Crypto-currency trading based on twitter and news data

Team members

Chenchu Gowtham Yerrapothu 1211212168
Harshit Kumar 1211074797
Mihika Shah 1211198050
Shaik Mohammed Faizaan 1211182307
Vaishnavi Raj 1211126121
Venkatesh Magham 1213176170

1) Introduction:

Stock market prediction has been an active area of research for a long time. The Efficient Market Hypothesis (EMH) states that stock market prices are largely driven by new information and follow a random walk pattern. Though this hypothesis is widely accepted by the research community as a central paradigm governing the markets in general, several people have attempted to extract patterns in the way stock markets behave and respond to external stimuli [1].

Cryptocurrency is an alternative medium of exchange consisting of numerous decentralized crypto coin types. The essence of each crypto coin is in its cryptographic foundation. Secure peer to peer transactions are enabled through cryptography in this secure and decentralized exchange network. Since its inception in 2009, the Bitcoin has become a digital commodity of interest as some believe the crypto coins' worth is comparable to that of traditional fiat currency. Considering the exchange rates of cryptocurrencies are notorious for being volatile, our team strives to develop an effective trading strategy that can be applied to a variety of cryptocurrencies [2]. In our project, we use Twitter data (most popular and extensive data for such endeavors) for understand the trend of cryptocurrency prices.

Online social networks, like Twitter, are enabling people who are passionate about trading and investing to break critical financial news faster and they also go deep into relevant areas of research and sources leading to real-time insights. Recently Twitter has been used to detect and forecast civil unrest [3], criminal incidents [4], box-office revenues of movies [5], and seasonal influenza [6]. Stock market news and investing tips are popular topics in Twitter. In this project, we collected a 3-year period Tweets ranging over one million tweets which contains keywords relating to cryptocurrencies such as "Bitcoin", "Ethereum" and, "CryptoCurrency" [10].

Past research has shown that real-time Twitter data can be used to predict market movement of securities and other financial instruments. The advantages of using Twitter include having access to some of the earliest and fastest news updates in a concise format as well as being able to extract data from this social media platform with relative ease [2]. In our project, since the most current and established cryptocurrency is Bitcoin and Ethereum, we take up them to determine our trading algorithms. We have first collected the data from various sources such as Kaggle and Poloniex which provide open API's and datasets for such endeavors.

Since the data had lot of other information, we preprocessed the data to create our features and then used various supervised regression based learning algorithms such as linear regression, logistic regression, Neural Network for day-to-day prediction of the cryptocurrency prices. To achieve good results, we have incorporated rigorous error analysis and validation to ensure that good accuracy is achieved by the various models. Since initially our logistic regression model wasn't resulting good results we have used sentiment analysis to boost them. The project is also an evidence to support the claim that Twitter data is a good feature to develop advantageous crypto coin trading strategies. Through supervised machine learning techniques, our team will outline several machine learning pipelines with the objective of identifying cryptocurrency market movement [2].

Our trading strategy applies supervised machine learning algorithms including regression based techniques and neural networks to determine whether the price of a digital currency will increase or decrease within a day. The ways to deal with preparing these classifiers include utilizing direct content, also called tweets, from Twitter clients and utilizing outsider open-source feeling investigation APIs to rate the energy and cynicism of words inside each post. Both the preparation strategies end up being successful in evaluating the trajectories of prices for cryptocurrencies. To predict market movement to a granularity, a time series of tweets equal in length to the trading period is required one cycle beforehand. This time series of Twitter posts is used as an input to the classifiers [2].

Applying machine learning to cryptocurrency is a relatively new field with limited research efforts. Using Bayesian regression, Shah et al. achieved an 89% return on investment over fifty days of buying and selling Bitcoins [7]. Another approach predicted the price change of Bitcoin using random forests with 98.7% accuracy [8]. These approaches fail to consider the feelings of individuals about Bitcoin, and therefore, fail to harness these potential features in their learning algorithms. Twitter sentiment analysis has been widely researched. Bollen et al. utilized the Profile of Mood States (POMS) to predict the movement of the Dow Jones Industrial Average with 87.6% accuracy. Go et. al focused only on classifying tweets and used several approaches to achieve an accuracy of 84.2% with Multinomial Naive Bayes, 79.2% with maximum entropy, and 82.9% using a support vector machine [9]. Our project implements the studies conducted by various mentioned resources from the past and try to apply Twitter sentiment analysis for cryptocurrency markets.

