Stock Market Prediction from Public Sentiment on Twitter

Sean Healy
College of Engineering
University of Notre Dame
Notre Dame, Indiana 46556
Email: shealy3@nd.edu

Nick Fahrney
College of Engineering
University of Notre Dame
Notre Dame, Indiana 46556
Email: nfahrney@nd.edu

Conor Rinehart
College of Engineering
University of Notre Dame
Notre Dame, Indiana 46556
Email: crineha1@nd.edu

***Abstract*—Social media is a way for the world to stay connected but it can also be utilized as a powerful tool. Twitter sees 500 million tweets per day with people expressing their thoughts from all over the globe. All this information can be transformed into the largest collection of peoples beliefs and feelings that the world has ever seen. Investors and hedge funds are always looking to innovate and try new ideas that could potentially lead to lucrative trading patterns. If we could predict future stock market gains or losses based off public sentiment on Twitter that would be a very valuable resource. We collected tweets that included the word tesla or tesla and determined the collective sentiment of these tweets. With this data we we were able to generate an an average sentiment of the public on Tesla and then compare an adjusted sentiment score to Tesla's stock market performance. With a data set of 100,000 tweets spanning across multiple years we found that using tweets that specifically referenced Tesla there was a 41.67% correlation between the sentiment score the previous day and the stock market performance of Tesla the next.**

I. Introduction

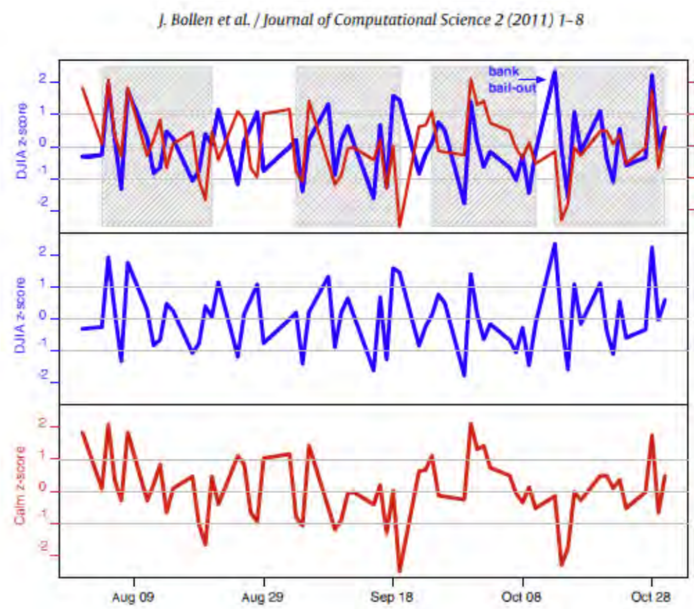
As technology has evolved, so have the ways in which people use it to improve their daily lives. With the advent of computers in the late 1900s we saw people begin to no longer trade on the stock market with solely their own ideas but develop algorithms to do the work for them. People have developed systems that can track patterns and recognize opportunities to buy and sell certain stocks. Computers have the huge advantages of being able to take in massive amounts of information and to perform analysis on it with blazing speed - two things that are extremely valuable in the fast-moving and multifaceted stock market.

Some time later, social media arrived on the scene and transformed the way that human beings interact with each other. We were suddenly able to access millions of people's thoughts and feelings with the touch of a button through Twitter, Facebook, as well as a multitude of other platforms. People have the freedom to talk about whatever they want on social media, anything from video game tips and tricks to in-depth discussions on the methodology behind a statistical model to breaking news. Could determining how people feel about a certain topic enable a person or system to get an inside edge on the the future movements of the stock market? Would it be possible to determine a company's change in stock price from the way people are talking about it on social media?

It is important to note that not all investors come from financial firms with large capitol and a propensity for in depth analysis of a company. Changes in stock prices are not always based on the balance sheets of companies, but also speculation by investors about future earnings, expansion, or crises (e.g., a pandemic). The stock market, as a result, can often function as a barometer of how the public feels about certain companies. Social media, and in particular Twitter, is simply another method for capturing people's thoughts, feelings, and opinions. This project aims to gain a deeper understanding of the interconnectedness of these two mediums so that one can be used to predict the other.

