Abstract

This project aims to explore if the analysis of the information posted online can have any value for the stock forecasting models. The data which was used in the report was taken from different verified accounts from Twitter, between the 1st of January 2013 and the 31st of December 2021. Based on the data acquired I wanted to extract relevant features which can be useful for the prediction model. I have used methods for both analysing textual data as well as checking the importance of the tweet based on the community response. In the experimentation phase I was building the model which would based on the tweets be able to predict the future price movements. The model did not improve its results when I have introduced the labels extracted from tweet classification as features. The final model obtained had an accuracy of around 52%. Through the project the main issues that I was struggling with was data leakage and overfitting. I was able to improve the feature set to eliminate the data leakage however the second issue was not resolved. Based on the results obtained from the tweet classification we can see that the overall accuracy of the model has improved which can mean that in the future after adding new features and potentially expainding the dataset even further the result accuracy might increase.

Contents

1 Introduction

1.1 What is the stock market and how it works

The history of the stock market exchange starts in medieval Europe. In the late 1400s, Belgium became the center of international trade. Merchants discovered the pattern in how supply and demand can influence the price of various cargo bought during this time. So they started buying different goods hoping that the price would increase and they could profit by selling them later. Approximately two centuries later, the Dutch East India Company became the first publicly traded company. On the 17th of May, 1792, the Buttonwood Agreement was signed. This historical event was marked as the beginning of the New York Stock Exchange. Since then, people have been buying stock shares, hoping they will profit from them. Since then, the stock market exchange only grew stronger and bigger. With the development of newspapers, stock exchange information became more popular, which meant that stock trading became more popular. Through the years, information found in different newspapers influenced how the stock market behaved. In the early 2000s, with the newest developments in the Internet, newspapers transitioned into online websites. During this time, when the first social media platforms were introduced, people started creating an online community where they could share their opinions, which were often related to companies publicly traded on various stock market exchanges. In the most recent history, there were several different instances where social media heavily influenced the price, below I listed a couple of examples.

1.1.1 Coca-Cola during Euro 2020

Football is one of the biggest sports in the world. Football stars have a massive following in the community, with Cristiano Ronaldo being the most followed person on the planet. On his Instagram account, he had garnished around 500 million followers, more than anybody else on the platform. During the 2020 UEFA European Football Championship, which is commonly referred to as Euro 2020, Ronaldo took part in a pre-match interview where he removed two Coca-Cola bottles from the frame and decided to advocate for a healthier diet by saying in Portuguese, “Water.” This gesture was then quickly spread on various social media platforms like Twitter and TikTok. It is worth noting that the Euro 2020 took place in 2021 as the countries responsible for hosting the event said that it is too risky to hold the event in 2020 as it could spread the pandemic even more. I am mentioning it because, during 2021, TikTok was already a giant platform with around 700 million users compared to 400 million in 2019. Because of that, more people saw the actual portion of the interview rather than read about it in the online newspaper. This also helped to spread the message as well as it gave people more understanding about the situation. As a result of the interview, the price of Coca-Cola stock dropped by 1.6%, which is roughly calculated to 4 billion dollars.[1]

1.1.2 Reddit and GameStop

In January 2021, at the height of the pandemic, when people were mostly spending their time on social media, Reddit users on the r/wallstreetbets [2] subreddit started planning a scheme to increase the price of GameStop Corporation. They used a strategy called short squeeze; this technique is used to rapidly increase the price of a stock owing primarily to an excess of short selling. The event started shortly after a comment posted by a popular short seller, which indicated that the stock would decrease. Starting from January 11th, the price increased by over 1500% by January 27th; because of high volatility, the trading was halted multiple times. With the stock price increasing in unprecedented amounts, the short squeeze got enormous popularity, and it was trending on most of the social media platforms. The GameStop short squeeze later inspired other similar actions, most notably AMC Theatres, which was a company in a similar position as GameStop. This also led to the price increase.[3]

1.1.3 Conclusion

In conclusion, we can see that there have been occurrences in history when the stock price was heavily influenced by social media. We can also see that social media can be used to spread information about unplanned events that can influence the stock, as well as it can be used as a communication hub for a group that wants to coordinate a group action to influence a stock price.

1.2 Project motivations

In the most recent days, I have noticed more and more correlations between what is posted on social media and how the stock price behaves. Because of that, I have started exploring different social media platforms in searching for possible future behaviours in the stock. In this process, I have explored that Twitter can be the platform which is most suitable for finding this type of information. The information that I post in the tweets is usually short and concise; there is also a wide range of profiles which cover different topics. For each company that we want to research, we can find a Twitter profile which covers information related to this section of the market or economy. Apart from that, Twitter also can provide us with feedback about how the community resonates with the topic at hand. Because of that, I decided to pursue this topic further and uncover if there is a correlation between the news reported on Twitter and the future price of a stock.

1.3 Project Aims

The project's main aim is to explore different methods in which we can use tweets collected from various verified Twitter accounts to predict future stock price movements. In order to achieve this task, I had to complete a series of secondary objectives; the first one was to create a dataset with tweets which are related to the analyzed company. After that, the data has to be transformed

into a feature set that can later be used in the tweet classification algorithm. After that, the data from the prediction set was used as an additional feature alongside technical indicators to measure if the tweets could have a tangible impact on the future movement of the stock.

2 Literature review

In the recent years there have been several research papers focused on using various deep learning models including Long Short Term Memory networks and information taken from social media, particularly Twitter.

While reading the news we can deduce if the general mood of the messages posted online. The emotions from the news can influence our behavior. When the news about the war at Ukraine started in 2022 people living close to the Ukraine began to panic. We have seen an increase in the amounts of gold bought by individual investors. News started reporing about this phenomenon and the level of fear increased. Because of that "Twitter mood prediction for the stock market"[4] was one of the articles which inspired for this project. The researchers have tried to come up with a method to somehow measure the current mood on the stock exchange by using the sentiment from the tweets. They have found a model with accuracy of around 86%, which predicts the stock moving up or down.

There have also been approaches where the LSTM has been applied to predict stock prices, on of them being "Stock Market Prediction Using Long Short-Term Memory" [5]. Here the researchers are talking about how the stacked LSTM can improve the better mean squarred error value. By analysing the results we can see that the applied methodolgy allowed for better results when applying different time steps in the model.

