Stock Market Prediction based on Social Sentiments using Machine Learning

Tejas Mankar[#], Tushar Hotchandani[#], Manish Madhwani[#], Akshay Chidrawar[#], Lifna C.S[$]
[#] BE Students of Department of Computer Engineering
[$] Assistant Professor, Department of Computer Engineering
VES Institute of Technology, Chembur, Mumbai, India

***Abstract*—Machine learning and artificial intelligence techniques are being used in conjunction with data mining to solve a plethora of real world problems. These techniques have proven to be highly effective, yielding maximum accuracy with minimal monetary investment and also saving huge amounts of time.To add to their annual income, nowadays, people have started looking at stock investments as a lucrative option. With expert guidance and intelligent planning, we can almost double our annual revenue through stock returns. That said, stock investment still remains a risky proposition for the uninitiated. Exorbitant wages of the investment experts coupled with a general ignorance pertaining to the financial matters among the public, deters many from trading in stocks. The fear of losses also acts as a deterrent to many. These facts propelled us to harness the power of machine learning to predict the movement of stocks. Using sentiment analysis on the tweets collected using the Twitter API and also the closing values of various stocks, we seek to build a system that forecasts the stock price movement of various companies. Such a prediction would greatly help a potential stock investor in taking informed decisions which would directly contribute to his profits.**

Keywords—sentiment analysis, stock market prediction, Twitter API, machine learning.

I. INTRODUCTION

Nowadays, social media has become a mirror that reflects people's thoughts and opinions to any particular event or news. Any positive or negative sentiment of public related to a particular company can have a ripple effect on its stock prices. We seek to predict the stock market prices of various companies by performing sentiment analysis of the social media data such as tweets related to the respective companies. The paper explains how Twitter will help you become a better investor by making appropriate investment decisions with its knowledge of the market sentiment. Tweets collected using the Twitter API would correspond to diverse topics. The main problem in data processing would be to sift through these tweets and filter the ones relevant to us i.e. the tweets related to the companies whose stock movements we are interested in predicting.

First, we will collect the tweets and perform sentiment analysis of it. Corresponding to that time period, we shall analyze the stock values from past data and use a suitable machine learning algorithm to justify a valid correlation between the tweet sentiment and the stock values. Finally, with this training data, we will train our model and develop capability to make stock predictions for future, provided, the tweets are provided. Since the public reactions to any major event are available almost instantaneously on any social media, their mood can be captured quickly and an estimate of the volatility in stock prices can be determined, thus providing an almost real time forecast similar to some weather forecasting models.

II. RELEVANCE OF THE PROJECT

This project is quite relevant as it guides people who possess limited know-how of investments and finance into making well informed decisions regarding stock market investments.It bypasses the need for hiring investment experts who command exorbitant wages to guide our financial decisions by providing a simple solution which can be accessed by anyone having a computer or a laptop and an internet connection. Stock market trends for a given time frame can be analysed easily even by the uninformed. Popularizing this machine learning option provides cheap alternative to various stock market investment guidance agencies which are in vogue today. The project puts in a small effort to assist the inexperienced investors and prevent from suffering heavy capital loss.

III. LITERATURE SURVEY

A thorough literature survey was performed to get a better understanding of the topic, analyze the previous models developed, note their advantages and drawbacks, and highlight the necessary developments. The survey conducted is summarized in Appendix 1. The methodology adopted is discussed in detail in the following section.

IV. PROPOSED METHODOLOGY

The proposed methodology can be summarized in the following modules :

A. Data Collection:

For tweet collection, Twitter provides robust API. There are two possible ways to gather tweets: streaming API and search API. To overcome the limitation of streaming API, we have used search API. The search API is REST API which allow users to request specific query of recent tweets.The search API allow more fine tuning queries filtering based on time, region, language etc. The request of JSON object contains the tweet and their metadata. It includes variety of information including username, time, location, retweets. We have focused on time and tweet text for further analysis

purpose.An API requires the user have an API key authentication. After authenticating using key, we were able to access through python library called Tweepy. The text of each tweet contains too much extraneous words that do not consider to its sentiment. Tweets include URLs, tags to others and many other symbols that do not have any sentiment value. To accurately obtain tweet's sentiment we need to filter noise from its original state.

