Mehul Vishal Sadh, Shivani Shivanand, Shonit Gangoly, Shraddha Ingale, Xinsheng Li

Ramakrishna Koganti

Course - 2212-CSE-5301-003

3 May 2021

Impact of Donald Trump’s tweets over stock market

**Abstract**

The Internet in the recent years has seen great advances in web technology and immense growth, with massive data being accessed and created by the users. And this data is cleverly being used by companies to get feedback on their products and services. Twitter is one of the most important online communities, since they allow people to share and express their opinions on a variety of topics, with 65 million new tweets a day according to recent figures. The field of sentiment analysis of twitter data has seen a lot of progress. And our project aims to analyze the tweet sentiments of influential celebrities in order to determine the correlation between those tweets and stock market trends. We will be using exploratory data analysis to analyze the datasets and understand the effects of the Donald Trump’s tweets on the stock market.

**Introduction**

Social media has developed and become a part of daily life. According to survey, social media now has more than 3 billion people across the globe. Social media is one of the sources of information about company’s financial condition and stock market.

Celebrity engagement with brands and promotion of a particular brands is nothing new. Celebrity endorsements influence everything from the beauty products we used to the cars we drive. So, its not new that the effect would be same for stock market.

The inspiration came from Kylie Jenner’s tweet about Snapchat in 2018 and the effect it had on their stock. The star posted negative comment regarding snapchats new update and company lost around 6 percent in firm value.

We have taken Twitter as one of the social mediums of cyber universe because of the following reasons:

• The best social media which satisfies both social interaction and informative content is twitter.

• Considering the vast user base tweets allows information to spread as fast as possible like a virus.

• Free information highly relevant for investors and other stock market followers.

• All major press and news release institutions have twitter accounts and helps us to get most up to date information.

**Methodology**

1. Data Pre-Processing

We begin with data preprocessing to create a custom data frame that can be used for further visualizations as well as to apply this to the required models. This required us to remove duplicate tweets, dates in the file, combining tweets together since Twitter has a 260 characters limit. We also remove tweets that are not withing the trading hour of the S&P 500 index.

Text, application

Description automatically generated

Fig 1. Code for processing tweet data.

Our next step involves the pre-processing of the stock data. For pre-processing of stocks data we will be adding some changes to the data frame.

Text

Description automatically generated

Fig 2. Code for processing stock data.

Text

Description automatically generated

Fig 3. Code for combing stock and tweets data.

1. Exploratory Data Analysis

Next, we go about performing feature extraction on our data set. The most important factors that in a tweet are the Sentiment of the tweet (Negative/Positive), Number of likes of the tweet, Number of times the tweet was retweeted by other users. These factors would give us an indication of how well the tweet was received by the populous thus indicating their factor in change in some stock value.

To accomplish that, we used VaderSentiment which is a function used by the nltk library, on his tweets. This function helps us score the tweet on a scale of 0 to 1. Where 0 is being an extremely negative tweet and 1 being an extremely positive tweet.

Graphical user interface, text, application, email

Description automatically generated

Fig 4. Code for sentiment analysis.

Chart, line chart

Description automatically generated

Fig 5. Average sentiment of Trump’s tweets per month

After finding the average sentiments that Donald Trump’s tweets contain, we can use them to gain insights on his tweeting frequency and the most frequented words that he likes to use while tweeting. This will help us target tweets that are more political in nature or in general contain references towards the economy.

Chart, histogram

Description automatically generated

Fig 6. Top-30 most tweeted words of Trump

We can also visualize his most frequented words by using WordCloud. This function works by arranging the most frequently used words according to their size. The bigger the size of the of the words, the more frequently they have been used. As we can see words like campaign, Arlington, congress, spy, Myer family will have an impact on the further analysis that we do. We will be storing these words in a dictionary and using them to filter out the tweets and then correlate them to the stock charts.

Text

Description automatically generated

Fig 7. Word could for Trump’s tweets

Now we will be performing topic modelling. In machine learning and natural language processing, a topic model is a type of statistical model for discovering the abstract "topics" that occur in a collection of documents. Topic modeling is a frequently used text-mining tool for discovery of hidden semantic structures in a text body.

Table

Description automatically generated with medium confidenceText

Description automatically generated

Fig 8. Most frequent word and trigram word

1. Model Building

We will be using a series of Machine Learning models to correlate the tweet and stock data. These models will train based on tweets and stocks used as vector values for input.

1. Long Short-Term Memory Regression

LSTM Regression is a recurrent neural network that is trained by using Backpropagation through time and overcomes the vanishing gradient problem. There are two major functions in LSTM:

a.) Tanh Function: Standardizes values from -1 to 1. This helps with reducing extreme values when calculating vector values of text words.

b.) Sigmoid Function: Scales the values from 0 to 1. This helps us in giving relevance to the words. Closer it is to 1, more relevant it is.

The model calculates vector values and classifies it on scale of 0 and 1 where it decides the relevancy of the values.

Table

Description automatically generated

Fig 9. Summary of the LSTM model.

Performance of the Model:

Table

Description automatically generated

Fig 10. Performance of the LSTM model.

1. LSTM Regression with word vectors + 30 min Price History inputs

The performance of the model:

Table

Description automatically generated

Fig 11. Performance of the LSTM+Vector model.

1. Naïve Bayes Classifier

Naïve Bayes Classifier works by using the most frequented words. It calculates the probability of the word with respect to how many times it appears in a document. In our case a tweet. From this it creates a frequent word dictionary that classifies the words based on probabilities. This is very useful when working with textual data. It follows the following probability formula : .

The performance of the model is shown like following:

Table

Description automatically generated

Fig 12. Performance of the Naïve Bayes model.

