INTERNATIONAL
HELLENIC
UNIVERSITY

Stock Market Prediction using Sentiment Analysis

Nousi Christina

SID: 3308190020

SCHOOL OF SCIENCE & TECHNOLOGY

A thesis submitted for the degree of

Master of Science (MSc) in Data Science

January 2021
Thessaloniki – Greece

INTERNATIONAL
HELLENIC
UNIVERSITY

Stock Market Prediction using Sentiment Analysis

Nousi Christina

SID: 3308190020

Supervisor:	Assoc. Prof. Christos Tjortjis
Supervising Committee Members:	Prof. Panagiotis Bozanis
	Dr. Dimitrios Karapiperis

SCHOOL OF SCIENCE & TECHNOLOGY

A thesis submitted for the degree of

Master of Science (MSc) in Data Science

January 2021
Thessaloniki – Greece

Abstract

Application of both Machine Learning (ML) and sentiment analysis from microblogging services has become a common approach for stock market prediction. In this thesis, Microsoft's stock movements are analyzed using historical and sentiment data. In particular, 90.000 tweets were collected from Twitter and 7.440 tweets from StockTwits covering the period from 16 – 07 – 2020 to 31 – 10 – 2020. Historical data were also mined from the Finance Yahoo website at the same period. The sentiment analysis of social media data was conducted using two Python libraries including TextBlob and VADER (Valence Aware Dictionary and sEntiment Reasoner). We also implemented multiple machine learning models including KNN, SVM, Logistic Regression, Naïve Bayes, Decision Tree, Random Forest and MLP. Our results indicate that when using tweets from Twitter with VADER as sentiment analysis tool, SVM is the ML algorithm which gives the highest f-score equal to 75.9% and Area Under Curve (AUC) equal to 65%.

Acknowledgments

I would like to thank my supervisor, Dr. Tjortjis Christos, for supporting and mentoring me during my dissertation. The help, the advice and the guidance, he has provided me with, were paramount for the completion of this thesis.

I am also grateful to my parents and colleagues for their constant support and encouragement during my studies for the MSc of Data Science.

Contents

List of figures

Chapter 1

1 Introduction

Stock market has become an essential part of a country's economy as it is a way to make investments and gain high capital (Billah, et al., 2016). A stock market is a network of economic transactions where activities of buying or selling shares take place. An equity market or share market mirrors the ownerships claims on businesses. This may include shares which emanate from public stock exchange or from individual trade, for example shares of private companies sold to investors. Trade in stock markets is defined as an activity of transferring money from small individual investors to large trader investors, including banks, companies etc. However, investments in the stock market are characterized as a highly risk activity because unpredictable behavior is noticed (Gurjar et al., 2018).

Stock market prediction could be paramount for the investors if it is achieved successfully. The efficient prediction of stock markets may offer investors a helpful guidance in order to take the appropriate decisions and measures whether to buy or sell shares. The definition of stock market prediction is the act of trying to identify the future stock's value (Gurjar et al., 2018). For many years, a lot of methods for predicting stock market have been introduced, but they can be classified into four small categories. The first one is fundamental analysis in which is based on published financial statements. The second one is technical analysis in which the predictions are achieved thanks to historical data and prices. The third one involves ML methods applied to a huge amount of data derived from multiple sources. The last one is sentiment analysis in which the predictions are made of published news, articles or blogs (Y. Huang et al., 2018). The combination of the last two categories is much newer than the other two and through studies and researches it seems to have a significant effect on making the appropriate decision whether to buy or sell a stock (Y. Huang et al., 2018).

The purpose of this thesis is to apply ML techniques and sentiment analysis in order to predict Microsoft's stock movements. Sentiment analysis is achieved using available information of microblogging platforms such as Twitter and Stocktwits, as they provide important information about people's emotions. The most common theory is that when the public's sentiment is positive for a company, then the stock prices tend to rise and vice a versa. However, when taking into consideration other economic factors, this theory is not always true. In this thesis, multiple ML techniques are applied to 90.000 tweets and 7.440 stock tweets which were extracted from Twitter and StockTwits respectively between 16 – 07 – 2020 and 31 – 10 - 2020. Historical data were also mined form Yahoo Finance website taking advantage of the open and close values of the Microsoft's stocks. The best prediction you can have about tomorrow's value is having today's open and close values.

1.1 Goal and Research Questions

The goal of this research is to study the state of the art concerning stock market prediction using sentiment analysis. In order to have a comprehensive overview of the issue, we focused on the following research questions:

(Q1) Which are the most prominent theories for stock market prediction?

(Q2) Which are the classic approaches for stock market prediction? In which cases have these approaches been used?

(Q3) In which cases of stock market prediction, have ML techniques been used? Which of these cases involved sentiment analysis?

(Q4) What needs to be involved in the design and development of a stock market prediction model?

The first question aims to provide a brief overview of the theories of stock market prediction including the efficient market hypothesis and random walk theory. The second question analyses classic approaches including the technical and fundamental approach, focusing on case studies in which these approaches have been used for stock market prediction. The third question emphasizes on the cases studies in which multiple ML techniques have been applied for predicting stock prices along with case studies in which both ML methods and sentiment analysis have been implemented for forecasting stock movements. The last question focuses on the development of our stock market prediction model. Particularly, it informs about model performance and predictions of Microsoft stock movements.

1.2 Contributions

In this thesis, we propose a method for stock market prediction, which uses historical and sentiment data. To train our predictive model, we worked with real stock data about Microsoft, the well-known technology company.

The contributions of this thesis are summarized as follows:

1. We extracted data from Twitter, StockTwits and Finance Yahoo and conducted preprocessing as detailed in Chapter 3.

2. Sentiment analysis on data from Twitter and StockTwits was done using two Python libraries, including TextBlob and VADER, as described in section 3.2.1.

3. We trained and tested our model using seven ML algorithms, as presented in section 4.2.

4. We evaluated seven ML algorithms for stock prediction as discussed in section 4.3.

1.3 Thesis Outline

In Chapter 2, the theories and classic approaches for stock market prediction are discussed. The case studies of stock market prediction using only ML approaches or both ML and sentiment analysis approaches are also included in this chapter. Chapter 3 discusses the dataset used in stock market prediction model development, as well as the data preparation process. In addition, the methodology of the proposed model is reviewed in detail in this chapter. The results obtained, the performance of the proposed model and the prediction of stock market movements are presented in Chapter 4. Finally, the conclusion of the research and proposed directions for future work are included in Chapter 5.

Chapter 2

2 Literature Review

Various researches have been conducted in the field of stock performance prediction (Bohn, 2017). In this chapter, we review theories for stock market prediction including the Efficient Market Hypothesis (EMH) and the Random Walk theory. In addition, we review the classic approaches used for the prediction of future stock movements concerning the technical and fundamental analysis. Then, previous studies using just ML techniques or ML techniques combined with sentiment analysis to predict stock market returns are reviewed.

2.1 Theories of Stock Market Prediction

There are various theories for predicting stock market prices; two of them are the best known (Falinouss, 2007). The first one is the Efficient Market Hypothesis (EMH) (Fama, 1960) and the second is the Random Walk Theory (Malkiel, 1999).

2.1.1 Efficient Market Hypothesis (EMH)

The Efficient Market Hypothesis (EMH) asserts that share prices represent all available information and expectations of the market. In other words, when new information enters the market, it is directly expressed in stock prices (Alam, 2017). So, the best estimate of a company's intrinsic value is its current market prices (Attigeri, et al., 2015). When the market accepts the new information, the system instantly is converted into the unbalanced condition and the new prices remove the predicted correct change (Attigeri, et al., 2015).

The given information that is prementioned may be fundamental or non – fundamental information (Fakhry, 2016). In other words, fundamental information is yields or macroeconomic factors in the sovereign debt market. On the other hand, non – fundamental is information from news (Fakhry, 2016).

The theory of Efficient Market Hypothesis is divided into three forms: Weak, Semi – Strong and Strong (Ajekwe, et al., 2017).

1. *Weak*. In weak EMH, only past data are considered in the current price, such as historical prices.
2. *Semi – Strong*. The semi – strong form includes all the historical and current data and all the public information, such as profit prognoses and sales forecasts.
3. *Strong*. The latter one is the strong form which contains all public and private information, such as insider information in the stock price.

The weak form is widely used in many research studies (Ajekwe, et al., 2017). According to Ajekwe et al. (2017), some tests of weak form efficiency were carried out in Nigeria. Monthly data on 59 randomly stocks were utilized from 1981 – 1992. It was found that the Nigerian market comply with the weak form when ten lags exist in return data. After conducting further research, it was concluded that the Nigerian stock market is of weak form efficiency (Ajekwe, et al., 2017).

2.1.2 Random Walk Theory

Another theory for predicting stock market movements is the theory of random walk. The theory states that changes of stock prices are independent through time, they have the same distribution and may be characterized by a random process, for example tossing a coin (Ajekwe et al., 2017). Changes of stock prices happen when new information arrives, since information arrives in a random way and stock prices change unpredictably (Ajekwe et al., 2017).

Concerning the stock market, it is claimed that the prediction of stock prices is impossible when the prices are determined randomly (Falinouss, 2007). In other words, prices literally take a random walk. Random walk indicates to investors that by taking extra risks is the only way to outperform the market. The market is said to be successful if the prices adjust easily and without bias to new information (Ajekwe et al., 2017). Moreover, it is important to mention that the random walk theory has the same foundation with the semi – strong form efficiency because all public information is available to everyone (Falinouss, 2007).

According to Ajekwe et al. (2017), correlation tests were carried out to investigate weekly prices of 21 selected Nigerian Firms from July 1977 to July 1979. It was concluded that stock prices fluctuations were not correlated, but they took a random walk. After further researching, the price behavior of 30 stocks has been tested from 1977 to 1980, utilizing Monday closing prices after conforming for cash dividends and script issues. Finally, it was inferred that the stock prices followed a random walk (Ajekwe et al., 2017).

2.2 Stock Market Prediction using Classic Approaches

According to Dow Jones theory, the fluctuations of the market price are developed in trends (Picasso et al., 2019). Thus, researchers have presented techniques in order to forecast market trends and evaluate stocks that led to the creation of two different major types of stock market prediction methods: technical analysis and fundamental analysis (Picasso et al., 2019).

2.2.1 Technical Approach

The primary purposes of the technical analysis approach are the assessment of investment, the anticipation of stake holders' thoughts and the detection of buying or selling opportunities based on available information about the historical price and volume (Huang, 2019). In other words, technical analysts try to extract patterns and use stock prices and various types of mathematical indicators computed by the historical stock prices and volume in order to predict the future stock prices (Picasso et al., 2019). All this information about the stock market is extracted from charts. However, the main disadvantage about charts and time series data is that they only present the event and not the reason why it happened (Falinouss, 2007). Furthermore, it is important to emphasize that technical analysis is more preferable for short term forecasting (Huang, 2019).

There are also three general assumptions underlying technical analysis. Particularly, the three premises are that market action discounts everything, prices move in trends and history repeats itself (Bohn, 2017).

1. *Market action discounts everything:* The first premise states that anything that can affect the price fundamentally, politically or psychologically, is actually discounted and mirrored in the price of that market (Bohn, 2017). This means that all fundamental information that affects the price is indicated in the historical prices. Technicians also believe that the price movements should affect demand and supply. For instance, if the market prices are rising, then demand should surpass supply and the fundamentals have to be bullish (Bohn, 2017).

2. *Prices move in trends:* The second assumption supports that price movements which follow a trend, regardless the time being observed, they will continue following the existing trends than reversing. It is preferable, stock prices continue moving a past trend than moving irregularly (Bohn, 2017).

3. *History repeats itself:* The last premise maintains that the future is a repetition of the past. The price movements tend to be characterized by the human psychology, such as fear or hope. Technical analysis takes into consideration chart patterns in order to analyze human emotions and as a result to understand stock market movements. This helps technical analysts to identify whether the market is bullish or bearish (Bohn, 2017).

Furthermore, it is vital to mention that most of the existing studies for stock market prediction are based on technical analysis. According to Huang (2019), a study was proceeded for stock market prediction using feed – forward neural network, in 1990. Some technical indicators and macroeconomic indices, such as interest rates and foreign exchange rates were used as inputs in order to predict their model. The model was tested and checked the existence of buying or selling signals of the TOPIX index from January 1987 to September 1989. The outcome of the study was that the neural network model can accomplish superior profit, through the buy – and – hold strategy (Huang, 2019).

Another study was conducted for forecasting the following day trend of NASDAQ and NIKKEI indices, utilizing ANFIS model and Recurrent Neural Network (RNN) (Huang, 2019). For both models, the previous closing price was considered as an input in order to predict the next day’s closing price. The training dataset consisted of data from 1971 to 1998, while the test dataset was constituted from data from 1998 to 2002.

Finally, it was induced that the return rate for the ANFIS model is higher than the RNN model and the buy – and – hold strategy for both indices (Huang, 2019).

2.2.2 Fundamental Approach

Fundamental analysis is based on financial data that companies have to publish on a regular basis, such as financial status, yearly report, balance – sheets, income statements etc. in order to forecast if the stock price increases or decreases in the future (Nti et al., 2019). The approach of fundamental analysis tries to investigate economic factors that may influence stock prices and determine the company's true value (Bohn, 2017). In particular, fundamental analysts analyze economic factors and the movements of stock prices using three dimensions, taking into consideration the economy, the industry and the company. Nevertheless, fundamental analysis is more suitable for mid – term and long - term stock market forecasting when the time horizon is a quarter, a year or longer (Nti et al., 2019).

There are also some financial ratios which are useful for comparing companies with different size, but in similar sectors. When defining the performance of a company, it is vital to exclude the size of the company because the real profit is a function of the percentage price change and not a function of absolute price change (Bohn, 2017). The most significant financial ratios used in fundamental analysis are profitability ratios, liquidity ratios, debt ratios, asset utilization ratios and market value ratios (Bohn, 2017).

1. *Profitability ratios.* This ratio computes how well the company is able to gain profit.

2. *Liquidity ratios.* The liquidity ratio measures the company's ability to pay off its immediate debt obligations.

3. *Debt ratios.* The debt ratio is a financial ratio which evaluates the ability of the firm to pay off the debt liabilities over time. Its definition is the proportion of total debt to total assets.

4. *Asset utilization ratios*. Asset utilization ratio calculates the efficiency of the company to use its assets.

5. *Market value ratios*. Market value ratio assesses the current stock price of the company and it mirrors the market value of the share and the company.

It is essential to mention that if the ratios for a stock are better than those of a company, this does not necessarily mean that the stock should be purchased. One reason is that even though the stock movements of a company may fluctuate better than similar ones, the whole market, the industry and the sector may underperform (Bohn, 2017).

Thus, as prementioned, the purpose of the fundamental analysis is to evaluate the stock price using available financial ratios, which are made public. According to Huang (2019), a feed – forward neural network model was developed based on some financial ratios. As an input, seven attributes were selected concerning the historical PE ratio, prospective PE ratio, market cap, EPS uncertainty, Return On Equity (ROE), cash flow yield and the last feature is a factor that is determined by the weighted average of estimated historical prices. For the study, 25 stocks were collected, and the data were extracted from Q1 1993 to Q4 1996. It is important to cite that there are 16 observations for each stock. The training set was defined by the first 10 observations while the test set consisted of the other 6 observations (Huang, 2019). However, because of the limited data, another perspective was studied. In particular, the training set was composed by the first three quarters, while the test set was comprised by the following quarter. Stocks with the highest estimated returns were selected for a portfolio. The experimental results indicate that the proposed model is capable of choosing portfolios which surpass 10 out of 13 quarters and generate higher returns. However, the conclusion is that the experiment could not be entirely conducted due to lack of data (Huang, 2019).

Another study by Namdari et al., (2018) proposed a Multi - Layer Perceptron (MLP) neural network model and a hybrid model using financial ratios and historical data with a view to predicting stock market fluctuations. The authors selected 12 financial ratios of 578 technology companies whose data is accessible through the NASDAQ website. The data were collected from June 2012 to February 2017. Afterwards, the authors developed another MLP neural network model based on technical analysis, which means that it is formed only by historical data regarding the same companies. Their purpose was to compare the two models and to choose the one with the best accuracy for predicting future stock movements. Finally, it was concluded that MLP model based on fundamental analysis had greater accuracy (64.38%) than the MLP model based on technical analysis with 62.84% accuracy. Thus, fundamental analysis was the appropriate method for predicting future stock movements (Namdari et al., 2018).

2.3 Stock Market Prediction using Machine Learning Approaches

Diverse ML and data mining techniques have become more common in recent years for the prediction of stock market movements. Now, related works which have implemented ML for determining the future value of a stock, will be studied.

Deepak et al. (2017) studied the use of various ML algorithms based on the stock market prediction and they concluded that Artificial Neural Network (ANN) was the most efficient algorithm with the highest accuracy. In their study, they implemented Support Vector Machine (SVM) with the use of Radial Basis Function (RBF) kernel algorithm. According to Deepak et al., the reason why they utilized the SVM algorithm was because it was considered to be the most appropriate algorithm for time series prediction, let alone for forecasting shares. The SVM algorithm is one of supervised learning and it draws a hyperplane which separates the data into two classes. The RBF algorithm is a type of neural network, it is a feed – forward one and it is a non - linear supervised learning which depends solely on radial distance from a point (Deepak et al., 2017).

Deepak et al. collected the required data from the yahoo finance website covering the period from 2014 to 2016. Then, they determined the input parameters which are done using historical stock data. Afterwards, they decided which of the parameters will be included. The selected ones were the open high, close high and moving averages values. The last step was to combine the four different feature lists of four Bombay Stock Exchange (BSE) listed companies using SVM and calculating the accuracy. The accuracy achieved was up to 89%. The conclusion was that the SVM algorithm played an important role for generating custom features and the use of RBF kernel contributed for having a grated accuracy in outcome (Deepak et al., 2017).

Rasel et al. (2016) proposed three different ML approaches including ANN, SVM and K – Nearest Neighbors (KNN) for predicting equities 1 day ahead, 5 days ahead and 10 days ahead. The purpose of the study was to predict the closing price of a stock. The extracted data were historical data of Wal – Mart stores Inc., a listed company in the New York Stock Exchange (NYSE), ranged from 2010 to 2015. The data set consisted of 10805 instances. The number of the selected attributes was five, including date, open price, close price, high price and low price. The date attribute was determined as ID and the close price attribute as the label. The data set was divided into two parts, the training set and the test set. The training set was composed of 80% of the data, illustrated the period from 2010 to 2014, while the test set contained the rest 20% of the data, including the period 2014 – 2015 (Rasel et al., 2016). Then, a windowing operator was developed in order to change the time series data into generic data. In addition, two measures were computed to calculate the rate of error for the three models. The two evaluation metrics were the Mean Average Percentage Error (MAPE) and the Root Mean Square Error (RMSE) which were applied only on the test set. Finally, it was inferred that the 1 day ahead which forecasts the price of 1 day ahead, is the best predictor among the three models and the model with the lowest rate of error was the ANN (Rasel et al., 2016).

Y. Huang et al. (2018) examined multiple ML techniques to predict stock prices. They tried to forecast using ANN the Indian stock market prices, in particular the National Stock Exchange (NSE). First, they implemented linear regression with a view to predicting the opening price of the stock for the following day utilizing the closing price of the stock on the previous day. Thereafter, SVM regression was applied to forecast the difference between close and open prices of the stock for the next day (Gurjar et al., 2018).

According to Gurjar et al., the ML technique used for predicting the Indian stock prices was ANN with backpropagation. They trained the ANN model by employing historical stock data. Furthermore, some features were extracted from the historical stock data, such as foreign exchange rate, NSE index, moving averages, Relative Strength Index (RSI) etc. for achieving a higher accuracy. In addition, some factors were used in the model for stock price predictions, including moving averages, stochastic oscillator, standard deviation and on – balance volume. In particular, moving averages are indicators which remove the noise from data and it is based on past prices. Simple moving averages were used for 1 day, 7 days and 15 days. The stochastic oscillator is another indicator which calculates the difference between the close price and the range of the stock’s prices through a period of time. The standard deviation indicates how far a set of data is from its mean and the on – balance volume is an indicator that uses volume flow to forecast changes in stock price. In the end, the authors state that ANN outperforms the linear regression and they should have added more stocks from NSE stocks for achieving better results (Gurjar et al., 2018).

Choudhry et al. (2008) compared the system based on Genetic Algorithm (GA) and SVM with the stand alone SVM system. First, the purpose of the study was to predict the stock prices of three companies including Tata Consultancy Services (TCS), Infosys and Reliance Industries Limited (RIL). The data used, were extracted from the Yahoo Finance website covering the period from August 12, 2002 to January 18, 2008. In total, 1386 trading days' data were collected. The features that the authors obtained were the opening, the highest, the lowest and the closing values of the stock prices. The data were split into training, validation and test set. The 60% of data constituted the training set, the 20% was used for the validation and the remaining 20% was used for testing the system (Choudhry et al., 2008).