3 Professional and Ethics consideration

3.1 BCS Code of Conduct

3.1.1 Public

In accordance to the guidelines outlined in point a) I must deeply consider the ramifications of this project on privacy and and well-being of others. The data collected from Twitter will be extracted only from organizational accounts. Before storing the data will be rechecked to make sure that none of the scrapped messages belongs to personal accounts.
Following the guidance of point b) Twitter has an API which gives open access to extract data. Direct permission was not given but Twitter's privacy policy states that published information can be analyzed. [6]

3.1.2 Professional Competence and Integrity

Following points a),b),c) I have analysed the project with my supervisor and we believe that it is within my abilities. Following the accordance with point d) I will not breach any legislation relating to my project. As said in point e) I will regularly update my supervisor with progress and status of my project. I will not cause any harm to others, as well as I shall not accept any form of bribery nor will I ever encourage it as said in point f) and g). [6]

3.1.3 Duty to relevant authority

The only relevant authority for this project is the School of Informatics at the University of Sussex. Following guideline in point a) I shall make sure that I meet the requirements set by the university throughout the duration of the project. As stated in point b) I shall seek to avoid any possible conflict situations by delivering my work on time. While working on this project i accept professional responsibility for my work and the ramifications it could have. Following point d) I shall not share any confidential information about the project which I am currently working on unless I am given permission by the Relevant Authority or it is required by Legislation. In accordance with point e) the withholding of necessary or important information is completely prohibited and therefore i will ensure any vital information about my project is relayed to the correct authority when necessary. [6]

3.1.4 Duty to the Profession

Following points a),b) I must undertake this project with respect and professional standards. Following the guidance of point c) I shall work on my project respecting the guidelines outlined in the BCS Code of Conduct. As said in point d) I shall treat every member of BCS and other professionals with whom I will work with respect. [6]

3.2 Ethical review

After consulting my supervisor and the research ethics integrity and governance team this project will not require an ethical review. I have taken preventive steps which ensure that an ethical review is not needed. All the data taken from Twitter is taken from the major organizational account and checked before storing it. The checking algorithm includes searching for links to individual accounts and usernames within the text of the message.

4 Collecting the dataset

4.1 Criteria for choosing the company

For this project, Apple will be the company which will be analyzed. The company was chosen based on several different criteria:

- Popularity - Apple is one of the world's biggest companies, making it a popular topic of discussion on different social media platforms, in this case, Twitter. This ensures that a lot of data is available for analysis, which is crucial for training different machine learning models.
- Innovation - Apple is one of the leading innovators in the technology industry. Each year they release new products and announce their upcoming projects. Those developments can have a significant impact on the stock price.
- Liquidity - in order to produce an effective forecasting model, the stock data can't be static, and the market has to be efficient. As a result, Apple stock is highly liquid, with millions of shares traded daily. This can ensure that the stock technical indicators can provide better feedback.

4.2 Collecting the data from Twitter

Before writing the code to scrape the relevant tweets, I had to set the criteria for which profiles the data would be scrapped and what keywords should be used in the search. The accounts used in the search were chosen by their following and relevance. There are two types of accounts. The first one is technology-related accounts which tweet about the newest available technology and inform us about recent developments and announcements made by Apple. The other one is related to the world and stock-related news. Those accounts inform about the newest worldwide events that can relate to the analyzed company. In order to scrape the information, I have used the snscrape library. It allows for scraping from a specified account with a keyword. The keyword selection was another part, and the keywords have to be precise in order to make sure that the tweets are somehow related to Apple. Because of that, I have decided to use names of different products and technologies developed by Apple; those include words like iPhone, iPad, Mac, IOS, macOS, and iCloud. Alongside the content of the tweet data and other essential information like the date of the post and the account which posted the tweet, I have also decided to add information about the reply count, retweet count and like count. That additional information can provide information about tweet relevance, whether the tweet is an ad or it was posted by a bot. Again, this helped with filtering the dataset only to contain relevant information. The data was later stored inside a Comma-Seperated Value file to be easily accessible for future experiments.

4.3 Obtaining stock data

Stock data was obtained from the Yahoo Finance website and then stored in a CSV file. This file contains daily information about the stock market. Each row contains information about the opening price, highest daily price, lowest daily price, daily closing price, and the volume of the stock traded each day.

5 Technical analysis

5.1 What is technical analysis?

Technical analysis is the methodology used by traders and investors to evaluate price changes and forecast stock movements. The prediction is based on analysing statistical trends and indicators and is based on the theory that past trading can predict future markets. It has a number of advantages, and it allows traders to identify trends and patterns which might not be evident from the fundamental analysis. The set of rules established by different technical indicators can also help to manage risk effectively. Despite the many benefits, using technical analysis can provide, it also has its own limitations. The fact that the technical analysis is based only on past trading activity accounts for external factors such as changing economic situations, political events or even natural disasters.

5.2 Stock market indicators used in this project

5.2.1 OBV - on balance volume indicator

This indicator measures the flow of volume of a stock, it is measuring the relation between how the price changes and the trading volume. When the OBV value increases this means that the buyers are in control, while if the value is declining the sellers have the control. OBV can help machine learning models understand buying and selling pressures. [7]

5.2.2 MACD - moving average convergence divergence

MACD is another trend-following indicator. It is calculated by subtracting long-term EMA from short-term EMA.

$$MACD = 12DayPeriodEMA - 26DayPeriodEMA$$

Thanks to this formula we can calculate if the MACD value is positive or negative. Based on the MACD traders can determine whether the bearish or bullish momentum is high.

One of the limitations of MACD is the trend reversal. The MACD indicator is known for producing false positive results. Meaning that it predicts trend reversals which eventually do not happen.[8]

5.2.3 RSI - Relative Strength Index

It measures the speed and change of price movements, this allows it to provide information on whether the stock is overbought or oversold. The measured value is on a scale of 0 to 100. If the RSI value is above 70 then the stock is considered to be overbought, meaning that the stock has experienced a big price increase, and the buying pressure is weakening. It creates a “bubble” which

leads investors to believe that the stock is overvalued. However, if the RSI value is below 30 then the stock is oversold, meaning that the stock experienced a significant decrease in price and the buying pressure is increased. [9]

5.2.4 BB - Bollinger Bands

Bollinger bands consist of three separate lines, a simple moving average - SMA, and two standard deviations which are located below and above the SMA. Observing those lines can provide useful feedback on how the price fluctuate over time. When the lines, bands begin to move closer to each other this indicates low volatility. This means that the price is mainly stable, there are not any significant fluctuations. Often a low volatility phase precedes a big price movement. When the lines widen they can indicate high volatility. In this time the stock experiences rapid fluctuations which can lead to increased uncertainty and it could also signal that the stock is oversold or overbought depending on the stock price. Incorporating Bollinger Bands into the LSTM model can significantly increase the model's ability to recognize high and low volatility periods which can lead to better detection of reversals and breakouts. [10]

5.3 Research related to use of technical indicators in stock market forecasting

In order to increase their changes to improve the performance of stock market forecasting models the technical indicators were introduced. However since then the research suggest that more than 90% of investors still loose their money while investing on the stock exchange. This introduces the question if using technical indicators can improve the performance and accuracy of the model. Based on the results from an article - "Stock Prediction Based on Technical Indicators Using Deep Learning Model" [11] we can see that the models performance has increased when using various technical indicators as features in the model. Due to this fact and the analysis done previously I have decided to include described above indicators to increase the ability of the model.

6 Working with textual data

6.1 What is preprocessing

Preprocessing is a critical step in machine learning algorithms; it involves cleaning, transforming, and preparing datasets before they can be used in a classification mechanism. It is a crucial step because it ensures that the data collected from various sources, which can contain missing or erroneous values, is transformed into data that can provide a viable source of information for the machine learning model.

