Stock prediction using Twitter

Aishwarya Anand (A20331867), Vivek Vijaykumar Bajpai (A20361204)

Illinois institute of technology

1. Introduction

Over the past 5 years, significant research is going on for predicting the stock market. But still the question remains same. “can we predict stock market”? Stock market is driven by multitudes of dynamics, like news, feelings, beliefs. Although, in today’s age there is a good possibility of predicting the stock market through fastest growing social networking tools on the Internet, namely Twitter. In a related work done by Johan Bollen1, Huina Mao1 and Xiao-Jun Zeng in 2010, using Twitter mood, they predicted the opening and closing direction of DJI with an accuracy of 87.6%, but the same system displayed an accuracy of mere 3.6% for another experiment over 20 days.

Our approach and statement is slightly different. Instead of predicting stocks for an Index (group of companies), we are predicting next day closing value for a specific company, like AAPL, MSFT etc. Our work seeks to explore on underlying relationship between sentiment of a company and its stock values. We aim to create a system that provides investors real time feedback about sentiment towards a company and effects on its stocks value. We have taken Apple Inc. for the case study.

2. Data

We are collecting data from Twitter API we have 34 keywords specific to Apple Inc. , for which, we are creating a search request and capture all the tweets returned. For a single keywords, on an average, we captured 3 MB of tweets per day. Instead of storing data in raw json format we stored the data in csv file for easy data lookup and search. We stored creation timestamp, screen_name, name, text, description, language etc. of each tweets that we received from twitter API . Overall we captured more than 1.5 GB of raw data. We found that there were lots of repeated tweets in our raw data and hence decided to write a filter to remove the repeated tweets. Logic for filtering is simple. First we check for language of tweets. If the language of the tweet is English then we add the tweet in a list named filtered tweets, if that tweet text is not present in our filtered tweets list. On Filtering we were able to get 1.5 K unique tweets for each day. Overall we got more than 17K of filtered tweets.

In our system, we have written two classifiers namely Advertisement classifier and Sentiment classifier , which requires labeled data. We have labeled more than 1 K of tweets for both advertisement and sentiment classifier. All data is present in a structured way in “Data” folder of project directory. “Data” directory is furthered divided in four directory namely “rawdata”, “filterdata”, “labeldata” and “noadvertismentdata”. “rawdata” directory contains csv files downloaded from twitter, “filterdata” directory contains all the filtered data after running filter on csv data, “labeldata” directory contains labeldata used by the classifier and ‘noadvertismentdata’ directory contains data that is processed through filtering and by advertisement classifier.

In a bigger picture, our data look like as follows:

Total Dataset:

