Stock Market Prediction using machine learning and linear regression

[1]Piyush R. Kapse, [2]Swapnil K. Meshram, [3]Prafullakumar Jha, [4]Ketki Bhandarkar,[5] Prof. A.D.Bhange
[1]Student, [2]Student,[3]Student, [4]Student, [5]Assistant Professor

[1]Computer Technology, [2]Computer Technology, [3]Computer Technology, [4]Computer Technology, [5]Computer Technology
[1]K.D.K. College of Engineering, Nagpur, India, [2]K.D.K. College of Engineering, Nagpur, India, [3]K.D.K. College of Engineering, Nagpur, India, [4]K.D.K. College of Engineering, Nagpur, India, [5]K.D.K. College of Engineering, Nagpur, India

Abstract- *The aim of the paper is to examine a number of different forecasting techniques to predict future stock returns based on past returns and numerical news indicators to construct a portfolio of multiple stocks in order to diversifythe risk. We do this by applying supervised learning methods for stock price forecasting by interpreting the seemingly chaotic market data.*
The paper being build is being built to predict the future stock prizes for the investor to predict the future stocks, and this is done by the usage of the social medium platform by the use Twitter API. This is being used to calculate the sentiment over the market places and accordingly could be used to predict the prices. The concept being applied could be achieved by using the platform such as python for programing, deep learning for studying the data available of the companies and provide and accuracy and twitter as a platform to carry the sentiment analysis.

Keywords- *machine learning, Twitter API, supervised learning, python, sentiment analysis.*

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. PROBLEM STATEMENT

This paper 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 paper puts in a small effort to assist the inexperienced investors and prevent from suffering heavy capital loss.

III. LITERATURE REVIEW

A thorough literature survey was performed to get a better understanding of the topic, analyse 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 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 allows user 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 tweetand their metadata. It includes variety of information including username,time,location, retweets. We have focused on time and tweet 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 tweet contains too much extraneous words that do not consider to it's sentiment value. To accurately obtain tweets 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 stop words. 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 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 sentiment meaning to text. To accomplish all these processes, we use regular expre4sions that matches these symbols. This all forms that necessary tweetcorpus.

B. Feature Extraction Module:

After gathering large tweets corpus, we have built and train classifier foe 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 for training. We have determined the 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 implementation, we have used pythons Natural Language Toolkit (NLT) which allows us to calculate with conditional frequency and frequency of each feature.

After calculating feature score, we rank the feature in order score and choose top n feature for training and classification. A feature reduction helps to improve speed of classification.

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 tweets regarding each company and generate another dataset which conatins positive, negative, neutral as well as total tweets of each day as a feature matrix. On other side we have taken such stock market historical data for each day and have calculated market market up as well as down direction and took it as a label for dataset. In case of stock market historical data, we have used Python's yahoo-finance library.

D. Prediction Module:

After training to 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 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.
(3)Authentication is required for accessing real time Twitter data.

VI. CONCLUSION

The main objective of this stock market prediction is to use this technique to find a relevant investment using the prediction and make the investor a better approach by using the prediction algorithm, and share the ideology about how the investment would benefit the investor by using the twitter sentiment analysis and the linear regression method. We could opt for a better predictive analysis.

REFERENCES

[1] OliveriaNuno, 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 Stcok Market Prediction with the TEXT analusyis*" (2017)
[3] Li, iaodong,et al "Empirical analysis: stock market prediction via extreme learning machine". *Neural computing and applications* 27.1(2016): 67-78.