6.2 Link removal

Preprocessing is a critical step in machine learning algorithms, it involves cleaning, transforming, and preparing dataset before it can be used in classification mechanism. It is a crucial step because it ensures that the data collected from various sources, which can contain missing or erroneous values, is transformed into data that can provide a viable source of information for the machine learning model.

6.3 Mentions removal

Preprocessing is a critical step in machine learning algorithms, it involves cleaning, transforming, and preparing dataset before it can be used in classification mechanism. It is a crucial step because it ensures that the data collected from various sources, which can contain missing or erroneous values, is transformed into data that can provide a viable source of information for the machine learning model.

6.4 Tokenization

Tokenization is one of the most important steps in text preprocessing, and it breaks given text into individual words called tokens. It enables us to perform operations on tokens rather than the entire string simultaneously. Here I used a special version of the tokenizer from the Natural Language Toolkit called TweetTokenizer [12]. Compared to more popular counterparts, this tokenizer is created to work on Twitter data. Tweets contain unique characteristics compared to normal text data. It is designed to preserve tweet components like hashtags as separate tokens as well as gives support in working with popular social media abbreviations. Based on that, it is more suitable to work for tweet data and its unique structure, which as a result, leads to more accurate analysis.

6.5 Stopwords removal

This part removes the most commonly encountered stopwords that usually don't carry any significant meaning or sentiment. Removing those stopwords can allow models to focus on more significant words as well as it reduces the dimensionality.

6.6 Stemming

Stemming is one of the crucial steps of preprocessing in Natural Language Engineering tasks. Its main goal is to reduce words to its roots. Through this process we can decrease the dimensionality in the dataset as well as potentially improve efficiency and effectiveness of text analysis [13]. For this project I have decided to use Snowball Stemmer [14]

6.7 Number removal

This step involves removing numerical values from the text. Tweets often contain numerical values, and those values do not contribute to the sentiment analysis. Removing numbers from the text can also reduce unnecessary noise in the data and improve the accuracy of the future model.

6.8 Short words removal

This preprocessing method helps with identifying words which are negated and later modifying the sentiment of the word that follows. Handling negations is an essential process as it can improve the accuracy of different sentiment analysis models.

6.9 Transforming the text to lowercase letters

Similairly to the methods mentioned above, transforming the text into the lowercase letters reduces noise and the complexity of the model as well as it allows for more accurate sentiment analysis; it ensures that the algorithm does not assign a more weight to certain words which have a capitalized letter in their contents.

6.10 Sentiment score

Incorporating sentiment analysis as a feature for tweet classification models can possibly enhance the model's ability to capture emotional context of the message. In the instance of this project I wanted to capture the emotional context surrounding financial markets and technology forums.

6.10.1 VADER approach

Sentiment analysis can provide a valuable insight into tweet sentiment. In this project I decided to follow a hybrid approach. I have combined two different techniques, VADER - Valence Aware Dictionary and Sentiment Reasoner[15] and Harvard IV-4 dictionary. VADER is a rule-based sentiment analysis tool designed for social media. It looks at grammar and syntactical structure of the text to produce the compound score. Harvard IV-4 dictionary is a collection of positive and negative words, which allows for an easier scoring approach based on the presence of different words in the text. It doesn't look at the context of the message which can change the meaning of words. By combining those two techniques I am aiming to harness the advantages of those two methods and achieve a better sentiment score. It allows for both, the presence of positive and negative words as well as overall context of a given tweet.

6.10.2 Using BERT

Alongside the sentiment obtained from VADER and Harvard IV-4, I have also obtained another sentiment from a pre-trained model - roBERTa [16]. This model was created to extract the sentiment value of the given tweet and returning if it is positive, negative or maybe neutral to the context. I decided to incorporate another feature which is focused on a sentiment of a tweet because I wanted to include another variable and possibly analyse the impact of incorporating differently extracted sentiments in the classification model.RoBERTa is a model which was trained on around 58 million tweets. Because of the large training set the aim for using this model was to capture more intrinsic linguistic patterns than the hybrid approach to sentiment analysis described above. However, it also has its limitations, the model is very sensitive to the context in which the words appear this can lead to troubles with identifying sarcasm or irony.

In conclusion, based on the previous research roBERTa has demonstrated potential in classifying the sentiment of the tweets in various domains, including stock market forecasting hence why I am using it in production of the final classification model.

7 Analysing other features extracted from Twitter

7.1 Why are those features used how can they improve the model

In order to evaluate how the tweet resonates with the community, social media platforms like Twitter allows users to interact with the content posted on the website. On Twitter, every user can either like the post, comment on the post - reply to it, as well as share the post to their main page. Each of those measures can indicate something different; for example, if the tweet has a low number of likes but a high number of replies, we can try to interpret the text as a controversial post, and the community does not necessarily agree with what it represents.

7.1.1 Tweet's like count

Likes are an important measure, and it signifies endorsement and agreement on the platform. The number of likes can indicate the extent to which the tweet resonates with the audience. This can potentially influence the investment decisions in the future. A tweet with a big number of likes and positive sentiments can mean that it can inspire influence in the company and can cause the price of the stock to rise. On the other side, if a tweet has a negative sentiment and a significant amount of likes can mean that the investors may be more cautious

when making their next financial decisions. In summary, incorporating likes as a features can help to measure the strength of the tweet and provide valuable information for the classification model.

7.1.2 Tweet's reply count

Tweet replies measure user engagement and interactions. It helps to understand if the tweet gained some sort of traction amongst the platform users. Though, according to the ethical requirements, I did not use or analyse any of the replies, which were mostly posted by individual accounts rather than organizational accounts, they can still offer a valuable insight for the classification model. The number of replies can indicate the level of attention and curiosity surrounding the related stock. A high number generally shows that the post has sparked interest with the community, which can lead to increased market volatility in the future.

7.1.3 Tweet's retweets count

The number of retweets can measure the reach of the tweet; it reflects the extent to which the tweet is shared on different accounts. Sharing and reposting tweets mean that it is shared with a greater and broader audience. The number of retweets a given tweet receives is a valuable indication of the degree to which the content of the post is deemed relevant by the Twitter community.

7.1.4 Events

One of the key reasons why Apple was chosen to be analysed in this project is the fact that they are one of the leading companies in technological innovations. Every year they release new products which can potentially change the technological world. Because of that, I have accumulated a list of the most important events and product releases during the analysed time period. Based on that, I looked for tweets which contained information about new products or important announcements. I have decided that based on the previously found dates, I searched through tweets which were posted either one week before or after the date. This allowed me to capture both speculations about new products happening before the data and the impressions after the event.

7.1.5 Ticker

Based on the analysis of the data before the initial preprocessing and feature selection, I have noticed that the appearance of the Apple stock ticker - $AAPL can suggest that this tweet could possibly have a stronger relation to the stock market itself than a tweet that does not contain one. Because of that, I have marked tweets which have this symbol in their content and used it as a feature.

7.1.6 Tweet count

Twitter was one of the first platforms which incorporated trends on their platform. In other social media platforms like Meta, the platform only shows their user's content which is chosen beforehand by the user themself. Twitter, apart from showing users the content from profiles that the user follows, also shows what is trending. Because of that, content that is currently popular on the platform can be shown to every user. Incorporating tweet volume as a feature can express how popular the company is at the moment.

