Twitter-Based Stock Sentiment

Final Project Report

Vidhey Oza

Abstract

Sentiment analysis using natural language processing is helpful in understanding the intentions behind written texts. Stock markets are highly sensitive and volatile to market sentiments, and hence understanding this sentiment is crucial to understanding its future trends. Social media platforms like Twitter are one of the best current sources of text where such sentiment analysis can be applied to understand sentiment about a particular stock. This project showcases a simple stock sentiment predictor that extracts sentiment from the most recent tweets using the Twitter APIs related to the given stock and puts it beside a basic mathematical stock price predictor to see how strong these sentiments are with respect to conventional prediction models. There is great value for such models that can accurately predict sentiment one or more days before the price follows that sentiment.

Introduction

Prediction in stock markets is a well-explored domain where mathematicians, computer scientists and data scientists have all designed various algorithms that give an edge on how the stock will perform in the future. But the stock market follows the future and not the past, making these prediction models incapable of capturing the true essence of the current stock market.

Recently, a novel approach of sentiment analysis using language models has entered this domain, kickstarted by authors of [1]. The premise is that the stock market is highly sensitive to the market sentiment about the particular stock, and this market sentiment is best captured by taking the voluminous data available on social media platforms and analysing the positivity or negativity of each data point to make a more educated guess. And Twitter is one of the most popular social media platforms, that enjoys the attention of market pundits and individual stock investors alike.

With these foundations in place, the goal of this project was to design a stock sentiment predictor that collects recent tweets from Twitter on the fly and extract general sentiment about the particular stock being discussed. That combined with the high volume of posts about that stock, the end goal was to predict the market sentiment, and see whether that is

an indication to how the stock price changes. I have developed a simple stock sentiment predictor that uses the Twitter APIs to extract recent tweets about any stocks from NASDAQ, and predict the general sentiment of that stock.

Dataset

The datasets used to develop this predictor came from 2 sources: the tweets were extracted by the Twitter APIs, and stock information and price points were taken from Yahoo Finance.

Twitter APIs [2] can extract different data and metadata about tweets and trends from their platform. The student/developer edition includes a read-only version of the APIs, which was sufficient for the task. The APIs go through a two-step verification for secure access to their vast collection of their social media platform data, where each step uses a public and a private key for two-way verification.

The first step is an account level key-set, where Twitter communicates with the developer and vice-versa via a Twitter account. Every API call that takes goes through that Twitter account. The second step is a deeper app level key-set, where each app deployed by the developer has a unique identifier between Twitter and the developer, and is used for convenience for both parties, and is also an extra set of locks to track and prevent malicious use.

Using the APIs themselves is straight-forward and involves just a few lines of code in Python after authentication is established. The API used in the project performs a simple search query on the Twitter database for a given string and extracts the most recent tweets containing that string. Since the student version of the API allows a fairly limited number of queries per month, only about 100-200 tweets were extracted while developing, but more than 1000 tweets were extracted for testing.

Stock prices were extracted using an open source library created by [3] as a developer-friendly solution to extract stock price data from Yahoo Finance [4]. This data was used to create a base model that would be used to gauge the stock trends up to 2 years in the past and understand its history by creating a simple linear regression model that predicts stock prices up to 2 months into the future. I performed experiments on 5 different stocks: Apple, Google, Tesla Motors, American Airlines and General Electric. This choice was in line with expectations on which stocks would be the most discussed on

Twitter so that enough data points could be gathered for all of them across multiple days, and also stocks that might have varying sentiment in the last 2 years.

This dataset was used to create a contrast between the usefulness of mathematical models used for stock price prediction and the intuitiveness of language models used for sentiment analysis.

Methods & Experiments

For the base price prediction model, a simple linear regression model was used that was trained to take features like previous closing price, volume, high and low as input, and use them to predict the next-day closing price of the stock. This model was chosen not to be the best price predictor but to act as a comparison on what general direction the price would follow in the coming days and weeks.