Choudhry et al. applied GA to select the most important indicators as input features for the SVM. The total number of features selected was 35. Also, they correlated the stock prices of various companies to predict the price of a stock. It was observed that TCS stocks were highly correlated with the stocks of similar industries, for example with Infosys. Furthermore, prediction performance was estimated with the calculation of the hit ratio. Hit ratio is the percentage of times where the prediction system was correct. Thanks to hit ratio, it was concluded that the GA – SVM hybrid model outperformed the SVM. For instance, for Infosys, the hit ratio of GA – SVM was 60.3% while the hit ratio of SVM was 56.7%. Thus, the concept of correlation and the GA implication was two important factors for improving the performance of SVM model (Choudhry et al., 2008).

Billah et al. (2016) proposed an improved Levenberg Marquardt (LM) algorithm of ANN for stock market closing price prediction. The data were extracted from Dhaka Stock Exchange (DSE) with a time ranging from January 2013 to April 2015. The selected features obtained from the historical data are the following: daily opening price, closing price, highest price, lowest price and total number of stocks traded. Then, the data were preprocessed for cleaning and removing any noise (Billah et al., 2016).

Billah et al., also, implemented two other algorithms including the Adaptive Neuro Fuzzy Inference System (ANFIS) and traditional LM algorithm, in order to explore which of the three models achieve the best stock prediction performance. To measure the accuracy and the performance of the models, Root Mean Square Error (RMSE) and the coefficient of multiple determinations (R^2) were used. RMSE values which are near to 0 predicts less error while R^2 values near to 1 mean higher correlation. The authors inferred that the improved LM algorithm was much more efficient than RMSE and traditional LM algorithm. In particular, the results demonstrated that the improved LM algorithm had the highest R^2 value and the lowest RMSE value which was 53% less than the other methods. In addition, the authors calculated the time and memory needed for running the three models. The improved LM algorithm showed to have the best performance concerning the time and memory needed. Specifically, the memory and time needed for the improved LM algorithm were 54% and 30% less than the traditional LM algorithm and 59% and 47% less than ANFIS. Thus, the improved LM algorithm performed not only better stock prediction but also less memory computation and computing time than the traditional LM algorithm and ANFIS (Billah et al., 2016).

Somani et al. (2014) conducted a survey on prediction stock market prices using Neural Network, SVM and Hidden Markov Model (HMM). In this paper, the authors implemented one model for forecasting the shares of three companies including ICICI, SBI and IDBI. First, they trained the data by developing the model of HMM with the help of the Baum - Welch algorithm which uses Expectation – Maximization (EM) with a view to opting the optimal parameters for the HMM model. Afterwards, the data was tested by using the Maximum a Posteriori approach (MAP). Once the model is tested, the authors utilized a metric to estimate the performance. The metric was the Mean Absolute Percentage Error (MAPE) which is the average absolute error between the actual stock prices and the predicted ones in percentage. In particular, the MAPE value for ICICI was 2.1, for SBI was 1.7 and for IDBI was 2.3. The results show that they achieved a good performance, and it was observed that when the training data was increased then the performance was decreasing (Somani et al., 2014).

Liu et al. (2018) approached a Term Memory Long – Short (LSTM) neural network model for extracting feature value, analyzing stock data and predicting stock prices. LSTM is a kind of time recurrent neural network (RNN) which is suitable for handling and forecasting significant events of interval and long delay in time series. In this paper, the authors use the LSTM neural network algorithm with a view to predicting short term stock price changes. The stock historical transaction data was extracted from the stock data interface of JoinQuant platform. The experiment was conducting using the historical data of CSI 300 Index including the time period ranged from 2014-05-18 to 2017-01-29. The features selected from the historical data of CSI 300 were open, close, low, high, volume, money, limit_up and limit_down values. Then, they constructed some features by performing some calculation. In particular, they calculated the following indexes (Liu et al., 2018).

- *Moving Average (MA)*
- *Exponential Moving Average (EMA)*
- *oc = (close – open) / open*
- *oh = (high – open) / open*
- *ol = (low – open) / open*
- *ch = (high – close) / close*
- *cl = (low – close) / close*
- *lh = (high – low) / low*

The prementioned features were used as the features of the training samples with the time period from 2014-05-18 to 2016-12-25. On the other hand, the test samples are the prementioned features of the closing data of CSI 300 from 2016-12-26 to 2017-01-29. In addition, the authors used not more than three layers of the stacked LSTM model because when the layers are more than five, more computing resources are needed. The experimental results showed that the accuracy of the single layer LSTM model was 0.66 while the accuracy of the three - layer LSTM model was more than 0.78. So, it was concluded that the more the layers, the better prediction performance. However, the only drawback was that if the layers are higher, the computational resources need will increase. Finally, this paper states that the prediction performance could be improved if more feature values were extracted for training the LSTM model (Liu et al., 2018).

Shah et al. (2018) presented two neural network models namely LSTM and Deep Neural Network (DNN) for predicting the daily and weekly stock fluctuations of the Indian BSE Sensex index. As dataset was defined the historical data of Tech Mahindra stock which illustrated the period from 1997 to 2017. First, the authors applied the two models for predicting only the daily closing prices of the stock. In order to compare the two models, two metrics were used including RMSE and forecast bias. RMSE is a suitable metric for analyzing time series predictions. The RMSE value is ideal when it is equal to zero. Forecast bias relates the model's bias with the true values. A model that has a large positive forecast bias value, it means that the model overpredicts the true data. So, after measuring the RMSE and the forecast bias, it was concluded that the DNN generated a lower RMSE value than the LSTM and LSTM produced lower forecast value than the DNN. However, both models presented a good prediction performance.

Then, this study proceeded with the weekly stock predictions (7 trading days ahead). The authors compared the two models by using a metric called Directional Accuracy (DA). DA correlates the direction of each prediction with the direction of true data for a particular time period. The results indicated that the LSTM model outperformed the DNN. Although DNN demonstrates a good performance on stock prediction, it has a difficulty when recognizing the quick changes of the time series data. It is important to be emphasized that a measure was taken in order to avoid the overfitting issue. In particular, they stopped training the models when loss function values of the training and validation data stabilized. By conducting this action, not only the overfitting evaded but also the data generalized. To sum up, this study could be improved by using more features like daily volume, volatility, fundamental ratios etc. and not only price data as it is used in this paper.

Patel et al. (2015) focused on prediction of stock market movements and stock prices indices. They compared four models namely ANN, SVM, random forest and Naïve – Bayes. The data used was historical data of CNX Nifty, S&P Bombay Stock Exchange (BSE) Sensex, Infosys Ltd. and Reliance Industries from Indian stock markets. The historical data was extracted from National Stock Exchange (NSE) India and BSE India websites and was ranged the period from January 2003 to December 2012. The authors employed two approaches for making the prediction. In particular, the first approach involves the calculation of ten technical parameters by using the stock trading data such as open, high, low and close prices. These ten technical parameters were presented as continuous values, which represents the actual time series, and were used as inputs to the predictor models (Patel et al., 2015). The ten technical indicators are the following:

- *10 days' Simple Moving Average (SMA) and Weighted Moving Average (WMA).* They are used for short term prediction and when the price is above the average then the trend is up.
- *Stochastic oscillators like STCK%, STCD%, and William R%.* When the stochastic oscillators have an upward trend, then stock prices go up.
- *Moving Average Convergence Divergence (MACD).* When the MACD indicator increases, then the stock prices also go up and the opposite.
- *Relative Strength Index (RSI).* It produces values from 0 to 100. If the RSI value is up to 70, then the stock is overbought and it likely go down in the future.

- *Commodity Chanel Index (CCI)*. It computes the difference stock' price change and its average price change. When its value is positive, then it means that the prices are above average which indicates a good performance.
- *Accumulation/Distribution oscillator (A*/D). It also underlines the stock trend meaning.
- *Momentum*. It determines the rate of increase or fall in stock prices. When the momentum produces positive values, it means that the stock trend goes up.

The second approach involves the transformation of the ten technical indicators to trend deterministic data which are discrete values. In other words, the ten technical indicators were normalized having values between +1 and -1, where +1 demonstrates up price movement while -1 indicates down price movement. So, the trend deterministic data is the new input to the four models. The estimation of the prediction performance was carried out by computing the accuracy and F – measure. The results have shown that the models which learned from continuous – valued inputs had a decent performance but when the models learned from trend deterministic data, the performance was further improved. Particularly, when the survey conducted with continuous – valued data, Naïve Bayes model produced least performance with 73.3% while Random Forest showed the best performance with 83.56% accuracy. On the other hand, when the models learned by using trend deterministic data, ANN had the least performance with 86.6% accuracy while Naïve – Bayes exhibited the highest performance with 90.19%. The authors, as future work, proposed the use of macroeconomic variables like inflation, interest rate etc., multiple variables like highly possible to go up, less possible to go down etc. and also achieving long term stock prediction (Patel et al., 2015).

Afterwards, Patel et al. (2015) conducted another study on predicting stock future values. They used historical data from January 2003 to December 2012 of two stock market indices namely CNX Nifty and S&P BSE Sensex. They used the same ten technical indicators as input for the predictor models, as they did from the previous study. In addition, they implemented two approaches. The first one was the single stage approach where the authors applied three models including ANN, Support Vector Regression (SVR) and Random Forest. The second approach was the two stage fusion approach in which the authors utilized SVR – ANN, SVR – SVR and SVR – RF. For both approaches, 1 – 10, 15 and 30 days in advance prediction experiments are performed. The performance of the two models were evaluated by using the Mean Absolute Percentage Error (MAPE), Mean Absolute Error (MAE), relative Root Mean Squared Error (rRMSE) and Mean Squared Error (MSE). The results exhibited that when the number of days ahead were increasing then the error values were going up (Patel et al., 2015). Moreover, the two - stage fusion approach presented a better performance than single stage approach for almost all prediction days. The performance became better when ANN and RF were hybridized with SVR. On the other hand, when the SVR was hybridized with itself, the performance was moderate. The best performance was presented by SVR – ANN model. As future work, the authors proposed the idea for taking into consideration the news related to company performance, government policies etc., as they affect the stocks' prices. Considering the sentiment of the news in combination with the stock trading data, the stock predictions become more accurate (Patel et al., 2015).

2.4 Stock Market Prediction using Sentiment Analysis Approaches

Social media, which represent the public sentiment about current events, have a great impact today than ever. Many studies have used Twitter or StockTwits in order to apply sentiment analysis in the field of trading strategy. In this section, previous works, which have combined ML approaches with sentiment analysis for the purpose of stock market prediction, will be elaborated.

Pagolu et al (2016) implemented twitter sentiment analysis and ML techniques and analyzed the correlation between the company's stock market fluctuations and the sentiment of the tweet texts. They extracted 250.000 tweets by using Twitter API. The tweets were ranged the period from August 31, 2015 to August 25, 2016. The content of the tweet texts referred to Microsoft. The authors used some keywords in order to filter out the desired content including $MSFT, #Microsoft, #Windows etc. They wanted to extract the sentiment of public not only about the company's stock but also about the products and services provided by Microsoft. Furthermore, the authors collected the stock opening and closing prices of Microsoft the period from August 31, 2015 to August 25, 2016, which were derived from the Yahoo Finance website. It is supported that the weekend and holidays, the stock market does not function. Thus, the authors replaced the missing values by using a technique by Goel. The missing values were approximately equal to the average of opening and closing prices. Furthermore, the tweets were preprocessed following three stages. The first one was tokenization, the second one was stop word removal and the last one was regex matching for removing special characters. At the tokenization stage, the tweets were split into words as a result to form a list of words for each tweet. At the stop word removal stage, words such as a, is, the, with etc. were removed from the tweets as they don't express any sentiment. And at the last stage, the authors removed the symbol (#) from the hashtags or when the tweets contained URLs were replaced by the word URL or when a user was addressed by the symbol (@) was replaced by the word USER (Pagolu et al, 2016).

Pagolu et al (2016) implemented two textual representations involving the N – gram representation and word2vec representation for feature extraction. In the first one, the tweets were split into N – grams and the features represented either the string of 1s or 0s, where the 1 displayed the presence of the N – grams of the tweets while the 0 displayed the absence. In the second representation, every word of the language was corresponded to a unique vector which they summed up and they produced a resultant vector of 300 dimensional vectors of all words in a tweet. The resultant vector constituted the features of the model. However, the word2vec representation was selected for the model as it is more effective when it has to do with large datasets. The tweets were classified into positive, neutral and negative using the features of the word2vec representation and were trained using the RF algorithm. The accuracy with the word2vec representation was 70.2% while the accuracy with N – grams was 70.5%. Even though the accuracy with N – representation was higher, the word2vec was selected as it behaves better with large datasets (Pagolu et al, 2016).

In addition, Pagolu et al (2016) labeled the stock price data of Microsoft properly. In particular, if the stock price of the previous day was higher than the stock price of the current day, then the current was labeled with a value of 0 else labeled with 1. So, the authors gathered the tweets and the stock prices in order to train the algorithm and to find if there is any correlation between the sentiment and the stock prices. The training set was the 80% of the total data while the rest was the testing set. Using Logistic regression algorithm, the accuracy was 69.01%. On the other hand, they trained the model using 90% of the data as training test with the help of LibSVM algorithm. They achieved 71.82% accuracy. The results showed that the performance with large dataset was good and also there was a good correlation between the stock market fluctuations and the tweet sentiment of the public. As future work, the authors suggest the use of stocktwits data and a dataset with more than 10.000 tweets (Pagolu et al, 2016).

Batra et al. (2018) focused on the Apple's stock market prediction using the combination of stocktwits data and market data. The stocktwits data were extracted from StockTwits through API which is a social networking website where people can post financial related tweets. The tweets were ranged the period from 2010 to 2017. The features retrieved were tweet id, user id, time, tweet text, retweets, sentiment of user for that tweet (bullish or bearish) etc. On the other hand, the market data were derived from Yahoo Finance website from 2010 to 2017. The attributes selected were open price, close price, low and high price volume and adjusted close. Then, the authors preprocessed the tweets by removing stop words, applying tokenization and removing some symbols such as @, #, $, URLs, extra spaces and punctuations expect $ which symbolizes the ticket of the company name. Afterwards, the market data were preprocessed. Because the stock does not function at weekends, the authors replaced the missing values by calculating the average of previous and next day values in order to find the current's day value. Furthermore, they constructed another attribute which included the stock price decision. In particular, they subtracted today's closing price from yesterday's closing price and if the result was positive, the price was increased and the person could sell the stock (Batra et al., 2018).

Batra et al. (2018) applied the SVM algorithm in order to forecast the public's sentiment. The 80% of data was the training set and the 20% represented the testing set. They classified the tweets into bearish and bullish. The accuracy achieved was 91.2% for the training set and 63.5% for the testing set. After that, the authors merged the sentiment and the market data in order to predict if a person will buy or sell a stock, by using the SVM algorithm. The final attributes selected for the model were three including date, stock price decision and sentiment. The training model performed 75.22% accuracy while the testing model presented 76.68% accuracy. To sum up, the results seem to be good, but they can be improved by increasing the size of the dataset (Batra et al., 2018).

Kordonis et al. (2016) conducted a survey whether the public sentiment is correlated with the stock market values for 16 most popular tech companies, for example Microsoft, Amazon, Apple Blackberry etc., employing the SVM and Naïve Bayes algorithms. The tweets were mined from Twitter through API and the features selected were tweet id, timestamp and tweet text which was constituted up to 140 characters. The stock data were extracted from Yahoo Finance API, including open, close, high and low attributes for each day. The tweet data were preprocessed by tokenizing them, removing stop words and unnecessary twitter symbols. They, also, used N – grams representation for feature extraction. The authors evaluated each unigram, bigram and trigram representation by applying the Pearson's chi – squared test. In this way, the most significant features were selected for training the model. In order to predict the public sentiment of the tweet texts, the authors applied the SVM and Naïve Bays algorithms. Using 7 – folds cross validation, they accomplished 80.6% accuracy with Naïve Bayes and 79.3% accuracy with SVM algorithm. Afterwards, they preprocessed the stock market data. Particularly, as it is prementioned before, the market is close at weekends and other holidays. Thus, the values were replaced by the average of the previous price value and the next price value. Moreover, two additional metrics were created enclosing the High – Low Percentage (HLPCT) and Percentage Change (PCT) which were calculated as it is shown below (Kordonis et al., 2016).

- $HLPCT = \frac{High-Low}{Low}$
- $PCT\ change = \frac{Close-Open}{Open}$

Those metrics were very important for finding the correlation existence between tweets and stock market. Then, the authors combined the tweets and stock data including the following features (Kordonis et al., 2016).

- *Percentage positive sentiment score*
- *Percentage negative sentiment score*
- *Percentage neutral sentiment score*
- *Close price*
- *HLPCT*
- *PCT change*
- *Volume*

Kordonis et al. (2016) applied the SVM algorithm for achieving the future stock market prediction. The results showed that the accuracy was 87%. In addition, the authors took into consideration the prediction errors which for all the tech companies were under 10%. In particular, the largest prediction error was noticed in the case of the Blackberry: 6.29%. On the other hand, nine out of sixteen tech companies accomplished prediction error under below 1%. In general, the average prediction error for all the tech companies was 1.668%. To conclude, the results demonstrated a good performance in which the public sentiment affected the stock market prices. As future work, the authors recommended using data with a wide range of dates, both from Twitter and the stock market and including intraday stock changes with a view to accomplishing a greater accuracy (Kordonis et al., 2016).

Hamed et al. (2015) were the first ones who investigated the correlation between Saudi tweets and the Saudi market index. The tweets data was selected from Mubasher company website in Saudi Arabia by using API. The Mubasher company is a stock analysis software provider in the region of Gulf. In total, they extracted 3335 tweets for 53 days, with a period ranged from 17/03/2015 to 10/05/2015. Also, they mined the closing prices of the TASI index from the Mubasher company website. First, the authors applied two methods for downloading the tweets. The first one was the method called "GET statuses/mentions_timeline" in which the twenty most recent mentions were returned for authenticating the user. The second one named "GET statutes/user_timeline" returns tweets which were posted recently by the screen_name or user_id parameters. The tweets were preprocessed by tokenizing them, removing stop words, suffixes and prefixes (Hamed et al., 2015).

After preprocessing, the tweets were classified into positive, neutral and negative by using ML algorithms including Naïve Bayes, KNN and SVM. The model was evaluated by computing the recall and precision values. Using 10 – folds cross validation, the accuracy of Naïve Bayes was 69.86%, the accuracy of SVM was 96.6% and the accuracy of KNN was 96.45%. The best recall was accomplished by SVM which was equal to 95.71% while the best precision was achieved by KNN which was equal to 95.91%. Afterwards, the authors tried to construct a one – to – one model which demonstrates the positive and negative sentiments in combination with the close prices of TASI index. They created a chart, in which during 24% of the time showed that when the TASI index falls then the negative sentiment increases. During 36% of the time, the charts indicated that when the TASI index rises then the positive sentiments augments. However, the existence of a stable fluctuation was noticed between positive sentiment, negative sentiment and the TASI index during 40% of the time. For example, when the negative sentiment was increasing, then the positive sentiment was falling and the TASI index was also reduced. To sum up, the results performed the existence of a good correlation between the sentiment and the TASI index. As a future work, they propose to predict the opening prices for the Saudi stock market (Hamed et al., 2015).

Mittal et al. (2012) examined the causative relation between tweets and the Dow Jones Industrial Average (DJIA) values. The tweets were mined from Twitter the period from June 2009 to December 2009. In total, they collected 476 million tweets posted by more than 17 million users. The features selected were the timestamp, username and tweet text. The DJIA values were extracted from Yahoo Finance ranged the period from June 2009 to December 2009. The attributes picked were the open, close, high and low values for a given day. The authors preprocessed the stock values by replacing the missing values with the help of estimating the average of the DJIA values on a given day and on the next day. Also, they removed periods in which the data was volatile and it was more difficult for them to make predictions. The tweets were classified into four categories namely calm, happy, alert and kind. The authors developed their own analysis code with a view to predicting the tweets' sentiment, as follows (Mittal et al., 2012).

1. *Word List Generation*. They created a word list with the help of Profile of Mood States (POMS) questionnaire. In particular, POMS questionnaire is a psychometric questionnaire in which a person is asked to rate his or her current mood by answering 65 questions on a scale of 1 to 5. Then, the answers were corresponded to 6 standard POMS moods including tension, depression, anger, vigor, fatigue and confusion. They followed a similar approach of N – grams representation.