Category	Number of Tweets
Filtered Tweets	17 K
Tweets without advertisement	9 K

Labeled Dataset

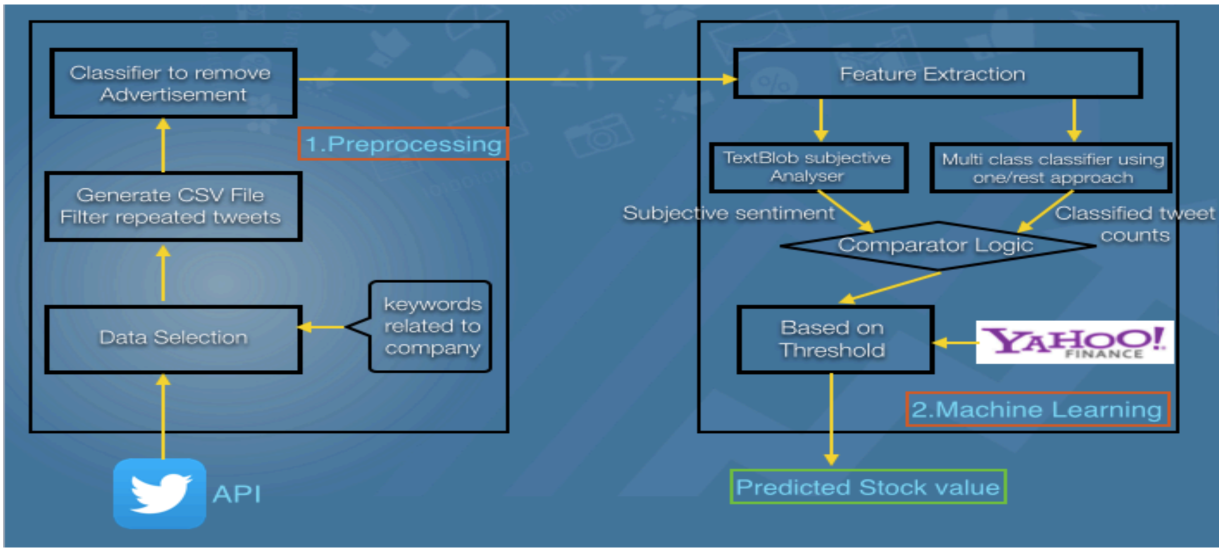
Labeled For	Number of Tweets
Advertisement classifier	700
Sentiment Classifier	800

We gathered the stock data using yahoo finance API. “apple.csv” file present in ‘data’ directory contains stock data and it contains all the necessary fields like opening value and closing value for each day.

3. Methods

We have tried and tested various methods and our final system architecture looks as follows.

`



Note: We have removed Granger causality test from the system architecture, as the result of granger causality test was good for regular data (one series exactly following the other) but when data becomes irregular, as in our case, the prediction was deviating with a high margin.

As we can see from the above diagram our system is divided in two parts

1. Preprocessing: Removes noise from data
2. Machine Learning: Sentiment analysis and Logic to predict closing value and direction of stock

Preprocessing is divided into three steps

a. Data collection: As mentioned in data section of this report we capture data from twitter API on 34 keywords.
b. Generating CSV file and Filtering : This part of system is also mentioned in data section of this report, in simple terms we store tweets in csv file and filter out the repeated tweets.
c. Advertisement Classifier: On looking at the filtered data we found that there were lots of tweets that are not related to the company's sentiments as they were advertisements Hence we decided to write a classifier that can filter out advertisement form our data set. For this we labeled tweets that are used by the classifier.