The mentioned machine learning models have different prediction accuracies and goals. Some of them predict stock price for the intended time-frame like [18], [19] and [20]. Time frames vary between next 20 minutes to up to next month. Works such as [10], [11], [12], [13], [14], [15], and [16] predict stock price direction for the next day. [17] aims to predict the price direction every 2-hours, and [10] aims to predict monthly direction.

2) Problem description:

A lot of research is going on to correctly predict stock price data. With the increase of cryptocurrency trading, there is a need to apply the same principles of predicting stock price prediction to predicting cryptocurrency prices. The goal of this project is to predict the cryptocurrency price from previous trading data and tweets which mention Bitcoin. We gathered the historical bitcoin data from poloniex.com and used [22] for collecting the tweets. We used linear regression, logistic regression and neural networks on the historical bitcoin prices and sentiment analysis on tweets for predicting the prices. Finally, we compare the accuracy and results of these models and analyze which model gives the best result.

3) Methodology:

Twitter Data: Tweets with the keyword *bitcoin* for each day was collected. We collected 1000 tweets for each day starting from Feb 19, 2014 to Oct 31, 207. This extraction was done by crawling over the internet for the given time periods [22]. We had to preprocess these tweets because there were unnecessary words. We removed all the stop words and punctuation marks based on the paper [1]. This data had 1000 tweets for each day and we further did sentiment analysis. After this step, A sample of the data looks like this -

Date	Tweets
2014-02-19	*last one promise deal put bid first us opening austin texas via launch blended price index ... (1000 tweets overall)*
2014-02-20	*trading touch gen item dollar h h last bid sell offer index track price ... (1000 tweets overall)*

Table showing tweets extracted from twitter

Linear Regression: Linear regression is a linear approach for modeling the relationship between a scalar dependent variable y and one or more explanatory variable (or independent variables) denoted *X*. The case of one explanatory variable is called simple linear regression. For more than one explanatory variable, the process is called multiple linear regression.

When the features in the dataset were plotted in a scatter plot against the target value, a healthy linear relationship was observed. Linear regression model using sklearn library has been implemented in python. The dataset passed into the model has the closing prices of the 1-10 days data as the independent features and 11^{th} day closing price as the target label (dependent variable). A test_train_split function from sklearn is used to split the data set into training and test sets randomly, in this case the data was split into 80% training data and 20% testing data. The Mean squared errors were and the R2 scores were calculated for training and testing data separately and the R2 score for training and testing data turned out to be close to 0.96. for checking on the over fitting data, the regularized versions of the regression like LASSO regression, Elastic net regression and Ridge regression were implemented. R2 scores were calculated and turned out close to 0.9 as well. To conclude the data was very well fitting onto the regression line.

Steps involved in logistic regression

Reading csv file
Separating the feature and target variables
Splitting into training and testing data
Fit the data on a line

Testing phase
Calculate R2 score

Linear Regression using Neural Network: An Artificial Neural Network (ANN) is a data handling worldview that is motivated by the way organic sensory systems, for example, the cerebrum, process data. The key component of this system is the novel structure of the data handling framework. It is made of countless interconnected preparing components (neurons) working as one to take care of issues. ANNs, like individuals, learn by case. An ANN is arranged for an application, for example, design acknowledgment or information order, through a learning procedure. Learning in organic frameworks includes changes in accordance with the synaptic associations that exist between the neurons. This is valid for ANNs also.