8 Creating the feature set for the classification algorithm

8.1 First feature set

After initial preprocessing of the dataset with tweets, now is the time to add all earlier described features to it. The first step is to calculate the mean average of like counts, reply counts and retweet counts for each of the profiles from which the information was scrapped. After that I have compared the values for each tweet with the average calculated for a given user and as signed them values as influential or not influential. This process was repeated for retweets and replies. Apple as a technology company constantly releases new products and soft ware updates. Because of that I have prepared a list of important moments that are related to newly released Apple technology between the 1st of January 2013 and the 31st of December 2021. To each date, I have assigned a set of keywords related to this event so that I can use them in a tweet search. Furthermore, in order to capture both the sentiment of anticipation of the event and the sentiment after the announcements, I have decided to perform the tweet search on post one week prior to and after the event. Based on the results, I have assigned each tweet an appropriate value. According to some research, the time of the day can be related to the stock price movements. Because of that, I have decided to look for tweets which were posted during trading time. In order to check if the tweet was posted during the trading time, I had first to change the date format in my dataset. While acquiring the data, the default timezone assigned to each date was the UTC; on the other hand, Apple is the company that is traded on the Wall Street Stock Exchange, which is localized in a different timezone - Easter Time. After changing the dates into an appropriate time zone, I then checked if each tweet was posted between 9.30 am and 4 pm, and based on that, I have assigned them values. When checking the initial dataset noticed that tweets which contain the Apple stock symbol, also known as a ticker, often contain information which is directly related to the stock data, therefore, can have a stronger connection to the actual stock price movements rather than a tweet which contains information about a new small update in a software or a new price discount. Because of that, I decided to look for those tweets and assign them different labels. The tweets were scrapped between the

beginning of 2013 up until the end of December 2021, which gives over 3000 days; however, the stock market is not open every day. It is closed on weekends and bank holidays. In this model, I have decided not to include the information for the days which are not included in the stock price data. The information was then grouped by dates, and the features were summed up. I have also added a new feature which holds the number of tweets posted each day; thanks to that, it can help the algorithm to find the correlation between the stock movements and the number of posts per day. After formatting the dataset into a suitable format for the classification, I have made the labels to predict. The prediction value was calculated if the stock price increased or decreased the next day.

8.2 Second feature set

The feature selection for the second model is fairly similar to the first model as it incorporates some of the features - trading time and checking if the tweet has a ticker in its content. However, there is a big distinction in how the tweets are being checked if they are influential. The previous model was based on the assumption that the Twitter profiles have the same audience all throughout the analyzed time frame. This is false, as Twitter rose by over 100 million users during this time. Instead of creating an average retweet, like and reply values and using the whole time frame to count the values, I have decided to count the monthly averages for each user and then based on that, I would calculate if the tweet is influential. Another difference from the first model is how the tweets are grouped by date. In the previous model, I have only used tweets from the dates that were in the stock data file. In the new model, I have decided to use all the tweets. Firstly I grouped them all by date; after that, I decided to check for each date if it appeared in the stock data. If it does not appear, then the tweets are moved to the next day. Additionally, I have also decided to change the prediction label. Instead of calculating the stock price difference between two days, I opted for calculating the difference between the opening and closing values each day. In order to prevent data leakage, I have also decided to push all of the tweets which were posted after 4 pm Eastern Time into the next day as they could potentially leak information about the future price leading to overly accurate results.

9 Exploring different classifiers

9.1 Decision Tree Classifer

The first classifier that I have explored is the Decision Tree classifier.The first classifier that I have explored is the Decision Tree classifier. This classification model is a supervised machine learning algorithm which can perform both classification and regression. The idea behind this model is to use the extracted features from the dataset to create yes or no questions and split the dataset to isolate all points that belong to each prediction class. By splitting the dataset

based on the binary response, we achieve a tree-like structure which is expanded by a new node each time the algorithm asks a question. The goal of the algorithm is to separate the dataset in such a way that all the leaf nodes, the nodes from which you don't split the data further, belong to a single class. However, this goal cannot always be achieved, and because of that, often, we see a mix of pure leaf nodes and mixed leaf nodes, which are assigned a predicted class based on the majority class in the node. One of the major reasons why this approach was used was because of its non-parametric approach, which does not rely on any previous assumptions about underlying data distribution, and this can make it more adaptable to real-world data, therefore, more effectively capturing complex relationships between tweets and stock movements. Decision trees are also suitable for larger datasets, which I thought can be important for large datasets. On the other side, Decision Trees are prone to overfitting as well as they can be biased towards the majority class and the stock price generally increasing in the analyzed time frame can lead to worse results.[17]

9.2 Random Forest Classifier

Another classifier used for the experiments was the Random Forest Classifier. It consists of a large number of individual decision trees that operate as an ensemble. Ensemble methods combine the strengths of multiple individual models to improve overall predictive performance. The Random Forest works on the principle of the wisdom of crowds meaning that the low correlation between models is key and can help to produce better results than the individual predictions. The predictions are made from each individual tree in which the majority class is assigned as the predicted class. Compared to decision trees, the random forest can help to reduce overfitting by averaging the predictions of multiple trees, which can lead to more reliable stock movement predictions. Decision trees are also more susceptible to small changes in data, and this can lead to unstable predictions. RF uses multiple trees, which improves the overall robustness and accuracy of the model.[16]

10 Analysing classification results

10.1 Introduction

In this chapter, I will try to explain the results which I have gotten from building the tweet classification mechanism, for which I have used a number of different approaches. As the problem at hand is a classification problem, I am going to use accuracy, precision, recall and F1-score to analyze collected results. Accuracy is the proportion of the correctly classified instances in the dataset, and it could be a powerful metric; however, it can be inaccurate when dealing with imbalanced datasets. Precision is the proportion of true positive instances out of the instances predicated as positive by the model. Recall is the proportion of true positive instances out of the actual positive instances in the dataset. F1

score is the harmonic mean of precision and recall; it is useful when dealing with unbalanced datasets because of its ability to provide a more balanced evaluation of the performance.

10.2 Experiment results

Throughout the experimentation process, I have run a number of experiments on the problem of tweet classification. The first experiment was created from the features obtained using the first model. This was meant to be a baseline score for future predictions. The initial accuracy was not very promising as the model was only able to achieve an accuracy of 50.99%, meaning that its ability to correctly classify the labels is only slightly better than just random guessing. After that, I have decided to hypertune this model with a number of different criteria hoping that this will give me a better understanding of how to further improve the model. I got a score of 54.41% as the accuracy score; however, the model was not able to correctly predict any of the instances of class 0. Because of that I have decided to apply an additional oversampling mechanism in the model to boost the accuracy.

	Precision	Recall	F1 score
class 0	0.00	0.00	0.00
class 1	0.54	1.00	0.70

Table 1: Experiment 2 results

In order to improve the F1-score, recall and precision I have used SMOTE - Synthetic Minority Over-sampling Technique to address an issue of class imbalance. This allowed me to obtain an accuracy score of 50%. From all of those three models I have learnt that the dataset which is inputted into the model can be a bit biased towards class 1, as in all instances the values of precision, recall and f1-score are higher than for class 0.