Machine Learning is compromised of various blocks

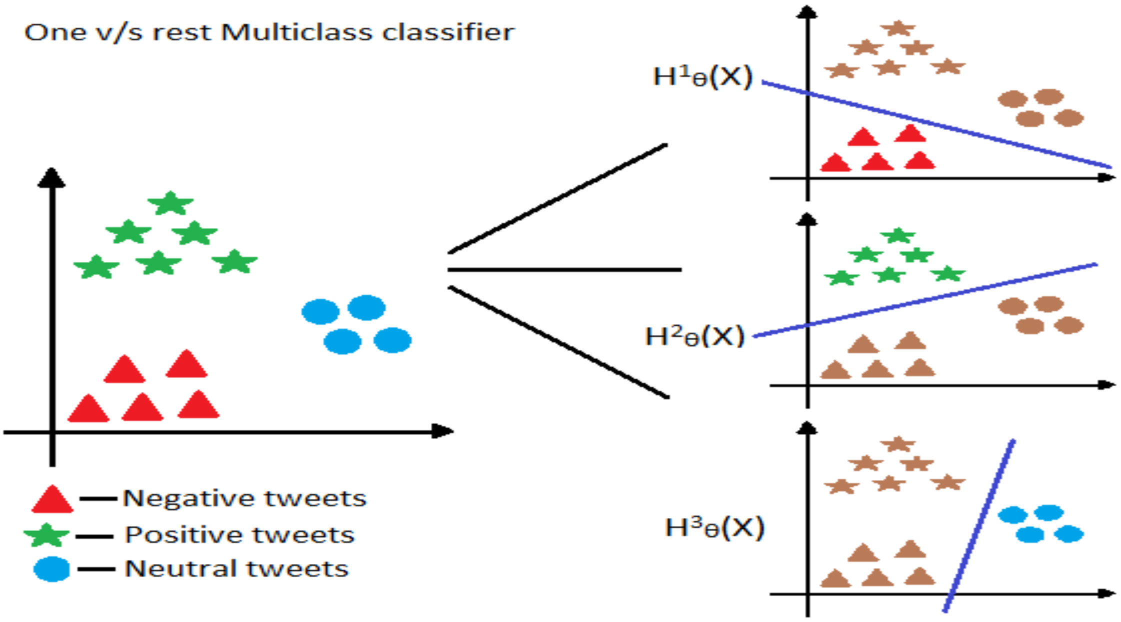
a. **Feature extraction:** In this part, we vectorize the tweets and make a feature matrix from them. A simple word tokenizer was selected, as it was giving the maximum accuracy during cross validation (accuracy =76%). Vectorization removes the words with document frequency less than 2. It uses n-gram of (1,1) as even though increasing the ngram was increasing the accuracy on training data, the accuracy on unknown data was getting reduced.
b. **Multiclass classifier using One vs Rest approach**: Predicting sentiment with accuracy is a regression problem, and we needed to divide the tweets into multiple classes. For this purpose, we have implemented a multiclass classifier using One v/s Rest approach. We tried to classify tweets into Positive, Negative and Neutral classes. The classifier works as follows.
   The multiclass classification (3 class) problem is divided into three separate binary classification problems. The first classifier tries to separate the tweets belonging to negative class, from the rest of all the tweets. The same approach is followed for the positive as well as neutral classes. Each classifier is governed by the following function

   $$H^{i}_{\theta}(x) = P(y=i|x;\theta) \qquad \text{where } i=(-1,0,1)$$

   We fit these binary classifiers with the training data, dividing the training data in 2 classes. While predicting we choose the class for a tweet, considering that the tweet actually belongs to the class for which, it has the maximum probability. It can be written mathematically as follows:

   $$\mathrm{Max}_i\, H^{i}_{\theta}(x) = \mathrm{Max}_i\, P(y=i|x;\theta)$$

   The following diagram shows it clearly:



c. **Text blob subjective classifier:** We wanted to use a third party classifier to keep a check on the aforementioned multiclass classifier. Decision for using Text Blob library was not easy. We tried various third party classifiers to do sentiment analysis. But all classifier were different from one another in output or in working. For example Opinion finder: Worked well for multiple line document, but not for individual tweets. We also tried Sentinet, but the output was not up to the expectation. We were hardly able to classify negative tweets. In Textblob, we tried the Naive Bayes Analyzer and the Pattern Analyzer .We choose Pattern Analyzer because results were more accurate than Naive Bayes Analyzer . Also, Naive Bayes Analyzer with text blob was not very efficient, considering the amount of data we had. Textblob gives polarity and subjectivity of a text, which provided a clear representation of the sentiment of each tweet.

Comparator Logic: As we have two sentiment classifier, each giving its own output and for prediction we need a single value that tells the sentiment of the day as a single value. For this purpose, we took average of sentiments given by both the classifiers and to calculated single sentiment value. We chose following approach.

1: For all days take mean for positive, negative, neutral sentiment values. Then we will get mean of positive, negative and neutral sentiment of all days

2: Now for each day subtract the mean with sentiment as below to get the positive, negative and neutral variation.

For Positive variation = Positive Mean of all days - today Positive sentiment

3: After summing up all the values of positive, negative and neutral variation we will get the variation of sentiment

d. **Prediction based on Threshold**: As mentioned above, we tried Granger causality for series prediction, but unfortunately we were not getting best result from that. So we tried another approach in which we a threshold based algorithm to predict the next closing value. We calculated the max and min of both sentiment of day and for difference between opening and closing value of stock value. Then we calculated distance between max and min for both sentiment and stock value difference and divided the distance of max and min for stock difference with the distance of max and min for sentiment that we called threshold value. We predict the next day value of stock by multiplying threshold value with sentiment of next day.