Neural Networks are a family of learning methods inspired by biological neural networks by modeling a system of interconnected neurons, which are tuned based on iterative learning. Feedforward neural networks connect a multi-dimensional input into one or more hidden layers of neurons before predicting an output. Dropout is used to prevent overfitting of the model. Hidden layers are modeled by affine transformation and final layers are modeled by SoftMax. In addition, a nonlinearity (such as the tanh function) is often used after each layer. Below is a visualization of a Feed-forward neural network with two hidden layers, like the one we implemented [21].

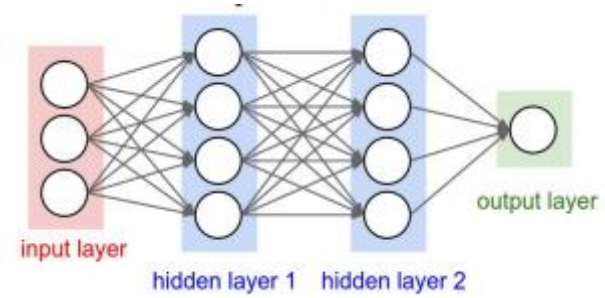


Figure shows neural network configuration with input layer containing 10 features, 10 neurons in each hidden layer and 1 output

We applied the Multilayer Perceptron based regression from the sklearn library of python on the dataset. and used the Adam solver i.e. stochastic gradient based optimizer for the weight calculation. We also utilized the k-fold validation and a constant learning rate on the dataset. Coming to the Implementation part, the dataset was randomly split into testing and training using the train_test_split function of sklearn. (80% training, 20% test) We obtained an R^2 score of 0.963. We used the MLP Regressor method of scikit learn to implement the neural network.

Logistic Regression: For logistic regression, our dataset consisted of 975 rows, where the input labels were [Day 1 - Day 10] closing prices and the target variable was Day 11. So, given the first 10 days closing data, we predict whether the price would increase or decrease on Day 11. For the actual implementation, we used scikit-learn's function to apply logistic regression. We split the data into training and testing data with 80% as training and 20% as

testing data. In order to give the best accuracy, we used k fold cross validation on the data using GridSearchCV() which uses the best parameters found in the training data for the test data. Here we used different values for C, and divided the data into 3 folds. The accuracy achieved after implementing logistic regression is approximately 61% and the average precision is 60%. The ROC score is 0.52.

Steps involved in logistic regression

Reading csv file
Separating the feature and target variables
Splitting into training and testing data
Defining parameters for GridSearchCV
Building the classifier
Testing phase
Finding accuracy
Calculate average precision
Calculate ROC Score
Plot precision-recall curve
Plot ROC Curve
Confusion Matrix
Classification Report

Sentiment Analysis: The top thousand tweets for each day were collected and stored for the past three years. For sentiment analysis, we used the python library TextBlob. This library has two methods which can be used to calculate the sentiment - *PatternAnalyzer* and *NaiveBayesAnalyzer*. We had to choose the one that best suited our purposes. *PatternAnalyzer* basically analyzes the pattern of the incoming sentence/tweet, based on its learning on the training data of movie reviews. But we are not looking for analysis of sentiment based on sentiment. And for our feature vector, getting the probabilistic values in more suitable and that is what would help in increasing the efficiency of algorithms. Thus, we chose to use *NaiveBayesAnalyzer*.

Sentiment(classification='pos', p_pos=0.548034588796245, p_neg=0.451965411203755)

This is what we get when we give a phrase as an input to *NaiveBayesAnalyzer*. The *p_pos* and *p_neg* are the probabilistic values of that phrases' positivity and negativity. They basically add up to 1. The attribute *classification* indicates the class the Analyzer classifies based on the *p_pos* and *p_neg* values. For calculating our feature vector, we use the *p_pos* value.

4) Results:

Various regression and classification techniques have been used and classification accuracies and R2 scores of each model were obtained.

Linear Regression: The R2 scores for training data and testing data are close to 0.96 which implies the data was well fitting into the regression line. R2 score is the proportion of the variation in the dependent variable that is predictable from the independent variables. The plots for the predicted labels vs true labels can be seen below for linear regression. R2 score ranges from 0 to 1; An R2 score of 1 indicated that the data can perfectly fitted on a straight line and an R2 score of 0 indicated that the data is scattered and is a poor idea to represent on a straight line.

