



The Power of Online Text Data

Leveraging social media & machine learning to
generate insights and make decisions

Business Administration and E-business

Master's Thesis

(CBUS02000E)

Contract no: 19586

Created by
Jamie Lee Jackson
(133972)
Makenzie McGraw
(133904)

Supervisor:
Liana V. Razmerita

Table of Contents

Abstract.....	2
1 Introduction.....	3
1.1 Research question	5
1.2 Topic	5
1.3 Delimitation and Scope	6
1.4 Structure of Paper	10
2 Literature.....	10
2.1 Theoretical Understanding of the Market & its behavior	13
2.2 Social Media & Machine Learning: Tools for harnessing text data	26
3 Methodology	44
3.1 Data & Data Collection	44
3.2 Pre-Exploration Processing	48
3.3 Data Exploration	50
3.4 Pre-model processing	63
3.5 Data frame Grouping	73
3.6 Model processing.....	78
4 Results	91
6 Final Thoughts.....	116
References.....	

Abstract

Our research's primary question aims to examine how social media can be used to gain insight into price movement. Secondly, its purpose was to scrutinize the methodological choices made by researchers within the areas of machine learning and text analytics and how they have applied their methods to comparable questions. This topic holds importance due to the growing access or availability of information online and our current ability to properly interpret and process this information. As we find in this research, text data found online can be immensely useful, but difficult to properly utilize. From the moment we began this paper, we realized the enormity of controversy that price analysis entails in relation to the stock market and its predictability. As researchers, we wanted to come to our own conclusion in terms of the plausibility of this task at hand. To inform upon these controversies we include the discussion of the efficient market hypothesis (EMH), random walk theory, adaptive markets theory, as well as other theories pertaining to behavioral finance. We found that a discussion of all three areas was necessary to gain insight into the possibilities and limitations of prediction within the stock market. Once these theories were established and discussed, we criticized and evaluated the latest works that have been conducted over the past twenty years in relation to stock market analysis and social media. Here we go over data used in the research as well the various methodological approaches taken. The many insights discovered in the previous works played an immense role in how we went about answering the primary question of this research. Once it was established how we would approach the collection and analysis of our data, we proceeded to the methodology. In this research we utilize data from both Twitter and Yahoo! Finance. We utilized existing datasets pertaining to Elon Musk's tweets as well as financial news from 2010-2020. Additionally, our data included historical stock data for Tesla (TSLA) and the NASDAQ Composite Index (^IXIC) for the same years. Our methodology includes various combinations of these four data sets. We look at Elon's tweets against both TSLA and ^IXIC, as well as the financial news data against both TSLA and ^IXIC. From there our methodology then employed a classification-based machine learning task where we used a logistic regression and a neural network. Overall, our results alluded to the impossibility of this task, at least with our resources. However, we did see significantly better performance with the combination of the financial news data and ^IXIC over any of the other combinations. Perhaps if an individual were to interpret information more efficiently this could in theory be possible for short term prediction. However, we do not possess the foresight to understand how the markets will behave in ten to twenty years from now.

1 Introduction

2021 began with an eventful start in relation to the stock market and the trading of certain securities. Game Stop made headlines in January 2021, as the stock price skyrocketed by over 1500%. This was particularly interesting as Game Stop (GME) historically has never seen such a high stock price, as well as the consideration that brick and mortar shops like this seem to be performing worse in recent years. We can see GME sees negative profit margins, negative growth in revenue (decreasing by -11% yearly over the last five years), and overall showing poor fundamental value (Chartmill, 2021). This leaves one with the question of what then is happening to cause this price explosion. This increase in price, is due to the online collaboration of retail traders. Much like the function of any hedge fund, where there are managers directing their analysts on how and what to trade, there are online bloggers directing their millions of followers on how and what to trade. One blogger on Reddit, wallstreetbets, with 9 million followers is said to have begun the buy and hold strategy of Game Stop stock that “sent it to the moon” or increased the price by over 1500%.

This event involving Game Stop piqued our interest into exploring how it is possible that an event such as this could take place. While also creating the question if it is possible to detect these events and apply prediction methods in relation to the stock movements. Current theories on the stock market and the market in general would state that the market is efficient, and it is not possible to predict price. As in a free-market system the securities prices should be controlled by the interaction of supply and demand. Additionally, all information should also be incorporated into the securities, therefore making it an impossibility to exploit the market. Is short term prediction of price movement a possibility? The proposed question is a well debated topic, wondering whether there is meaning behind the short-term fluctuations in stock price. The random walk theory would suggest that there are none, technical traders and behavioral finance theorists would assert that there is significance behind the movement.

There are known existing possibilities to manipulate the markets, this manipulation is done through artificially inflating or deflating security prices or influencing market behavior for personal gain. The firms or organizations that have the power to do this are heavily monitored and regulated by market forces such as government entities but have been able to slide under the radar in the past. It remains, when speaking about an established cooperation, there is a sense that the regulators have some sort of control over what, say, a hedge fund might be able to get away with, but what will happen when a new, decentralized entity is manipulating the market such as the

online retail traders. We would need to consider additional theories that could explain these market inefficiencies, which could in turn expand our view on how predictable the markets are.

For instance, the efficient market hypothesis (EMH), which relates strongly to the random walk theory, suggests that “all available information is incorporated into a security's price, and no investor has monopolistic access to the illusive information” (Malkiel, 2003). This would suggest that any investor cannot at any given point have access to all the information that could significantly impact stock prices. The rhetoric within the EMH stems from the fact that stock prices are driven by latest information as opposed to past prices. As the news itself is quite unpredictable this follows the logic that stock prices will follow a random walk. On the other hand, technical traders would argue that we could in theory analyse price charts through various techniques and use this information in real time to develop theories about the ways in which the market is likely to move. However, this does place emphasis upon the “short” term. Those that ascribe to the notion that human behaviour is inherently predictable would concur with this thought that we can predict. They believe that human emotion drives the decision-making processes and moods of individuals. Social media can manipulate this process and can be used as a tool that can both guide and map out the thoughts of individuals. The consideration of being able to predict in the short term still alludes to information inefficiencies, albeit short term inefficiencies.

The presentation of information efficiency as assumed by the EMH created an additional question, are the markets informationally efficient or have we never been able to efficiently intercept information? We shall further explore this idea further within this paper. With the growth of the internet and the emergence of web 2.0 (modern social media, user generated content etc.), is it possible our access to information is growing? This question is especially relevant when we consider information to be insights into people's daily actions, mindsets, and emotions. Today, we are all so connected through social media, in some ways more than others, we can communicate and live our lives through our computers, mechanisms for storing these thoughts and emotions. Not only are we connecting with friends and family in this time of being only online, but we are also at work, socializing and joining online communities. What are the implications of taking trading to social media and interacting in a real time environment with other retail investors? It is possible that large social groups can make or follow decisions and in turn manipulate the market. The recent event with GME would show us that yes, this possibility of manipulation does exist in this form, but the question of how we can capture and follow these events in the market through data remains.

The current work relating to social media analysis and stock prediction is quite fragmented in methodological approach and data usage. Much of the work that we came across pertained heavily to machine learning (ML) and sentiment analysis of social media platforms such as Twitter. The consensus in terms of predictability remains contested amongst these areas of research. In part this is due to the sheer magnitude of the data as well the bias and noise that often accompanies text analysis. It is our aim, whilst keeping previous works in mind, to examine these methodological choices, and to see why the data behaves this way. We would also like to see if social media can provide any insight at all into stock price movement in general. With the emergence of the interactive internet-based technologies, user generated input has become the lifeblood of social media. Social media truly is a digital representation of the ways in which we behave in our everyday lives. Through the protection of our computer screens our emotions and sentiment become very tangible qualities that can be tracked and analysed to an extent that has not been a possibility in the past. This now brings us to our research question.

1.1 Research question

How can social media be used to gain insight into price movement?

→ How can we predict stock price movement in relation to public sentiment?

1.2 Topic

The topic of this assignment overall seeks to question the different elements that can be combined to examine the movement of price on the stock market. Within this topic we seek to channel text data from social media platforms to gain insight into price movement. Whilst, we are not aiming to revolutionize, we do wish to understand how the different data features interact, and how the different machine learning algorithms treat the data. The goal is to process and clean the text data within python through using a series of functions and packages. We shall also visualize our data using Power BI as a means of gaining further insight into stock price movement. From there we will progress to our models. The aim is to start simplistically, with a logistic regression and to then work our way to a more complex model like a neural network. Lastly, as this paper falls within the realm of e-business, the topic shall be related to all societal implications as well as provide a critique of the methodological uses. By societal, this will refer to the organizational implications.

Motivation

Our motivation for this research stems from our interest within both machine learning (or AI) and the impact of social media on various aspects of the world. Moreover, the ability to predict based on one's ability to collect and transform data. In our studies, we have found that text data is one

of the most fascinating data types to examine and process in terms of its unstructured high dimensional nature. These interests coincided simultaneously with our fascination with social media and its ability to garner and capture the minds of millions. We thought that the best way to exhibit this interest would be to “exercise” the skills we have learned throughout our degree and apply it to a Twitter data set alongside stock price data.

Importance

Although we have alluded to this above, we shall merely state why we believe our topic is important. First, the attempts to gain insights about retail investors through using social media channels has been a growing trend. Several companies have already begun to start pondering the utility and tools themselves. Secondly, gathering and interpreting social media data has also been a growing trend over with the increased digital footprint that many leave online. This data thus becomes a currency that has redefined the ways in which people do business within the digital space. Lastly, having the machine learning tools to harness this information can thereby allow investors to react in real time.