For the sentiment analysis model, the framework provided by TextBlob was used so that experimentation across different vocabularies and analyzers could be performed without significantly changing the implementation for each experiment. I tried vocabulary models from NLTK like PUNKT, Word2Vec etc. so as to understand which of them closely resembled text from tweets.

TextBlob provides a basic Blob class structure that uses different models to easily create language model systems. I experimented on some of these models to gain an advantage on creating a more fitting sentiment analysis model. Models like tokenizers could not be altered since a sentence tokenizer would not be helpful for text from tweets, but the analyzer and parser models could be tested and altered as per requirement. For analyzers, the 2 best performing models were PatternAnalyzer and NaiveBayesAnalyzer with each model showing promise in different stocks but not all of them.

With this framework in place, I performed experiments by trying different string queries on the Twitter APIs. One of the simplest ways was to go with the full company name, but many companies have informal nicknames like AmAir for American Airlines or Tesla for Tesla Motors Inc. Another way was to use the ticker symbol, but this was a controversial choice for its own reasons: while market pundits would prefer using the ticker symbol as a sign of professionalism, common investors (and most people in general) would rarely use them in their tweets.

Using a zero threshold on sentiment score to classify a positive or negative tweet did not yield the best results, so I moved on to using a slightly positive threshold. This was mainly because the tweets were rarely highly negative about any of the chosen stocks but were moderately negative or neutral a lot, which was sometimes indicative of a future price drop in a lot of cases since it meant the companies were not performing as per expectations but because of the high reputation of the companies people were wary of being exceedingly negative.

With this sentiment model in place, it is important to ascertain the value of the domain expertise of stock markets, and understanding that while in general a negative market sentiment precedes a downward price trend and upward for positive sentiment, that was not always the case. When the scores were closer to zero, a low positive score also showed weakening confidence leading to increasing speculation in the markets about a possible downward trend, inadvertently creating that trend themselves.

Results & Discussions

I observed varying results for the different models tried for sentiment analysis, and it was especially difficult to quantify the output accuracy of the sentiment analysis models, since high domain expertise is required to understand what positive, negative and neutral market sentiments mean for the stock price itself.

For all 5 stocks tested, I first plotted the output of the linear regression model with the past values to understand the past trends and the future possibilities. Plots in Fig. 1 show the price of Google and American Airlines for the past 2 years and how they may perform according to the model for the next 2 months.

Now, for the sentiment analysis results, we followed the PatternAnalyzer model for getting the sentiment score on any input string (the entire tweet text in this case). This model gave the most consistent results across different time periods. NaiveBayesAnalyzer gave better sentiment scores on some tweets by smartly analysing the context of the tweet, but sometimes the simpler tweets were given overly neutral scores, making it a toss up for any threshold whether the tweet would be considered positive or negative.