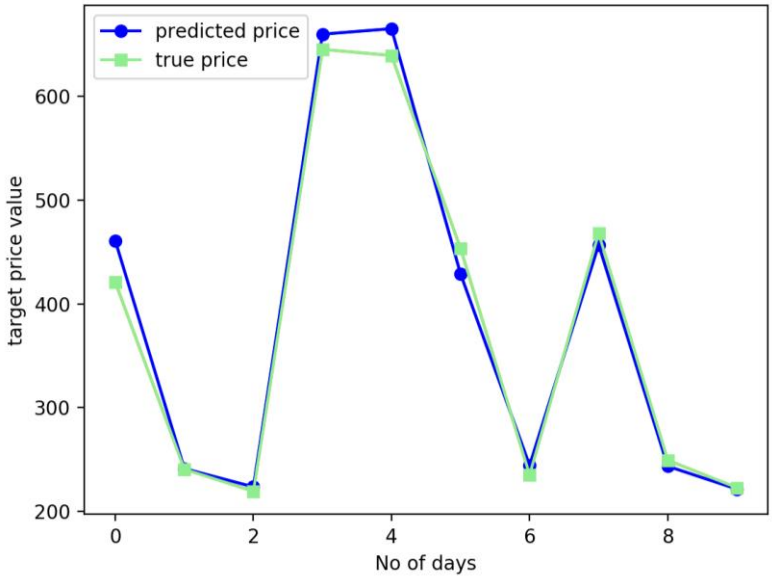


Figure shows how close the actual value is from predicted value for training data using linear regression, as a plot between Bitcoin price vs 10 randomly chosen days

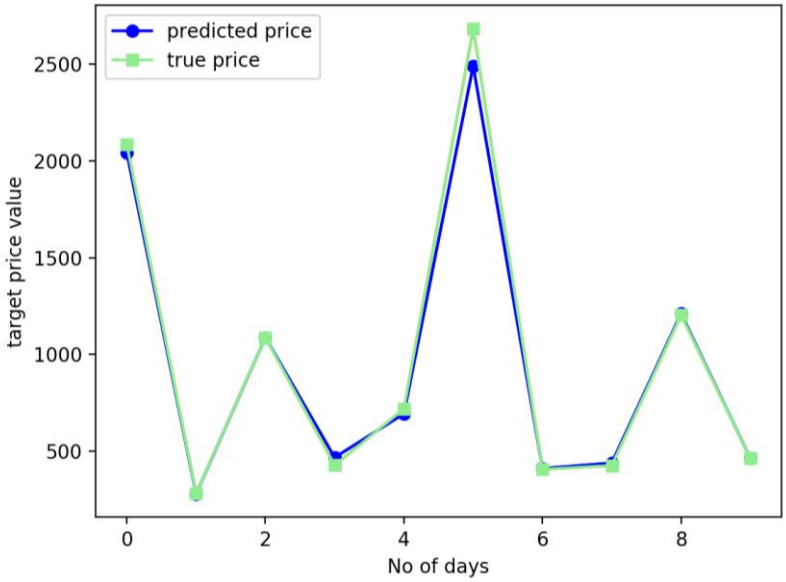


Figure shows how close the actual value is from predicted value for testing data using linear regression, as a plot between Bitcoin price vs 10 randomly chosen days

Neural Network: Since the test data contained 195 values, we trimmed the curve to show for 10 values only. Cross validation curve was plotted to measure the accuracy of the generated model.