	Precision	Recall	F1 score
class 0	0.46	0.50	0.48
class 1	0.54	0.50	0.52

Table 2: Experiment 3 results

Seeing that the Decision Tree Classifier is struggling with the task I have decided to explore some different options, the first one being the Random Forest Classifier. During the first experiment without any hypertunning it achieved an accuracy score of 54.63%, which was the biggest recorded in my experiments so far. On the other hand the classification report suggested a big imbalance and

bias towards class 1, which has achieved a recall score three times larger than class 0.

	Precision	Recall	F1 score
class 0	0.50	0.29	0.37
class 1	0.56	0.76	0.64

Table 3: Experiment 4 results

I have again tried to address this issue using various oversampling techniques like Random Over Sampler, however the accuracy of the results dropped to values around 50%.

	Precision	Recall	F1 score
class 0	0.46	0.50	0.48
class 1	0.54	0.50	0.52

Table 4: Experiment 5 results

After achieving results which were not especially helpful due to the overfitting and potential bias I have decided to try using another feature dataset. Similarly to previous experiments I have started with the Decision Trees. The first experiment has failed classifying the labels with only 46% accuracy however the other metrics showed that the new dataset might be better balanced.

	Precision	Recall	F1 score
class 0	0.43	0.45	0.44
class 1	0.50	0.48	0.49

Table 5: Experiment 6 results

After that I decided to apply some hypertunning to the current model. I have tried finding the best parameters for the maximum depth of the tree and the minimum number of splits. After training the model on the best parameters found and predicting the labels the models accuracy has improved to 50% with the other criteria suggesting that the model does not have a bias.

	Precision	Recall	F1 score
class 0	0.47	0.48	0.28
class 1	0.53	0.52	0.53

Table 6: Experiment 7 results

In this case with a more balanced dataset I have decided to try and explore how the Random Forest Classifier would have behaved in such a situation. Based

on the initial result suggesting that the new model is better behaved than the older model, with the accuracy increased to 54%, the precision values showed that this was the first model which had both metrics for class 0 and class 1 above 0.5.

	Precision	Recall	F1 score
class 0	0.51	0.29	0.37
class 1	0.55	0.76	0.63

Table 7: Experiment 8 results

In later experiments with Random Forest Classifier I have tried exploring various different options, one of them included exploring different vectorization techniques to better grasp the meaning from the textual data. TF-IDF - Term Frequency-Inverse Document Frequency is a more sophisticated approach: it considers not only words in one document but it considers its frequency across the whole corpus, giving bigger importance to selected words. Additionally it reduces the impact of common words. The accuracy obtained from this model was only slightly better than random guessing.

	Precision	Recall	F1 score
class 0	0.46	0.34	0.39
class 1	0.53	0.65	0.58

Table 8: Experiment 9 results

Being unsatisfied with the results obtained from the second dataset I have decided to explore deep learning classification techniques. The first model build employed an embedding layer for converting the text data into a dense vector, an LSTM layer which captures the sequential information and a concatenation layer which allows both textual and non-textual features to work on a prediction together. The created model was then compiled to handle a binary classification. It was trained initially on 100 epochs and the accuracy of the model was around 52% with the bias towards class 1.

	Precision	Recall	F1 score
class 0	0.45	0.10	0.17
class 1	0.53	0.89	0.66

Table 9: Experiment 10 results

This graph is showing us the model accuracy for both training and validation sets over a number of epochs. The validation set allows for us to learn how the

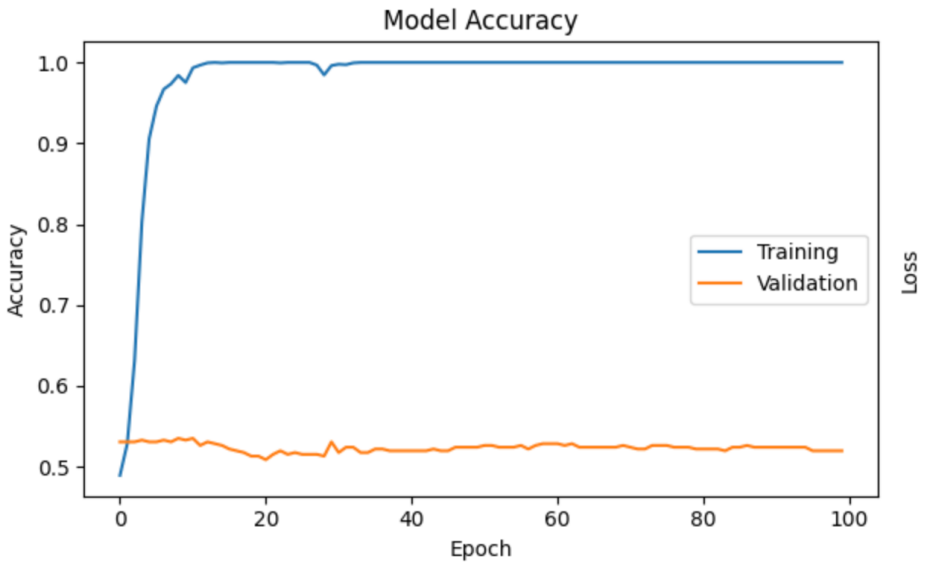


Figure 1: Experiment 10 - model accuracy over epochs

model is working on a set of unseen data. From the results obtained we can see that the validation line is fluctuating between 0.5 and 0.6 accuracy while the training accuracy rose from the value 0.5 for 0 epochs up to 1.0 for around 10 epochs. This can indicate overfitting in the model, which can cause the data to not respond well to the validation set

The figure presenting the model loss over epochs on both training and validation sets also suggest similar conclusions as the previous graph. The model seems to be overfitting with the training loss running down to 0 and fluctuating over the value afterwards.

Based on the two graphs we can see that the model is clearly overfitting and that the number of epochs, the number of times the algorithm goes through the dataset, indicates that the model might not be learning after around 10th epoch.

In order to reduce the overfitting in the model I have decided to create a new model with regularization applied to some of its layers. The accuracy result obtained from this model was around 50%. With the mean recall and F1-score around 0.5.