4. Experiment

Based on our experiment we got the following result

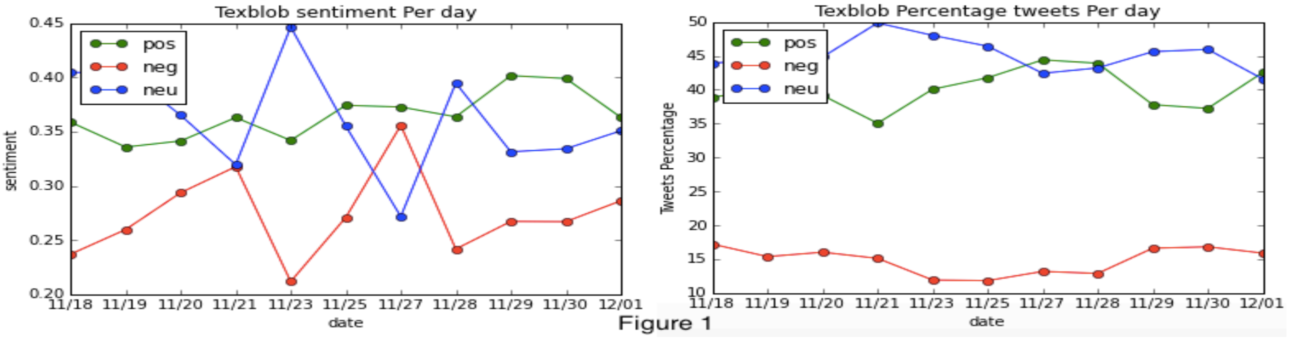


Figure 1

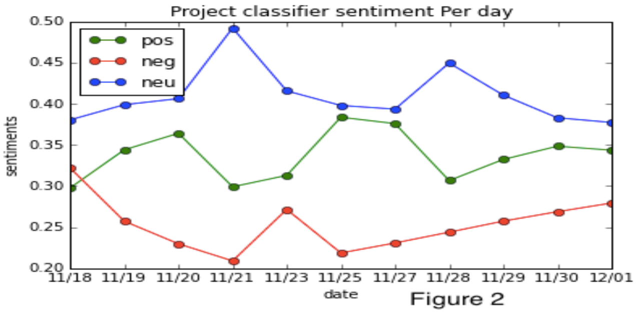


Figure 2

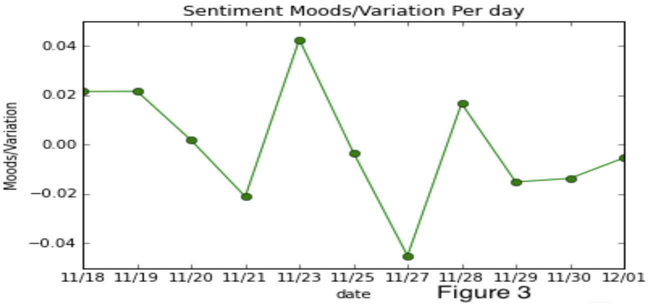


Figure 3

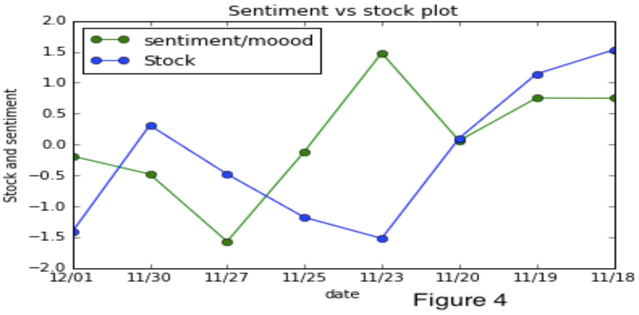


Figure 4

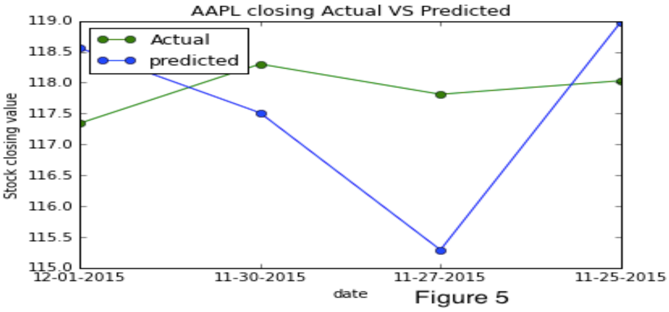


Figure 5

Figure 1: Represent overall sentiment for each day according to TextBlob
Figure 2: Represent sentiment for each day generated by our classifier
As we can see form the result of sentiment there are more positive sentiment than negative
Figure 3: Gives the combined result of sentiment for each day based on sentiment classifier and comparator logic
Figure 4: shows the direction of sentiment v/s stock direction for each day
As we can see **we got 6 days matching direction between stock and sentiment**
Figure 5: shows our predicted value in comparison for actual AAPL closing value for 4 days
Accurate Prediction is still a challenge.

5. Related work

As we know predicting stock market is still a challenge for economist and researchers. There had been various attempt made to predict stock market as discussed in following section:

1. **Johan Bollen and team:** It is one of the most famous paper on the topic which uses ‘OpinionFinder’ and ‘Google Profile of Mood States’ for analyzing the twitter sentiment. GPOMS provides 6 mood dimensions for any text, and they hypothesized that different moods will have different type of effect on the stock value. They also used an ANN based algorithm which improved the prediction with time.
2. **Tien Thanh Vu and team:** This paper discusses the approach that induces the lexicon automatically by association with “bullish” (a positive price outlook) and “bearish”(a negative price outlook) anchor words on the Web.