2. *Tweet Filtering*. They obtained tweets which contained words like feel, makes me, I'm or I am as they expressed a sentiment.

3. *Daily Score Computation*. A word counting algorithm was applied to estimate the score for every POMS word for a given day.

   Score of a word = $\frac{number\ of\ times\ the\ word\ matches\ tweet\ in\ a\ day}{number\ of\ total\ matches\ of\ all\ words}$

4. *Score Mapping*. The score of each word was mapped to the six POMS moods. Then, the six POMS mood was restricted to the authors' four mood states produced by some correlation rules. For instance, happy was resulted as the sum of vigor and negation of depression.

The authors significantly noticed that they should have compared the tweets across the days and not comparing the value of a mood against other. Furthermore, Granger Causality analysis was used in order to be identified which mood value can be used to forecast the future stock movements. Granger Causality analysis is a metric which indicates how much predictive information an attribute has about another for a specific time period. The p – value, also, was computed as it demonstrated the statistical significance of the authors' result. The lower the p – value, the higher the predictive ability and vice a versa. It was concluded that happiness and calmness were the most helpful moods for predicting the future DJIA values and particularly when using the past 3 or 4 days' data (Mittal et al., 2012).

After investigating the causality relationship between the past three days' moods and the present day's stock prices, they applied four ML algorithms including Linear Regression, Logistic Regression, SVM and Self Organizing Fuzzy Neural Networks (SOFNNs) to train and test the model. They performed six different combinations with the past 3 days' mood values in order to make sure any existence of the dependence of other mood states on DJIA, as follows (Mittal et al., 2012).

- $I_{CD} = Calm + DJIA$
- $I_{CHD} = Calm + Happy + DJIA$
- $I_{CAD} = Calm + Alert + DJIA$
- $I_{CKD} = Calm + Kind + DJIA$
- $I_{CHKD} = Calm + Happy + Kind + DJIA$
- $I_{CHAD} = Calm + Happy + Alert + DJIA$

The best results were presented by Calmness and Happiness which confirmed the previous Granger causality results. Concerning the ML algorithms, SVM and Logistic Regression presented the worst performance for all the mood combinations while Linear Regression performed a good result with the Happy and Calm moods and SOFNN had the best performance. In particular, it was observed that SOFNN had the best results with the combination Happy, Calm and DJIA with 75.56% accuracy. Generally, it was noticed that adding any other mood state, the accuracy was decreased. For example, in cases of I_{CHKD} and $I_{CHAD,}$ the accuracy was really low. As future work, the authors recommend for investigating the real public sentiment by using other social media platforms and including other languages except English (Mittal et al., 2012).

2.4.1 Sentiment Analysis Applications in other fields

In addition to stock market, there are other several popular fields where the sentiment analysis of social media data has a significant role in achieving a valid prediction.

Oikonomou et al. (2018) and Beleveslis et al. (2019) focused on election result predictions. Beleveslis et al. (2019) conducted a survey to predict the sentiment of a Greek tweet related to the Greek 2019 Elections. The authors applied a hybrid method which combines Greek lexicons and Twitter data filtered based on hashtags (Beleveslis et al., 2019). Oikonomou et al. (2018) created a model to forecast the results of the US presidential elections happened in 8 November 2016. The authors focused on three primary states of US including Florida, Ohio and N. Carolina and on two candidates: Donald J. Trump and Hillary Clinton. They extracted data from Twitter and the sentiment analysis was performed by TextBlob and Naïve Bayes. Their predictions proved accurate as the results were really close to their predictions and they correctly forecasted who would win the elections (Oikonomou et al., 2018).

Tsiara et al. (2020) concentrated on chart position for songs. In particular, the authors collected chart data including titles, artist names and rankings and Twitter data which are related to the top 10 songs and artists for each week. They gathered more than one million tweets and the sentiment analysis was performed by VADER. Their results indicated that there is a moderate correlation between the title of a song referred to Twitter posts and the success of the song on Billboard Hot 100 Chart for the next week. However, there is a weak correlation between the tweets that provide the number of mentions of an artist and the future performance of a song (Tsiara et al., 2020).

Koukaras et al. (2020) conducted a literature review on using social media data in healthcare. Thanks to the existence of social media data, people can detect, mitigate and predict diseases, virus outbreaks, vaccination decisions etc. The contribution of the sentiment analysis of social media data is very important as deaths are prevented, healthcare costs are decreased etc. (Koukaras et al., 2020).

Rousidis et al. (2020) have reviewed trending domains of social media prediction using recent literature. The authors analyzed several fields which have been categorized into three groups: Finance, Marketing and Sociopolitical. The finance category focuses on stock market or product pricing prediction. The marketing group includes the prediction of trends, behavior etc. and the sociopolitical one includes the prediction of elections, natural phenomena etc. Their conclusions show that not all the models' predictions are high accurate and seems that is based on the corresponding field. In particular, 53.1% of the examined fields achieved a valid prediction, the 18.8% did not and the rest seems to perform a plausible valid prediction (Rousidis et al., 2020).

Chapter 3

3 Data & Methodology

A major component for predicting Microsoft stock is collecting the corresponding dataset. One of the challenges was to gather the appropriate data from multiple sources and combine them together. It is also very important to prepare the dataset properly in order to use it for model training and testing. In this chapter, we discuss how we collected the appropriate data and how we preprocessed them to be ready for measuring our model performance. The data collection and pre-processing were developed in Python.

3.1 Data Collection

For forecasting Microsoft's stock price, we collected financial data from Yahoo Finance website and social media data, including twitter data from Twitter, and stock twitter data from StockTwits the period from 16 – 07 – 2020 to 31 – 10 - 2020. The stock market prediction will be separated into two parts. The first one includes the use of Twitter data and financial data and the other one includes StockTwits data and financial data.

3.1.1 Twitter Data

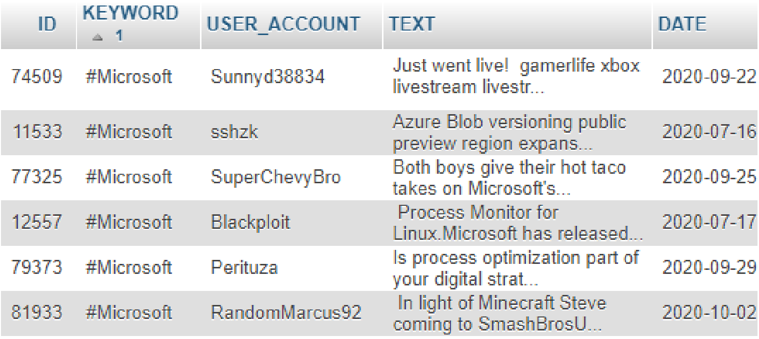
With over 200 million active users a month, Twitter[1] is one of the most popular social media and has become a wealth of data for those trying to understand how people feel about brands, products, and more (Serban et al., 2014). In this project, we collected Twitter data from Twitter in order to perform sentiment analysis. In other words, our goal was to measure people's opinion about the Microsoft and their products.

We created a developer account on Twitter website with a view to taking the appropriate Twitter credentials to connect with Twitter API. Importing the tweepy library and setting the access token and access token secret, the authentication of Twitter was achieved. The server returns a JSON object of tweets. After creating the API object, tweets were collected and filtered using some keywords like #Microsoft, #MSFT, $MSFT, #Microsoft365, #Windows, #MicrosoftNews, #MicrosoftSurface, #MicrosoftFlightsSimulator, #MicrosoftExcel, #MicrosoftAzure, #MicrosoftWord, #MicrosoftTeams, #MSInspire, and #MicrosoftStocks. Not only the opinion of people about the company's stock was extracted but also about products and services offered by Microsoft.

In addition, the tweets posts collected were written in English. What we extracted from each tweet post was the keyword, the user id, the user account, the date in which the

[1] www.twitter.com

tweet was created and the tweet text. All these information were stored in the MySQL database, as it is shown in Figure 1, because the total number of tweets collected were about one hundred thousand. Thus, using the Microsoft SQL server was easier to manipulate such amount of data.



ID	KEYWORD ▲ 1	USER_ACCOUNT	TEXT	DATE
74509	#Microsoft	Sunnyd38834	Just went live! gamerlife xbox livestream livestr...	2020-09-22
11533	#Microsoft	sshzk	Azure Blob versioning public preview region expans...	2020-07-16
77325	#Microsoft	SuperChevyBro	Both boys give their hot taco takes on Microsoft's...	2020-09-25
12557	#Microsoft	Blackploit	Process Monitor for Linux.Microsoft has released...	2020-07-17
79373	#Microsoft	Perituza	Is process optimization part of your digital strat...	2020-09-29
81933	#Microsoft	RandomMarcus92	In light of Minecraft Steve coming to SmashBrosU...	2020-10-02

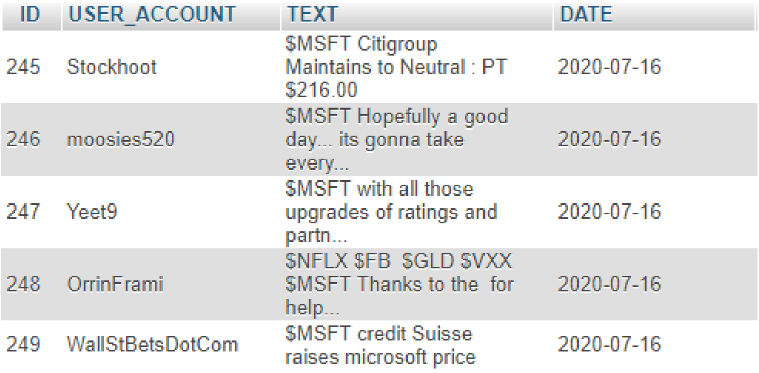
Figure 1: A sample of the Twitter data

3.1.2 StockTwits Data

In order to take advantage of the benefits of sentiment analysis in stock market industry, we extracted tweets from StockTwits[2], which is a social networking platform for finance. StockTwits users have stock and price discussions and they express their market sentiment with millions of traders and investors. StockTwits has attracted more than 40 million users worldwide. It also provides two APIs in order to collected data. The first one is the streaming API which furnish the latest 30 tweets of a user and the second one is the search API in which the extraction of tweets is based on language, time and company ticker (Batra et al., 2018). In our study, we use the search API, even though the queries have rate limit.

Importing the library named requests, the server returns a JSON object of tweets. The object contains the user account, the text of tweets and the date in which the tweets were created. The extraction of the tweets is based on the company ticker like $MSFT. The tweets collected were also stored in MySQL database, as it is shown in Figure 2, for manipulating data with an easy and efficient way.

[2] www.stocktwits.com

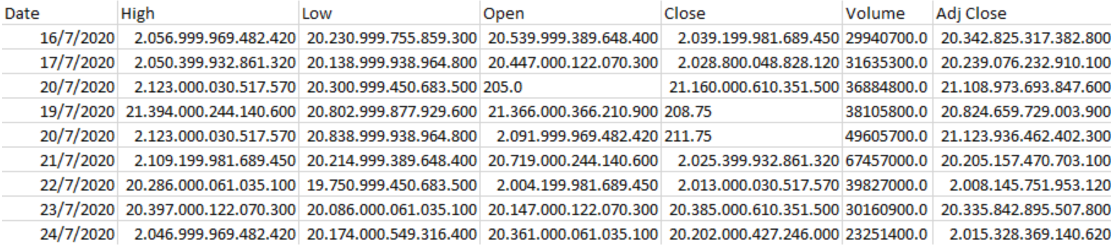


ID	USER_ACCOUNT	TEXT	DATE
245	Stockhoot	$MSFT Citigroup Maintains to Neutral : PT $216.00	2020-07-16
246	moosies520	$MSFT Hopefully a good day... its gonna take every...	2020-07-16
247	Yeet9	$MSFT with all those upgrades of ratings and partn...	2020-07-16
248	OrrinFrami	$NFLX $FB $GLD $VXX $MSFT Thanks to the for help...	2020-07-16
249	WallStBetsDotCom	$MSFT credit Suisse raises microsoft price	2020-07-16

Figure 2: A sample of the StockTwits data

3.1.3 Financial Data

The Microsoft's historical data, as it is demonstrated in Figure 3, was extracted from the Yahoo! [3]finance website in which there are myriads of international market data, up-to-date news, stock quotes or portfolio resources (Nann et al., 2013). The attributes that we collected are closing price, opening price, low and high price, volume and adjusted price. In our study, we focus mostly on the opening and closing prices.



Date	High	Low	Open	Close	Volume	Adj Close
16/7/2020	2.056.999.969.482.420	20.230.999.755.859.300	20.539.999.389.648.400	2.039.199.981.689.450	29940700.0	20.342.825.317.382.800
17/7/2020	2.050.399.932.861.320	20.138.999.938.964.800	20.447.000.122.070.300	2.028.800.048.828.120	31635300.0	20.239.076.232.910.100
20/7/2020	2.123.000.030.517.570	20.300.999.450.683.500	205.0	21.160.000.610.351.500	36884800.0	21.108.973.693.847.600
19/7/2020	21.394.000.244.140.600	20.802.999.877.929.600	21.366.000.366.210.900	208.75	38105800.0	20.824.659.729.003.900
20/7/2020	2.123.000.030.517.570	20.838.999.938.964.800	2.091.999.969.482.420	211.75	49605700.0	21.123.936.462.402.300
21/7/2020	2.109.199.981.689.450	20.214.999.389.648.400	20.719.000.244.140.600	2.025.399.932.861.320	67457000.0	20.205.157.470.703.100
22/7/2020	20.286.000.061.035.100	19.750.999.450.683.500	2.004.199.981.689.450	2.013.000.030.517.570	39827000.0	2.008.145.751.953.120
23/7/2020	20.397.000.122.070.300	20.086.000.061.035.100	20.147.000.122.070.300	20.385.000.610.351.500	30160900.0	20.335.842.895.507.800
24/7/2020	2.046.999.969.482.420	20.174.000.549.316.400	20.361.000.061.035.100	20.202.000.427.246.000	23251400.0	2.015.328.369.140.620

Figure 3: A sample of Yahoo! Finance data

3.2 Methodology

After gathering the appropriate data, the next step is to identify people's opinion about Microsoft and its product and services. In other words, we want to make sense of the Twitter and StockTwits data by doing sentiment analysis. In our research, we apply two sentiment analysis tools, for both Twitter and StockTwits, including the TextBlob and VADER in order to test which of the two gives the best result. Afterwards, we preprocess the data to be ready for building the ML models. In this section, our final goal is to merge the stock prices with the sentiment score into one table with a view to forecasting Microsoft's stock prices.

3.2.1 Sentiment Analysis

In our approach, we used two lexicons analysis including the VADER and TextBlob. The VADER is a lexicon and rule – based sentiment analysis tool which is a great tool when dealing with social media data. VADER does not just return the positive, neutral

[3] http://finance.yahoo.com

or negative values but it refers how positive or negative a sentiment is (Gilbert et al., 2014). Importing the library named “SentimentIntensityAnalyzer” and calling the polarity_scores() method, we obtain the polarity for each text. This method returns four scores, the negative, the positive, the neutral and the compound score. The compound score is the sum of all lexicon ratings, the negative, positive and neutral ones. As it is shown below, in Figure 4, most tweets from StockTwits are neutral. Particularly, 46.7% of tweets are neutral, 43.3% are positive and 10.0% are neutral. This means that most users are neither positive nor negative to invest money for buying or selling any Microsoft’s stock. In Figure 5, most of tweets are positive. In particular, 48.0% of tweets are positive, 28.0% are neutral and 24.0% are negative. In other words, people’s sentiment is positive about the Microsoft industry and its products. Thus, we can assume that people may be in favor of buying Microsoft’s stocks.

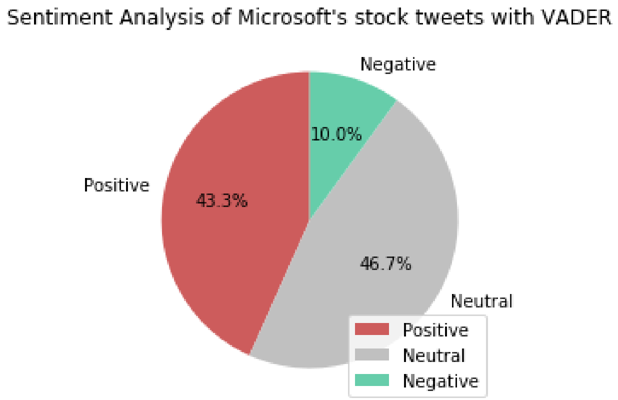


Figure 4: Sentiment Analysis of tweets from StockTwits using VADER

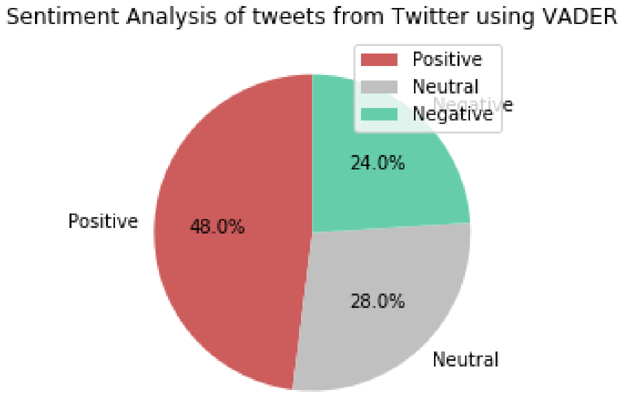


Figure 5: Sentiment Analysis of tweets from Twitter using VADER

On the other hand, the TextBlob is a simple python library used to perform sentiment analysis. It is a successful tool in which we can implement natural language processing easily and quickly (Loria, 2018). Importing the library named “TextBlob”, the tweets are classified into positive, neutral and negative. In Figure 6, most tweets from StockTwits are neutral. In particular, 53.3% of tweets are neutral, 40.0% are positive and 6.7% are negative. Similarly, in Figure 7, most tweets from Twitter are also neutral with 48.0%, while the positive tweets are 36.0% and negative tweets are 16.0%.

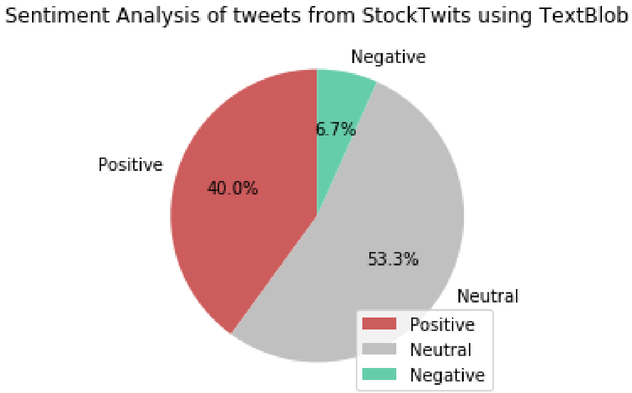


Figure 6: Sentiment Analysis of tweets from StockTwits using TextBlob

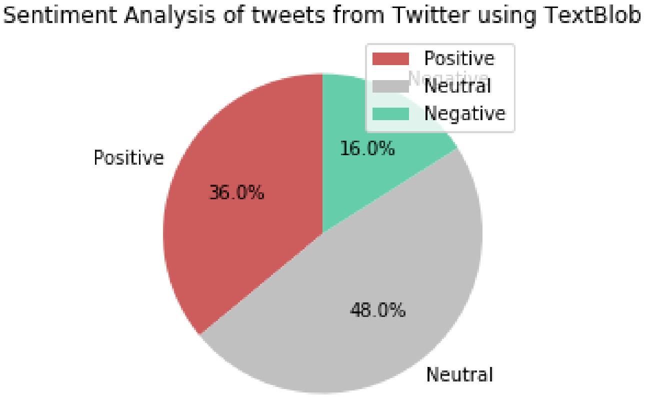


Figure 7: Sentiment Analysis of tweets from Twitter using TextBlob

3.2.2 Pre – processing

When posting a tweet, most users tend to use numbers, punctuations, URLs or special symbols so as to make their posts more interactive to the public. However, for our study, we preprocessed and cleaned our data in order our prediction to be more accurate. Particularly, before conducting the sentiment analysis, we removed some certain symbols like @, $, URLs, extra spaces and punctuations because they don’t add any value in the sentiment analysis.

After collecting the data, we standardized the sentiment score around the mean to be equal to 0 with standard deviation of 1. Importing the library “StandardScaler” from sklearn.preprocessing, we make sure that data is consistent and each data point has the same range and variability. Then, we observed that our data contains outliers which we tried to remove. We removed the data that are extremely positive or negative in order to mitigate any bias. In our research, we used as flooring the 10th percentile for low values and as the capping the 90th percentile for the higher values. We created some boxplots to detect the outliers and display the distribution of data. As it shown in Figure 8, there are some extremely positive and negative values, which were removed. Particularly, out of the 7.440 data points, the 5.406 ones were remained.

