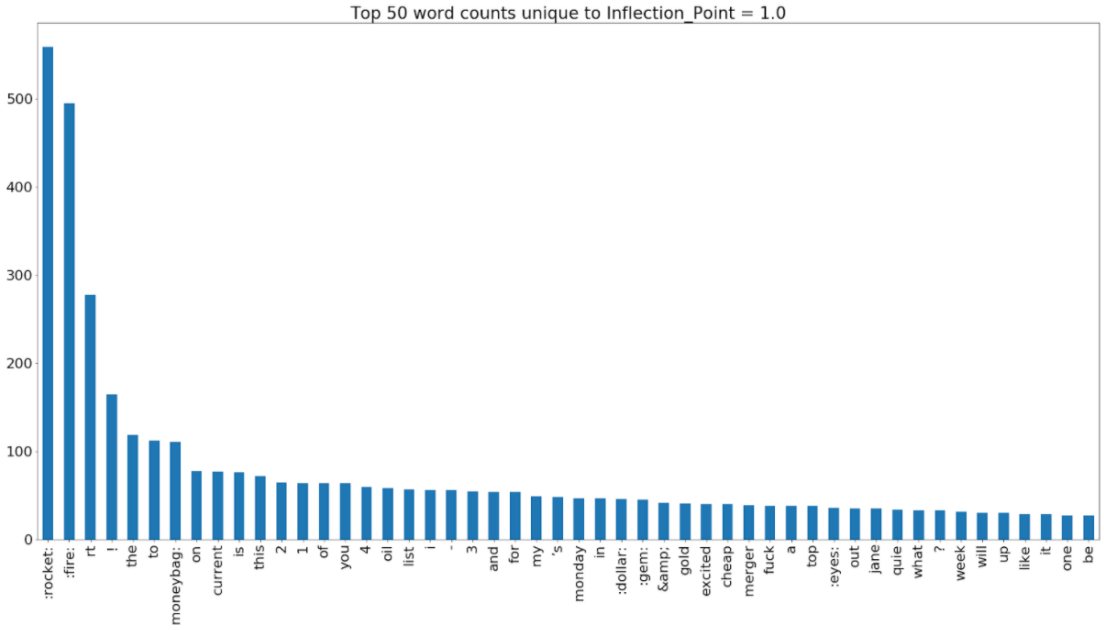


**Figure 2: Top 50 Word Counts for Tweets near Positive Inflection Points**

Figure 2 shows the results for the most popular words and emojis around a positive price inflection (aka the start of a pump). As seen, a high use of emojis such as the “rocket”, “fire”, and “moneybag” in tweets are a heavy indicator that the price is starting to head in a positive direction. Outside of emojis, words like “current” (as in an OTC stock upgrading to Pink Current status), “excited”, and “cheap” all pass along tones of hope to potential buyers in the stock that do not want to miss out on what is viewed as an easy money maker. To counteract that, the most important words in tweets at the negative inflection point were observed as well.

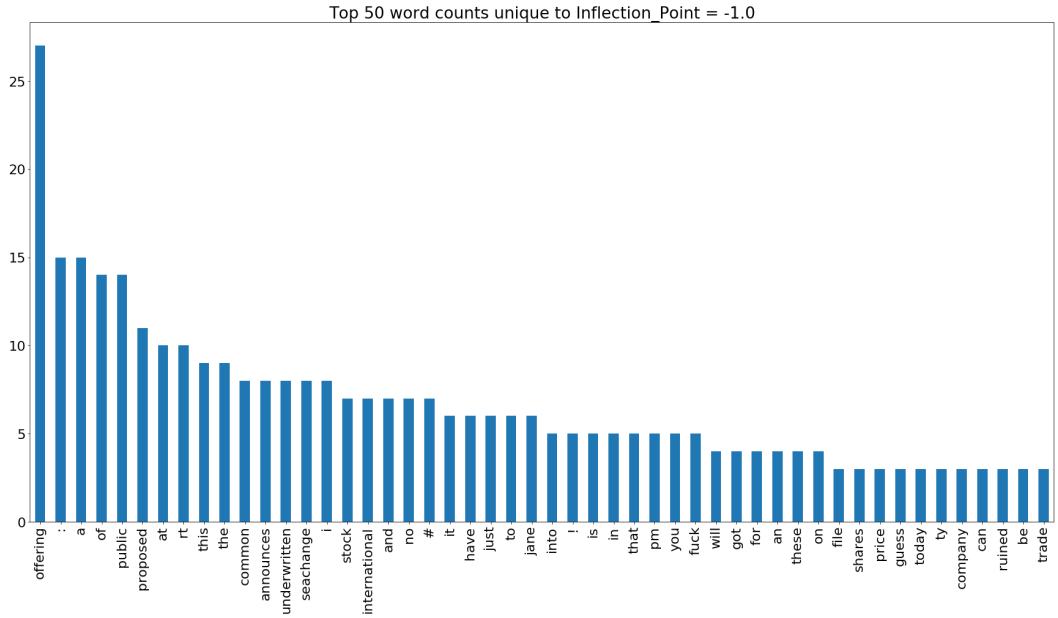


**Figure 3: Top 50 Word Counts for Tweets near Negative Inflection Points**

This exercise was done on the Inflection Points and the Sentiment Scores to provide popular words to use when determining the classification of tweets. Emojis were not as prevalent in the negative inflection points, but words such as “offering”, “proposed”, and “announces” stood out as common indicators of an impending negative shift in the stock price. These words seemed to correlate well with the old adage “buy the hype, sell the news”, with each word being related to either an announcement around the stock, or more specifically, a public offering of common stock.

Using these words to help score sentiment in our model, various models were explored in an effort to choose which model would be most accurate and efficient. The accuracy scores to these various models are provided below in Figure 4.

```
LogReg
0.8380758807588076
RF
0.8380758807588076
KNN
0.8177506775067751
SVM
0.8407859078590786
GNB
0.5047425474254743
```

**Figure 4: Results of Tweet Sentiment Predictions using Various Models**

From these results, it was determined that our future analysis would be performed with the Random Forest approach since there was a decent amount of overlap in the data (ruling out SVM) and could be deployed to larger datasets as needed.

## Analysis

Tbd…

## Conclusions

Tbd…

## Limitations/Future Exploration

Tbd...