1.3 Delimitation and Scope

As we embark our journey to discover how the analysis of social media can be used to determine stock movement, it is important to set the tone and outline our delimitations. The aim is to provide clarity of the direction of our research.

Research Philosophy

As we are using a mixed method, we shall use the pragmatism philosophy. As described by Saunders *et al.* (2019), pragmatists recognize that there are many ways of approaching problems within the world and undertaking research. They believe that no single view can give an accurate picture of a situation. This relates to our project as we shall strive to examine the various conflicting theories.

Approach to research

Our approach to this shall be deduction based, as we are testing theories with collected data. With a deductive approach, Saunders *et al.* (2019) states that it normally follows the following logic:

- **The first step is to set a hypothesis from a theory. In our case, we are exploring whether there is a relationship between social media and stock movement.**
- **Use existing literature, specify the areas in which the theory is expected to hold, and from there deduce several propositions.**

- **Examine the premise of the logic and the argument that produced them.**
- **Test the premises by collecting data and variables**
- **If the results are not consistent then the theory is false or rejected, or modified**
- **If the theory itself is consistent then the theory is corroborated.**

Methodological Choice

As we have adopted the pragmatic philosophy, with a deductive approach, it seems quite logical that we take a mixed method approach. The aim is to have one methodology support the other. Therefore, this involves the use of a concurrent embedded design. The main take away is that it will allow us to use both quantitative and qualitative elements to add to the area of research. This is required for this area of research as social media data is text driven. One could argue that the data itself could be construed as quantitative as the transformation of it involves tokenizing the data. Nevertheless, the very essence of the source itself is qualitative, and must later be transformed into quantitative. The benefit of using mixed methods is that we can allow certain understandings to be elaborated upon or confirmed. Moreover, double testing with two sets of data will give us the confidence to confirm or deny our hypothesis.

Subject Choice

The choice of subject came down to the availability of data. When initially examining short term prediction and social media there were several articles that referred to Elon Musk and his impact on stock price movement. As he is the CEO of Tesla, we chose to examine the possible impact he could have on Tesla's stock price through his Twitter account. As Tesla is listed on the Nasdaq stock exchange, we thought it would be prudent to include his impact on the Nasdaq Composite Index. Additionally, we were lucky to find a comprehensive financial news dataset which contains a lengthy list of financial news providers. We believe that the combination of these sources will result in finding valuable insights throughout this research. We will delve deeper into the intricacies of what a stock exchange and index mean in the literature review.

Overview of subjects

Elon Musk is the CEO and product architect of Tesla, Inc. while also being involved in founding and managing several other companies. Tesla Inc. is an American electric car company which produces electric cars as well as energy generating and storing devices. Tesla is headquartered in Palo Alto California, with factories in both California and Shanghai. Tesla was founded in 2003, produced its first model "The Roadster" in 2008 and gained popularity around 2013 after the

release of the “Model S”. The Model S was the world's first ever premium all-electric sedan with the longest battery range, and fastest acceleration time. Model S has since become the best car in its class in every category (Tesla Inc., 2021).

Elon has on several occasions been referred to as a “mover” of markets. His brazen behavior has cultivated a following of individuals that take his opinion regarding both Tesla and other companies seriously. There have been several instances where this behavior has been exhibited. For example, when Elon Musk’s declared his support for Bitcoin, soon after Bitcoin’s value jumped over 20%. Whether this relationship is completely causal is of course speculative, however his tweets regarding Doge Coin and Etsy also seemed to have caused these assets to shift quite significantly. As a case study for prediction, he serves as an interesting example. For one, we can look at how he leverages social media and in turn see the impact that his words may have on the price change. On the other hand, it could also shed some light upon the possibilities of social media. As the CEO of a major company, Elon seems to have a larger marketing bandwidth as opposed to those who run the social media for the company. It has left us curious and eager to examine him as a market disruptor as well as a CEO who embraces the power of social media in general. From an insight’s perspective, this can further guide us in our journey to understand the channeling of text analytics and text data. Moreover, we understand that Elon could be considered an enigma and his behavior could be seen as an outlier.

The main reason we chose to incorporate a secondary text dataset was to combat this question. This dataset is a compilation of financial online news and will be used as a point of comparison. By financial news, we are referring to the tweets made by major news outlets. Outlets such as Bloomberg, CNBS, Financial Times, Seeking Alpha and many more are included in this data. The tweets from the news contain both news and opinion articles. We will cross validate the datasets, so that there is not simply one text comparison to our two financial stock data sets. This will permit us to expand the boundaries of our research as well as attempt to quality ensure and remove noise from our data.

Nature of Research

The first step in this process will be to identify the nature of our research. According to Saunders *et al.*, (2019), the way in which we ask our research question will lead to an answer that is either, exploratory, descriptive, explanatory, and descriptive and explanatory. This is relevant to the delimitation and scope as it will outline the purpose and scope of our paper. We have concluded that the nature of our research shall follow an exploratory strategy.

An exploratory study is a means of asking open questions that allow you to gain insights about a certain topic. It often includes a search of the literature, interviewing experts, conducting in depth interviews, or conducting focus groups. It can be quite a flexible approach to research and can be paired with our research design. We will apply this through exploring the relationship between social media data and stock movement. According to some of the existing theory, all the variables are independent and uncorrelated towards one another. This prompted us to use the exploratory approach to explore if there is an existing relationship, and to develop our own opinion. Furthermore, our research could be taken one step further through an explanatory approach, provided a relationship does exist. This is evident within our secondary research question.

Choosing a time horizon

In terms of time horizon, we feel the need to set the scope for two areas. The first being our data, and the second our literature. In terms of data, we aspire to have a longitudinal study due to our ability to collect historical data from two sources. The sources in this case are an archived data base of tweets, and historical stock data. This data will be a historical view of the last ten years, from 2010-2020. In terms of literature, as social media is quite a new phenomenon it should be stated that we shall focus our research upon the last twenty years. Originally, we would have preferred to limit ourselves to the last 10 years of available research. When considering theory as well as previous works done on this topic, we found that it was necessary to extend this boundary to keep a good overview of the topic. Nonetheless, our desire is to ensure the most recent and up to date approach to understanding the relationship between social media and stock movement.

Establishing the Ethics

The next area of focus shall be upon the ethics of our research design. We believe that we do not have any ethical dilemmas in this case as we are using publicly available sources. However, Elon Musk himself does possess an ethical dilemma in his outlandish behavior. We aim to preface that by establishing or examining his effect, it does in no way mean that we condone nor encourage other individuals to follow in pursuit. This in part stems from the fact that on a few occasions Elon Musk has been accused of bordering on the edge of violating fair trade laws in the United States.

Another ethical dilemma we have considered is what happens in the event that it becomes easier to predict the stock market. If a company gains the ability to mine retail investors. Are we breaking an ethical boundary? We already know that companies try and predict our purchasing behavior online through our digital footprint and in doing so they encourage us to buy everything from

socks to chocolate. Companies possess a significant advantage to collect and store data, would this be a fair advantage to have over competitors or retail traders? Would they then volatility trade based upon the retail traders? By this we mean a company would predict the moment in which retail traders would seek to buy a stock before them. The company could in theory make money from this behavior. This also pertains heavily to the ethics regarding big data. Only the larger companies, with the massive amounts of computing power, would be able to undertake a task such as this. This would then leave some of the smaller companies at a disadvantage.

1.4 Structure of Paper

The aim of the following figure is to create an easy-to-follow outline of the structure of this paper, and the flow of the paper itself. It is both for the reader as well as a general guide to research.

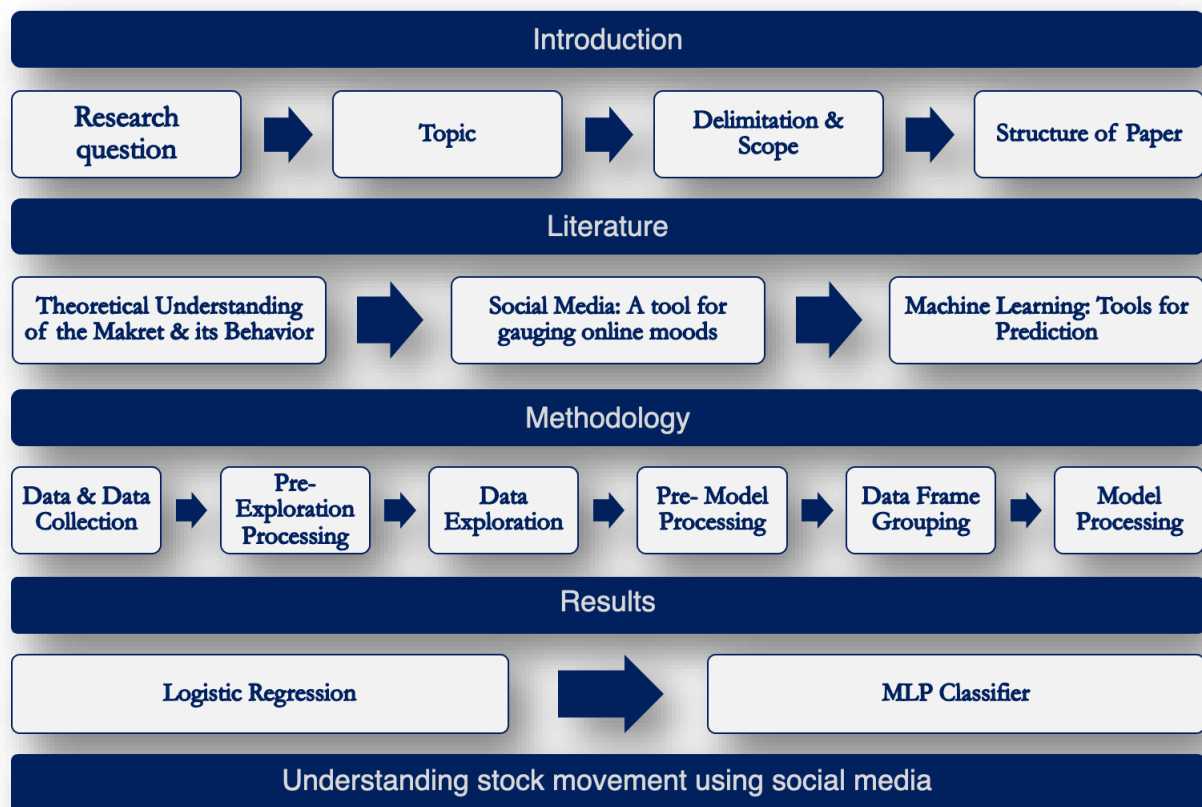


Figure 1.1. Structure of Paper

2 Literature

We shall approach our literature critically with the hope that it will provide the foundation upon which our research is built. The primary aim with this is to formulate our understanding of the respective areas of our research interest. Then from there gain insight into both the previous works