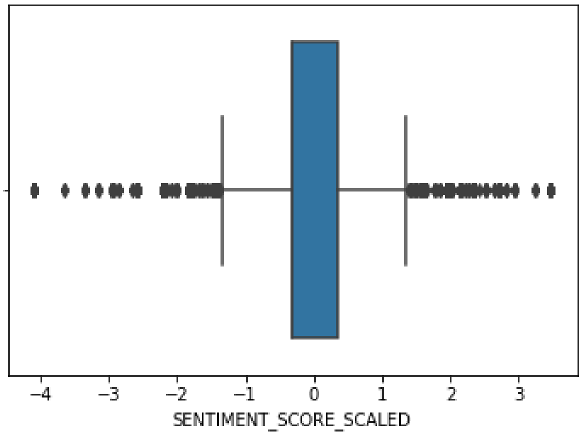


Figure 8: Boxplot for StockTwits using TextBlob

In Figure 9, we observe some extremely negative values in StockTwits data when using the VADER. Out of the 7.440 data points, we kept the 5.427 ones for our model. In addition, we can observe that distribution of the data points using the VADER is ranged between the [-2, 2] values. However, when using the TextBlob, the distribution of data points is ranged between [-1, 1].

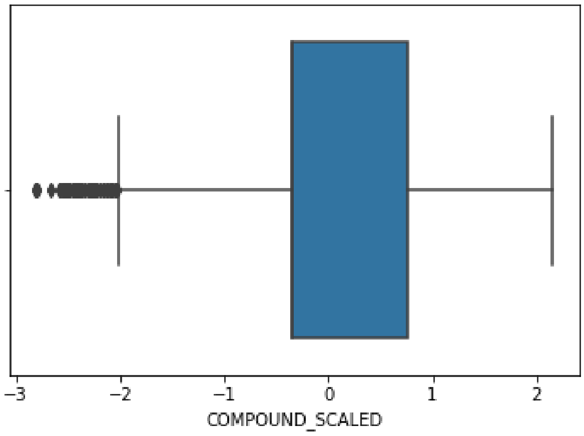


Figure 9: Boxplot for StockTwits using VADER

In Figure 10, we notice that there are some extremely negative and positive values in the Twitter data when using TextBlob. Here, the distribution of data is ranged between [-2, 2]. After removing the outliers, 81.338 data points were remained out of 90.000.

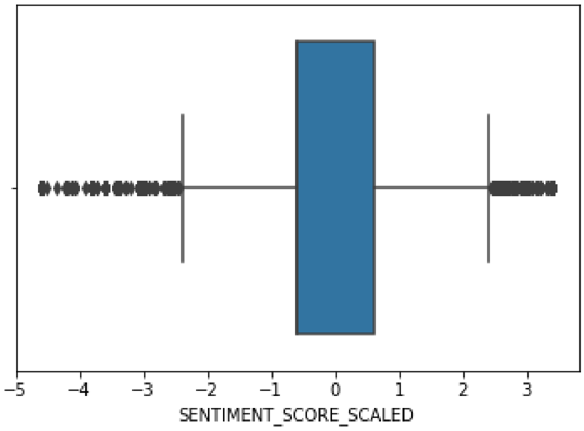


Figure 10: Boxplot for Twitter using TextBlob

In Figure 11, there are only negative points as outliers in the Twitter data when using the VADER analysis. Generally, we observe that when using the Textblob there are both positive and negative data points as outliers, but when using the VADER, only negative outliers exist. After removing the outliers, 72.241 data points remained out of 90.000.

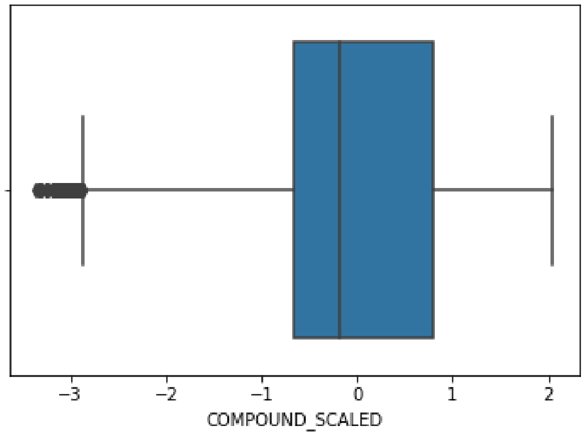


Figure 11: Boxplot for Twitter using VADER

After removing the outliers, we calculated the mean of the sentiment value for each day. Because when downloading tweets, we gather multiple sentiment values for one day. However, for our purposes, we need one sentiment value which represents the sentiment for each day. In total, we have collected 107 trading days. However, it was not possible to collect tweets for all the days, so the missing sentiment was replaced with the mean value. In addition, the stock data is missing for weekends or whenever the stock market is off. Thus, to replace the missing values, we used a function which linearly interpolates between known data with a view to obtaining the unknown values. Linear interpolation is a method for estimating unknown values that appear to be between the known values.

We also created a new variable called “Stock Change”, in order to decide whether the stock price will increased or decreased. To make this decision, the close price was subtracted from the open price and then divided by the open price, as it shown below.

$$\text{Stock Change} = \frac{Close - Open}{Open}$$

If the result of the stock change is greater than zero means that the stock change is positive, therefore the stock price is decreased and the person can buy Microsoft’s securities. We indicate this result equal to 1. Otherwise, if the result of stock change is negative, the stock price is increased and the person can sell the stock in order to earn profit. This result is indicated equal with -1. The Figure 12 depicts how the close and open price is distributed through the time. The red line represents the open price and the green line represents the close price. In general, the open price is very close to the close price with small differences. Important to mention that at the end of August and at the beginning of September, we observed high open and close prices in relationship with the other periods of time.

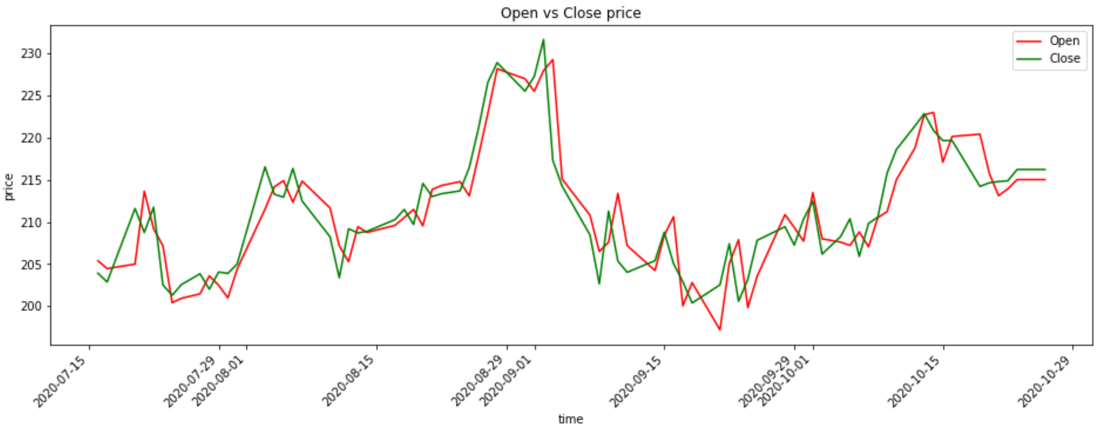


Figure 12: Microsoft's close vs open though the time

Finally, we have concatenated the Twitter data and stock data and respectively the StockTwits data with stock data. In total, we have 4 cases resulting from the use of Twitter data and stock data with TextBlob and VADER analysis. Likewise, the use of StockTwits data and stock data with TextBlob and VADER, too. For all the four cases, we have created a dataframe with two columns including the sentiment whose values range from [-1, 1] and the stock change which has two distinct values like -1 and 1. The index of this dataframe is the trading date. Having figured out final dataset, now we can build and train our classifiers for predicting the Microsoft’s stock market.

Chapter 4

4 Results

After collecting and preprocessing the data and developing the methodology, the next paramount step is to assess how good and appropriate is our model for predicting the Microsoft's stock prices. In this chapter, we will train our model and give an overview of our data through some plots. Afterwards, we will test our model's ability to predict stock prices by using some metrics, for instance F-score and Area Under the Curve (AUC). At the end of this chapter, we will forecast the fluctuation of Microsoft's stock market for each ML model.

4.1 Testing

In our study, a binary classification is implemented because it shows better results than a continuous one (Nabipour et al., 2020). In particular, the input variable is the sentiment value and the target variable is the stock change which has two distinct values including -1 and 1. In other words, it contains two values which represent the status of selling or buying the stock, respectively. In Figure 13 to Figure 16, we can see the distribution of the values of the target variable for both StockTwits and Twitter and for the two sentiment analysis tools, TextBlob and VADER. In the four cases, the number of the buying status is greater than the selling. This means that most people tend to buy the Microsoft's stocks and few of them to sell them.

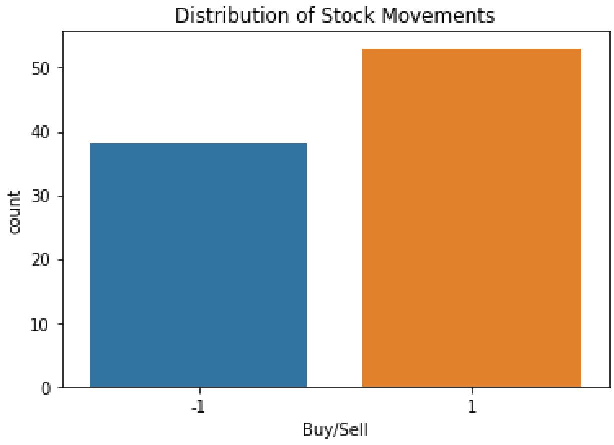


Figure 13: The distribution of stock movements using StockTwits with TextBlob

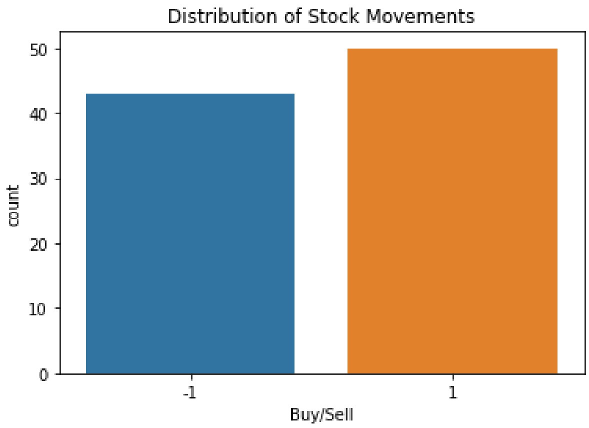


Figure 14: The distribution of stock movements using StockTwits with VADER

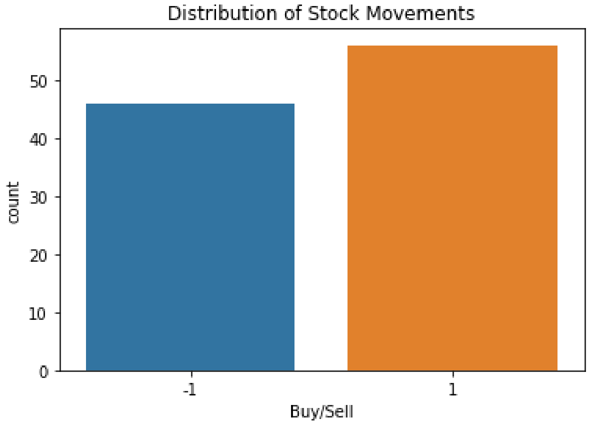


Figure 15: The distribution of stock movements using Twitter with TextBlob

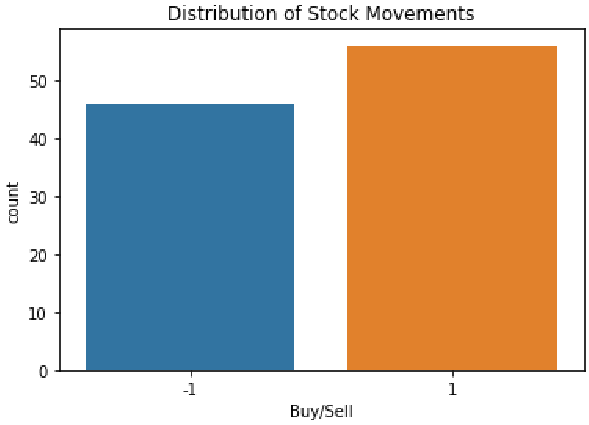


Figure 16: The distribution of stock movements using Twitter with VADER

In Figure 17 to Figure 20, the input and target variables are depicted through the time. The blue line represent the target variable which is the stock change and the red line is the input variable, otherwise the public’s sentiment about the Microsoft. In particular, in Figure 17, it is noticed that from July to August, the two lines are related as when the sentiment is positive, there is a rise in stock change. In other words, when the sentiment is positive, people tend to buy the Microsoft's stocks. From September to October, it is observed the same as in the previous period but with some deviations.

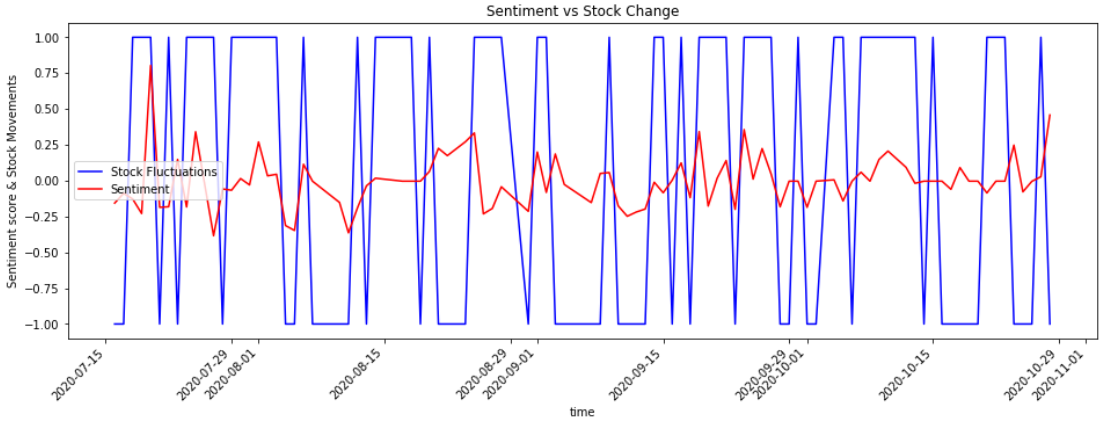


Figure 17: Sentiment & Stock Change through time using StockTwits with TextBlob

On the other hand, in Figure 18, we cannot observe a clear pattern in August. However, in July, it seems that the sentiment is related with the stock change. When the sentiment is positive, then there is a rise in stock change, vice a versa. In September and October, the two lines are related for some days as there are a lot of deviations.

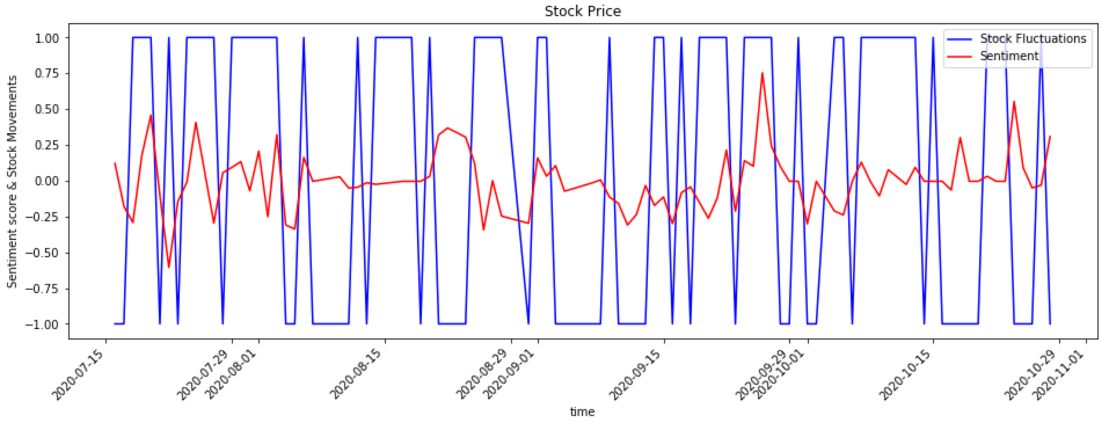


Figure 18: Sentiment & Stock Change through time using StockTwits with VADER

However, when Twitter data is used and analyzed with the TextBlob and VADER, as it shown in Figure 19 and Figure 20, the sentiment of most of the tweets are neutral. There are a lot of ups and downs which are really close to 0. This means that the use of twitter data may not be reliable information for predicting stock prices as the sentiment and the stock change are not related.

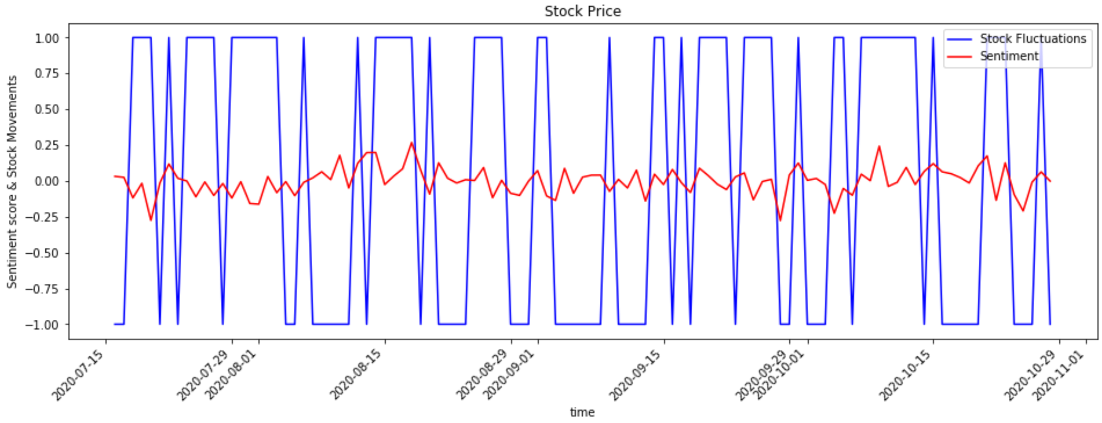


Figure 19: Sentiment & Stock Change through time using Twitter with TextBlob

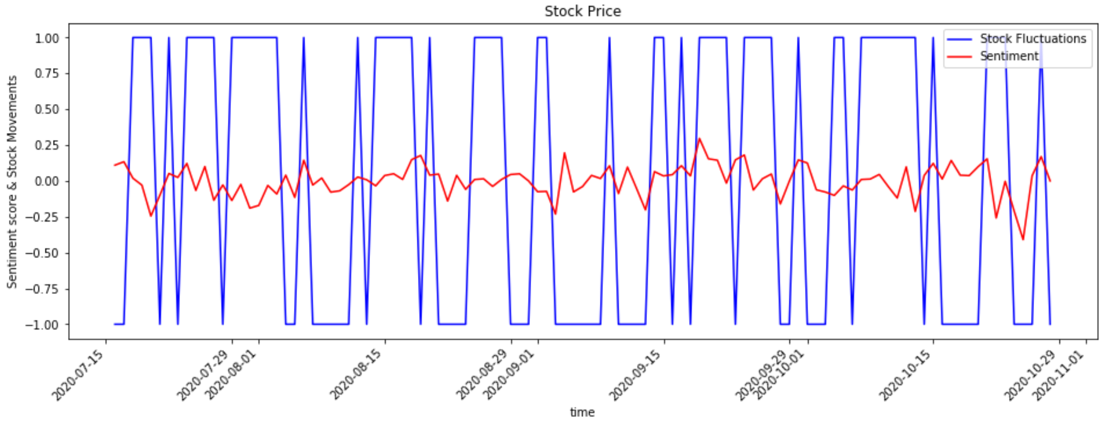


Figure 20: Sentiment & Stock Change through time using Twitter with VADER

After defining the input and target variables and their relationship, it is important to train our model and then to proceed to the testing. To test our model, we splitted our data into train set and test set. Particularly, 20% of our data is used for testing and the rest of 80% is used for training the model.

In the next section, we present the results of our research for each ML model and we evaluate our results using f - score and AUC.

4.2 Model Performance

In this section, we present the results for each model. This section will be divided into 4 subsections in order to perform our results by using StockTwits and Twitter with TextBlob and VADER as sentiment analysis tools. In our study, we apply two metrics including F-score and AUC area as they are the appropriate metrics for imbalanced data (Gurav et al, 2018).