From the model accuracy over epochs we can see that the models did not improve enough to be able to handle the overfitting issue. The validation line is still fluctuating around 0.5 value and the training line drastically rises up after

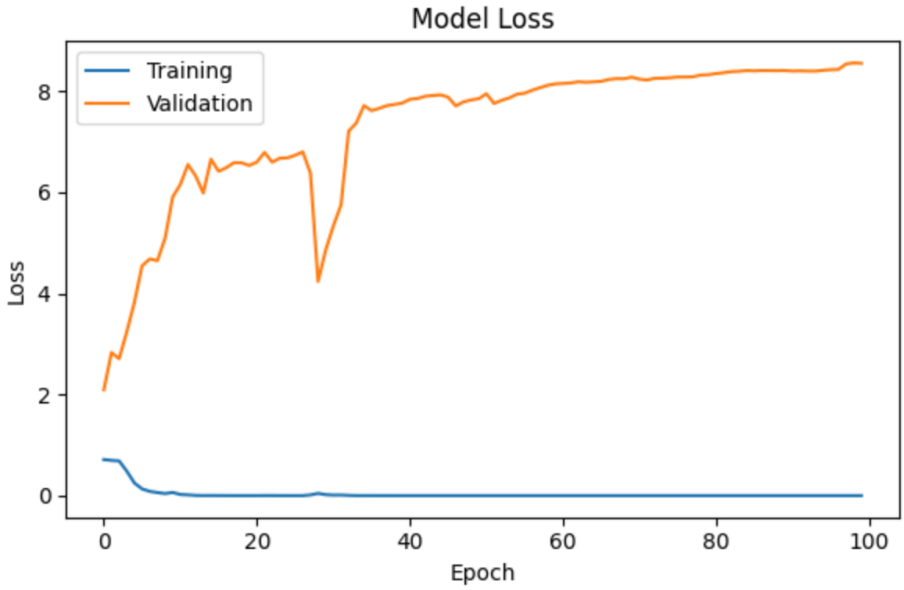


Figure 2: Experiment 10 - model loss over epochs

	Precision	Recall	F1 score
class 0	0.47	0.44	0.45
class 1	0.53	0.57	0.55

Table 10: Experiment 11 results

10 epoch and begins to fluctuate around 1.0.

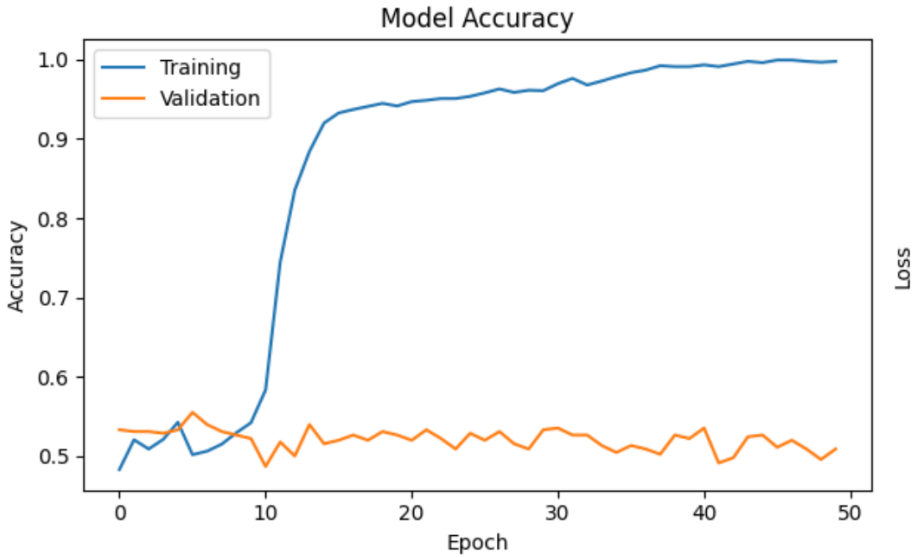


Figure 3: Experiment 11 - model loss over epochs

In the graph visualizing the model loss over epochs we can confirm that the model is in fact still overfitting, the lines are both on the same value until around 10th epoch and then they seperate from each other with the validation loss increasing and training loss decreasing and fluctuating close to 0.

After seeing models struggling and the overfitting issue being at hand I decided to change the model itself. Apart from implementing the regularization techniques applied in the previous model, I have also opted to include Early Stopping to further increase the models accuracy. It stops the learning process when the metrics start to degrade.

	Precision	Recall	F1 score
class 0	0.58	0.25	0.35
class 1	0.56	0.84	0.67

Table 11: Experiment 12 results

Based on the obtained results I have achieved the largest accuracy score yet - 56,4%. While looking at the results obtained from the recall and F1-score metrics I decided to look how many instances of each class are there in the testing set. Apart from the fact that we see that we have a more or less balanced testing set with the positive labels are the majority class with around 53% of all instances.

Figure 4: Experiment 11 - model loss over epochs

Since the early stopping stopped the algorithm we can see that the graphs for accuracy and loss over epochs is only drawn over the span of 10 epochs.

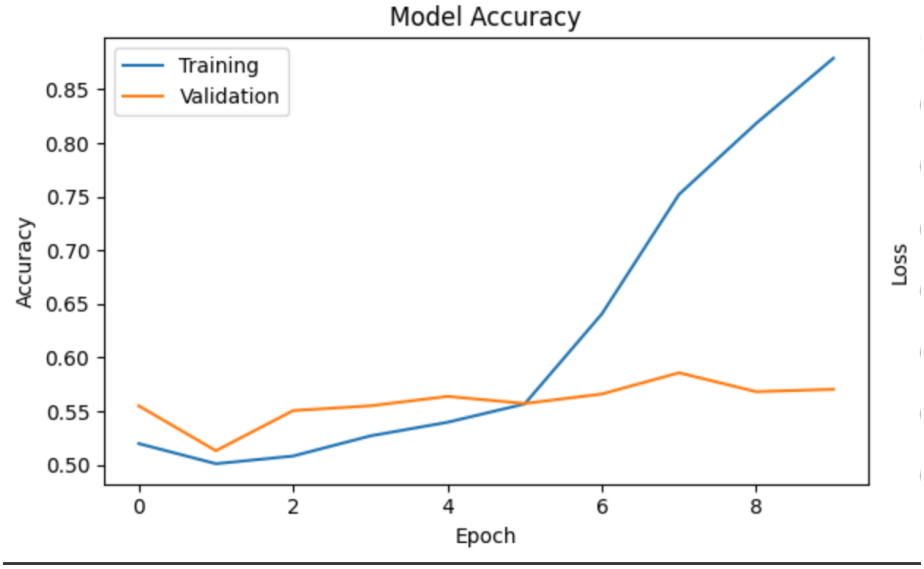


Figure 5: Experiment 12 - model accuracy over epochs

In the accuracy graph we see that similarly to the previous models that the validation and training lines are initially fluctuating over the same level to later split with the training accuracy increasing and validation accuracy fluctuating.

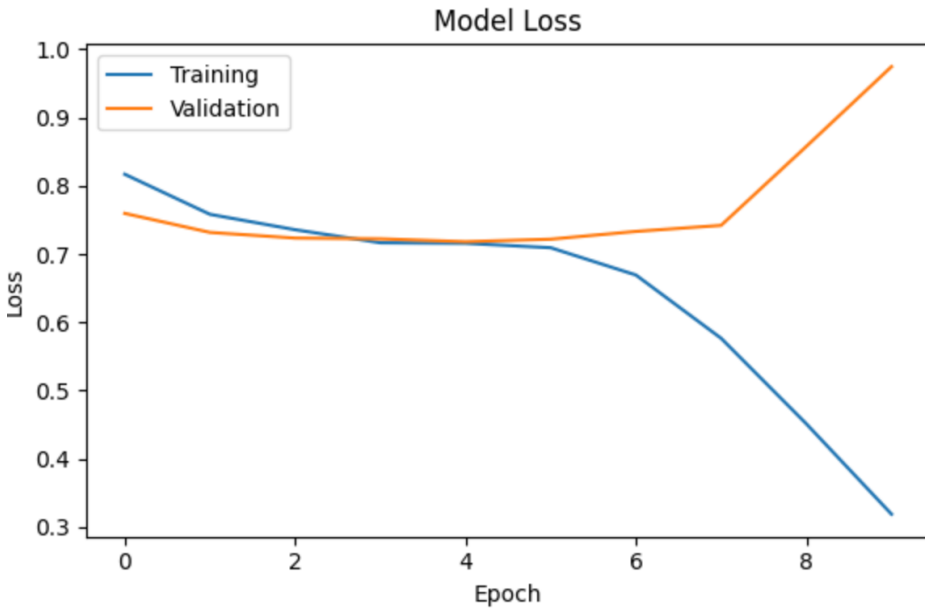


Figure 6: Experiment 12 - model loss over epochs

Figure 5 shows us similar results to the previous curves, as the model is overfitting with data over epochs.