as well as the trends that have emerged in line with this area. As we have chosen to have a deductive approach, this means that we will develop a theoretical framework which we will use as a means of evaluating our data at a later point (Saunders *et al.*, 2019). The aim of our literature review will be to evaluate the different inferences and trends pertaining to our subject, whilst recognizing any biases or oversights that may occur. From there we will present this in the way that we believe to be the most logical for the task at hand. We shall approach our literature review in the form of a funnel (Saunders *et al.*, 2019). We took some of our inspiration from the way in which Saunders *et al.* (2019) approach the literature review. In the following stepwise section, you will find our process interlaced with inspiration from Saunders *et al.* (2019).

Research Parameters

Before we deep dive into our strategy, we believe it is necessary to state our search parameters first. These will be used throughout our research strategy as a guide that will enable us to remain within a clear and concise boundary.

Parameter	Narrow
Language	English
Subject Area	Stock Market, Machine Learning, Sentiment Analysis, Finance
Business Sector	Automotive Industry & Technological Sector
Database	Copenhagen Business School Library, and Google Scholar
Publication Period	Last 20 years
Literature Type	Journals, books, websites

Table 1.2. Literature Search Parameters.

Research Strategy

1. Beginning on a general level: This shall involve researching the stock market, social media, and machine learning in general.
2. Overview: The areas shall be mapped out accordingly into broad themes and summarized based upon their respective areas. We have created a figure for this, and it can be found at the end of our literature review in figure xx.
3. Compare and Contrast: We shall evaluate all the areas of research accordingly, and their approach to the assorted topics.
4. The work shall then be narrowed down to illustrate the most relevant previous works. When delimiting the scope of the theoretical foundations, as mentioned in our parameters we tried

to limit it to the past twenty years. However, there were certain cases of citation that do reference theorists from before this time. However, this was in part due to the contemporary theorists understanding and the homage that they pay to their predecessors. In addition to this, we shall research additional specific aspects that were found in the overview.

5. We shall then provide a detailed account of our findings in the literature review.
6. We shall embellish upon the previous works and add additional elements to it that both criticize and provide context to the ways in which we will evaluate it.

Our Literature Structure

Now that our process has been outlined. We will outline the structure of the literature review, and what we will discuss.

→ Theoretical Understanding of the Market & its Behavior

- *Stock Market Mechanisms*
- *Efficient market hypothesis (EMH)*
- *Momentum and behavioral finance*
- *Adaptive Markets Hypothesis*

→ Social Media & Machine Learning: Tools for harnessing text data

- *Social Media: A tool for gauging moods online*
- *Machine Learning: A tool of prediction*

After our phases of literature, we believed that it was essential to examine theory that both corroborates and contradicts the areas of this research. This shall be reviewed in three parts that can be found within the Theoretical Understanding of the Market & its Behavior. We shall first begin with describing the efficient market hypothesis and random walk theory. These theories are viewed as quite compelling amongst theorists in their direct rebuttal of the ability to predict stock price based on information found in society. This pertains to the use of media data. The second part of this shall strive to address the latter theory that aims to understand the more human side of market movement through behavioral economics. The human side indicates that there are cases in which one could predict market behavior. In support of behavioral economics is the theory adaptive market hypothesis, which aims to connect the arguments of the EMH and its contesters. Whilst also outlining the theory, we will provide criticism of the approaches and use this information to inform our own research. The subsequent section shall endeavor to examine the previous works done in relation to stock price prediction and sentiment analysis using machine learning. In doing so, we shall both examine the methodological uses, as well as the findings of the

respective researchers. We shall critically evaluate the previous works as well as draw inspiration from it for our own research.

2.1 Theoretical Understanding of the Market & its behavior

There are many circulating theories about the stock market and its predictability. Our goal with this section of the paper is to inform on the main theories that may support or criticize our ability to apply machine learning prediction to movement within the stock market.

Stock Market Mechanisms

To ensure that the reader will follow mentions of stocks and the stock market, we believe that it is important to mention some of the key terms and concepts of stock market research. The main subjects we will cover being what are technical and fundamental traders. As well as the actual difference between a stock, a stock exchange, and a stock index. As these terms sound so similar, we feel that it is prudent to explain the nuances. As well as to explain why we apply certain methods later in this research.

Market Participants

Before we investigate the previous works regarding social media and stock market analysis, it is prudent to describe how investors make decisions. By decisions, we mean their reasoning behind what stocks they will buy and sell. We alluded to this prior in our introduction, where we referenced technical analysis. Stock market prediction in general is difficult due to the enormity of uncertainties involved. Investors typically fall into two categories when making decisions regarding the stock market itself. Patel *et al.* (2015) separate the analysis that investors take into two specific types: fundamental and technical. These are viewed as decision making tools for stock market decisions (Nti *et al.*, 2020).

Fundamental investors examine the value of the stock, performance of the industry itself and the economy. According to Investopedia (2021), analysts within their area study everything from the overall economy and the industry conditions to the financial strength and management of these companies. Thus, everything from “earnings, expenses, assets, and liabilities” will all be taken into consideration. Due to its unstructured nature, the automation of this type of analysis is difficult. Fundamental analysts use openly accessible facts about the stock to perform the analysis of the stock price itself (Nti *et al.*, 2020).

Whereas the technical analysts will examine the statistics. They will actively search and identify opportunities that emerge from statistical trends, such as the stock price and volume. *Volume* is the number of shares that a security changes hands over some period. This interval can vary between

daily, monthly, and yearly. Securities that have a more daily volume are more liquid as they are a bit more active. Volume is used as a measure of significance of a stock, as it can show interest in that security (Investopedia, 2021). *Stock price* can have different measures that appear from a normal trading day. Within a trading day, there is a high price, low price, closing price, and an adjusted close. The high and low price are quite self-explanatory. However, the adjusted close will amend the stock's closing price to reflect that stock's value after accounting. The closing is the raw price, this is the cash value of the transacted price before the market will close (Investopedia, 2021). Investopedia (2021) states that the core assumption accepted is that all fundamentals are accepted within the price, and as such there is no need to pay attention to them when you are a technical analyst. These analysts will seek to examine a stock's patterns over time. These modes of analysis are relics that have not possessed the same manoeuvrability of other types of analytical tools. It is in part why our journey has led us to outline machine learning practices and how it is employed within stock movement.

Stocks, Exchanges & Indices

According to Investopedia (2021), a *stock* or *security* can be referred to as equity that represents ownership of a piece of a company. Ji *et al.* (2020) state that a stock is a financial product that is characterized by its risk, return, and flexibility and is favored by investors. Units of stock are referred to as shares and a purchaser of a company's shares is referred to as a shareholder. The shareholders, or people who purchase a specific stock can obtain returns through a process of estimating the stock price trends. A stock will typically or ideally allude to the performance of an individual company. For example, we can all purchase a stock of Tesla and would then own a small fraction of the company's assets and profits. Before any stock can be available to investors, a company must take their company public. Taking a company public is done by the issuing of stock to an initial set of public shareholders. This is typically to raise funds for operational purposes in an initial public offering (IPO). An IPO is the process that leads to a company's stock becoming available for investors to trade on exchanges. Overall, the price of a stock is determined by the flow of supply and demand, which is tracked on a stock exchange (Investopedia, 2021).

This leads us to our next term, *Stock Exchange*. A stock exchange is the physical or virtual place where a security changes hand. There are exchanges worldwide that operate typically on normal working days and hours. These exchanges make it possible to buy and sell different financial instruments such as equities (stocks), commodities and bonds (Investopedia, 2021). These exchanges monitor the flow of supply and demand, while also providing a platform that creates enough liquidity for there to be efficient and fair trades. Some of the most well-known exchanges

are The New York Stock Exchange (NYSE) and the NASDAQ, these are currently the two largest exchanges in the world by market capitalization or total value (World-Exchanges, 2021). These exchanges can vary slightly in how they operate, for example NYSE is comprised of 500 of the largest US companies where NASDAQ is comprised of 3000+ companies both large and small. NYSE is traded both physically by people and electronically, whereas NASDAQ is solely an electronic exchange. An additional note is that trading days are in fact not 365 a year, there are around 253 trading days a year. However, this can vary based upon the number of public holidays throughout the year. This is an interesting fact to note, it will later be reflected in our dataset as we will see that the data are limited to the amount of trading days in a year. On trading days, when a shareholder places orders to buy or sell stocks, these exchanges are the executors of the swap. While the term stock exchange may seem obvious, it is important to distinguish between stock exchange and stock index.