II. Related Work

The hundreds of millions of users on these social media sites generate an unbelievable amount of information about a vast number of topics. With all this information publicly accessible online, companies and individuals alike have realized that they could find a use for it. In 2010 a group of Indiana University researchers began looking to see if there was a correlation between the collective public mood on twitter and fluctuations with the Dow Jones Industrial Average. They did not do a big sentiment analysis of all of the tweets coming through Twitter. They instead decided to narrow their scope and look only at tweets that were specifically conveying people's moods or how they were feeling. With this approach, they began to run tests to see if this could improve upon already existing models that tried to predict closing stock prices from other factors. They found that they could enhance these other models with an accuracy of 87.6% [1]. Overlaying the results of their work with the Dow Jones produces the following graphic:



[2]

While the group of Indiana University researchers were some of the first to utilize Twitter data to predict the stock market, they were certainly not the last.

Multiple European groups have utilized Twitter to predict future stock market results as well, this time with a greater emphasis on the popularity of tweets. A Dutch group looked into how tweets that were saying buy or sell in relation to a company would correlate to future performance of that stock. They found that if you were to further curate the tweets to those that were more popular, either by looking at the follower count of the person tweeting or the number of retweets a tweet received, the correlation was further improved. They tested this correlation by simulating trades using this model over the course of 21 weeks and the model consistently beat the stock market [3]. A German team that essentially used a method that was a combination of the Dutch group's and Indiana University's methods increased its portfolio by 36% over a 9 month time frame. The Germans looked into gathering data to determine the mood through twitter. They filter the tweets not for specific references to mood but instead for the popularity of the tweets. This allowed them to favor tweets that were more influential with users. Multiple studies and research teams have shown that twitter data, when used carefully and correctly, can be a valuable source of information for predicting the movements of the stock market.

III. Problem Statement

The goal of our work was to follow in the footsteps of the aforementioned researchers and see if we could construct a program that was capable of predicting the stock movements of a single company solely from Twitter data. We wanted to gather a large enough amount of data that the calibration of our system could come from back-testing, so that it would be immediately usable. We also knew that it was important for the results to be relative to the general stock market, so building that aspect in was part of the goal as well.

IV. Solution

The first step we had to take to kick off our work was to gather tweets from a particular day(s) and compare the measured sentiment of those tweets to the stock market performance that day and the days following. The sentiment analysis tool that we settled on using was textBlob. We chose textBlob because multiple studies had shown that it is one of if the not the best free sentiment analysis tools [4]. The way the tool works is as follows: it takes a string of text as an input, and then analyzes it to produce a "sentiment score." This score ranges from -1.0 to 1.0, with -1.0 being the most negative and 1.0 the most positive. A score of 0 would say that the input was a totally neutral block of text. Tweets are classified as positive if their score is greater than 0 and negative if less than 0.

The studies we consulted did however find that while the tool was fairly accurate when predicting positive sentiment, it was not as good at picking up negative sentiment. We saw this ourselves when looking at data sets involving tweets on topics that were certainly negative. Even when looking at keywords associated with deeply negative events, such as "Harvey Weinstein" during the "MeToo" movement, TextBlob would classify around 20% of tweets positive and 10% of tweets negative. Clearly there was a discrepancy in the data here so we performed a manual assessment. Analyzing the tweets ourselves, it was clear that there were more than 10% negative tweets in the data set. Looking into this we found the flaw was with TextBlob's rating system and subsequent classification. There were a large number of tweets graded out as barely positive (e.g., a sentiment score of .02) that were classified as positive despite being far closer to a neutral score than a 1.0, while all the tweets that were being classified as negative were much closer to a truly negative score (-1.0). It was clear that textBlob had a tendency to lean towards a slightly positive score as opposed to a slightly negative score when looking at a neutral tweet. Factual news headlines would consistently get classified as barely positive when they were clearly meant to be neutral. To combat this problem, we changed the inner workings of TextBlob to output the sentiment score that previously was only being used internally to classify the text as positive, negative, or neutral. From there, we changed the way we used the output from TextBlob - instead of only tracking percentages of positive, negative, or neutral tweets, we added all of the tweets' sentiment scores together to get a number that represented the overall sentiment score for the data set. This way every tweet had an equal weight to in determining the resultant stock prediction. Using percentages meant that a tweet with a .1 score would fully cancel out a tweet with a -1.0 score. Adding these numbers together meant that a tweet would only get counted proportionally to how positive or negative it was.