First step is to split the text by space, forming a list of individual words per text which is called as list of words. We will use each word in tweet as feature to train our classifier.
Next, we remove stop words from list of total words. We have used python's Natural Language Toolkit (NLT) library to remove stopwords.Stopwords contains articles, punctuation and some extra words which do not have any sentiment value. There is a stop word dictionary which check each word in list of tokenized words against dictionary. If the word is stop word then it is filtered out.

Now, tweet contains extra symbols like "@", "#" and URLs. The word next to "@" symbol is always a username which does not add any sentiment value to text, but it is necessary to identify the subject of the tweet. Words following "#" are kept as they contain information about tweet. URLs are filtered out as they do not add any sentiment meaning to text. To accomplish all these processes, we use regular expressions that matches these symbols.This all forms the necessary tweet corpus.

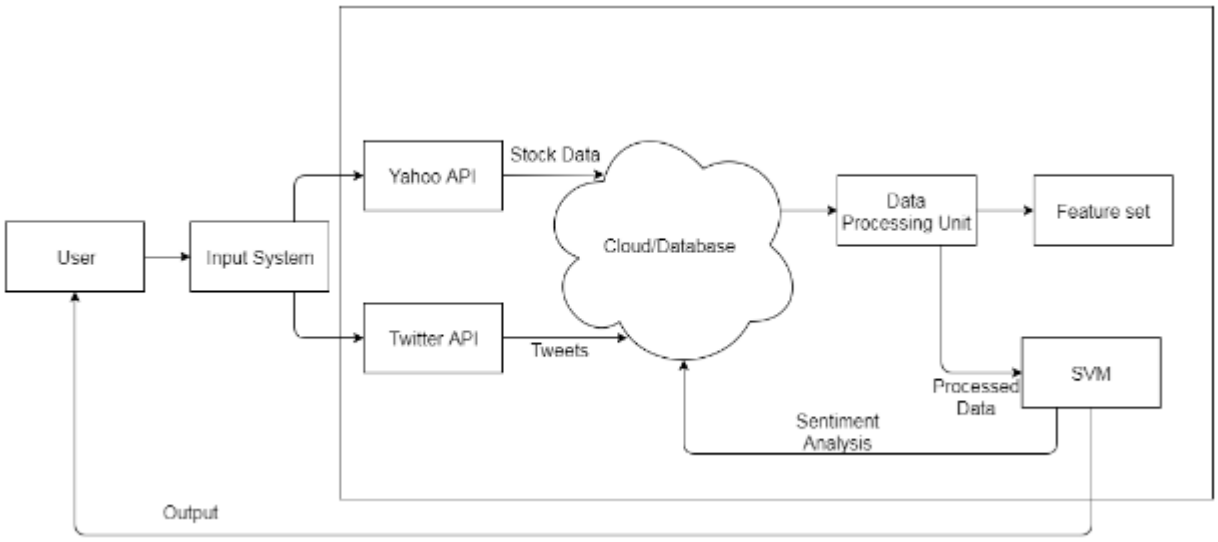

Figure 1. Proposed System Diagram

B. Feature Extraction Module:

After gathering large tweet corpus, we have built and train classifier for tweet sentiment analysis. We examine mainly two classifiers: Naïve Bayes and Support Vector Machine.For each classifier we extract the same features from the tweets to classify on it.To build feature set, we process each tweet and extract meaningful feature and create feature matrix by unigram technique.

For example, if positive tweet contains word "sorrow", a feature for classification would be whether or not a tweet contains the word "sorrow".

As explained the method above, the feature set grow larger and larger as dataset increases. After certain point, it becomes difficult to handle larger dataset. In this case it is not necessary to use every unigram as feature vector to train Naïve Bayes classifier and Support Vector machine. To avoid critical situation, we decided to use 'n' mostly significant feature for training.We have determined the n best features from larger set using chi-squared test. It scores each word of training data and separate n best feature to classify model. For the ease of implementation, we have used Python's Natural Language Toolkit (NLT) which allow us to calculate with conditional frequency and frequency of each feature.