3. **Anshul Mittal and team**: This paper discusses the similar approach as done by johan Bollean paper but instead of using 6 moods they used four moods Twitter data to find the public mood and the degree of membership into 4 classes - Calm, Happy, Alert and Kind (somewhat like fuzzy membership).

Unlike other research we are predicting stock market at company level instead of Index level, which is comparatively difficult to do. The focus of most of other papers is on predicting the direction of stock value but in out system we are trying to deduce the stock value as well. Another special feature of our system is the use of advertisement classifier, which is not present in other similar systems.

6. Conclusion and Future Work

Having a limited dataset we are able to predict direction of Apple inc. stock for 6 days out of 8 days with 75 % accuracy as the problem is very hard and it will be immature statement to make. But we can deficiently conclude that there is strong correlation between social media sentiment with the stocks value although we are unable to predict the actual closing stock value of Apple Inc. However we got advertisement classifier accuracy with 76% and sentiment classifier accuracy of 70% but there are large number of parameters on which stock value depends on and we are considering only few of them. One more factor that we noticed is the sentiment of investors and consumers could be different for the same news. For example: Tweets saying “All iPhones booked for next two months”.
In future we would like to include Klout score (importance of user) provided by user, so that we can give weighted to a specific tweets as well as it would be a great add-on to add GPMOS in our system.

7. References

Bollen, Johan, Huina Mao, and Xiao-Jun Zeng. "Twitter mood predicts the stock market." Journal of Computational Science
Tien Thanh Vu1,3 Shu Chang2,3 Quang Thuy Ha1 Nigel Collier3 “An Experiment in Integrating Sentiment Features for Tech Stock Prediction in Twitter”
Anshul Mittal, Arpit Goel “Stock Prediction Using Twitter Sentiment Analysis