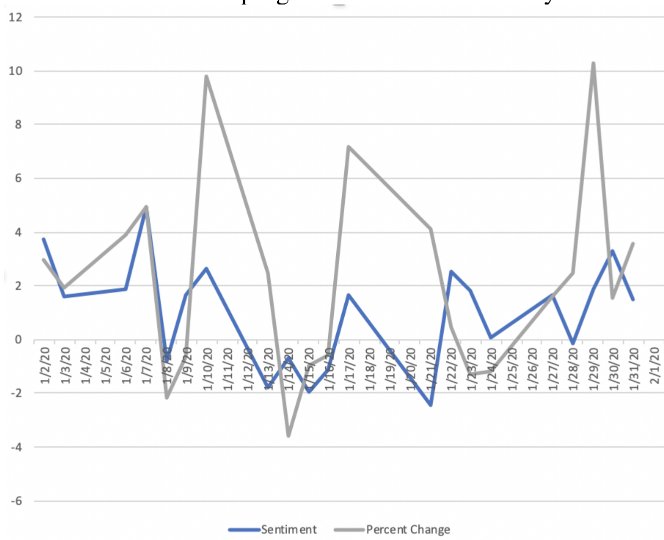
This number was still not perfect, however and problems quickly arose with simply adding up all the scores. The score that a particular time period would return could range anywhere from as low as the single digits to as high as the thousands. And this raised a valid concern, what is considered a ”positive” or ”good” score? If the score for a neutral data set came out to be 100, then it is highly probably that a negative data set could come out to a positive number as well. One of the cruxes of using social sensing data for predictions is that it needs to be tailored towards the subject material accurately. Every company is quite unique in not only its online presence, but also the public’s base perception of it. For instance a company that works with animals might have a more positive position in the public’s eye than a company that drills for oil; and another company might not have any public presence at all. This means that a score of 100 on a data set could be extremely positive for one company while entirely the norm for another. To remedy this, We needed to skew the data results in a way that better reflected the specific company that we were working with: in this case Tesla.

What we needed to determine was what a ”neutral” for our given stock translated to in terms of this all-in-one score; this way we could determine a qualitative analysis of the resultant data. Initially this was done by manually finding neutral days in the stock and feeding tweets from that day about the stock into our modified TextBlob. We were able to use the score from this neutral set as a comparative marker and determine whether a positive data set resulted in a higher score. While this method allowed for testing of the program and error checking, it was imperfect and could have been biased based on the neutral period chosen. Thus in the final iteration of the project we implemented a more thorough tool for assigning a neutral score. We created a new script that would have the sole purpose of creating a neutral baseline. The script would use the same tweet gathering a sentiment tools to analyze the entire history of a stock back to when twitter’s popularity first started. We used the yfinance package to gather the data from the stock market. This tool takes advantage of the yahoo stock market API, which we imported into a data structure from a selected time period. The data from this stock market was then split up into individual weeks over the course of time period chosen. Each week was analyzed for its sentiment as well as the sentiment of a couple of days prior. If the stock market saw a growth or loss within the range we determined to be ”neutral”, then the sentiment data for this week was added to a list. To evaluate the growth or decline of a stock, we needed to make sure it wasn’t simply an overall stock market trend. So the stock’s performance was always compared to the performance of the DOW Jones during the same time period. Therefore a positive stock performance would mean that the Tesla stock increased at a percentage greater than that of the DOW Jones and a negative stock performance meant the opposite. After the entire time period was evaluated for its score and the relevant neutral sets added to the list, the final step was to average the neutral weeks together. Whatever this average came out to be would be considered the baseline score for the stock. In the process of gathering this data, we analyzed over 100,000 tweets over the course of Tesla’s history on the stock market. To gather this data, many hours had to be invested in pure data extraction, and is not a realistically repeatable procedure. Thus the data collected was all input into an external file, including the average ”neutral” score, to be used by the main program in the future. This score would be pulled from the file and used as an offset that would be subtracted from any future data collections (per 100 tweets collected). This offset was absolutely essential to our project because it allowed us to compare a time period’s score against this ”neutral” score. We were then able to gather data from whenever we liked and offset it to a score that effectively conveys how much different than normal people are feeling about Tesla. Any data collected would now reduce to a score that was indicative of its meaning: a positive score meant stocks were expected to go up and a negative score meant stocks were expected to go down.