4.2.1 StockTwits with TextBlob

In this case, we utilize financial data from Yahoo in combination with StockTwits data. The sentiment analysis of StockTwits data was performed with the use of TextBlob. Our model was trained and tested through 7 ML algorithms including the KNN, SVM, Logistic Regression, Naïve Bayes, Decision Tree, Random Forest and MLP. In Figure 21, the f-scores are depicted for each model. The f-score demonstrates the model's accuracy or how correct our predictions are (Derczynski, 2016). In Figure 22, we show the ROC (Receiver Operating Characteristic) curves and the areas under the curves (AUC) for each model, too. In y-axis, we have the true positive rate and in x-axis, we have the false positive rate which both have values in the range [0, 1]. So, the AUC is the area under the curve of plot true positive rate and false positive rate. In other words, it describes the discriminatory power, how capable is the model to distinguish if the stock price will rise or fall. The greater the AUC, the better the performance (Bradley, 1997). As it is presented in Figure 21 and Figure 22, the f-scores of our ML models range from 53.8% to 66.7%, while the AUC areas range from 40% to 50%. Due to our need to predict both the positive and negative raise of Microsoft's stock price, we care equally about our true positives and true negatives. Thus, we take into consideration the f-score and the AUC area as evaluation metrics which are fundamentally different. The model with the greatest f-score, when using StockTwits with TextBlob as sentiment analysis tool, is the SVM and Naïve Bayes with AUC area equal to 50%. Similarly, the Decision Tree and Random Forest demonstrated a good performance with f-score equal to 61.5% and 50% AUC area.

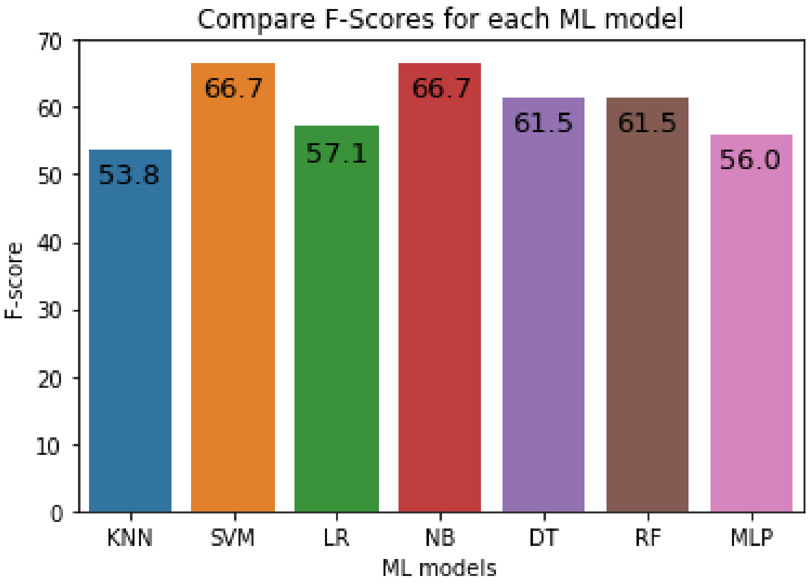


Figure 21: F-scores using StockTwits with TextBlob

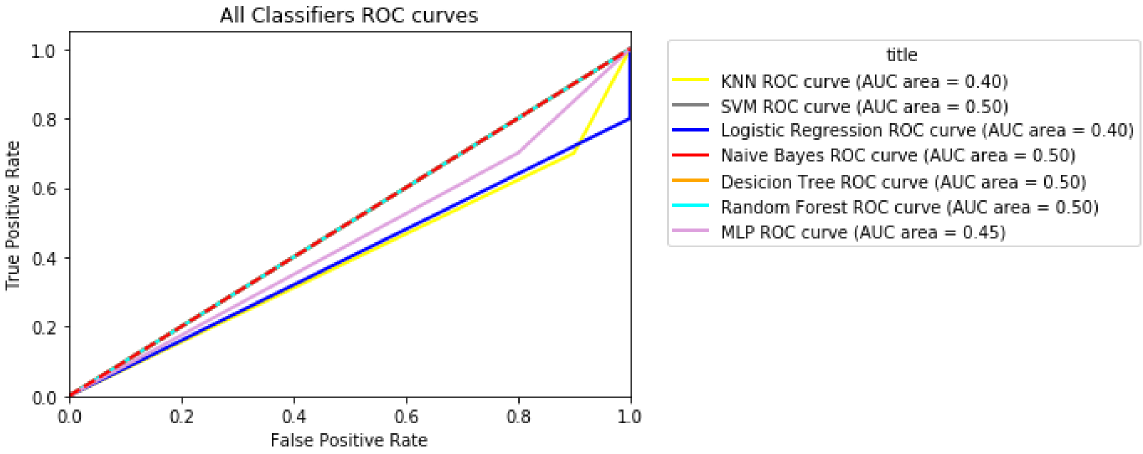


Figure 22: ROC curves using StockTwits with TextBlob

4.2.2 StockTwits with VADER

In this subsection, our data consists of financial data and StockTwits data. Here, the sentiment of StockTwits data is analyzed with the VADER. In Figure 23 and Figure 24, we observe that the range of f-scores is fluctuated from 45.5% to 66.7%, while the AUC areas are ranged from 40% to 50%. Four ML algorithms have the best f-scores including the SVM, Logistic Regression, Naïve Bayes and MLP equal to 66.7%. Their AUC areas is 50%, respectively. The next best ML algorithm is Random Forest which presents f-score equal to 58.8%. The Random Forest covers 45% of the AUC.

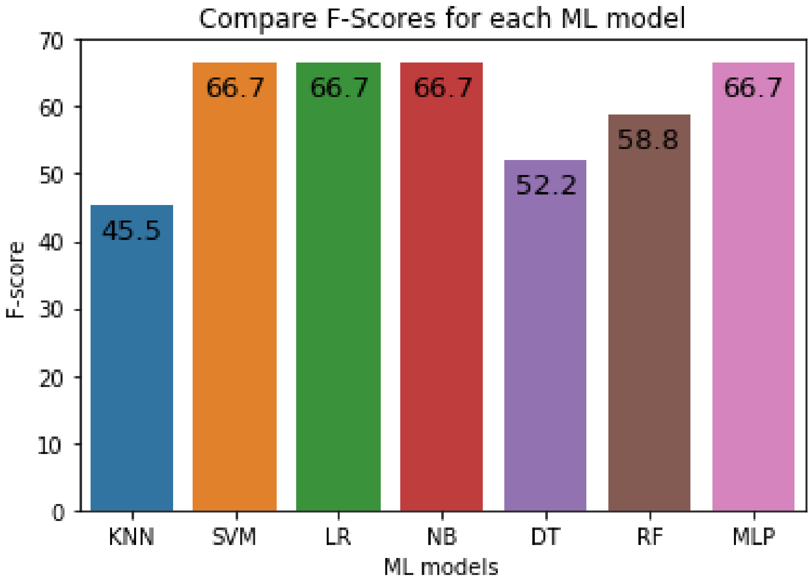


Figure 23: F-scores using StockTwits with VADER

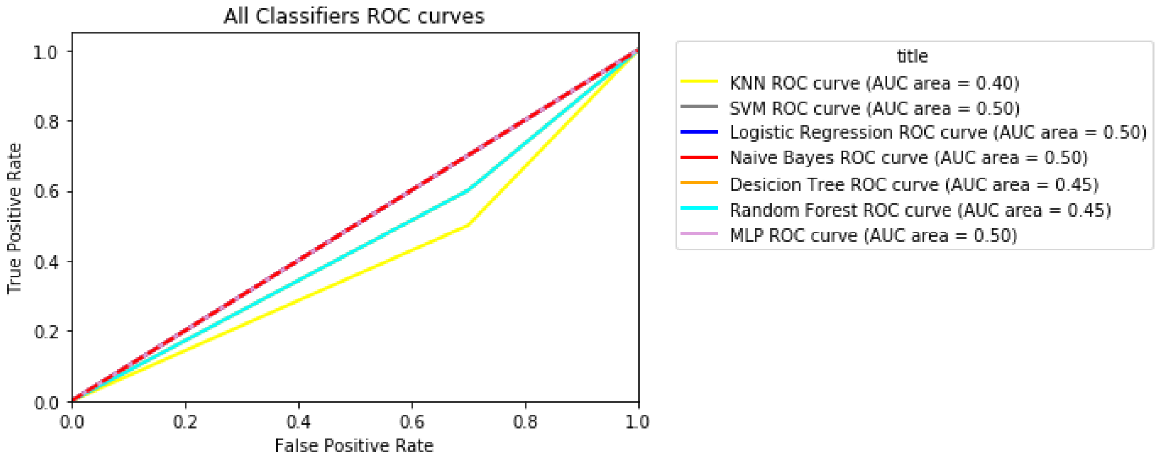


Figure 24: ROC curves using StockTwits with VADER

4.2.3 Twitter with TextBlob

In this case, we make use of financial and Twitter data and we apply the TextBlob as sentiment analysis tool. As it is shown in Figure 25, the f-scores are varied from 53.8% to 74.3%. The greatest f-score seems to have Naïve Bayes and MLP with 74.3% and KNN and Decision Tree with 72.0%. On the other hand, some AUC areas increases up to 50%. Generally, they range from 44% to 68%. KNN and Decision Tree have the best AUC metrics, so far, compared to the other cases. In particular, their AUC values are equal to 68%, respectively. This means that KNN and Decision Tree present the best discriminatory power. The Naïve Bayes algorithm is missing from Figure 25 because the f-score is equal to 0 and therefore there is no need to test further the f-score for this algorithm.

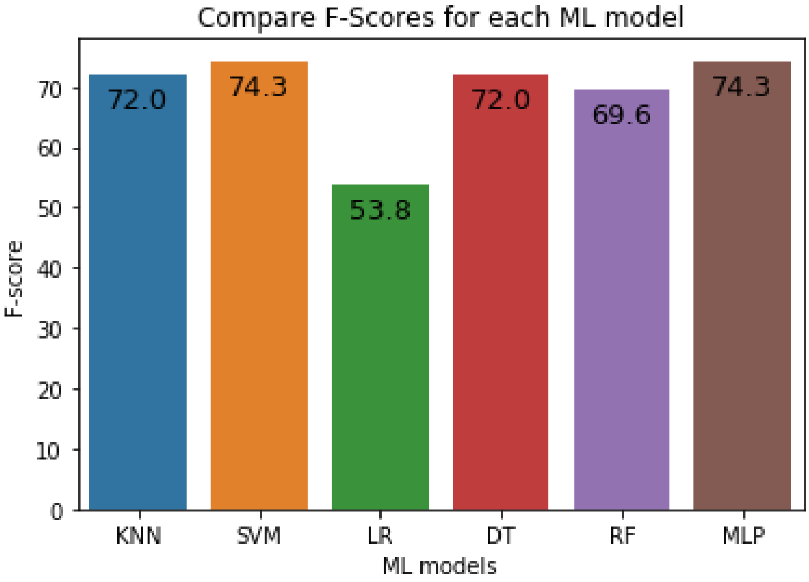


Figure 25: F-scores using Twitter with TextBlob

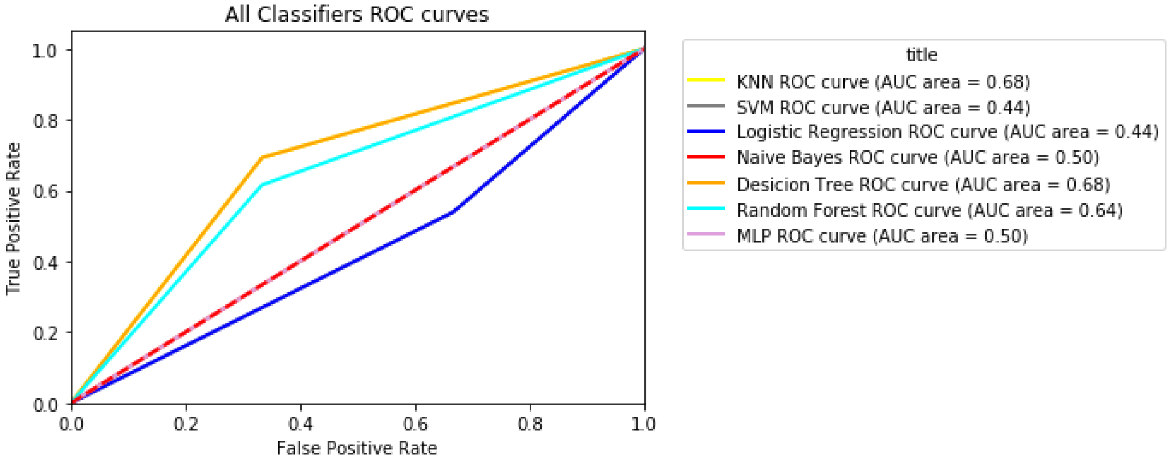


Figure 26: ROC curves using StockTwits with VADER

4.2.4 Twitter with VADER

In this subsection, financial and Twitter data are used. The sentiment analysis of twitter data was performed with use of VADER. As it is depicted in Figure 27, the f-scores were figured out from 23.5% to 75.9%. The best performance seems to have SVM with 75.9% while the second place takes Logistic Regression with 69.2%. However, in Figure 28, we notice that the AUC areas fluctuate from 38% to 65%. SVM achieves the greatest discriminatory power with 65% and the second one does the Logistic Regression with 62%. The Naïve Bayes algorithm is missing from figure 25 because the f-score is equal to 0 and therefore there is no need to test further the f-score for this algorithm.

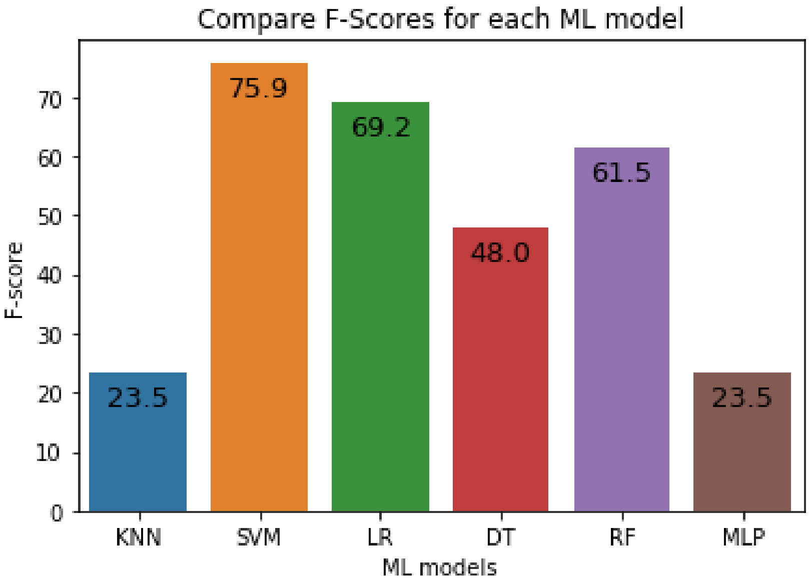


Figure 27: F-scores using Twitter with VADER

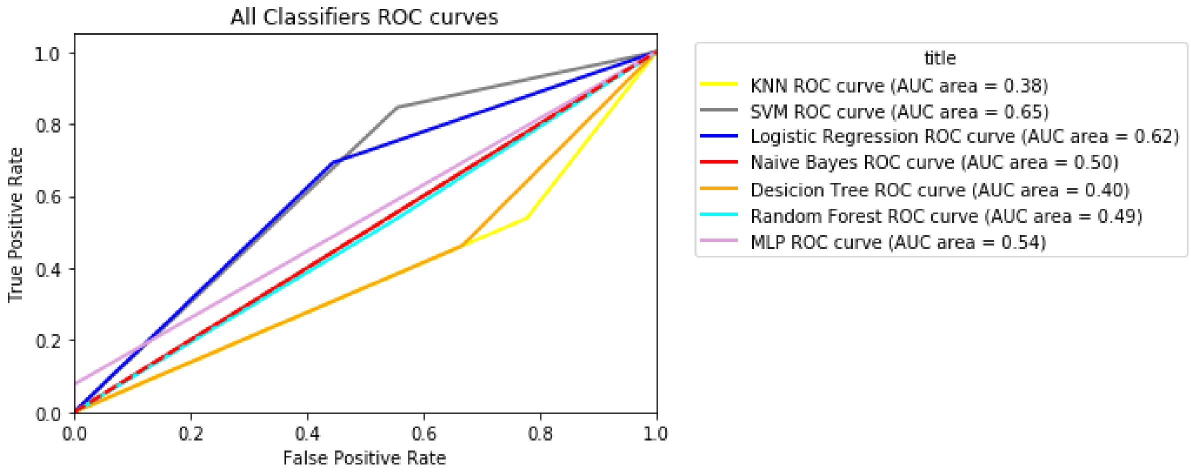


Figure 28: ROC curves using Twitter with VADER

4.3 Predictions

In this section, we present the predictions of the Microsoft's stock movements for each model. This section will be divided into 4 subsections in order to present our Microsoft's stock predictions by using StockTwits and Twitter with TextBlob and VADER as sentiment analysis tools.

In Appendix, the plots of predictions are shown for each case and for each ML algorithm during the period from 12-10-2020 until 31-10-2020, for about 20 days. The x-axis contains the dates while the y-axis contains both actual and predicted prices. The blue color, in y-axis, represents the actual price. However, the pink color represents the predicted price. When there is not both actual and predicted price in one day, it means that the predicted price is equal to the actual one. Otherwise, the predicted price is different from the actual one.

4.3.1 StockTwits with TextBlob

Figure 29, for KNN algorithm, shows that the predicted stock prices is increased for 17 out of 20 days, while in 28-12-2020, 29-12-2020 and 31-12-2020, the stock prices decrease. On the other hand, the actual prices have a decrease trend for 10 days. The actual prices were predicted correctly for 8 out of 20 days, in which the 7 days have an increasing trend.

Figure 30 which refers to SVM and Figure 35 to MLP, the 10 days demonstrate that the Microsoft's actual stock prices decrease. However, the predictions show that the stock prices will rise for all days. Also, for 10 out of 20 days, the stock prices are predicted to be increased correctly.

The Logistic Regression and Naïve Bayes, in Figure 31 and Figure 32, depict that there are 10 days in which the stock prices increase and the other 10 days decrease. In Logistic Regression, the prediction was successful for 8 stock market days in which the price rise positively while in Naïve Bayes, the successful prediction was for 10 days which also has an increasing trend.

According to Decision Tree and Random Forest, in Figure 33 and Figure 34, the 12 out of 20 stock market days, Microsoft's stock prices increase while there are 8 days which have a decreasing trend. The prediction was successful for 10 days, in which two of them show a negative movement especially in 17-10-2020 and 28-10-2020.

4.3.2 StockTwits with VADER

In Figure 36 which refers to the KNN algorithm, the stock market fluctuations drops for 10 days out of 20. The prediction worked well for 8 days, in which the stock prices are predicted to be up for 5 days.

According to the SVM, Logistic Regression, Naïve Bayes and MLP, in Figure 37, Figure 38, Figure 39 and Figure 42 respectively, the stock prices move positively for 10 out of 20 days. The prediction was correct for 10 days in which forecasts that the Microsoft's stock prices will go up.

In Figure 40 and in Figure 41, when applying the Decision Tree and Random Forest algorithms, the stock prices increase for 10 days. It predicts successfully that the prices will rise up for 6 days while they will be down for the other 3 days.

4.3.3 Twitter with TextBlob

In KNN, in Figure 43, the stock prices tend to move up for 10 days, in which the prediction is successful for 9 out of 10 days. For SVM, in Figure 44, the prices have an increasing trend for 13 days in which the prediction is correct for all the 13 days. In addition, when applying Logistic Regression, in Figure 45, the actual prices rise up for 12 days. The algorithm predicts correctly for 10 days, in which the 7 days show that the prices will increase and the other 3 days the prices will decrease. In Naïve Bayes, in Figure 46, the actual prices are going down for 9 days. The predictions for these days are correct and the stock prices will fall. In Decision Tree, in Figure 47, the prediction is correct for 15 days in which the stock market prices will increase for 9 days and the other 6 days will decrease. Similarly, the Random Forest, in Figure 47, forecasts correctly for 16 days in which the stock prices goes up for 10 days. Last but not least, the MLP, in Figure 48, predicts successfully that the stock prices will rise up for 13 days.