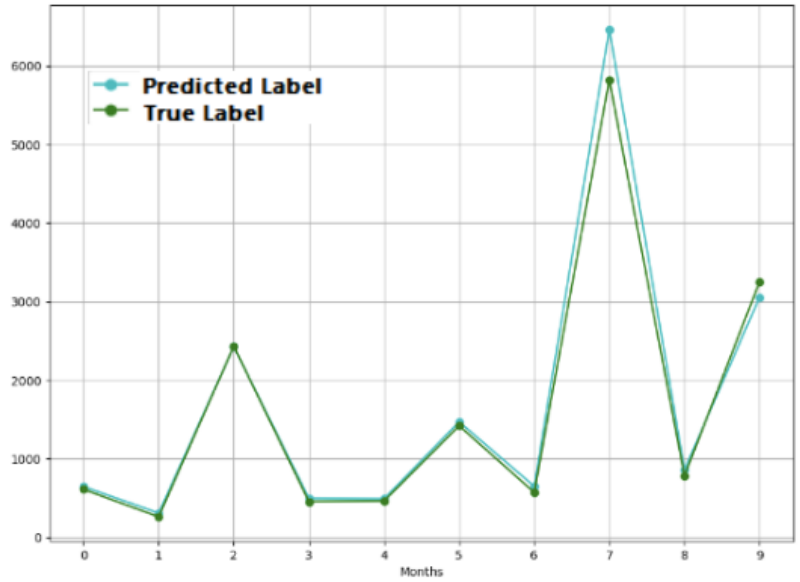
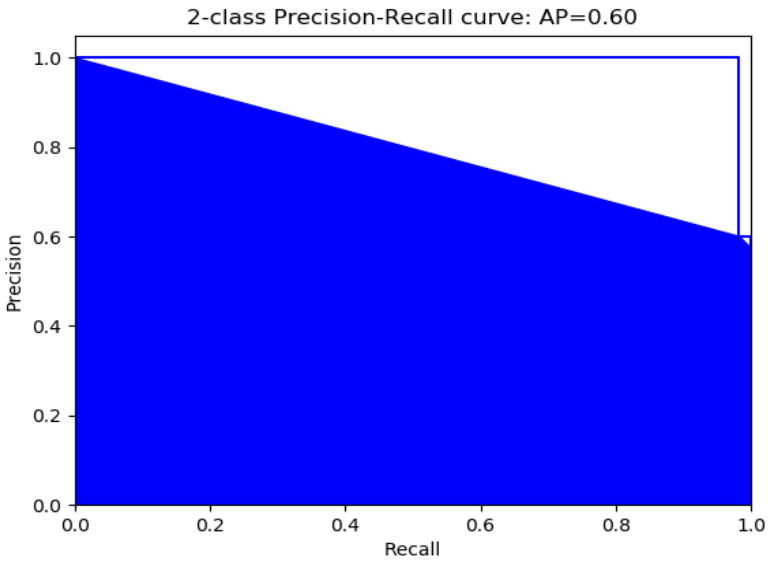


Figure shows how close the actual value is from predicted value for testing data using neural network regression, as a plot between Bitcoin price vs 10 randomly chosen days

Logistic Regression: These are the results obtained for logistic regression:
Accuracy: 0.626
ROC Score: 0.542



Precision vs Recall curve for logistic regression task on bitcoin prices (without sentiment analysis)

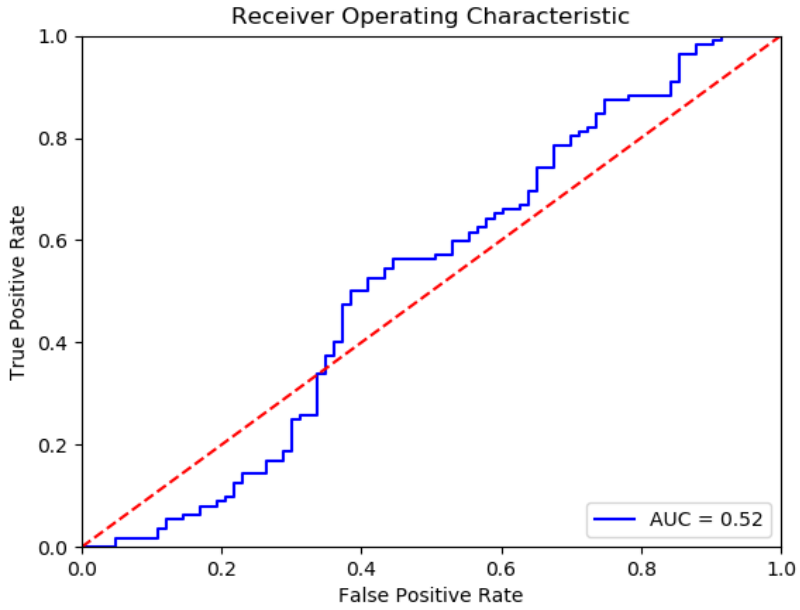
Confusion matrix: [[7 63]
[10 115]]