One problem we had to deal with was the method of collection for that massive data set. We initially tried to use the official Twitter API, Tweepy, but found that Tweepy was very limited in its capabilities. For instance, Tweepy only allows collection and analysis of tweets from the past week. We needed the freedom to look at tweets from more specific time periods. There needed to be the possibility to look at tweets that were older than a week in order to get a substantial amount of data for tuning our system. We found a GitHub project by user Jefferson-Henrique called GetOldTweets that allowed us to get tweets from as far back as we needed. This python project uses Twitter’s browser search feature to collect tweets from a set time period [5]. The tool, after searching for the desired keys, travels back to the time period specified in the program, and increments forward tweet by tweet, returning each one into a data structure for further development.

As an aside: for these initial experiments, we chose to focus on the electric car manufacturer Tesla. A few factors went into this decision. For one, Tesla is an up and coming company, growing rapidly since its inception in 2003 and its IPO in 2010, giving us an interesting ten year window of stock movement to analyze. This ten year window also has the benefit of twitter being popular throughout the entire lifespan of the Tesla stock. A side effect of the Tesla’s growth is more and more corresponding buzz about the company in the news and on social media. Tesla is a not only a car company, but also a tech company that specializes in the up and coming electronic car industry. People like to talk about cutting-edge companies like Tesla, making its social network presence quite large. Another factor we considered was the activity of the company’s founder and CEO, Elon Musk, on Twitter. Elon Musk has become somewhat of a celebrity in recent years due to his strong online presence as well as his involvement in a number of innovative companies (Tesla, SpaceX, etc.). Due to his popularity, along with the fact that he is Tesla’s CEO, his opinion and statements have a huge impact on Tesla’s stock performance. On multiple occasions, Musk has tweeted things that have strongly affected its stock price. For example, he once tweeted that he thought Tesla’s stock price was too high, and it quickly dropped in response. We thought that some of this Twitter-based activity would be interesting to gather data on.

It is also important to add that we added a feature to our system that would specifically target tweets that were more popular, either through number of favorites or number of retweets. When the tweets are collected from the twitter search feature by the program, they return with not only the text, but a lot of other data on the tweet as well. Then, before the tweet is added to the our list for sentiment analysis, we check to see if it meets the popularity criteria we selected. We did this in an attempt to capture the tweets that may be more influential/indicative of major changes or news. The data from these tweets was more accurate than the data from tweets that were less popular, however not many tweets met the criteria and the run-time of the program did increase heavily.



V. Evaluation

We found that the baseline sentiment score per 100 tweets for Tesla was 7.859439851693139. This score can be subtracted per 100 tweets from any data set to adjust the results to account for the stock's normal performance. In our gathering of the data for the baseline, we also determined the rate at which Twitter sentiment correlates with Tesla stock price overall. To do this, we took all the data from the historic tweets, which was saved in an external file, and tested whether or not each week had a correlating sentiment and stock score. If a week, after being adjusted to the baseline, had a sentiment and stock score that were both of the same sign (positive, negative, or neutral), then this was considered a success. The total successes was divided by the total weeks for the correlation value. For Tesla, we found a correlation of 41.67%. That means that historically, Tesla stock follows Twitter sentiment 41.67% of the time.