A *stock index* refers to the grouping of company stocks that will present a gauge on a certain sector (i.e., technology) of the stock market and how it is performing. Visually, we can tell the difference between a normal security and an index by looking at its Ticker. A ticker is a unique series of letters that are assigned to a stock (Investopedia, 2021). Where a normal stock like Tesla will have a ticker TSLA, an index will contain the symbol ^, as we can see on the S&P 500 whose ticker is ^GSPC. According to Investopedia (2021), the market index is a hypothetical portfolio of investment holdings that often represent a segment of the market, the price of which is gathered from the underlying stocks. Each market index itself, such as S&P 500 (^GSPC) contain a set of individual stocks that are listed on a stock exchange, in this case, all stocks found on the ^GSPC are the largest US companies listed on either the NYSE or NASDAQ stock exchange. Any stock that appears on an index influences the indexes price; the impact each stock has on the index is determined by its weight relative to other stocks on that same index. This weight can be assigned through calculations, typically based on the price of a stock or the value of the company itself, also referred to as a company's market cap. Market cap refers to a security's number of outstanding shares multiplied by the price of the outstanding shares (Investopedia, 2021). In short, companies with the largest market cap have the greatest impact on the index itself. Although the S&P 500 uses market capitalization to weigh its securities, other indices can utilize alternative methods to do this.

Overall, the stock market can be such an overwhelming place for anyone who is not an expert, due to this it is important to understand why one would choose to analyze one piece of it over another. Specifically, in this project we chose include both a company listed on the NASDAQ Stock exchange, and a NASDAQ index as a focal point of our research. Initially this was quite

confusing to sort out, so for the reader we believe it is important to note the differences between the uses of the term NASDAQ.

Nasdaq Inc. is referring to Nasdaq the holding company, a holding company is a company that is created to buy and own the shares of other companies. Typically, these companies do not manufacture or sell products, instead they hold the controlling stock in other companies (Investopedia, 2021). In this case Nasdaq Inc. provides and operates a global electronic marketplace for buying and trading securities, one of their exchanges is referred to as the NASDAQ Stock Exchange. We should note that Nasdaq Inc. has absorbed many exchanges in the US, for example Philadelphia Stock Exchange (PHLX) is now referred to as Nasdaq OMX PHLX. So, Nasdaq Inc. should be thought of as an umbrella covering many exchanges. Additionally, the company Nasdaq Inc. is traded on a The NASDAQ Stock Exchange as NDAQ. This ticker represents the corporation but does not represent the performance of a market like say, The NASDAQ Composite Index (^IXIC) would.

Indexes as Indicators

Now that we have walked through what the differences are between a stock, a stock exchange, and a stock index, we feel it is important to discuss further the relevance of a stock index. As we will be utilizing the NASDAQ Composite Index (^IXIC) in this research, we will focus specifically on the evaluation of this index. Due to this methodological choice, we believe it is important to explore why an index can be a good indicator of the market or a sector of the market. Additionally, why this would be useful when comparing to social media and online news data.

A market index is simply a portfolio of securities that represent a section of the stock market. As discussed above, the indexes derive their value from the underlying securities. This grouping of securities can become a useful benchmark for a specific sector of the market, or perhaps an overview of the market. Often investors will follow these indexes to track securities performance and use them as tools to aid in investment decisions. In the United States, usefulness of an index can vary from looking at the largest stocks by market cap (S&P 500), to looking at how an entire exchanges stocks are performing (NASDAQ Composite). Due to the indexes being a widely used tool for both investment managers and retail traders/investors, we can see this as a good gauge on the market and perhaps by extension, its participants. Specifically, the NASDAQ Composite, as it is looking at the wide variety of securities listed on the NASDAQ Stock Exchange.

Furthermore, indices can be used as a tool to capture broad sentiment of the market due to the number of securities within them. They can provide good snapshots of the overall health of the

market which can be used as a good historical overview of market health. However, due to the index's modification overtime, stocks are added and taken off. The index reflects different values or securities over time. This should be considered when comparing historically, as an index will not be reflecting the same securities today as they did twenty years ago. However, when thinking about comparing financial news data to stock price, we consider it relevant to compare this sentiment from news of a given period, to the overall market health of a given period.

Additionally, The NASDAQ Stock Exchange is known for having a high number of securities on their platform, both large and small cap, creating an effective way to gauge the overall market. The NASDAQ Composite Index specifically measures all Nasdaq domestic and international based common type stocks listed on The Nasdaq Stock Market (NASDAQ Composite, 2021). It is important to note that smaller companies will not have as sizeable of an impact, due to the weighing of securities by market capitalization. A disadvantage of having a mix of small and large cap companies on a weighted index is that there will be a substantial impact on the index performance if say Apple Inc. (APPL) were to have a difficult day. In figure xx below we can see both the top securities listed on this index and the overall industry breakdown. When looking at the industry breakdown, it is interesting to note that the technology sector takes close to fifty percent of the weight on this index.

TOP 10 SECURITIES BY WEIGHT			INDUSTRY BREAKDOWN		
TICKER	SECURITY	WEIGHT	INDUSTRY	WEIGHT	SECURITIES
AAPL	APPLE INC.	9.89%	Oil & Gas	0.82%	79
MSFT	MICROSOFT CORP	8.58%	Basic Materials	0.46%	52
AMZN	AMAZON.COM INC	7.52%	Industrials	6.13%	344
FB	FACEBOOK INC	3.42%	Consumer Goods	7.78%	203
GOOG	ALPHABET CL C CAP	3.27%	Health Care	10.02%	908
TSLA	TESLA; INC.	3.09%	Consumer Services	18.98%	318
GOOGL	ALPHABET CL A CMN	2.99%	Telecommunications	0.89%	19
NVDA	NVIDIA CORPORATION	1.60%	Utilities	0.77%	17
PYPL	PAYPAL HOLDINGS	1.37%	Financials	6.19%	784
INTC	INTEL CORP	1.25%	Technology	47.95%	403

All Information as of 03/31/2021

Figure 2.1. NASDAQ Composite, 2021

Due to the heavy weight of technology companies on this index, it is possible that news pertaining to technology in general will be most related to the movement of this index. It is interesting to note that there is no automotive industry included in the breakdown in figure xx, while Tesla is one of the top 10 largest companies on the index. According to Investopedia (2021) Tesla does not fit into an established sector. As its stock acts like a technology company with its extraordinarily growth patterns, yet it is fundamentally an automotive company which requires high levels of

capital to operate. By including both Tesla and the NASDAQ Composite as points of comparison, we will be able to see how each varies when compared to various text data. We will essentially be able to compare the performance of the text data to a ticker pertaining to the automotive/technology sector (TSLA), and a ticker that is looking at a range of sectors (^IXIC).

Efficient Market Hypothesis Theory

Before we can discuss previous works about stock price predictability in relation to online sentiment, we must discuss the theory that would not consider this relationship. Eugene Fama (1970) is often regarded as the “founder” of this theory through his reviewing of previous works and his providing of the standing principles of an efficient market. Though we will not go into depth regarding Fama’s (1970) works, we found that it was relevant to mention where the theory emerged from, as well as the contradictory nature it possesses in relation to our research.

According to Lo (1999, 2004) the efficient market hypothesis can be demonstrated through the principles of supply and demand. The demand curve represented by the customer who would like to make money in the market, ideally acting rationally to maximize their own returns, but each having their own personal limitations such as income. The supply curve then representing individual producers' outputs, which incorporates the price of their product. Lo (2004) relates the supply and demand of the markets to the three P’s, price, probabilities, and preferences. Price relating to the supply, demand relating to the optimization of preferences, and probabilities relating to how consumers and producers will act in the market in the future. Theorizing that the three P’s create an equilibrium across all markets (Lo, 1999). Through the individual actions of market participants making decisions in their own best interests, this market equilibrium is created. This leads to the concept of the random walk, and its underlying principle of information efficiency.

Lo (2004) describes this idea in simple terms which makes the EMH and random walk easy to grasp. A person is walking down the street and sees a 100-dollar bill (new information), this person goes to pick up the 100-dollar bill, but their companion (an economist in Lo’s example) says not to bother, as if it were a 100-dollar bill, someone would have already taken it. This example shows the essence of the hypothesis. That if a market participant would see new information that would benefit them in the market, no benefit would be realized because in an information efficient market, these benefits would have already been realized. Information efficiency leads to an efficient market, which ideally should have unpredictable price movements, also referred to as a random walk.

The random walk theory approach to price movements asserts that “all subsequent price changes represent random departures from previous price” (Malkiel, 2003). Earlier studies reviewed by Fama (1970) found that a random walk was a good model for understanding stock price, but due to information efficiency one is unable to forecast price change. The idea of this theory is based on the thought that there is a constant and un-interpretable flow of information that is constantly and immediately affecting the price of stocks. The more efficient this flow of information is, the more random the price change. Therefore, in a market which is informationally efficient, i.e., all market participants know all relevant information illustrated by Lo (2004) above; price would be unpredictable. Malkiel (2003) supports this stating that when the latest information arises, the information spreads very quickly, and it is incorporated into pieces of security immediately without delay. Both Roberts (1967) and Fama (1970) tested this hypothesis by placing structure on various information sets available to market participants.