10.3 Conclusions

Summarizing, thanks to those experiments we can see how different classification models work on the given data. The models where the Decision Tree classifier was used show the results which are usually more balanced where the difference in recalls and F1-scores is usually small. This was seen when I was using the first and the second feature dataset.

When using Random Forest classifiers on both feature sets the accuracy of the models improved. Unfortunately, with the increasing accuracy the model's difference between recall scores and F1-Scores increase. This behavior can indicate that the model is imbalanced.

Because of low results from both classifiers I have decided to explore if using neural networks can improve my model accuracy. In all of the classification results obtained I could see that the model was overfitting with data, after analysing a number of different solutions I was not able to handle this issue. Because of that I have decided to use my results acquired from the best accuracy, which is the model with 56.4% accuracy. The labels generated from this will be used as features for next forecasting algorithm.

11 Data Leakage

Data leakage is a critical issue that needs to be considered during the development and evaluation of different forecasting models, particularly in the domain of stock market forecasting. It occurs when we use information which might not be available at the time as training data for the model, meaning that the model is learning to predict future data based on the data acquired from the future. This can lead to false results, which typically are overly optimistic. Stock forecasting algorithms are heavily reliant on historical stock data to predict future trends and fluctuations in stock prices. Because of that, it is important to maintain the integrity of the data during the classification process. In this project, I have employed several different steps to prevent data leakage in my models. In the beginning, I checked if the features which are being used in the model do not rely on future data. Here it is important to note that some of the technical indicators can unintentionally cause data leakage. That is why instead of creating various technical indicators like Moving Average Convergence Divergence or Bollinger Bands for the entire data at once, I split the dataset into training and testing sections, and then later, I created the indicators separately for each of the sets. Splitting the data into the training and testing sets also has to be done using appropriate methods. The default train test split function from the sklearn python library randomizes the data, which can cause a situation where we could be using future data to predict past data. After splitting the data, it is also important to remember to apply different techniques like vectorization and scaling separately on training and testing datasets, as vectorizing or scaling the features in full can lead to potential data leakage.

12 Analysing the results acquired from the Long Short Term Memory network model

12.1 What is Long Short Term Memory network

Short Term Memory network is a type of recurrent neural network that is extremely well suited for capturing temporal dependencies in time series data due to their capability to store and retrieve information over an extended period of time.

The whole structure consists of a number of memory cells, which are responsible for storing and maintaining information over time. Depending on the signal received it can gather, preserve or delete the information. Alongside cells we can also find three gating mechanisms:

- Input gate - responsible for controlling the flow of new information into the model, it determines which parts of the input should be stored inside of memory cells. The gate itself produces values from between 0 and 1. If the value is low and close to 0 then we can say that the gateway is closed, and the flow of information has been reduced. On the other hand

if the value is high, close to 1, this indicates that the gate is open and the information can flow freely.

- Forget gate - this gate controls how much of the information stored inside of the memory cell should be retained or discarded. Similarly to the input gate it also uses sigmoid activation and produces the values from between 0 and 1. If the created value is low then the information should be forgotten. On the other side if the created value is high then the information should be remembered.
- Output gate - this gate determines which information from the memory cells should be used as the output of the LSTM unit.

12.2 Baseline models

In order to accurately access the impact of adding tweets as features into stock market forecasting algorithm I have to create a couple of models which would by the starting point of the experiments. In those models I have used the features extracted from the second feature set. This was done due to the potential data leakage with creating the event feature as theoretically I will not be able to know the names of the newly released technologies, hence capturing the speculation before the event can lead to data leakage.

The first baseline model was focused on predicting the future closing price. Based on the results from an image which portraits the comparison between actual closing price and the predicted closing price from the whole testing data we can see that the lines look similar. The slopes on both lines are aligned. In order to evaluate the model I wanted to check if both of those prices lines are either rising or falling each day. In this case the accuracy of this model was at around 47%.

Another baseline model was made to be a classification model which was predicting if the price of the given stock will increase or decrease in set up number of days. With a mean accuracy of around 50% in a span of 10 days.

12.3 Analysing the results

The final model uses both technical indicators as well as tweet classification results as a feature. The label for prediction for this algorithm is created using an algorithm which calculates if the price increased or decreased during a given number of days. Because of the way the prediction label is formulated I am able to evaluate the performance of the algorithm for both short and long term prediction. The results obtained from the prediction process suggest that the model is not doing well with the first model acquiring the accuracy of only slightly above 50%. From those results we also know that the model has a precision accuracy of over 60% for predicting instances of class 1, and the recall indicating that only 38% of all instances of class 1 are classified appropriately.

The next step in the project was to try and improve the model, one of the experiments was trying to predict if the price will go up or down in a two day

time span. The final accuracy obtained from this model was set to around 52%. Here again the model started suggesting that the results are imbalanced, as the recall and F1-score are higher for class 1.

When experimenting with different models and the time spans which can be used in order to create prediction labels I have noticed that the algorithms which include a higher time window than 3 days don't necessarily provide relevant results even if the accuracy is high. Even though the ability of the algorithm to predict the labels is at nearly 60%, those algorithms cannot balance out the F1-Scores and recalls. The model is usually better equipped to predict the labels of class 1 than the labels of class 2.

12.4 Conclusion

The results obtained from the experiments do not suggest that the experiment was successful. The maximum result which I was able to achieve had the accuracy of 52% and the model used in this prediction was predicting if the price will go up or down in a span of two days. When we compare this model to the baseline models we can see that this model is better suited for making predictions than the algorithm which was used to predict the price change. On the other side, the model is performing worse than the model which is predicting the price change label using only technical indicators. When we compare the predictions for stock movement changes up to three days time we can see that the baseline algorithm is performing much better with accuracy scores of around 56%.