4.3.4 Twitter with VADER

According to the Figure 49, the actual prices of the KNN algorithm go down for 8 days and the prediction correctly shows that the prices will fall for those 8 days. In SVM, in Figure 50, the actual prices of the Microsoft's stocks are going down for 9 days in which the 4 days are predicted correctly that the prices will go down. The Figure 51 depicts that the stock market prices are moving down for 9 days and the Logistic Regression algorithm predicts correctly that the prices will drop for 5 out of 9 days. Regarding Naïve Bayes, Decision Tree and Random Forest, in Figure 52, Figure 53 and Figure 54 respectively, the actual prices show an increasing trend for 13 days in which the 9 days are predicted correctly. And the MLP, in Figure 55, demonstrates that the actual prices move up for 13 days and the correct predicted prices are increasing for 12 days.

However, the results produced in subsections 4.2 and 4.3 may be invalid and not reliable for two reasons. Firstly, Python libraries including TextBlob and VADER calculate wrong sentiment score for many tweets. For instance, they usually perceive a negative tweet as positive and vice a versa, in consequence of having an invalid training set. In addition, another threat to the validity of our results is spam accounts, fake accounts and bots which are involved in our Twitter and StockTwits data. Hence, the spread of misleading information and therefore a wrong-calculated sentiment is possible to be included in our dataset. Moreover, it is observed, in our dataset, that for a particular date, the sentiment is positive and the stock movement is going down. The existence of no relation between the sentiment and the stock movement is possible due to the two prementioned reasons.

Chapter 5

5 Conclusions and Future Work

5.1 Conclusions

In this thesis, we investigated the issue of stock market prediction using ML methods with sentiment analysis. We dealt with Twitter and StockTwits data in combination with financial data. The sentiment of the microblogging data was extracted with the help of two Python libraries, TextBlob and VADER. All tweets and stock tweets were stored in MySQL Database. After proceeding with the appropriate preprocessing steps, we evaluated our model performance, in each of the 4 cases, using seven ML algorithms including KNN, SVM, Logistic Regression, Naïve Bayes, Decision Tree, Random Forest and MLP. The evaluation of the model was performed with two metrics, the f-score and AUC.

In the case of using StockTwits data with TextBlob as sentiment analysis tool, SVM and Naïve Bayes perform the best f-score (66.7%) and AUC equal to 50%. The prediction of these two algorithms is successful for 10 days in which the stock prices increase.

In addition, when using StockTwits with VADER, SVM, Logistic Regression, Naïve Bayes and MLP result in the best f-score with 66.7% as well as the best predictive power as the prediction was correct for 10 days in which forecast that Microsoft stock prices will go up. Also, these algorithms perform one of the top AUC rate (50%), which means that they can discriminate effectively the increase and decrease in Microsoft's stock price.

In the case of using Twitter with TextBlob, SVM and MLP achieves the greatest f-score (74.3%) and it predicts successfully for 13 days. However, Random Forest with 69.6% f-score seems to perform the most correct predictions as it forecasts correctly for 16 days in which the stock price goes up for 10 days. Furthermore, KNN and Decision Tree have the greatest discriminatory power with 68% and f-score equal to 72%, while MLP presents AUC with 50% and SVM presents 44%.

Lastly, in the case of using Twitter with VADER, SVM results in the best f-score (75.9%) and the greatest discriminatory power (65%). Indeed, SVM presents the most correct predictions for 15 days in which the predicted prices increase for 10 days.

To sum up, the use of Twitter data, and particularly with the use of VADER as sentiment analysis tool, produces the greatest predictive and discriminatory power. This means that our model can correctly predict the true positive and true negative points. In terms of f-score, SVM presents the best performance (75.9%) while in terms of AUC, it performs the best discriminatory power (65%). Finally, Microsoft stock price movements are predicted correctly to go up for most days.

5.2 Future Work

There are several aspects of our research we could improve in the future. Due to the limited time, we cannot claim that our research is accurate enough for the intended use. Further steps need to be considered for our research improvement. Firstly, we could only collect tweets from twitter users who have a lot of followers because they have potentially great influence on Microsoft stock. In addition, we could exclude fake twitter accounts as they mislead the right calculation of the sentiment. Moreover, more data and more trading dates could potentially improve our model performance. Lastly, it would be very important if we applied a ML algorithm such as Naïve Bayes for calculating the sentiment score. It would be more valid if we trained and then tested our twitter data rather than using a library. Many times, the TextBlob and VADER perceives for example a positive comment as a negative thus producing a wrong sentiment score. So, training and testing our data will help to improve our model.

References

Ajekwe, C. C., Ibiamke, A., & Haruna, H. A. (2017). Testing the random walk theory in the Nigerian stock market. *IRA-International Journal of Management & Social Sciences*, *6*(3), 500-508.

Alam, S. (2017). Testing the weak form of efficient market in cryptocurrency. *Journal of engineering and applied sciences*, *12*(9), 2285-2288.

Attigeri, G. V., MM, M. P., Pai, R. M., & Nayak, A. (2015, November). Stock market prediction: A big data approach. In *TENCON 2015-2015 IEEE Region 10 Conference* (pp. 1-5). IEEE.

Batra, R., & Daudpota, S. M. (2018, March). Integrating StockTwits with sentiment analysis for better prediction of stock price movement. In *2018 International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)* (pp. 1-5). IEEE.

Beleveslis, D., Tjortjis, C., Psaradelis, D., & Nikoglou, D. (2019, September). A Hybrid Method for Sentiment Analysis of Election Related Tweets. In *2019 4th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM)* (pp. 1-6). IEEE.

Billah, M., Waheed, S., & Hanifa, A. (2016, December). Stock market prediction using an improved training algorithm of neural network. In *2016 2nd International Conference on Electrical, Computer & Telecommunication Engineering (ICECTE)* (pp. 1-4). IEEE.

Bohn, T. A. (2017). Improving long term stock market prediction with text analysis.

Bradley, A. P. (1997). The use of the area under the ROC curve in the evaluation of machine learning algorithms. Pattern recognition, 30(7), 1145-1159.

Choudhry, R., & Garg, K. (2008). A hybrid machine learning system for stock market forecasting. *World Academy of Science, Engineering and Technology*, *39*(3), 315-318.

Deepak, R. S., Uday, S. I., & Malathi, D. (2017). Machine learning approach in stock market prediction. *International Journal of Pure and Applied Mathematics*, *115*(8), 71-77.

Derczynski, L. (2016, May). Complementarity, F-score, and NLP Evaluation. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)* (pp. 261-266).

Fakhry, B. (2016). A literature review of the efficient market hypothesis. *Turkish Economic Review*, *3*(3), 431-442.

Falinouss, P. (2007). Stock trend prediction using news articles: a text mining approach.

Fama, E. F. (1960). *Efficient market hypothesis* (Doctoral dissertation, Ph. D. dissertation, University of Chicago, Graduate School of Business).

Gilbert, C. H. E., & Hutto, E. (2014, June). Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *Eighth International Conference on Weblogs and Social Media (ICWSM-14). Available at (20/04/16) http://comp. social. gatech. edu/papers/icwsm14.vader. hutto.pdf* (Vol. 81, p. 82).

Gurav, U., & Sidnal, N. (2018). Predict Stock Market Behavior: Role of Machine Learning Algorithms. In *Intelligent Computing and Information and Communication* (pp. 383-394). Springer, Singapore.

Gurjar, M., Naik, P., Mujumdar, G., & Vaidya, T. (2018). Stock market prediction using ANN. *International Research Journal of Engineering and Technology (IRJET)*, *5*(03).

Hamed, A. R., Qiu, R., & Li, D. (2015, December). Analysis of the relationship between Saudi twitter posts and the Saudi stock market. In *2015 IEEE Seventh International Conference on Intelligent Computing and Information Systems (ICICIS)* (pp. 660-665). IEEE.

Huang, Y. (2019). Machine Learning for Stock Prediction Based on Fundamental Analysis.

Kordonis, J., Symeonidis, S., & Arampatzis, A. (2016, November). Stock price forecasting via sentiment analysis on Twitter. In *Proceedings of the 20th Pan-Hellenic Conference on Informatics* (pp. 1-6).

Koukaras, P., Rousidis, D., & Tjortjis, C. (2020). Forecasting and Prevention Mechanisms Using Social Media in Health Care. In *Advanced Computational Intelligence in Healthcare-7* (pp. 121-137). Springer, Berlin, Heidelberg.

Liu, S., Liao, G., & Ding, Y. (2018, May). Stock transaction prediction modeling and analysis based on LSTM. In *2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA)* (pp. 2787-2790). IEEE.

Loria, S. (2018). textblob Documentation. *Release 0.15*, *2*.

Malkiel, B. G. (1999). *A random walk down Wall Street: including a life-cycle guide to personal investing*. WW Norton & Company.

Mittal, A., & Goel, A. (2012). Stock prediction using twitter sentiment analysis. *Standford University, CS229 (2011 http://cs229. stanford. edu/proj2011/GoelMittal-StockMarketPredictionUsingTwitterSentimentAnalysis. pdf)*, *15*.

Nabipour, M., Nayyeri, P., Jabani, H., Shahab, S., & Mosavi, A. (2020). Predicting Stock Market Trends Using Machine Learning and Deep Learning Algorithms Via Continuous and Binary Data; a Comparative Analysis. *IEEE Access*, *8*, 150199-150212.

Namdari, A., & Li, Z. S. (2018, June). Integrating fundamental and technical analysis of stock market through multi-layer perceptron. In *2018 IEEE Technology and Engineering Management Conference (TEMSCON)* (pp. 1-6). IEEE.

Nann, S., Krauss, J., & Schoder, D. (2013). Predictive analytics on public data-the case of stock markets.

Nti, I. K., Adekoya, A. F., & Weyori, B. A. (2019). A systematic review of fundamental and technical analysis of stock market predictions. *Artificial Intelligence Review*, 1-51.

Oikonomou, L., & Tjortjis, C. (2018, September). A Method for Predicting the Winner of the USA Presidential Elections using Data extracted from Twitter. In *2018 South-Eastern European Design Automation, Computer Engineering, Computer Networks and Society Media Conference (SEEDA_CECNSM)* (pp. 1-8). IEEE.

Pagolu, V. S., Reddy, K. N., Panda, G., & Majhi, B. (2016, October). Sentiment analysis of Twitter data for predicting stock market movements. In *2016 international conference on signal processing, communication, power and embedded system (SCOPES)* (pp. 1345-1350). IEEE.

Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2015). Predicting stock and stock price index movement using trend deterministic data preparation and machine learning techniques. *Expert systems with applications*, *42*(1), 259-268.

Patel, J., Shah, S., Thakkar, P., & Kotecha, K. (2015). Predicting stock market index using fusion of machine learning techniques. *Expert Systems with Applications*, *42*(4), 2162-2172.

Picasso, A., Merello, S., Ma, Y., Oneto, L., & Cambria, E. (2019). Technical analysis and sentiment embeddings for market trend prediction. *Expert Systems with Applications*, *135*, 60-70.

Rasel, R. I., Sultana, N., & Hasan, N. (2016, October). Financial instability analysis using ANN and feature selection technique: application to stock market price prediction. In *2016 International Conference on Innovations in Science, Engineering and Technology (ICISET)* (pp. 1-4). IEEE.

Rousidis, D., Koukaras, P., & Tjortjis, C. (2020). Social media prediction: a literature review. *Multimedia Tools and Applications*, *79*(9), 6279-6311.

Serban, I. V., González, D. S., & Wu, X. (2014). Prediction of changes in the stock market using twitter and sentiment analysis.

Shah, D., Campbell, W., & Zulkernine, F. H. (2018, December). A comparative study of LSTM and DNN for stock market forecasting. In *2018 IEEE International Conference on Big Data (Big Data)* (pp. 4148-4155). IEEE.

Somani, P., Talele, S., & Sawant, S. (2014, December). Stock market prediction using hidden Markov model. In *2014 IEEE 7th Joint International Information Technology and Artificial Intelligence Conference* (pp. 89-92). IEEE.

Tsiara, E., & Tjortjis, C. (2020, June). Using Twitter to Predict Chart Position for Songs. In *IFIP International Conference on Artificial Intelligence Applications and Innovations* (pp. 62-72). Springer, Cham.

Appendix

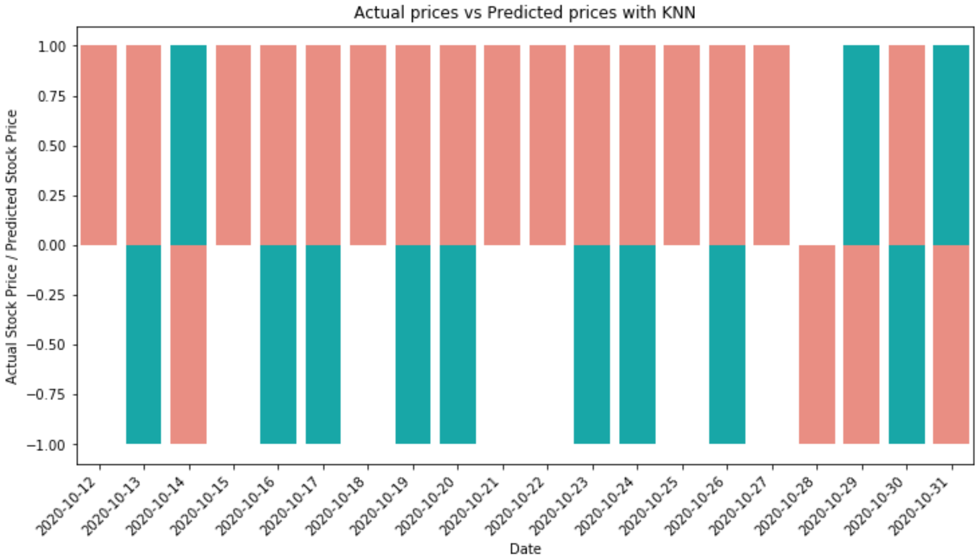


Figure 29: KNN - prediction vs actual price using StockTwits with TextBlob

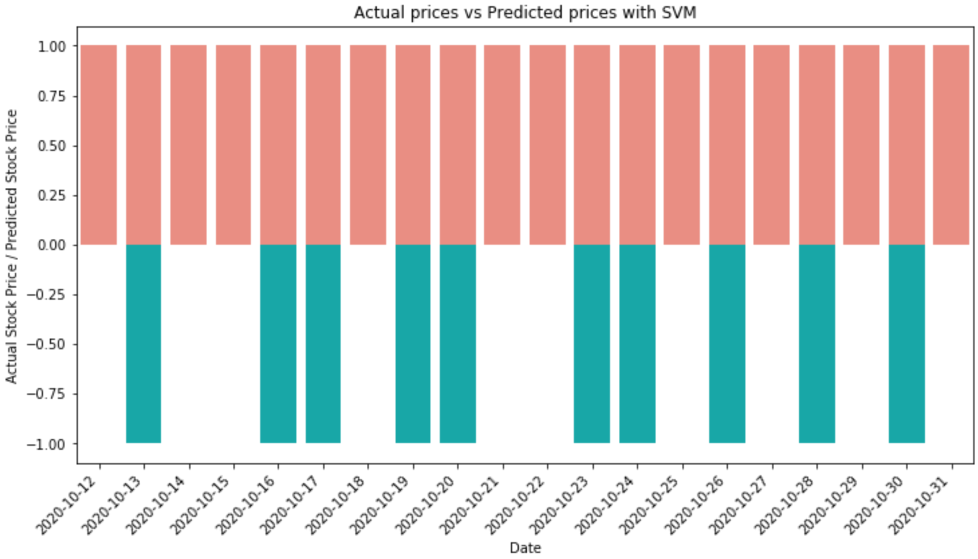


Figure 30: SVM - prediction vs actual price using StockTwits with TextBlob

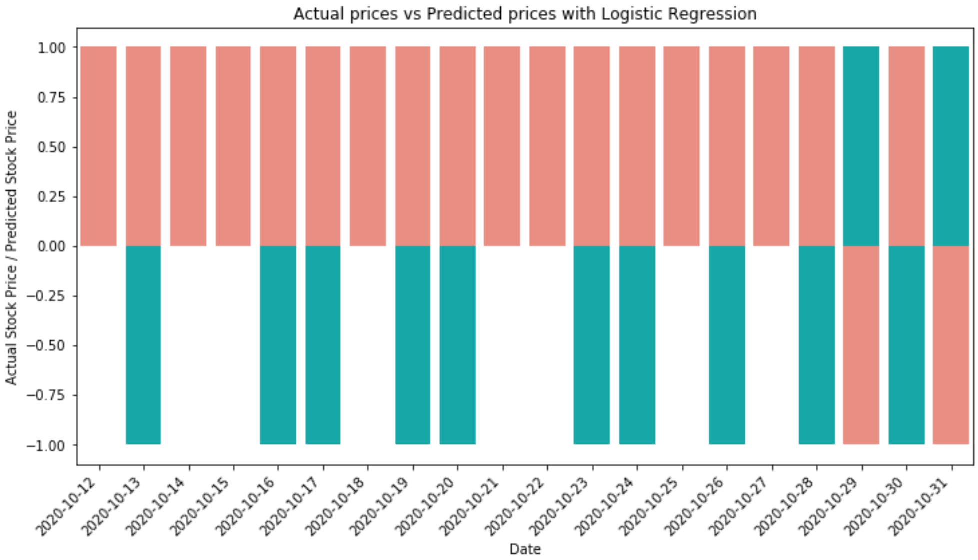


Figure 31: Logistic Regression - prediction vs actual price using StockTwits with TextBlob

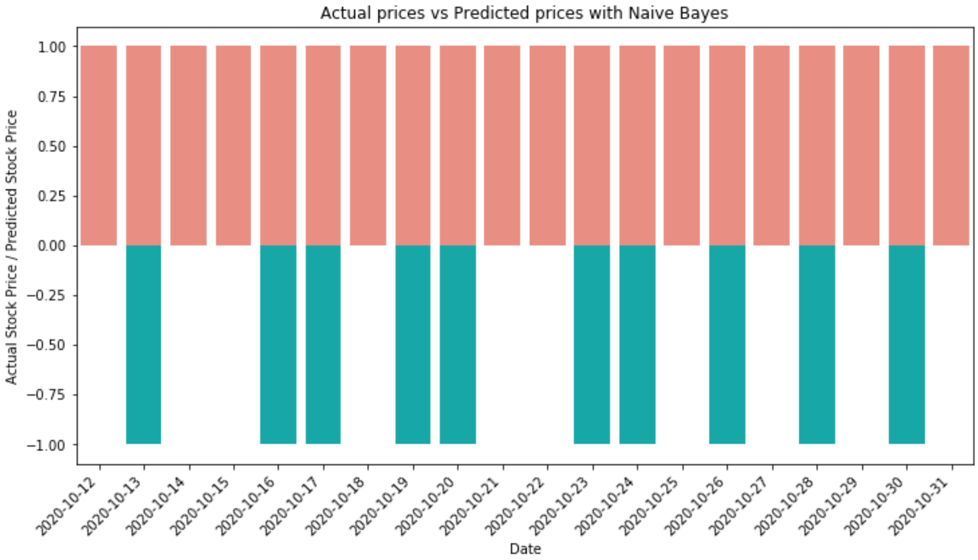


Figure 32: Naive Bayes - prediction vs actual price using StockTwits with TextBlob

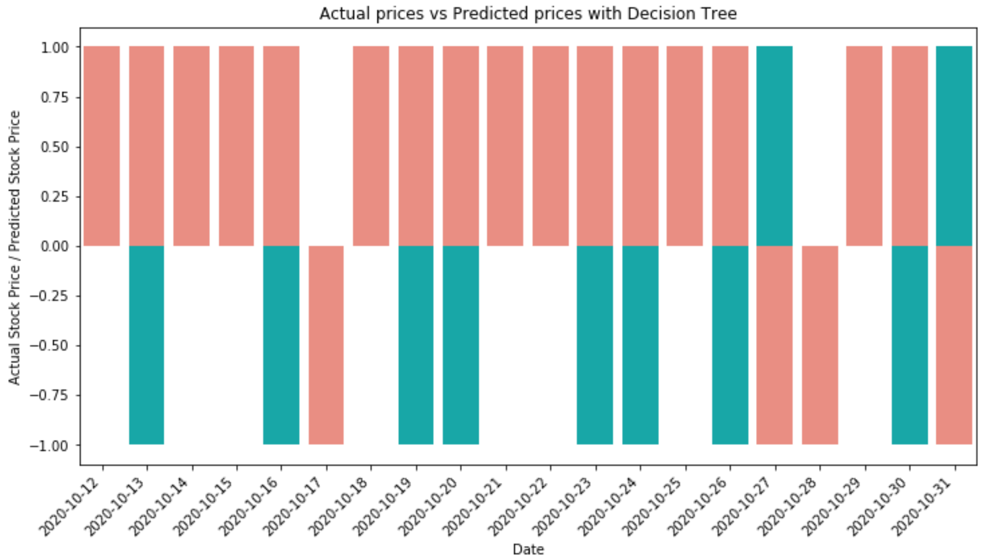


Figure 33: Decision Tree - prediction vs actual price using StockTwits with TextBlob