The random walk theory suggests that anyone’s guess is as good as the experts and that when it comes to returns, because price change is not a predictable variable due to information efficiency. The theory holds due to the belief that driven by profit, all investors act on any new information that can be used to their advantage in order to maximize their opportunity. Thereby making the market even more efficient by incorporating their new information (Lo, 2004). Therefore, this theory would suggest investing in a broad-based index fund such as the NASDAQ Composite index, as it would yield a stable return due to efficient markets. In the same vein theorists would suggest that neither a technical analysis nor a fundamental analysis will yield any predictive results. Both Malkiel (2003) and Satchell (2007) state that predictive methods will not help investors to yield any greater returns than those that could be found from holding a randomly selected portfolio of stocks.

This most basic logic of this theory of course does make sense, as one individual could not intercept all information and process said information simultaneously to gauge what the best possible next move is. In this sense, information efficiency then exists. It should be specified that these theorists do allude to certain inefficiencies in the market that leave it vulnerable, many of which will be outlined in the proceeding section.

Behavioral Finance & Momentum Theory

This section's purpose is to display and examine the theoretical disagreements that exist in relation to the EMH mentioned above when incorporating human behavior into economics. The plausibility of this area became a reality with the expansion of the medias reach. The concept of momentum and the incorporation of human behavior over randomness seemed to resonate with

theorists due to the observed irrational exuberance of investors in cases of market anomalies. Malkiel (2003) states that behavioral finance thus became prominent as a branch of momentum. Thus, this section will dive into the behavioral aspects of the price movement.

The very essence of behavioral finance serves as a contradiction to that of the EMH, in the notion that people are not always rational, and markets are not always efficient. Thus, we believe it is imperative to examine this area as behavioral finance may in turn shed some light upon why individuals do not always make the decisions they are expected to make. As the EMH seeks to view the rational side of people, behavioral economists view individual market participants as simply humans, capable of being irrational and often privy to biases, heuristics, emotions etc.

Moreover, Dhankar and Maheshwari (2016) states that behavioural finance seeks to view share prices as deviating from their true fundamental value, and the existence of this deviation is due to investors not acting rational. The premise behind behavioural economics follows the notion that we ought to draw inspiration from psychology and finance to understand stock behaviour. This understanding has driven much of the work around searching for models that could explain behaviour and psychological biases. Nofer (2015) notes that since the 1990s finance researchers have aimed to show that the stock market is driven by psychology.

Aside from this, Nofer (2015) progresses to briefly describe the observed market abnormalities pertaining to periods of time that serve to contradict the EMH. These can be viewed as technical or seasonal. Nofer (2015) uses the example of calendar anomalies by explaining the January effect. He states that when the month of January takes place, the returns are typically higher compared to other months due to tax-loss selling. This is where investors aim to avoid their taxes through a process of selling shares that have performed badly throughout the year. Malkiel (2003) adds to this by stating that traders often strategically examine these patterns and utilize them to their advantage. In cases such as this, patterns such as the January effect will cease to be useful for investors once it has received considerable publicity as the anomaly will perish.

As mentioned, prior, anomalies can also be considered technical. Nofer (2015) further elucidates upon this by using the momentum effect as an example. Dhankar and Maheshwari (2016) describe it as the product of mass and velocity. In this context, it is the observed tendency for asset prices to rise further and falling prices to keep falling. The term of “momentum” thus makes sense in this context. Satchell (2007) expounds upon this in his work by describing the momentum effect as “out of sample”. This means that people will behave according to perceived success. In the context of social media, the hyper influx of content makes this a reality. This perceived success,

particularly relating to online chatter, is also referred to by Lo (2004) as noise in the market, which acts like information but is in fact a deception.

This relates quite well to the psychological aspect of behavioural finance and the way that the investor behaves. Malkiel points out that the existence of short-term momentum could be consistent with psychological feedback mechanisms (2003). When viewing the stock market from a psychological perspective, momentum can increase when individuals observe an increase in stock price, leading to investors to join in and increase momentum. Of course, we know that most people will not invest time to see returns, or if they do, they cannot deploy this strategy consistently over time and expect consistent returns. The main assumption behind the short-term predictability of stock price movement is that investors will under/overreact to latest information, not grasping the importance of the added information when it is first available, but later when the effects have already taken hold.

Lo (2004) illustrates another behavioral aspect of economics, behavioral bias. Behavioral bias according to Lo refers to irrational probability beliefs that leave the “believer” exposed to exploitation from the savvy investor, who has more consistent probability beliefs. Although behavioral biases do exist and create market inefficiencies, Lo states that they are not sustainable as market forces will exploit the biases until they no longer yield profits (2004). One part of behavioral bias can be seen in traders that act on what they assume to be new information but is just noise in the market. If enough market participants believe that the noise is indeed information, it could appear that participants are acting irrationally. Black (1986) and Lo (2004) refer to these investors as noise traders. Noise traders can demonstrate why markets cannot be informationally efficient but have varying degrees of inefficiency. This degree of inefficiency “determines the effort investors are willing to expend to gather and trade on information, hence a non-degenerate market equilibrium will arise only when there are sufficient profit opportunities, i.e., inefficiencies, to compensate investors for the costs of trading and information-gathering” (Lo, 2004). Lo then goes on to explain that the profits captured by these investors who are trading information, come from the loss of the noise traders.

Nofer (2015) states that behavioural finance researchers often refer to two types of investors. The first being rational arbitrageurs, those not prone to sentiment. Then there are noise traders that rely heavily upon sentiment and other information that is deemed nonfundamental. Noise traders often follow trends and sometimes over or under react to the news. This is quite like what Malkiel (2003) describes as the bandwagon effect, or “irrational exuberance”. Nofer (2015) asserts that noise in the market can create a substantial impact as noise traders follow other noise traders

creating positive feed-back loops, which will reconfirm their noise with actions. Like Malkiel's (2003) outline of psychological feedback systems.

Whether referring to behavioural or cyclical inconsistencies in the market, one cannot deny that there are indeed inefficiencies. The question remains, where do these inefficiencies take hold in the efficient market hypothesis and how can this theory be reconciled to incorporate the mentioned flaws above. We believe that this could imply that predictability is plausible within certain situations. This pertains heavily to the information efficiency within the market and our growing access to information.

Adaptive Markets Hypothesis

As discussed earlier, in an information efficient market, a real 100-dollar bill would not be laying around for me to pick up. Nevertheless, perhaps, someone had in fact not paid attention to their situation enough, to realize that they had lost their 100-dollar bill. A bill that I have just found, leading me to be 100-dollars richer. In turn leaving the person who lost the money to come to regret their forgetfulness, enforcing the need to hang on tighter to their money in the future, a normal human learning process.

The adaptive market hypothesis tries to explain the contradictions between efficient markets and behavioral aspects of economics. This reconciliation comes from an innovative approach which gives a favorable light on both hypotheses discussed above. Lo (2004) presents an evolutionary approach to market behavior which aims to connect the EHM and behavioral perspectives. This theory applies the principles of competition, reproduction, and natural selection to social interactions to explain human behavior in an economic and financial context (Lo, 2004).

Lo draws a parallel to his evolutionary psychology approach to a conclusion made by another researcher, Niederhoffer (1997). This researcher compares financial markets to an ecosystem with dealers as “herbivores”, speculators as “carnivores”, and floor traders and investors as “decomposers” (2004). Lo (2004) describes these roles as organisms in a cycle, as clearly there are roles where one is successful, and where one fails. As humans or organisms on this planet, our goal is to survive and thrive. Biology would suggest that throughout all living organisms' lifetimes, we have adapted progressively to survive in our environments. Lo's (2004) theory would consider the market and its participants to also be subject to the process of biology, but in a social context. He reiterates research that suggests that natural selection is not only in our genetics, but also present in our social and cultural evolution thereby making way for an explanation of the ever-changing financial markets.

Through this evolutionary framework, Lo (2004) explains the process an individual's evolution in the market. First by mentioning that the usefulness of the evolutionary psychological framework becomes plausible due to the work of Herbert Simon who suggests that individuals are not capable of the extremely elevated level of optimization that would be necessary to make consistent rational decisions in the market. Simon (1995) calls this “Bounded rationality” where individuals have a limit to their rationality, but in a hypothetical efficient market, individuals should have an “unbounded rationality”. Simon describes this as contradictory to the EMH as, individuals are incapable of consistently making rational decisions, and goes as far to say that individuals eventually end up settling with their decisions. Simon calls this satisficing, as opposed to making the best decisions, individuals will allow themselves to make decisions that are good-enough but not the best, or not- rational (Lo, 2004).

What the EMH does not explain is how a market formed of participants with bounded rationality could produce a market that is completely rational. This is what Lo's evolutionary perspective attempts to explain, that the process of human decision making is one of trial and error. He suggests that participants make their rationally bounded decisions and learn from these decisions based on positive or negative reinforcement of their decision's outcome (Lo, 2004). Throughout time these decision processes develop with new challenges and new circumstances.

Lo takes this concept of evolution one step further by comparing availability of profits to the availability of natural resources for humans. The rarer the resource and the more humans who need that resource, the fiercer the competition is, eventually causing a decline in that resource and the population and re-starting the cycle. Lo (2004) is suggesting that the cycle of the market mimics the aspects of competition only with profits instead of natural resources and consists of multiple types of competitors or species (such as retail traders or hedge funds) (2004).

This theory can then be related to investment strategies in that they go through cycles much like natural resources can and that obvious market inefficiencies such as bubbles and market crashes are participants learning and adapting to new conditions. When thinking about the market and its participants from this perspective, it seems quite logical that the best at the game will survive the longest, or the one with the most resources. Thereby eventually removing the worst traders from the competition as there is no way for them to stay in the competition without improving, as Lo coins “Survival of the richest” (Lo, 2004). The last important aspect of Lo's adaptive market hypothesis is the idea that emotion plays a key role in the trader's success, contrary to the belief that of the EMH where emotion is left out of the equation. In earlier research Lo and Repin (2002) discussed evidence that securities trader's autonomic nervous system is highly correlated with

market events, suggesting that this emotional response is key in being able to assess financial risk through the channeling of this emotion (Lo, 2004).