After calculating feature score, we rank the feature in order of score and choose top n feature for training and classification. A feature reduction helps to improve speed of classification. We find tweet sentiment value using Python's AFINN library for training datasets labels. For example, the processed tweet is "#Peter lost the election as he was being charged with corruption".In this example, we have two negative keywords which results in the whole tweet sentiment being negative.

C. Training Module:

The generated data is used as training dataset to train the model for sentiment analysis. On inspecting the model on test dataset, we receive the tweet sentiment labels as an output. We will use this dataset for stock market prediction. We calculate total available stock tweets regarding each company and generate another dataset which contains positive, negative, neutral as well as total tweets of each day as a feature matrix. On other side we have taken stock market historical data for each day and have calculated market up as well as down direction and took it as label for dataset. In case of stock market historical data, we have used Python's yahoo-finance library.

D. Prediction Module:

After training our classifier, we move on to an application to look at correlation between tweet sentiment and stock market prices on each day scale. To do so, we have collected stock data as well as tweet data for same timeline as explained above. In addition, we focus on specific company stocks gathered daily data for each. After justifying a valid correlation, we are able to predict the stock values.

V. Challenges

The following challenges needs to be addressed: (1) Historical Twitter data cannot be obtained, unless it is saved by someone, so data has to be collected over a duration of a fixed number of months starting from the present date and time; (2) It is necessary to filter out required data from the stream of unrelated tweets; and (3) Authentication is required for accessing real time Twitter data.

VI. Conclusions

Based on the comparative study that we performed, Support Vector Machine proved to be the most efficient and feasible model in predicting the stock price movement, in favour of the sentiments of the tweets. Using Machine Learning techniques for prediction purposes is inexpensive compared to the ground survey that would have been conducted otherwise to gauge the public mood.

Cloud services will enable us to collect large amount of data and also store it in real time when we will get the data directly from the REST API. Classification of tweets as positive, negative and neutral gives a good overview of public mood. Other factors that are affecting the public mood will be subsequently studied and incorporated in the system.

VII. References

[1] Oliveira, Nuno, Paulo Cortez, and Nelson Areal. "The impact of microblogging data for stock market prediction: Using Twitter to predict returns, volatility, trading volume and survey sentiment indices." *Expert Systems with Applications* 73 (2017): 125-144.

[2] Bohn, Tanner A. "Improving Long Term Stock Market Prediction with Text Analysis." (2017).

[3] Li, Xiaodong, et al. "Empirical analysis: stock market prediction via extreme learning machine." *Neural Computing and Applications* 27.1 (2016): 67-78.

[4] Sorto, Max, Cheryl Aasheim, and Hayden Wimmer. "Feeling The Stock Market: A Study in the Prediction of Financial Markets Based on News Sentiment." (2017).

Appendix 1. Comparative Study on Methodologies studied

Paper	Pros	Cons
[1]	GPOMS gives a precise indication of public mood relative to 6 dimensions. Daily up and down stock price closing value changes is projected with an accuracy of 87.6%. Sentiment analysis is inexpensive as compared to ground surveys.	Inclusion of all the six mood dimensions in forecasting decreases the accuracy of prediction. Insufficient knowledge of 'ground truth' for the various mood states.
[2]	SVM gives better accuracy, around 64.10%.	Speed is an issue in prediction in both methods. Small dataset and less training period may have hampered the overall accuracy.
[3]	K-ELM has best accuracy and is the fastest technique.	K-ELM requires very high CPU specifications. But, cannot process parallel stock predictions.
[4]	Experiment was conducted and results were derived, individually for headlines and the summary of news articles. Dividend yields and price- earnings ratios are analyzed to find patterns that indicate who to invest .	Results prove that the stock market cannot be accurately predicted. Results of test runs do not show strong prediction capabilities; the results are weak, which further support the Efficient Market Hypothesis.