Graphing a comparison of the correlation data would not have been very informative or helpful as the size would have needed to be much larger than is possible/feasible. Instead we gathered tweets every day the stock market was open in January 2020 and compared the sentiment score there to percent change in Tesla's stock. The sentiment score listed is not the same day as the stock market's but is shifted back a day. This is to help us determine if the tweets from one day are predictive of the stock's performance the next day. If it is, we could confidently saw that our system has some value as a stock predictor. This experiment yielded the results that are graphed above.

As you can see, there is definitely some correlation between the changes in sentiment and the changes in stock price. We found that the price seems to be a little more volatile than the sentiment, which means that sentiment doesn't change as drastically as the price tends to. Tesla saw explosive growth in January, so it is possible that this situation, as well as those like it in the past, is a slight outlier and may be causing miscalculations in the baseline or data collections. The correlation is not perfect, but it does have the ability to predict correctly if the factors are correct. For example, from the beginning of the month to about the 9th, there is a very tight correlation between sentiment and stock price. The sentiment perfectly captures a bad day that the company had on the 8th, with nearly identical changes in each metric.

VI. Discussion/Limitation

In this section we will briefly discuss our results and the factors that affected them or limited them, as well as some possible future steps.

As far as our correlation score of 41.67%, we believe that this number would be higher if we took larger sets of tweets from each time slice in the company's history. The collection of historic tweets took a very long time collect, but given more time or processing power, we could gather even more tweets to hopefully increase our correlation or at least make it more accurate. Along with this, the time slices chosen when collecting historic data could have been more specific. Perhaps going by a time slice of smaller than a week, we would get data that more accurately reflected to the continuous fluctuation of the stock market. Another improvement could be to look into the time periods themselves and make sure they represent meaningful data. For instance, the recent Corona virus outbreak has led to a hasty decline in the entire stock market but is not reflective of how the stock market would act in a more "normal" time. A stock's fluctuation could be incorrectly correlated to its company's sentiment when in actuality an external source has caused it to drop or rise. Doing this, however, would be very unique to each company and would not be possible to fully generalize. Overall though, improvements could definitely be made to the baseline tool which would lead to more accurate results in the main program's function as well.

Similarly, a large increase in the quantity of gathered and analyzed tweets would be helpful for the actual implementation of the tool (an example of which is represented by the graph). In the graph, we gathered 50 tweets a day for a month. The program that does this isn't all that fast. Gathering 1000 tweets a day, for example, would probably take nearly an hour per day. This was one of the most limiting factors that we dealt with in the late stages of our project, when we were collecting most of our data. If we wanted to use our tool to actually advise on when to buy or sell stock, we would want something that is a little faster, which leads into one of the steps we could take if we were to continue this project.

If we were to continue this project in the future, we could attempt to optimize Jefferson-Henrique's GetOldTweets program. It is not clear that this can be done in any significant manner, but it would be worth looking into. The program gathers some information that we don't really care about, and maybe cutting out some of the excess fat from the program so that it only does the things we need it to, we could potentially make it faster. Adding to this, we the program could parse the twitter website for criteria more specific than just the date a tweet was tweeted. For instance, our program was often implemented using a minimum favorites requirement per tweet. This helped spread the tweets out over the time period as well as make each tweet more meaningful as it would represent the thoughts of all the people that favorited it and not simply the tweeter themselves. Currently, the program does this parsing after the tweet is already collected into a data structure, and in doing so, wastes a lot of time in an unnecessary data transfer. Instead, perhaps it would be possible to add a favorites criterion when parsing the website itself. This in theory should also speed up the program and allow us to collect more data since we would be able to cutout excess processing time.

Another step we could easily take in the future would be to run full baseline developments for companies in other industries to see if our system works better on certain companies. Any company that has a large public presence would be a good candidate. All we would need for this is a lot of time for data collection, as the code is currently very generalizable.