Through the discussion of the AMH the lines between the EMH and the behavioral critiques of the EMH seem to have been blended, due to the cooperation of evolutionary psychology and economics. While Lo's solution to the question of the efficient market seems to have some merit, the theory still would need to be researched in order to provide evidence of evolutionary markets. Lo goes further in his hypothesis to break down four implications of the AMH which helpfully depict how the market can be seen as adaptive over efficient through the actions of the participants.

The first implication as stated by Lo has to do with risk/reward preferences and that these are not constant but shaped throughout time due to the forces of natural selection. Overall risk preference is what will drive individuals to make decisions in the market and these preferences can be swayed due to outside forces such as regulation, but also size and preference of the population (Lo, 2004). An example of risk preference changing could be an entire market experiencing a bubble, some participants are forced out due to significant losses, creating a change in risk preference in the entire market due to the change in market participants.

The second has to do with occasional opportunity for arbitrage. By the EMH, such opportunities do not exist, but Lo would suggest that without such motivation to exploit the market to make additional profits, the price-discovery aspect of financial markets would collapse (Lo, 2004). Lo mimics the notion of information opportunities disappearing once they are exploited, but looking at the markets from a cyclical perspective, these opportunities will constantly arise and be exploited. The third is that investment strategies develop and change over time. This implication is relatable to Malkiel's (2003) the above-mentioned January effect, where investment strategies will have different performance based on when they are implemented and which environment, they are implemented in. The fourth is the notion that innovation is key to survival. Suggesting that rather than holding a 'sufficient' degree of risk, the AHM would suggest that risk/benefit varies over time and to maximize reward, one should adapt to the changes in the market.

Theory – Closing Thoughts

Relating these different perspectives to real world events is useful to see if in-fact we can simply point out market inefficiencies. To re-iterate market efficiency is referring to the degree to which the aggregate decisions of all market participants accurately reflect the intrinsic value of public companies and their share prices at any given time (Investopedia, 2021). As the efficient market hypothesis assumes that the market is efficient, and that securities are priced in a timely manner

when there is new information, there should be no under or overvalued stocks. With this in mind, we can consider an event that occurred in January 2021 when a stock called Game Stop (GME) experienced a dramatic increase in value by %1500. Throughout the first quarter of 2020, GME experienced a decline in value of their stock, speculated to be due to the structure of their business which is like that of a Block Buster and stores closing due to the Covid19 pandemic. In the beginning of 2020, around March the company stock was valued between 2-4 dollars per share. At this time there was a high volume of short interest in GME held by hedge funds, which could in theory drive the price of the share further down, as these positions are betting against the shorted stock. The interesting turn in GME's history is when retail investors conspired to buy and hold the shares, with no interest of selling them. The goal of this was to drive the price of the share high, so that the hedge funds who held the short positions would need to pay large sums in order to close their positions due to the lack of supply/sellers in the market. This initiative to drive the price up eventually came to fruition in January 2021 when the price went from 17 dollars on January 3rd to 347 dollars on January 26 known as the GME short squeeze. Due to this short squeeze major hedge funds such as Marvin Capital experienced huge capital losses, in this case around 50% of Marvin Capital's capital was lost to this short squeeze (Bloomberg, 2021).

The circumstances for this price increase are quite unique, as it occurred due to the existence of a blog on a platform called Reddit. A blogger called wallstreetbets with 9.7 million followers suggested vehemently to their followers that they buy and hold GME for the sole purpose of taking money from the hedge funds who shorted GME. It was widely stated that these traders did not care if they lost money (of course this would not be the case for all involved) but that they would gladly lose the money invested to see the hedge funds crash or some similar sentiment. To combat this the trading platform, hedge funds and government regulation forces immediately sought to act to correct this inefficiency. This worked momentarily as the price was driven down to around 50 dollars. This initial increase in price met momentary resistance because of the trading platform Robinhood's refusal to let their clients (retail traders) buy more shares of GME. The only action that could be taken by retail traders interested in or holding GME, was to sell shares. The traders mentioned here cannot be considered rational, albeit following their own strategy, the rationality assumed by the EMH is not in line with what was seen with GME. Still in April, we can see that the price for GME is historically high, over 150 dollars and Wallstreetbets followers are still holding shares for the sake of it. The other side of the coin is the poor risk assessment done by the hedge funds who shorted GME and similar stocks. Due to the nature of shorting activities, the one holding the short position can lose infinitely, as the price of a security could infinitely go

up, but not down. These activities together caused a huge disruption in the market that requires a second look at the market's efficiency.

Theorists that support the EMH disregard the impact of both noise and momentum in the market. The EMH suggests that noise traders only have short term effects, and that eventually the market forces take positions against them until market equilibrium is reached. The EMH would similarly disregard the impact of momentum or drive for profit by saying that these factors could not significantly impact the market, and that market forces (such as hedge funds, government, or supply & demand) will always return the market to “rational” prices (Lo, 2004). It is interesting to note that regulatory forces (government) or highly influential investors (hedge funds) are considered market forces. More specifically, that these market forces are considered powerful enough by the EMH to overcome behavioral biases. As we can see from current events, these regulatory forces, while powerful, may not be able to return the market to equilibrium.

Overall, there are many interesting takes and discussions on the market and its level of efficiency. The primary argument that we operate within an efficient market with rational participants has had its doubters since the theory first came in the late 70's. Our main reservations with the efficient market hypothesis are the presented behavioral aspects of the market participants. When attempting to find a relationship between price movements and social sentiment, there would need to be some underlying relationship between human action and the stock market. Due to this we considered Lo's reworking of the EHM to be an interesting take that would consider a relationship between sentiment online and the movement of price. As we are living in a technologically advancing world, we should expect that our ability to capture and interpret information might be greater than it was thirty years ago. Due to this we would expect that the way we interpret our markets could also adapt. Though this does not necessarily mean that stock movements are becoming predictable as our information interpretation becomes more efficient. There are merely aspects of our growing store of online data that can lead to insights about the movements of the market.

2.2 Social Media & Machine Learning: Tools for harnessing text data

The following sections will outline the use of social media as potential market disruptor in the game of stock prediction. As well as the modern applications of machine learning to harness the power of social media and online media in general.

Social Media: A tool for gauging moods online

With the exponential growth of social media throughout the last couple of decades, the rapid spreading of personalized content and personalization of information has never been so

prominent. As a result of this, social media platforms have become a fruitful source of information to be tapped into. As content is personalized and spreads quickly, it allows researchers to explore the minds of the users and apply their findings to various fields that concern human interaction. For example, companies now know their customers more intimately than ever before, capturing everything from their shopping habits to how the public perceives the brand online. These platforms not only open a window to follow a single user, but follow groups and observe how they grow, form and influence. Over the last two decades, we have seen a boom of information, first from our computers, now from devices in our pockets. With the rise of the internet and social media, information has exploded as a resource, available for everyone who has the means to gather it. It is interesting to think of social media as a tool that could be used tap into the groups of information all over the world, giving insights into how people are feeling and learning how to interpret those feelings and how they influence the world at the time. As discussed above, there is a question of how behavioral biases can influence market efficiency, tools such as social media can give insights into how these biases will form and affect price movements.

In the past, the understanding of stock price has been limited within the realm of econometric tools. Many theorists, however, have strived to prove the inefficiency of these models and how they neglect to take into consideration the plethora of information that we have within society. Ji et al. (2021) discussed the limitations of previous econometric models that were once used to understand price fluctuations. The belief is that previous models were incapable of understanding all the underlying elements that could impact price itself. This is what has brought us to the examination of social media. We will not delve deep into some of the econometric tools and their understanding of price fluctuation. However, we will continue this section with the aim of gaining an overarching understanding of social media, and some of the tools that emerged in relation to price movement and prediction using social media data.

Overview of Twitter as a Social Media Source

Today we can find various social media platforms consisting of microblogs (Twitter), content communities (YouTube), and social networks (Facebook). These applications according Kaplan and Haenlein (2010) are based on the logical foundations of web 2.0 which allows the creation and exchange of information. For clarification, Web 2.0 merely refers to websites that emphasize user generated content through a participatory culture. The platforms themselves can typically have their data sources tapped and utilized for multiple purposes. A user on one of these platforms can generate copious amounts of data, as they can create as much as they would like. Understandably, it would be quite cumbersome for an individual to not only search through the entirety of one

users' Twitter page, but to also derive meaningful information from it. Thanks to the modern technology that gave us social media, this tedious task can be avoided by using data mining techniques. Platforms such as those mentioned above, have created API developer pages that allow this content to be made publicly accessible (Nofer, 2015). By making this type of data publicly accessible creates value for both researchers, individuals, and companies alike. While a company's motives may be slightly more insidious, researchers like us can create great insights into the world around us because individuals post valuable information about themselves which can be used to generate insights (Nofer, 2015).

Researchers Nti *et al.* (2020) approached the prediction of stock price by using multiple sources online, including online news, tweets, Google trends and forum discussions and then progressed to obtain sentiment from the text objects. The conclusion from their work indicated that the social media source that generated the most utility was in fact the platform Twitter. Macy *et al.* (2015) echoes the usefulness of Twitter as a data source. They highlighted the general scope of Twitter's data capabilities and indicated why it is more valuable than say other types of sources. They progressed to state that opinion sharing is one of the primary reasons for this richness and that interactions amongst users can be seen in the form of retweeting or following. The value in this case stems from the hyper expansion of thoughts. Inversely, we may not see this in the New York times where interactions are limited.