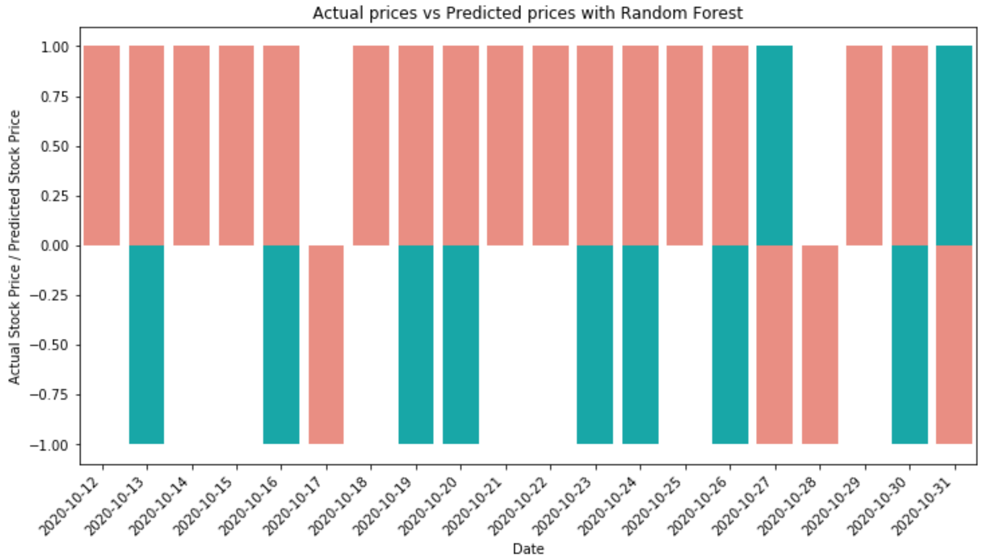


Figure 34: Random Forest - prediction vs actual price using StockTwits with TextBlob

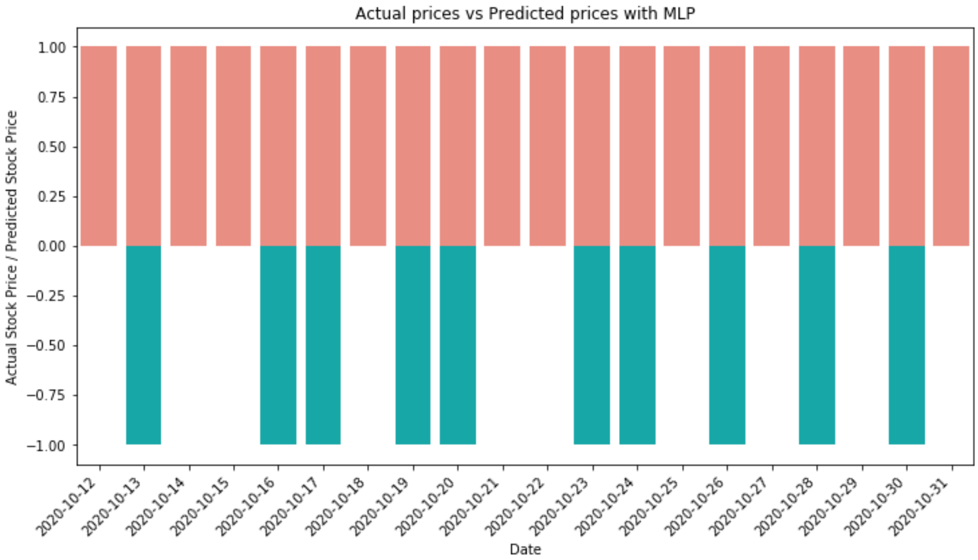


Figure 35: MLP - prediction vs actual price using StockTwits with TextBlob

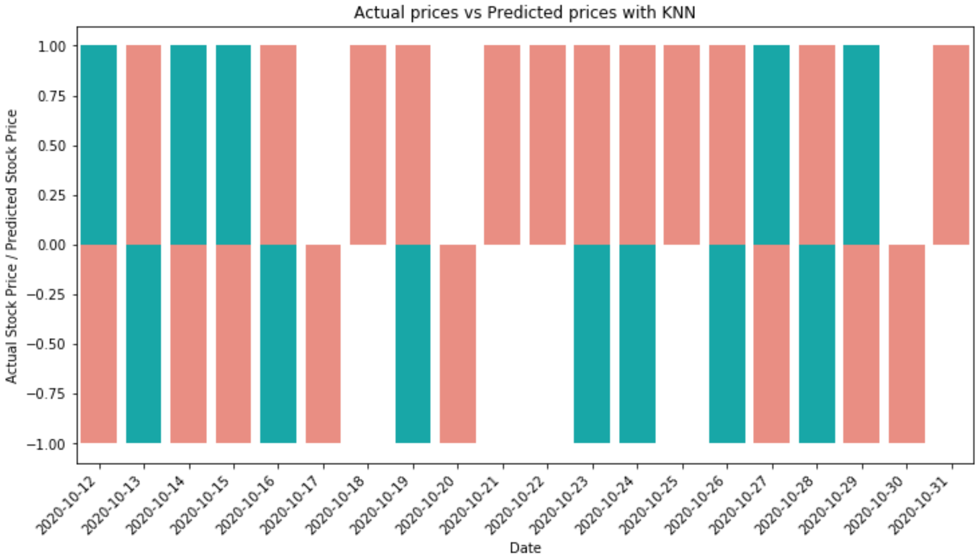


Figure 36: KNN - prediction vs actual price using StockTwits with VADER

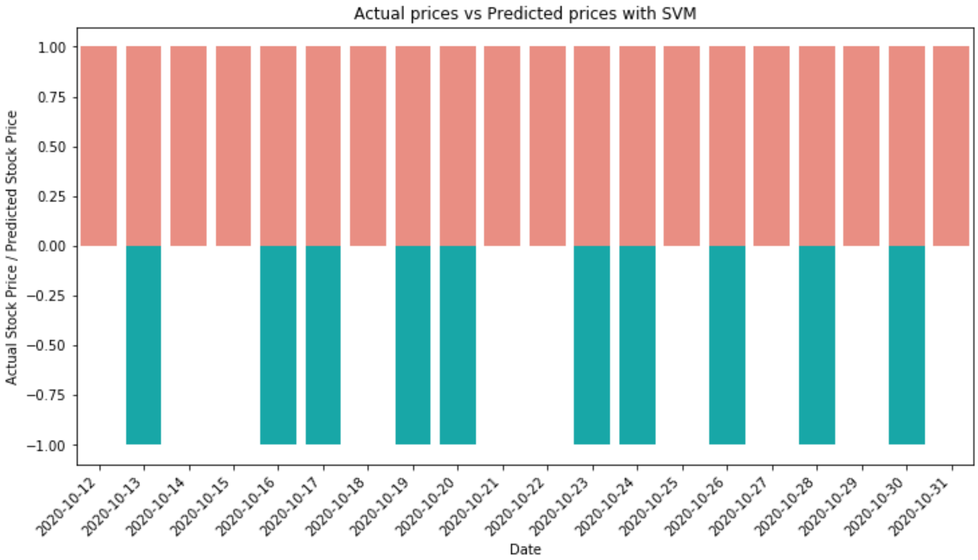


Figure 37: SVM - prediction vs actual price using StockTwits with VADER

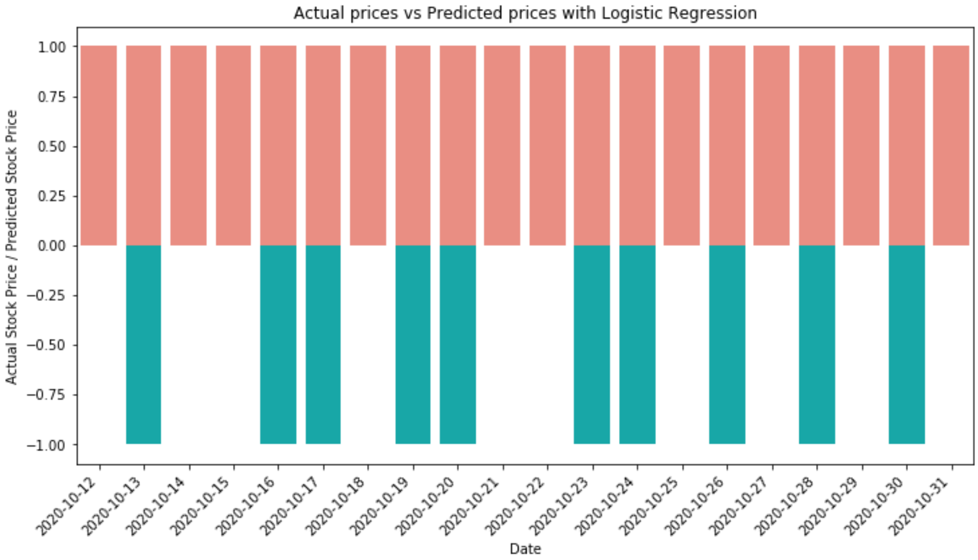


Figure 38: Logistic Regression - prediction vs actual price using StockTwits with VADER

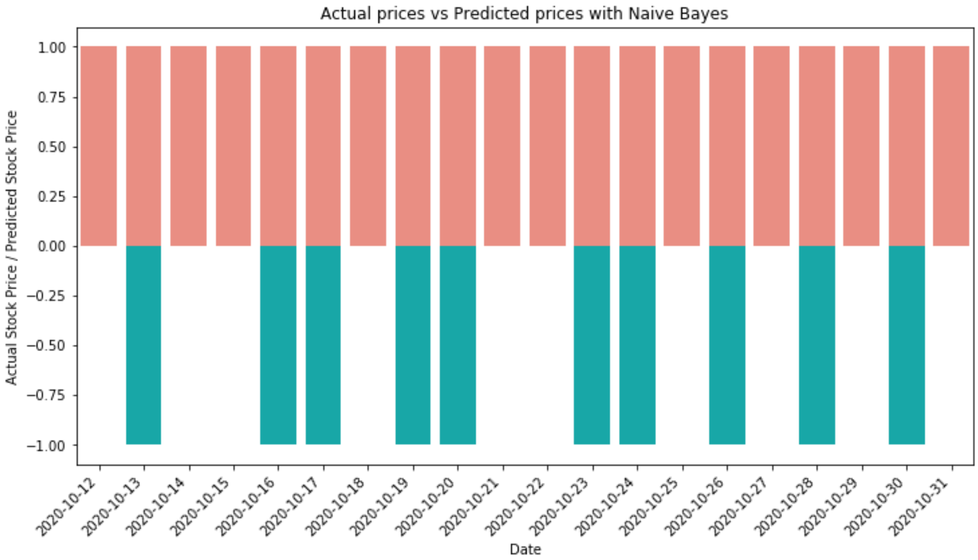


Figure 39: Naïve Bayes - prediction vs actual price using StockTwits with VADER

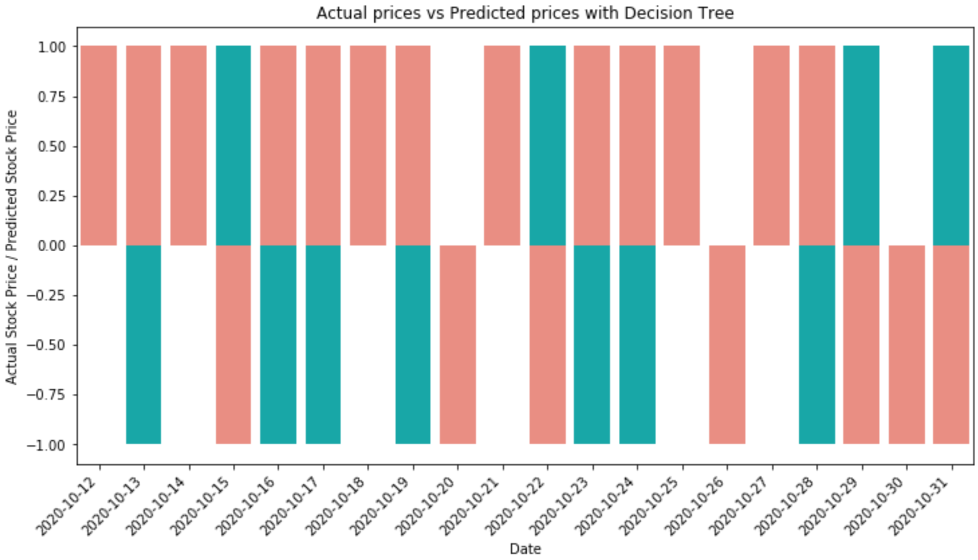


Figure 40: Decision Tree - prediction vs actual price using StockTwits with VADER

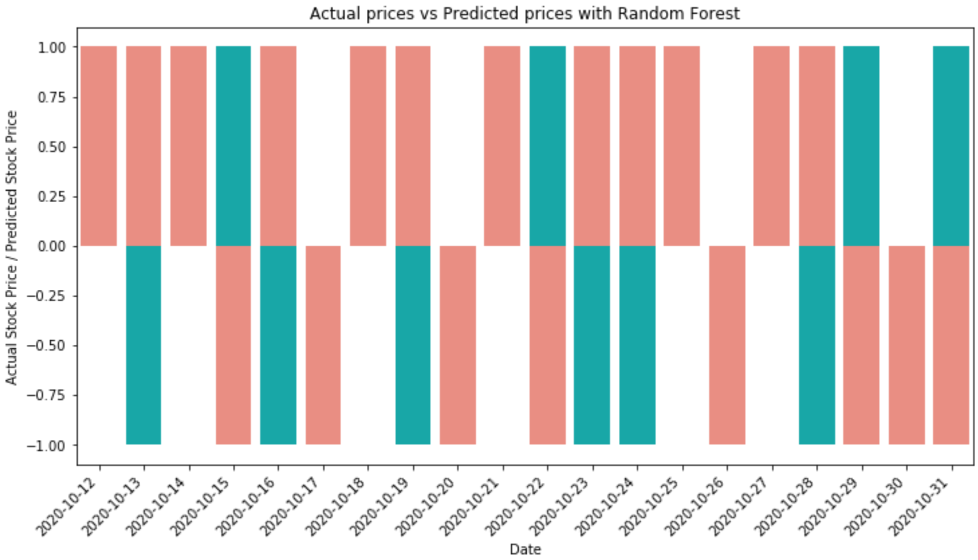


Figure 41: Random Forest - prediction vs actual price using StockTwits with VADER

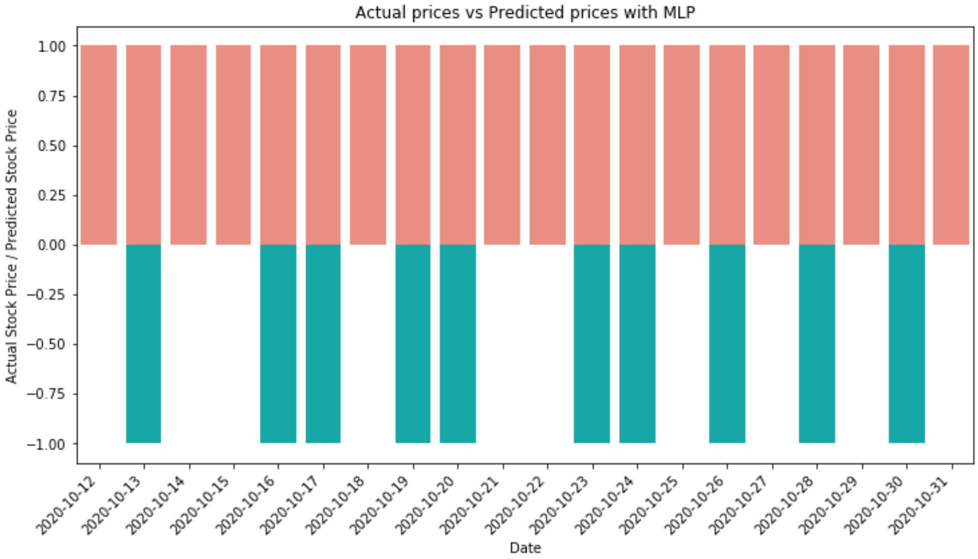


Figure 42: MLP - prediction vs actual price using StockTwits with VADER

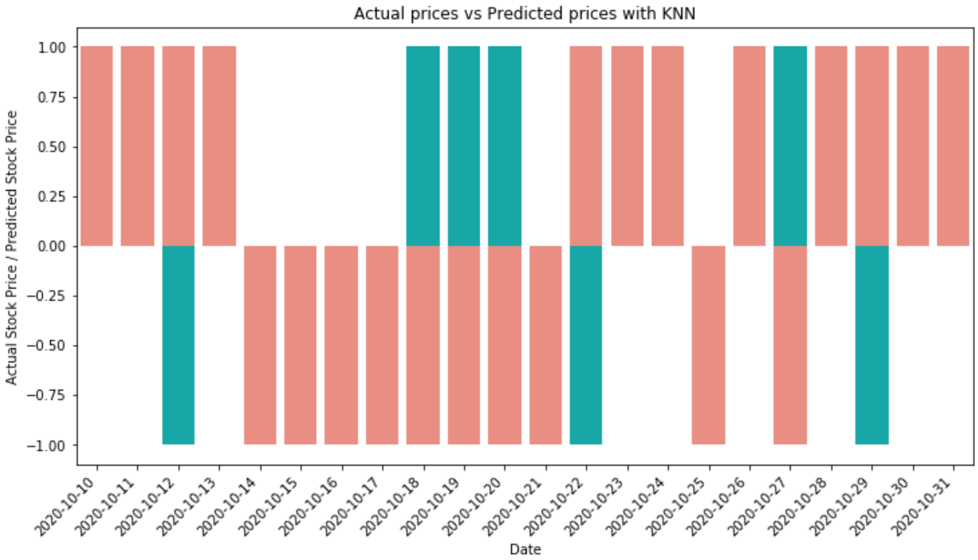


Figure 43: KNN - prediction vs actual price using Twitter with TextBlob

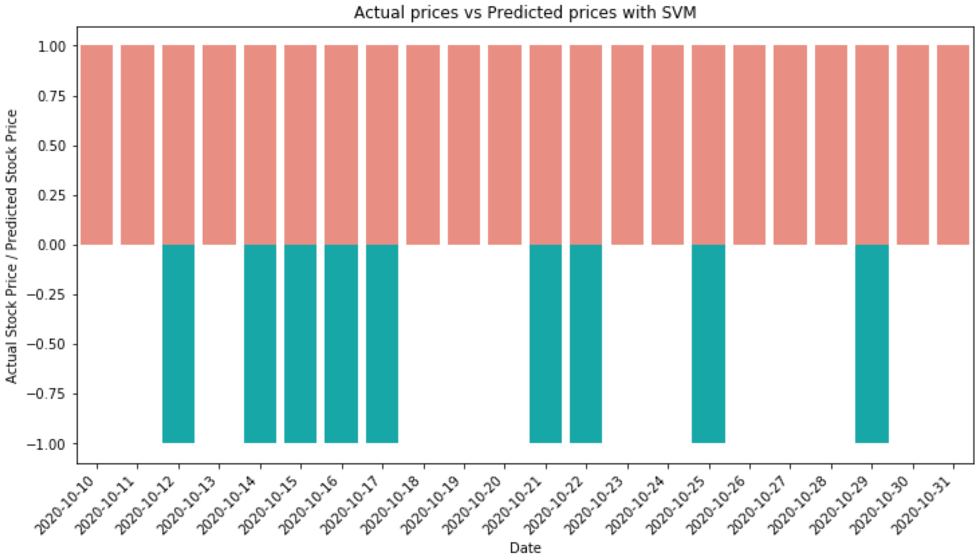


Figure 44: SVM - prediction vs actual price using Twitter with TextBlob

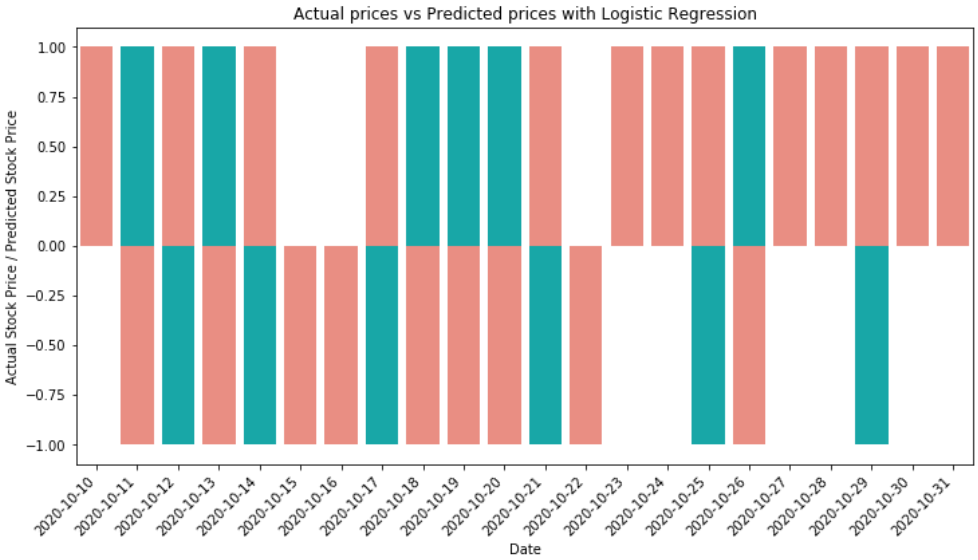


Figure 45: Logistic Regression - prediction vs actual price using Twitter with TextBlob