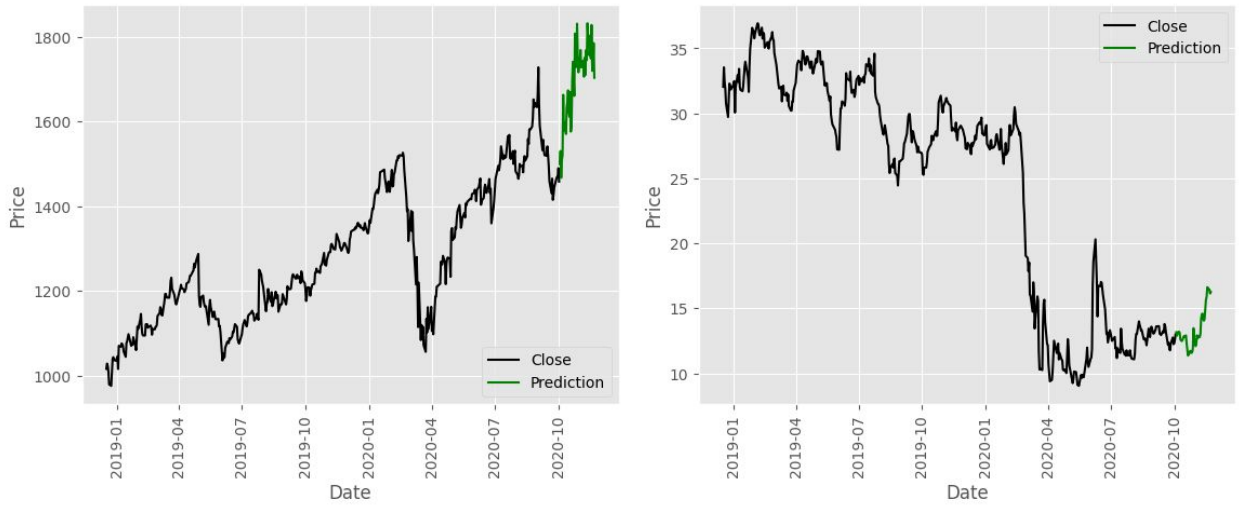

Fig 1: Stock prices from beginning of 2019 to end of 2020. Left – GOOG: Fairly stable stock price with a March-April 2020 dip due to the COVID sentiment in the stock market. Right – AAL: One of the most popular airlines with a March 2020 dip that it hasn't yet recovered from due to the uncertainty in domestic and international travel.

Some of the examples of tweets and their scores are given below:

- `A coalition of U.S. states was expected to file an antitrust lawsuit against Alphabet Inc's Google alleging that it altered the designs of its search engine to the disadvantage of rivals that offer specialized search results - Politico`
    - `PatternAnalyzer sentiment score: -0.100`
    - `NaiveBayesAnalyzer sentiment score: -0.063`
- `Alphabet Inc. (GOOG) surprised the market with Q2 result. Bain capital changed the rating to Buy, as Alphabet Inc. (GOOG) has great fundamentals and its currently trading at a PE of 0.97 with a PEG of 0.48.`
    - `PatternAnalyzer sentiment score: 0.505`
    - `NaiveBayesAnalyzer sentiment score: 0.561`
- `American Airlines Group Inc. (NASDAQ:AAL) Expected to Announce Quarterly Sales of $4.09 Billion`
    - `PatternAnalyzer sentiment score: -0.050`
    - `NaiveBayesAnalyzer sentiment score: 0.056`

In examples like above, there are tweets whose mild negativity is overpowered by high positivity, giving the stock an overall low positivity. Also, many tweets are generally neutral, hence bringing the average sentiment very close to zero. This is the reason why I chose to try out different thresholds to classify a negative from a positive tweet.

Due to the difference in training methodology for the PatternAnalyzer and NaiveBayesAnalyzer, not only do they predict slightly different scores but they also need

their own thresholds in order for them to more consistently classify the tweet sentiment. PatternAnalyzer was consistently better than NaiveBayesAnalyzer at predicting tweets with more complex texts and was hence the model of choice for final predictions.

Conclusion & Future Work

Sentiment analysis is a useful tool in understanding the intentions behind written texts, and when used along with the existing prediction models helps in gauging the credibility and intensity of those predictions. Stock markets are highly sensitive and volatile to market sentiments about any particular stocks, and one of the prime examples of that is the Tesla stock that fluctuates as high as 5% every week, which is exceedingly high for such a large corporation. This points to a larger aspect of incorporating domain knowledge in language models so as to put the predictions in context to forecast the trends of the stock markets around the world.

For future work, there can be some improvements in methodology and more extensive experiments can be performed. Since it is tricky to use the full company name or the ticker symbol for querying the Twitter database, perhaps using a combination of those approaches or using common names of those companies may help in better data extraction. Also, giving preference to stock market jargon in the model vocabulary and applying domain knowledge may aid in better understanding of a tweet with more technical keywords that are not used commonly.

References

1. *Bollen, Johan, Huina Mao, and Xiaojun Zeng. "Twitter mood predicts the stock market." Journal of computational science 2.1 (2011): 1-8.*
2. *Twitter API Documentation:* https://developer.twitter.com/en/docs/twitter-api
3. *YFinance, a Python library by ranaroussi:* https://github.com/ranaroussi/yfinance
4. *Yahoo! Finance:* https://finance.yahoo.com
5. *TextBlob, a language model library:* https://textblob.readthedocs.io/