In addition, Macy *et al.* (2015) describes the various aspects of tweets. By aspects, we are referring to what a tweet typically contains. We found this description to be quite meaningful as it will serve as a point of departure within our methodological choices. Macy *et al.* (2015) state that twitter provides a way for individuals to create and see tweets on their feed. Feed refers to their user page. This is where the user posts their information to, as well as receives latest information from. Users interact with other information by who they follow and whose tweets they like, by doing this, users are following conversations of their choosing.

The content on the platform can come in many forms, these include original content, directed tweets, retweets and quote tweets. Directed tweets refer to if a user is tagging a specific user in the beginning of their tweet which can typically indicate a conversation between users, these tweets might not hold any special interest for this project. Whereas a retweet can be a simple re-post of another user's content that the user finds interesting or relevant. Retweets are particularly interesting as it represents the transmission of the message throughout the platform, through this we could see the importance of different tweets Macy *et al.* (2015). Quote tweets are like a retweet

but include a user's own content with that tweet. Additionally, the number of likes a Tweet receives could show supplementary importance of a tweet on the platform.



Figure 2.2. Screenshot of Elon Musk's tweet.

From this figure xx we can see that Twitter focuses on the number of retweets, Quote tweets and Likes for measurements of a tweets success and these are metrics that should also be included to derive the importance of a tweet and the impact of the tweet and its sentiment in our study. Now that we have established the value of Twitter as a source of data, we find it relevant to mention one of the many uses of Twitter in a business context.

Social Media and CEO's

We believe it imperative to examine some of the literature regarding Twitter and CEOs as our subject for this paper revolves around Elon Musk. We discovered a couple of interesting opinions regarding CEO's leveraging their social media to yield value. The first being Malhotra & Malhotra, (2016), and their study on how CEO's can use their social data. They believe that many CEO's do not effectively harness their social media accounts. They provide an example of Elon Musk, and outline his overactive behaviour online. In doing so, they detail that he tweets numerous times a month, and the tweets themselves are comprised of both new and exciting information. They use the example of March 2015, where Elon Musk tweeted about a new Tesla product. He hinted in this tweet that it would not be a car. In doing so, he added a level of mystery that could easily intrigue his followers and investors. The news in this context was not disclosed through any other outlet. Twitter was the only platform that had this valuable information. Cases like Elon's are what prompted to Malhotra & Malhotra (2016) to claim that CEO's have an enormity of power through their Twitter accounts. They can readily bypass the "waiting time" that it takes for a company to spread their news and have a significant amount of control over the general narrative.

Granted their understanding of this is purely speculative and based upon their opinion that it will yield success. They did not provide any grounding information that can validate their assumption and merely declared a series of benefits that individuals can obtain from leveraging their Twitter accounts. They assert that by using Twitter it can directly result in creating positive sentiment around the business. In addition, they found that business related tweets were strongly correlated with a positive movement. The limitation in this context arises from the fact that they quite readily use the word correlation without any indication of the tools that were used to obtain these results. This makes it difficult to obtain a definitive answer regarding the weight of CEO's. Also, their emphasis upon Elon Musk leaves us pondering the legitimacy of this argument. Are CEOs in general capable of this type of power? Can a CEO have a direct impact on stock price?

Other researchers refute this claim that all CEO's can possess this power. For example, Strauss & Smith (2019) rebut this notion in their deep dive of Elon Musk as a case study. Their study touches upon how communication regarding a specific corporate event may in term frame the event itself as well as the market reactions. Their area of focus was on Tesla and various channels of communication. What was thought-provoking was that they chose a multi-method event study that combined text analysis and the abnormal returns of price. The event itself that they chose was the release of a new battery pack for Model S and X in 2016.

Although it is not prediction based, the study itself alludes to the correlative/causal relationship that exists between Elon Musk and the share price of Tesla. Their belief is that most of the studies pertaining to prediction, and such have not focused on the complex dynamics that occur within a given event. Accordingly, their study focuses upon price movement from a micro perspective within a period that focuses upon the constant stream of information that pours out onto the internet. Their rebuttal of the above stems from the overall limitations and discussion of their study. They conclude that this type of CEO communication should not be applied broadly across all listed companies. Thus, Elon in his social media prowess may be considered the exception as opposed to the rule.

We find that Strauss & Smith's (2019) work should be taken lightly. This is exhibited in the inherent limitations found within their methodology. The first limitation is that they chose to use two coders to analyse tweets in real time. By real time, we mean that they examined events as they were happening and recorded and coded the data accordingly. If machines have a challenging time predicting price with the cornucopia of information they receive, there is a limited chance that two human brains can readily receive and analyse this effectively. It could also be argued that what they have done simulates a real time investor; however, it is inadequate from a data handling

perspective. In addition, for a task of determining price fluctuation, it seems inappropriate. This comes from the mere fact that it does not possess the same kinds of checks and balances that machine learning provides. By checks and balances we refer to the cross validation and removal of bias. Lastly, even though they were quite limited in their approach they did notice that announcements made by corporations themselves can trigger the participants in the market to react immediately. This thereby evokes a stream of additional news reporting.

It leaves us wondering regarding the extent of the work concerning the impact of CEOs on the stock market. Is Elon Musk indeed a market anomaly, or is he merely a first mover on a trend that has yet to reach fruition? These are questions that we unfortunately cannot answer due to data constraints. However, we think that it is fascinating to posit.

Machine Learning: A tool for prediction

As we have already outlined the previous types of analysis (fundamental and technical) typically associated with the stock market. It makes sense that we shall outline machine learning and how it is often seen as the next phase of determining investor behavior. Whenever we hear the words machine learning and artificial intelligence, our brains immediately picture something along the lines of a machine from *The Terminator*. Alas, the area itself is a lot less complicated and significantly less scary. Typically, machine learning merely refers to code that can be applied quite broadly to generate insights from data. The data in this instance that we shall place emphasis upon is social media data. With this being said, the following section will elaborate upon what machine learning essentially is as well as the ways in which it is used in conjunction with social media data in relation to the stock market.

Machine Learning - What is it?

Machine learning has been described by Müller & Guido (2016) as the intersection of statistics, artificial intelligence, and computer science. The very essence of machine learning lies in its extraction of meaning from data. It does this through a process of prediction where the application is fed a series of data points, and from there it will deduce patterns and make informed decisions based upon these patterns. Several types of machine learning models are used in this process to find meaning within the data. The types of algorithms that are used are known as supervised and unsupervised learning. With supervised learning, the algorithm itself knows both its input and output data. With unsupervised learning, the model only knows the input data. It will then guess an output based upon patterns within the data. We have decided to describe this area as a large

portion of stock market analysis in the last decade involves the development and deployment of machine learning models for prediction.

To relate this back to technical and fundamental and how they will be used in the progression of our paper we can look at Nti *et al.* (2020) outline of the modern understanding of how technical and fundamental analysis can work alongside one another. They use the following figure to display an overview of how this can work in conjunction with predictive models. Fundamental and technical data both serve as input datasets and the output will be a predictive value. Machine learning is viewed as the evolutionary next step in terms of analytics tools.

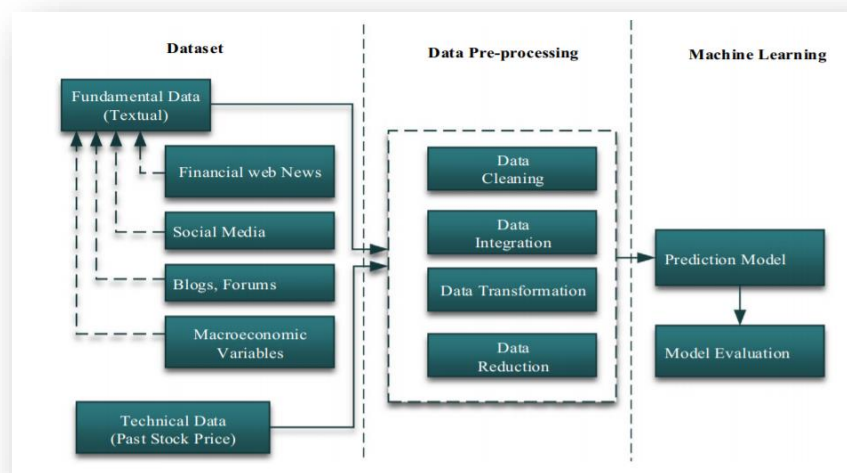


Figure 2.3. Technical, fundamental and machine learning taken from Nti, Adekoya, & Weyori's (2020)

How is this related to Artificial Intelligence?

As we mentioned above, machine learning is a subset or intersection of artificial intelligence. Often artificial intelligence and machine learning applications are viewed as synonymous with one another. However, with artificial intelligence, we are referring to systems that seek to imitate human behavior. Machine learning focuses heavily upon the use of data and has a narrow scope in terms of capabilities. Artificial intelligence conversely focuses upon the possibilities that can emerge from technology, whether this may pertain to engineering or computer science. It is the overarching field that encourages advancement within its subsets. If there are advancements within a machine learning algorithm, this is quite positive for the field of artificial intelligence in general.

To make this a little clearer, we will give an example of how and why machine learning is a branch of artificial intelligence. Machine learning algorithms rely on different models for different tasks.