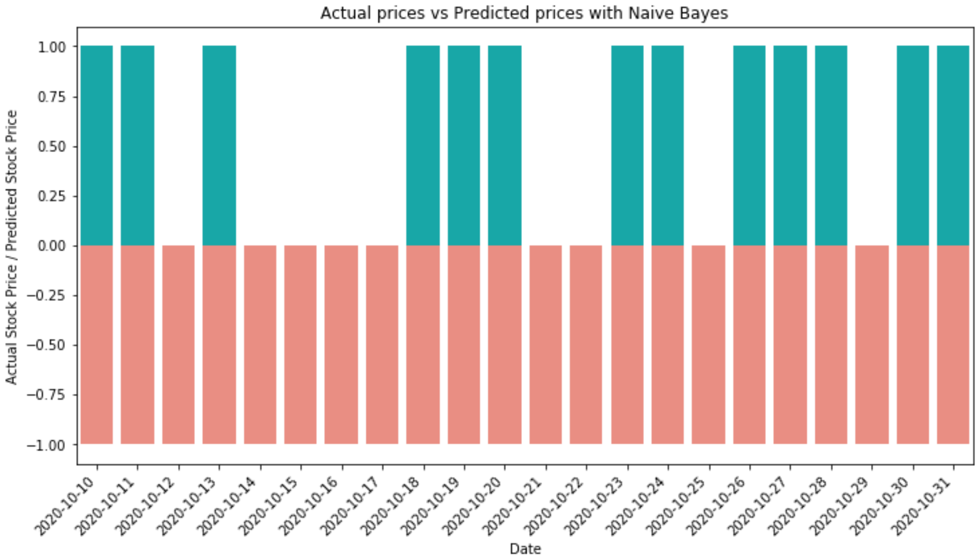


Figure 46: Naive Bayes - prediction vs actual price using Twitter with TextBlob

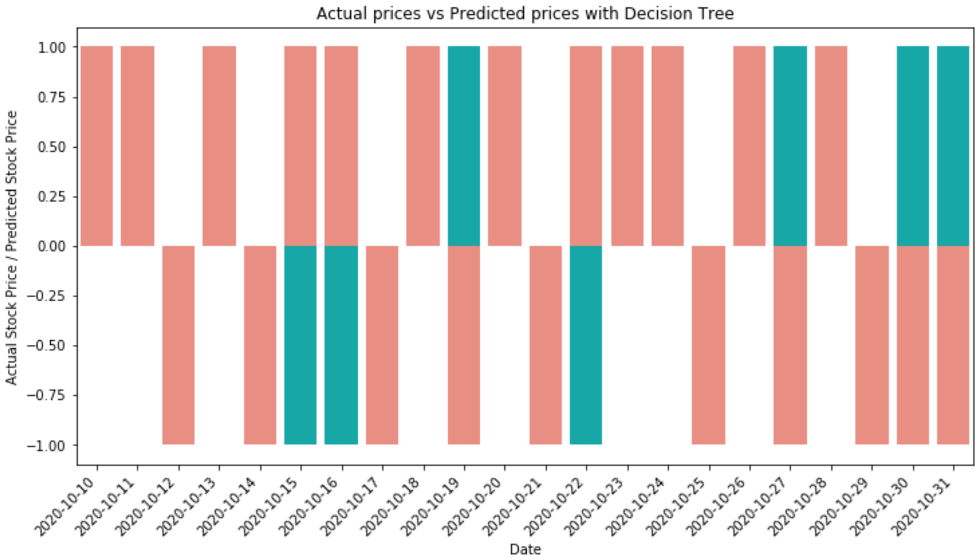


Figure 47: Decision Tree - prediction vs actual price using Twitter with TextBlob

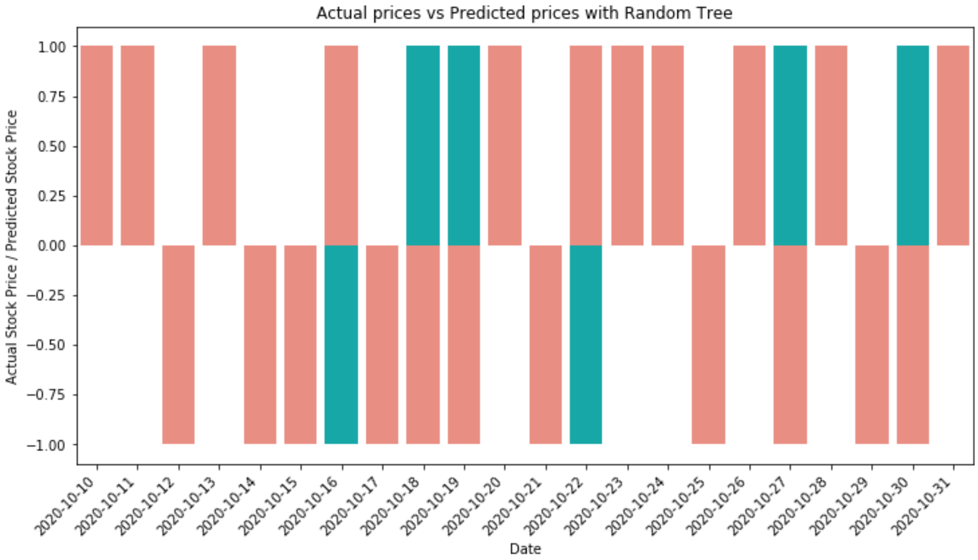


Figure 48: Random Forest - prediction vs actual price using Twitter with TextBlob

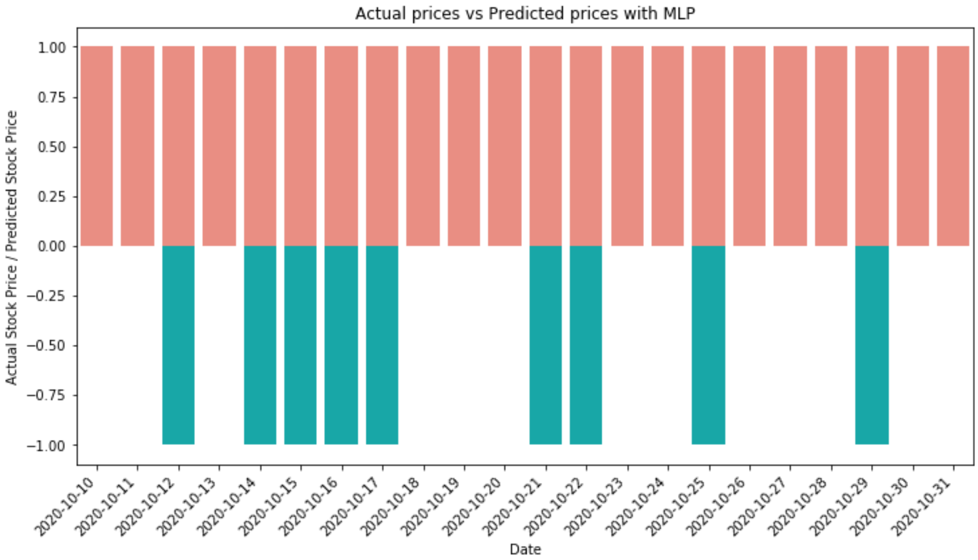


Figure 49: MLP - prediction vs actual price using Twitter with TextBlob

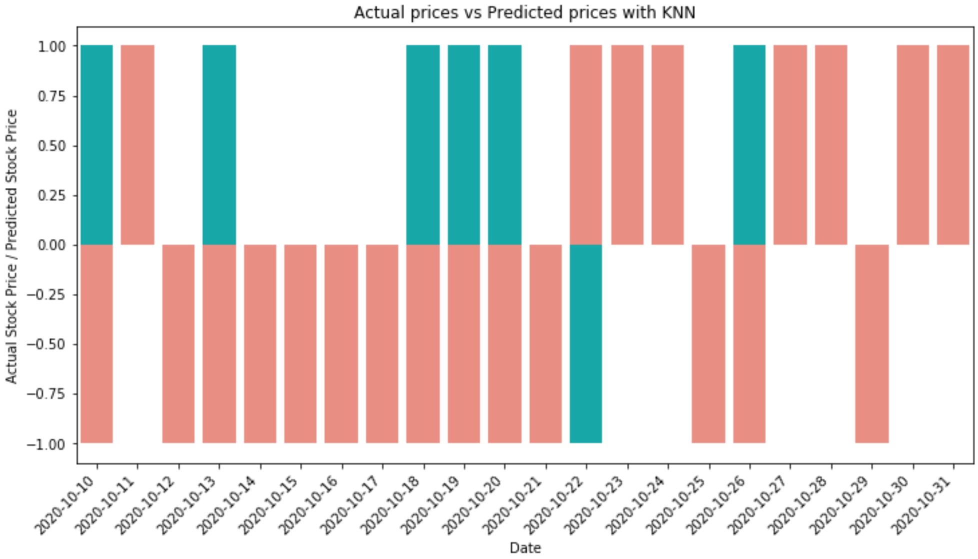


Figure 50: KNN - prediction vs actual price using Twitter with VADER

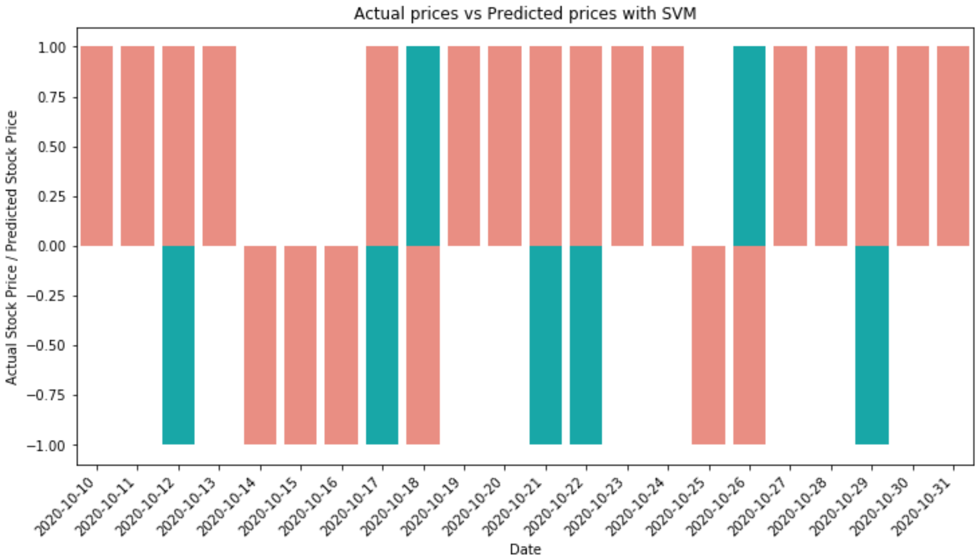


Figure 51: SVM - prediction vs actual price using Twitter with VADER

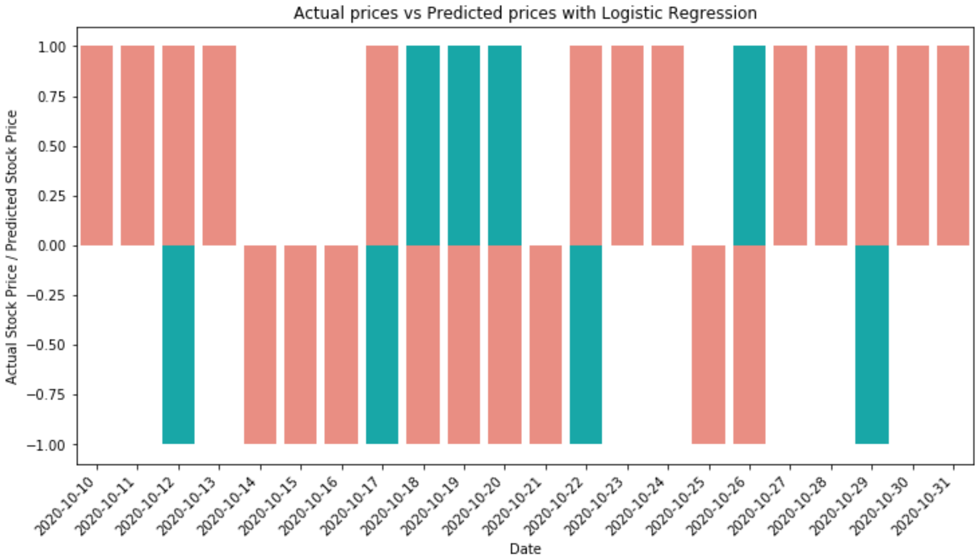


Figure 52: Logistic Regression - prediction vs actual price using Twitter with VADER

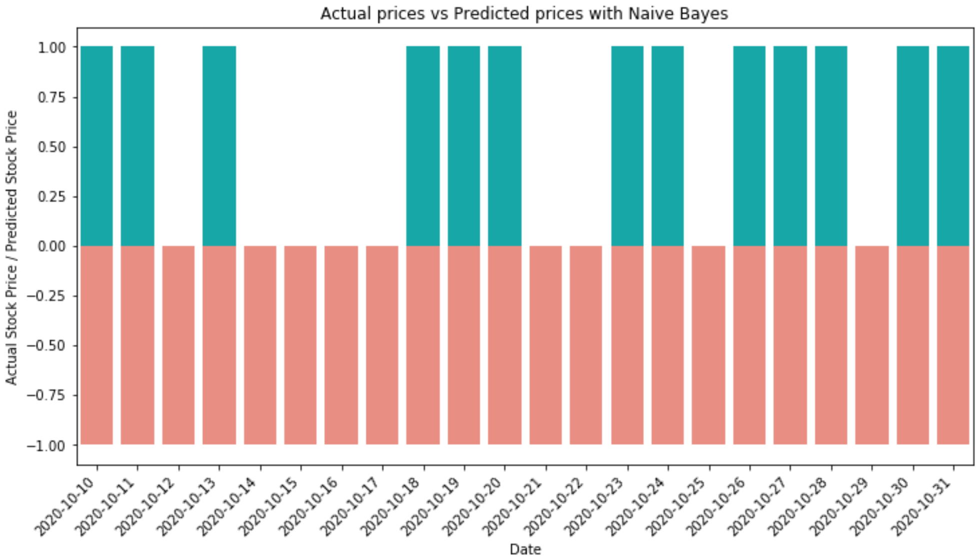


Figure 53: Naive Bayes - prediction vs actual price using Twitter with VADER

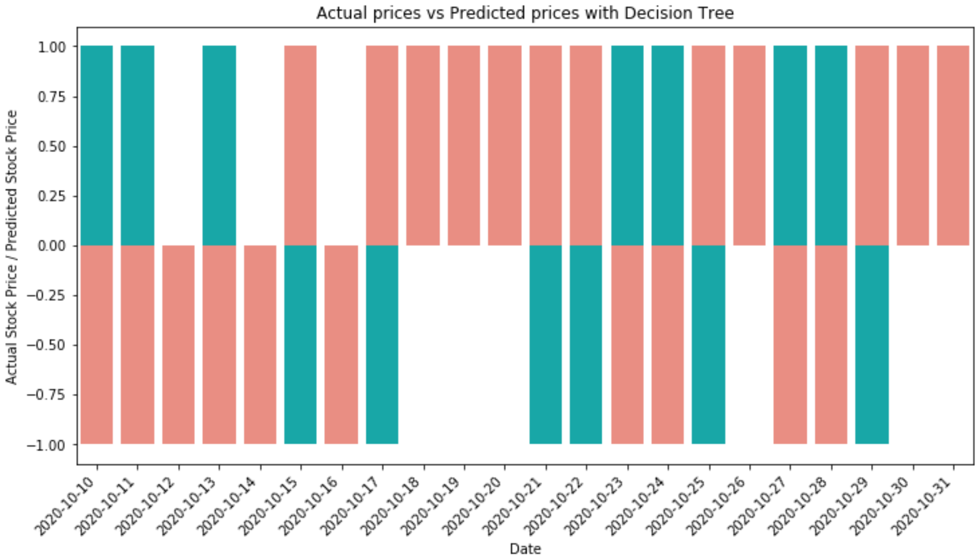


Figure 54: Decision Tree - prediction vs actual price using Twitter with VADER

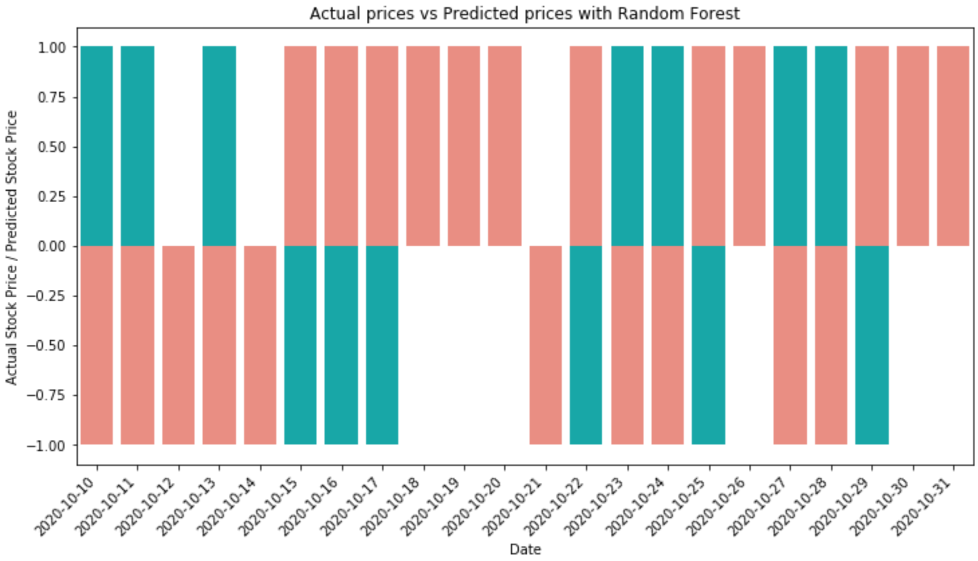


Figure 55: Random Forest - prediction vs actual price using Twitter with VADER

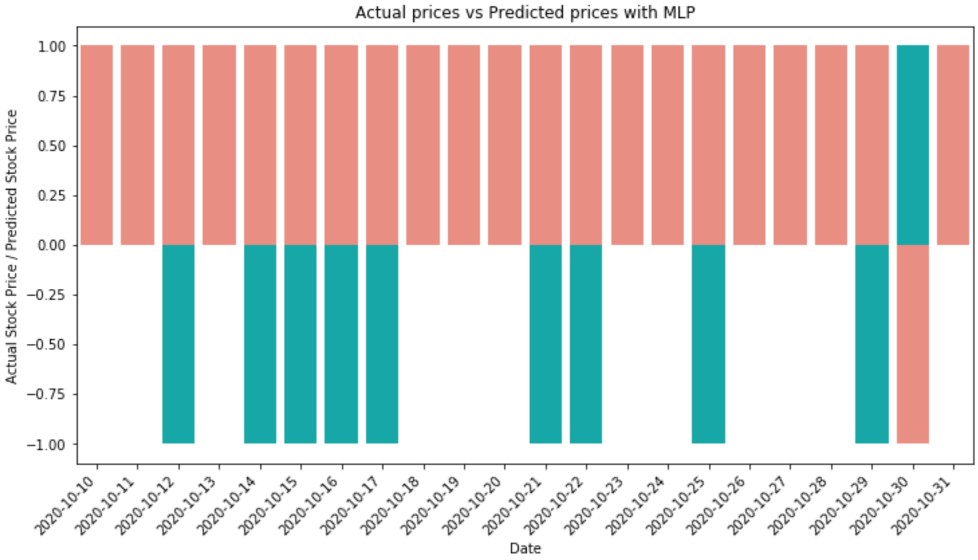


Figure 56: MLP - prediction vs actual price using Twitter with VADER