When trying to analyze the classification results for the model where the time prediction window is larger than 4 I can see that the tweet model is preforming better, at least that is what the accuracy is suggesting, however when we look at class distribution in the test sample we can see that there is more of the instances of class 1 - where the stock price increased during the analyzed time span. This imbalance in the testing set can lead to inaccurate and overly optimistic results, and this can be seen in this case:

The accuracy of the model is suggesting that the algorithm is working well, however we can see the disproportion in other metrics than accuracy. The model after creating the labels had twice as many instances of class 1 than class 0. This can be reflected in how precision behaved,

13 Limitations of the model

The main limitation about the model that can be noticed is the need to provide more data into the algorithm. From analyzing the results we can see prediction algorithms struggle with predicting the labels on the testing set. This has happened due to the fact that the stock price of the analyzed company is rising over time in most of the test cases. This leads to the results which can suggest some sort of bias or imbalance in the dataset.

As presented in the introduction, information posted on social media can have the ability to influence the stock market. I have tried to incorporate the

possibility for the model to recognize this kind of unexpected information, by analyzing the contents of the tweet as well as by measuring the public response to the message. This method is not full proof and newly posted information which can have a negative effect on the stock market can be skipped by the algorithm due to the fact that it might have been missed because of the unusual response from the community.

14 How the model can be improved in the future

Based on the obtained results we can see that the model could use some improvement. The main issue about the model seems to be the tweet prediction itself, the dataset is not balanced as we can see in most of the classification experiments. Because of that the first improvement that should be considered is building a larger dataset with tweets. Looking at a couple of research papers I can see that often scientists use a dataset with tweets which are not strictly related to one company. Therefore building a larger dataset which also contains tweets about different companies from the same market sector can help to train the future models.

Another possible solution to improve the accuracy of tweet classification algorithms is to include topic modeling as a feature. It enables the discovery of hidden patterns within large volumes of text data. This can be particularly useful for detecting latest trends emerging in the tweets and it can lead to gaining a better understanding about tweets' impact on the stock market. To achieve that we could use topic modeling technique - Latent Dirichlet Allocation. This unsupervised learning method can be crucial to understand the sentiment and opinions expressed in the tweets.

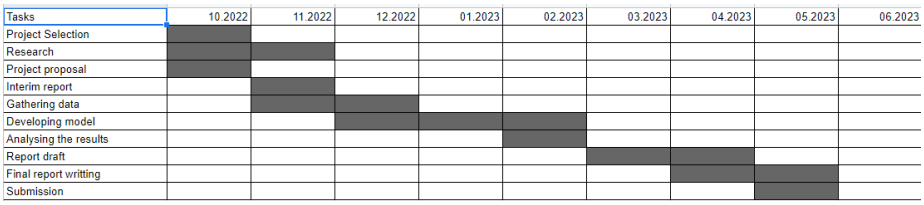
When working with the dataset of tweets which are related to a given company some of the scraped information might be irrelevant to the project. Companies often post ads related to their new products and analyzing those tweets in the model might decrease the accuracy. In order to improve the final model one could implement a tweet spam detection mechanism so that we can make sure that the dataset is only combined with more relevant tweets to the task.

15 Conclusion

Summarizing, this project was a more difficult task than I have expected, the results obtained from the final models show that the model is not performing with its accuracy only achieving a score of around 52%. This result indicated that the model is only performing better than one baseline algorithm where we are predicting future prices of the security. The tweet classification algorithm has achieved more optimistic results as I was able to improve it accuracy from below 50% to 56%. This can indicate that there is a future possibility of improvement in this model. Unfortunately, due to the time constraints I was not able to implement any other techniques which I am talking about in the section

covering techniques which can be applied to increase the performance in the future. Those improvements can help models to grasp a bigger understanding for the models as well as provide a number of additional features.

16 Project plan



16.1 Meeting log

Meeting 1 - 05.10.2022
Introduction meeting, we talked about project planning and the project proposal.
Meeting 2 - 02.11.2022 We talked about the interim report and its structure. We also discussed the research that we have done so far.

16.2 Completed work

Twitter API - October
I have developed a Python script which allows me to scrape tweets from selected user account. It also rechecks if there is not any personal information hidden inside extracted messages.

Stock data - November

I have developed a script which allows for getting stock data from a specified company and time period. The data is stored inside a CSV file.

First LSTM model - November

I have created first LSTM model for stock forecasting. Which gave me a lot of inside knowledge about how it should work in the future.

Building the first classification models - January

I have created the first decision tree classifier for the tweet binary classification problem.

Improving the accuracy of the model - February, March

I have explored different approaches to the tweet classification problem and hypertunned the model.

Building the stock prediction model - March

Based on the results from the classification I have began building the prediction model.

Data leakage ruinned models - March/April

I have noticed that the models which I have previously build are possibly leaking data, because of that I have redone the models on a new feature set.

Writting the report - April till now

I started writting the report based on the information acquired from the experiments.

References

[1] The Atlethic Staff. Ronaldo's coca cola gesture followed by $4bn drop in company's market value. `https://theathletic.com/news/cristiano-ronaldo-coca-cola-euro-2020/HO8sDjMJgPLh/`, 2021.

[2] Reddit. r/wallstreetbets. `https://www.reddit.com/r/wallstreetbets/`.

[3] Wikipedia. Gamestop short squeeze. `https://en.wikipedia.org/wiki/GameStop_short_squeeze`, 2021.

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

[5] M. Ukrit, Saranya Ak, and Anurag Rallabandi. *Stock Market Prediction Using Long Short-Term Memory*, pages 205–212. 01 2020.

[6] Bsc code of conduct. *https://www.bcs.org/membership-and-registrations/become-a-member/bcs-code-of-conduct/*.

[7] Adam Hayes. On-balance volume (obv): Definition, formula, and uses as indicator. `https://www.investopedia.com/terms/o/onbalancevolume.asp`, 2022.

[8] Brian Dolan. Macd indicator explained, with formula, examples, and limitations. `https://www.investopedia.com/terms/m/macd.asp`, 2022.

[9] Jason Fernando. Relative strength index (rsi) indicator explained with formula. https://www.investopedia.com/terms/r/rsi.asp.

[10] ADAM HAYES. Bollinger bands®: What they are, and what they tell investors. https://www.investopedia.com/terms/b/bollingerbands.asp.

[11] Manish Agrawal, Piyush Shukla, Rajit Nair, Anand Nayyar, and Mehedi Masud. Stock prediction based on technical indicators using deep learning model. *Cmc -Tech Science Press-*, 70:287–304, 09 2021.

[12] Python nltk — nltk.tweettokenizer(). https://www.geeksforgeeks.org/python-nltk-nltk-tweettokenizer/.

[13] Al-Khafaji Dr Hussein K and Habeeb Areej Tarief. Efficient algorithms for preprocessing and stemming of tweets in a sentiment analysis system. *IOSR J. Comput. Eng*, 19(3):44–50, 2017.

[14] Snowvall stemmer - nlp. https://www.geeksforgeeks.org/snowball-stemmer-nlp/.

[15] Can python understand human feelings through words? – a brief intro to nlp and vader sentiment analysis. https://www.analyticsvidhya.com/blog/2021/06/vader-for-sentiment-analysis/.

[16] Francesco Barbieri, Jose Camacho-Collados, Luis Espinosa Anke, and Leonardo Neves. TweetEval: Unified benchmark and comparative evaluation for tweet classification. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1644–1650, Online, November 2020. Association for Computational Linguistics.

[17] Decision tree classifier. https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html.