We applied grid search cross validation. This is an algorithm that is used to estimate and tune the hyper-parameters of the model.

Model Performance after Grid Search Cross Validation:

Table

Description automatically generated

Fig 13. Performance of the Naïve Bayes cross validation.

1. Support Vector Machine

An SVM is a classifier that works in linearly separable binary sets. In our case we have two features, 1 for positive correlation and 0 for negative correlation. It uses Hyper plane line that calculates the max distance from each data point. The closer the value is to the Hyper Plane, the more related it is to that class.

The performance of the model is like following:

Table

Description automatically generated

Fig 14. Performance of the SVM model.

1. Random Forest Regression

Random Forest is an upgraded version of Decision Trees. DTs work by creating path trees using two values:

1. GINI Index
2. Entropy Gain

These two values are used to prune and evaluate the purity of the nodes in the tree. The tree then finds the optimal path from root to child node.

In the case of random forest classifier it uses a multitude of Decision Trees. It then finds either the best valued tree or the average value of the trees and then classifies the values.

The performance of the model is like following:

Table

Description automatically generated

Fig 15. Performance of the Random Forest model.

Comparison of all the 5 models:

Table 1. Comparison of all the 5 models

| Model | Precision | Recall | F1 | Final accuracy |
| --- | --- | --- | --- | --- |
| LSTM Regression | 0.39 | 0.40 | 0.39 | 39.94% |
| LSTM Vector | 0.38 | 0.33 | 0.34 | 33.11% |
| Naïve Bayes | 0.38 | 0.41 | 0.37 | 41.05% |
| SVM | 0.36 | 0.40 | 0.35 | 40.22% |
| Random Forest | 0.31 | 0.39 | 0.31 | 39.11 |

**Final Analysis**

After applying the various models, we can now do correlation on the data sets to view whether Trumps Tweets affect stock market.

Acquiring the Top 10 topics that Donald Trump usually tweets about, we find a common set of string characters that he frequently uses.

Chart, bar chart

Description automatically generated

Fig 16. Common topics of Trump’s tweets.

We also find the top 10 tweets with positive and negative price increase:

Graphical user interface, text, application

Description automatically generated

Application

Description automatically generated with medium confidence

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Fig 17. Top-10 tweets with positive and negative price increase.

The correlation graph between trump’s tweets and SPX returns is shown as following, it shows that the trump’s tweets truly influence the stock price.

Chart, line chart

Description automatically generated

Fig 18. The correlation graph between trump’s tweets and SPX.

**Reference**

[1]. <https://www.kaggle.com/vivekam101/how-trump-manipulate-stock-market?scriptVersionId=0>

[2]. <https://www.kaggle.com/shailaja4247/sentiment-analysis-of-tweets-wordclouds-textblob>

[3]. <https://www.kaggle.com/luciesteel/donald-trump-tweets-visualization>

[4]. <https://www.kaggle.com/doughersak/donald-trump-tweet-statistics>

[5]. <https://www.kaggle.com/ironicninja/quick-eda-on-all-of-trump-s-tweets>

[6]. <https://github.com/ParInsights/Predicting-President-Trump-Tweets-Stock-Market-Analysis->

[7]. <https://github.com/briangohjw/Trump-Tweets-Stock-Market-Correlation/blob/main/notebooks/Pre-Processing.ipynb>

[8]. <https://www.bloomberg.com/features/trump-tweets-market/?srnd=politics-vp>

[9]. <https://www.tradersmagazine.com/departments/commentary/how-trumps-twitter-became-the-ultimate-stock-market-indicator/>

[10]. <https://au.finance.yahoo.com/news/this-trump-tweet-punctured-the-stock-market-234301067.html>

[11] https://machinelearningmastery.com/crash-course-recurrent-neural-networks-deep-learning/

[12] http://www.ijrar.org/papers/IJRAR190I005.pdf

[13]. Hochreiter, Sepp, and Jürgen Schmidhuber. "Long short-term memory." Neural computation 9.8 (1997): 1735-1780.

[14]. Wang, Lipo, ed. Support vector machines: theory and applications. Vol. 177. Springer Science & Business Media, 2005.

[15]. Rish, Irina. "An empirical study of the naive Bayes classifier." IJCAI 2001 workshop on empirical methods in artificial intelligence. Vol. 3. No. 22. 2001.

[16]. Biau, Gérard, and Erwan Scornet. "A random forest guided tour." Test 25.2 (2016): 197-227.

**Appendix**

Pre-Processing:

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated with medium confidence

Text

Description automatically generated with medium confidence

Graphical user interface, text, application

Description automatically generated

Exploratory Data Analysis:

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Text, application, email

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Text, chat or text message

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Models Implemented:

Model 1: LSTM Regression

Text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

A picture containing text

Description automatically generated

Model 2: LSTM using Word vectors and 30 min Stock Data

Text

Description automatically generated with medium confidence

Text

Description automatically generated

Table

Description automatically generated

Model 3: Naïve Bayes Classifier

Graphical user interface, text, application

Description automatically generated

Model 4: SVM Classifier

Text, application

Description automatically generated with medium confidence

Model 5: Random Forest Classifier

Graphical user interface, application

Description automatically generated with medium confidence

Final Analysis:

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application

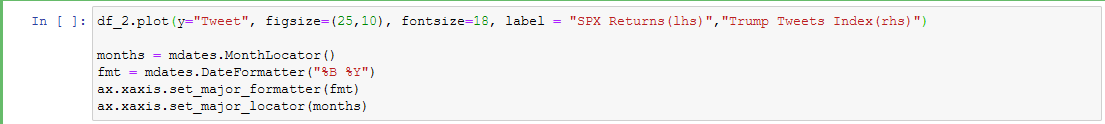
Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated



Chart, line chart

Description automatically generated