There are also clearly more factors at play in the public's sentiment of a company that would not show up in a twitter scan for its name. In the instance of Tesla, the CEO, Elon Musk, has a large public presence and plays a huge role in how well the company's stock does. This is something we took somewhat into account, however, there is room for a larger incorporation. There also could have been a change in an entire industry that may not show up in that company's specific mentions. For example, a popular celebrity could die in a car accident, causing the entire automobile industry to take a small dip. These factors are hard to implement due to the shear vastness of everything that has influence on a company.

It would also be interesting to take into account larger, more overarching factors that could be affecting our chosen company to produce an adjusted sentiment score. For example, and mentioned previously, the current pandemic has had an unprecedented influence on the stock market, affecting different companies in different ways, leading to more volatility in a lot of them, while increasing the value of companies conducive to widespread stay-at-home orders. It would be interesting to see if the program could factor in the current/past volatility of the market to either produce more accurate results or at least provide a disclaimer of possible discrepancies. There are plenty of things like this we could explore that could make our system more effective as a stock predictor.

One last limitation of the program is in twitter itself. While Twitter produces millions of tweets a day, the percentage of the population who are active on the platform is actually quite small. This leads to the very real possibility that the sentiment being gathered is not fully representative of the country, and thus can result in incorrect predictions. While there is no perfect remedy to this, it could be possible to gather from other social networking sites to diversify the data and hopefully gather sentiment from different demographics.

VII. Conclusion

We set out to determine if we could use social media to predict the movements of the stock market, hoping that the similarities between the two mediums would lead to a correlation. We found that there was in fact a correlation between tweets and stock performance of our primary chosen company, Tesla. We had hoped that the discovered correlation would have been very strong, but our results revealed that our initial implementation had many flaws. As stated previously, we recognize that there are many limitations in the current design of our project, and we fully expect that a further developed version of our project could produce a correlation even stronger. The implications the link between stocks and social media could shape the way that investors trade stocks in the very near future (and already is starting to, as we talked about in our section Related Work). We believe that there is a tremendous amount of value to be gained from further research in this particular area and in all of the outside-of-the-box ways that social media data can be applied to other aspects of our lives. As the world continues to embrace the new age of technology and online social connections, we fully expect social networks to continue to be utilized as a source of data and study.

Acknowledgment

The authors would like to thank Dr. Dong Wang for the excellent guidance and suggestions he provided for our group, as well as the Teaching Assistants for Social Sensing and Cyber-Physical Systems. We would also like to thank GitHub user Jefferson-Henrique, the creators of the TextBlob software, and the developers behind the yfincance tool for their contributions to our project.

VIII. Links

Our GitHub repository link is: https://github.com/shealy3/SocialSensingProject

References

[1] Keller, Jared. *"Predicting Stock Market Changes Using Twitter."* The Atlantic, Atlantic Media Company, 17 July 2013, www.theatlantic.com/technology/archive/2010/10/predicting-stock-market-changes-using-twitter/64897/.

[2] Bollen, Johan, Huina Mao, and Xiaojun Zeng. "Twitter mood predicts the stock market." Journal of Computational Science 2.1 (2011): 1-8.

[3] Team, Editorial.*"Researchers Develop Algo That Uses Twitter to Predict the Stock Market."* Finextra Research, Finextra, 24 Jan. 2018, www.finextra.com/newsarticle/31564/researchers-develop-algo-that-uses-twitter-to-predict-the-stock-market.

[4] Intellica.AI. “VADER, IBM Watson or TextBlob: Which Is Better for Unsupervised Sentiment Analysis?” Medium, Medium, 19 Sept. 2019, medium.com/@Intellica.AI/vader-ibm-watson-or-textblob-which-is-better-for-unsupervised-sentiment-analysis-db4143a39445.

[5] Henrique, Jefferson. ”GetOldTweets-python” Jefferson-Henrique, GitHub, Nov. 21, 2018, https://github.com/Jefferson-Henrique/GetOldTweets-python.