The confusion matrix is of the format [[True negative False negative]
[False positive True positive]]

Classification Report:

	Precision	Recall	fl-score	Support
Negative Examples	0.41	0.10	0.16	70
Positive Examples	0.65	0.92	0.76	125
avearge / total	0.56	0.92	0.76	195

Recall is the number of actual positive retrieved / total positives retrieved
Precision is the number of relevant instances retrieved / total instances retrieved



ROC plot for logistic regression task on bitcoin prices (without sentiment analysis). We see that the Area under curve is 0.52, which means that this classifier a poor classifier

5) Conclusions and future work:

Conclusions: To summarize, we have done both regression and classification. All the results are given in the tabular format below. For classification techniques, using Sentiment of the twitter data has seemed to work better for both NNs and Logistic regression. It increased the accuracies by around 6 % and 8 % in NNs and Logistic regression respectively. Coming to regression, we were achieving a good fit using both linear regression and Neural Networks. They both achieved a similar R2 score of around 0.96.

Model	Accuracy
Logistic regression without Sentiment Analysis	60 %
Logistic regression with Sentiment Analysis	68 %
Logistic regression using NNs without sentiment analysis	73 %
Logistic regression using NNs with sentiment analysis	79 %
Linear regression	R2 score - 0.96
Linear regression using NNs	R2 score - 0.963

Table showing the model and the results (accuracy & R2 score)

Future work:

- We collected around thousand tweets for every day with using the keyword 'Bitcoin'. There was too much data but we didn't know how to exactly represent it. We plan on using LSTM cells to represent each tweet. So around thousand tweets would give us thousand extra features. We would try using these tweets and see how the results carry.
- We have used just the twitter data for sentiment. But we believe it's better to use the news data as it represents the crowd beliefs better.
- We would like to use the same algorithm to predict stock prices and see how better or worse it works.
- Our predictions now are for every day closing time. There is a limited data for us if we do it on daily data as neural network architectures tend to need lot more data. We plan on using minute by minute data and do the predictions accordingly.
- Bitcoin was the only cryptocurrency we used for predictions. Even though Bitcoin is the most popular cryptocurrency we would like to try our algorithm on other cryptocurrencies like ethereum, litecoin etc.

References

[1] Mittal, Anshul, and Arpit Goel. "Stock prediction using twitter sentiment analysis." Standford University, CS229 (2011 http://cs229. stanford. edu/proj2011/GoelMittal-StockMarketPredictionUsingTwitterSentimentAnalysis. pdf) 15 (2012).

[2] Lee, Kari, and Ryan Timmons. "Predicting the Stock Market with News Articles." Colianni, Stuart, Stephanie Rosales, and Michael Signorotti. "Algorithmic Trading of Cryptocurrency Based on Twitter Sentiment Analysis."

[3] Harshavardhan Achrekar, Avinash Gandhe, Ross Lazarus, SsuHsin Yu, and Benyuan Liu. Twitter improves seasonal in- 18 H. alostad et al. / Directional Prediction of Stock Prices using Breaking News on Twitter fluenza prediction. In HEALTHINF 2012 - Proceedings of the International Conference on Health Informatics, Vilamoura, Algarve, Portugal, 1 - 4 February, 2012., pages 61–70, 2012.

[4] Sitaram Asur and Bernardo A. Huberman. Predicting the future with social media. In Proceedings of the 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology - Volume 01, WI-IAT '10, pages 492–499, Washington, DC, USA, 2010. IEEE Computer Society.

[5] Ryan Compton, Craig Lee, Jiejun Xu, Luis Artieda-Moncada, Tsai-Ching Lu, LalindraDe Silva, and Michael Macy. Using publicly visible social media to build detailed forecasts of civil unrest. Security Informatics, 3(1), 2014.

[6] Xiaofeng Wang, Donald Brown, and Matthew Gerber. Spatiotemporal modeling of criminal incidents using geographic, demographic, and twitter-derived information. In International Conference on Intelligence and Security Informatics, Lecture Notes in Computer Science. IEEE Press, IEEE Press, 2012.

[7] Go, Alec, Lei Huang, and Richa Bhayani. Twitter sentiment analysis . Entropy 17 (2009).

[8] Madan, Isaac, Saluja, Shaurya, and Aojia Zhao, Automated Bitcoin Trading via Machine Learning Algorithms, Department of Computer Science, Stanford University.

[9] Shah, Devavrat and Kang Zhang Bayesian Regression and Bitcoin . http://arxiv.org/pdf/1410.1231v1.pdf. 6 Oct. 2014. Web. 12 Nov. 2015.

Bollen, Johan, Huina Mao, and Xiaojun Zeng. "Twitter mood predicts the stock market." *Journal of computational science* 2.1 (2011): 1-8.

[10]Alostad, Hana, and Hasan Davulcu. "Directional prediction of stock prices using breaking news on Twitter." Web Intelligence and Intelligent Agent Technology (WI-IAT), 2015 IEEE/WIC/ACM International Conference on. Vol. 1. IEEE, 2015.

[11] Johan Bollen, Huina Mao, and Xiaojun Zeng. Twitter mood predicts the stock market. Journal of Computational Science, 2(1):1–8, 2011.

[12] Michael Hagenau, Liebmann Michael, and Dirk Neumann. Automated news reading: Stock price prediction based on financial news using context-capturing features. volume 55, pages 685; 685–697; 697, /2013 06. doi: 10.1016/j.dss.2013.02.006 pmid:.

[13] MI Yasef Kaya and M. Elif Karsligil. Stock price prediction using financial news articles. In Information and Financial Engineering (ICIFE), 2010 2nd IEEE International Conference on, pages 478–482. IEEE, 2010.

[14] Stefan Lauren and S. Dra Harlili. Stock trend prediction using simple moving average supported by news classification. In Advanced Informatics: Concept, Theory and Application (ICAICTA), 2014 International Conference of, pages 135–139. IEEE, 2014.

[15] E. L. Lehmann. Nonparametrics :statistical methods based on ranks. Holden-Day, San Francisco. E. L. Lehmann, with the special assistance of H. J. M. D'Abrera.; ;24 cm; Includes bibliographical references and index.
[16] Yuexin Mao, Wei Wei, and Bing Wang. Twitter volume spikes: analysis and application in stock trading. In Proceedings of the 7th Workshop on Social Network Mining and Analysis, page 4. ACM, 2013.
[17] Arman Khadjeh Nassirtoussi, Saeed Aghabozorgi, Teh Ying Wah, and David Chek Ling Ngo. Text mining of newsheadlines for forex market prediction: A multi-layer dimension reduction algorithm with semantics and sentiment. Expert Systems with Applications, 42(1):306–324, 1 2015.
[18] Jigar Patel. Predicting stock market index using fusion of machine learning techniques. Expert Systems with Applications, 42(4):2162; 2162–2172; 2172, 2015. doi: 10.1016/j.eswa.2014.10.031 pmid:.
[19] Robert Schumaker and Hsinchun Chen. Textual analysis of stock market prediction using financial news articles. AMCIS 2006 Proceedings, page 185, 2006.
[20] Tien-Thanh Vu, Shu Chang, Quang Thuy Ha, and Nigel Collier. An experiment in integrating sentiment features for tech stock prediction in twitter. 2012.
[21] Greaves, Alex, and Benjamin Au. "Using the Bitcoin Transaction Graph to Predict the Price of Bitcoin." (2015).
[22] - https://github.com/Jefferson-Henrique/GetOldTweets-python