A neural network for instance, can build a complex model that seeks to mimic the ways in which neurons within the human brain interact. To do so, machine learning models predict based upon a series of data. Input data are fed into the model, and from there the calculations are applied. The result is an output that makes an estimate based upon patterns it has noticed. We can see in this rather broad example that machine learning algorithms use models to replicate human behavior.

With the progression of technology and social media, these tools are heavily ingrained within our society to the point in which they are thought to be ubiquitous with our everyday lives. They are found within recommendation systems, the personalized feed we see on our Facebook pages, the recognition of your friends in a social media post, etc. It makes sense that the machine learning applications that are successful are those that seek to automate common decision-making processes. It is this reason this tool is often used within stock price prediction. Stock price prediction using machine learning applications seeks to replicate the rationale that an investor makes on an investment.

Text Analysis in Machine Learning

As we have now covered machine learning and what it entails, it is now imperative to outline text analysis, or text mining. Machine learning for text analytics involves using a set of statistical techniques that allow individuals to identify parts of speech, entities, sentiment etc. These techniques can be expressed as a model that can then be applied to other text. It can be also used to deduce meaning from clusters of data.

Text data requires a specific approach to machine learning due to its extraordinarily complex nature. Aside from being unstructured, the data itself is highly dimensional and may often contain noise (Aggarwal *et al.*, 2012). What adds further complexity to understanding the data is processing it on a syntactic, semantic, and pragmatic level. The syntax deals with the of the ways in which the words are arranged. The semantics refers to the general meaning behind the word. Lastly, pragmatics handles the meaning of the entire sentence. These elements can make it problematic to process and model the data. This process can be made even more so cumbersome when the dataset itself is quite large. Large volumes of text data, otherwise referred to as corpora, make it difficult computationally speaking to handle (Brownlee, 2020). Therefore, there is a significant amount of pre-processing is often required before undertaking this sort of analysis. With the above in mind, it is especially important to consider the diverse types of text analytics.

This is what brings us to text sentiment analysis. Upon researching some of the previous works in relation to stock market prediction using social media, we observed that a substantial portion of

the research revolved around text sentiment analysis. It is also commonly referred to as emotional polarity computation and has become a “flourishing frontier in the text mining community” (Li & Wu, 2010). The entire purpose of sentiment analysis is that it allows the researcher to determine the general mood or attitude of the person or writer in relation to a particular topic. The purpose of sentiment analysis in this context would be to assign meaning to a tweet.

Why is it important to outline the above? These definitions and parameters are vital to understand for a multitude of reasons. Often artificial intelligence, machine learning, sentiment analysis, and text mining can be viewed as either the exact same or vastly different entities from one another. It is important to distinguish between them as it can enable us to narrow our literature down even further. Moreover, it can aid us in our critique and evaluation of the methods that have been used to predict stock market movement. It can even allow us to identify why this is such a highly contested area.

Previous methods using Social Media

This section will now deep dive into the different methodologies and perspectives surrounding text analytics and the stock market. As we began our research, we noticed that the methodology varied significantly according to each respective researcher. The differences themselves ranged across several areas: data types, model selection and analysis type, and learning approach. In addition, the choice of data type varied significantly. This can range from stock index to the origins of the text data. Lastly, the most noteworthy way in which the previous works differed from one another was in the choice of machine learning model and learning type. So, whether they used supervised or unsupervised learning.

Sentiment Analysis Techniques

Before diving into the research below, we think it is first import to inform the reader on the basic understanding of sentiment analysis. In its most basic uses, sentiment analysis is used to derive meaning from text data, such as tweets. Specifically, sentiment analysis is looking to find subjectivity and polarity from the text. A subjective text statement would contain non-factual information, whereas an objective statement would carry the opposite. Separately, sentiment polarity is looking to see if the text has a positive, negative, or neutral sentiment (Dhaoui *et al.*, 2017).

Most researchers within stock prediction and social media analysis chose to analyse text data using sentiment, but sentiment analysis itself can be thought of as umbrella of analysis, therefore not specific to one technique. Dhaoui *et al.* (2017) outline two prominent approaches to automated

sentiment analysis. The first involves classification using a lexicon of weighted words. Lexicon-based sentiment analysis refers to the use of an existing dictionary, within that dictionary all words are pre-labelled with both their polarity (positive, negative, or neutral), subjectivity, as well as mood and other sentiment indicators. Text phrases can then be mapped to this lexicon and categorized. This is done through tokenizing the individual words within the text based on the lexicon, where the words are then combined to find a final sentiment score. We shall dive deeper into lexicon-based sentiment later in this section. This approach is typically used amongst marketing researchers as it does not involve any pre-processing or training of a classifier.

Alternatively, there is the machine learning approach to sentiment analysis. As this is still sentiment analysis, the technique is looking for the same results, but simply using a different method. The text data in this case will come with a pre-labelled data set with sentiment polarity as well as subjective and objectivity. A classification model will then need to be built, as its job is to then to learn the assigned labels and be able to predict future sentiment polarity, subjectivity, and objectivity. This machine learning approach to sentiment analysis is also used quite often amongst marketing researchers however it possesses a training phase of the data that is either conducted by the researchers themselves or by the sentiment software itself.

There is a lot of disagreement amongst researchers in terms of which approach should be used as it can impact the accuracy of sentiment classification itself. Lexicon approaches in some instances are viewed as less effective than machine learning. However, there are some cases in the literature that advocate for a combined approach indicating that using either approach on their own is not optimal (Dhaoui *et al.*, 2017). There have been several attempts where researchers have attempted to combine both, and these studies use a lexicon-based sentiment to label the data and then use this as a training set for the machine learning models. This was a prominent trend within the literature regarding stock market analysis and sentiment. The following shall outline the two distinct forms of analysis and how in some cases they have been combined with one another.

Lexicon based approaches

The first, lexicon-based approach involves finding a seed list of opinion words. This involves either using a dictionary-based approach that searches a dictionary for certain words, be those synonyms or antonyms (Madhoushi *et al.*, 2015). This is a corpus approach which starts with a list of opinion words and then finds other words in a large body of text to assist in finding words with context specific orientations. If the word does not have an exact match, then the dictionary will search for synonyms or antonyms that could match. For example, if a tweet has the word “stupendous,” and the dictionary does not contain this, then the next best step is to search for a synonym. In this

case, the synonym could be “amazing” or “great” or it could simply just take the root of the original word. Nguyen *et al.* (2015) state that with this technique of forming root words, or simplifying the data, the dictionaries will use the simplified data to identify the sentiment of the text. This is done by giving the words negative and positive labels, the phrases are then counted and given weights based upon their level of negativism. This type of sentiment often includes what is described as three polarity classes (Hasan *et al.* 2018). These are positive, negative, and neutral. These words typically have a score associated with them. The scale itself is found between -1 and 1. The closer the word is to -1 the more likely it is to be a negative word. The closer the word is to 1, the more likely the word is going to be positive. In our examination of the previous work in the area, we found that a sizable portion of the research attempted to utilize dictionaries in their endeavor to extract meaning from the text data. This is in itself somewhat limiting to apply broadly to a data set as sometimes words are misspelled or the dictionary itself may not contain the words. Whilst researching, we did not actually uncover any researcher than solely focused upon lexicon. There were a few instances in which researchers did employ this; however, it was solely used as a means of criticism, or used as a means to an end. By this, we mean that the researchers chose to combine the lexicon approach with a machine learning algorithm.

Combined Approaches

When thinking about yourself, or human emotion in general, it is quite hard to restrict emotion into the three categories as mentioned above, let alone the derived context or if the person is serious or sarcastic. This led us to wonder if there were other approaches beyond this limiting three mood categorization of text data, whose creator was human. The work that resonated with us when examining was that of Mittal and Goel (2012) and Bollen *et al.* (2011). Their papers view current sentiment analysis approaches as under-developed. They criticized it quite heavily due to its binary nature. They argued that when they attempted to separate social media content into categories it was simply insufficient to model human emotion according to two or three categories. What we found useful from their work was their combining of lexicon and machine learning tools.

In the case of Mittal and Goel (2012), they ran into issues when performing sentiment analysis, finding that **OpinionFinder** or **SentiWord** were not suitable for prediction. To briefly explain, **OpinionFinder** and **SentiWord** are software packages for sentiment analysis which can identify binary emotional polarity of a text (Bollen *et al.*, 2011). These are traditional lexicon-based approaches that can merely be applied to a data set. They found the binary nature to sentiment to be quite limiting. Due to this they created their own word list for sentiment analysis based on Profile of Mood States questionnaire. Here we have quite a clear and succinct example of

researchers attempting to find accuracy in the merging of two areas of analysis. Mittal and Goel assert that POMS is an established psychometric questionnaire which asks a person to rate his/her current mood by answering 65 different questions on a scale of 1 to 5.

For example, rate on a scale of 1 to 5 how tensed you feel today? (Mittal & Goel, 2012). These 65 words are then associated with the standard 6 POMS moods tension, depression, anger, vigor, fatigue, and confusion. The authors then took the original moods and created their own formulas for deducing what happiness, calm, alert, and kind would entail. Then all the tweets in a dataset would be mapped according to the specific words.

Table 1: p-values obtained using Granger causality analysis with different lags (in days)

Lag	Calm	Happy	Alert	Kind
1	0.0207	0.4501	0.0345	0.0775
2	0.0336	0.1849	0.1063	0.1038
3	0.0106	0.0658	0.1679	0.1123
4	0.0069	0.0682	0.3257	0.1810
5	0.0100	0.0798	0.1151	0.1157

Figure 2.4 Mittal & Goel (2012) p-values

We can see that the value improves for the moods Calm and Happy as the lag of days increases but this is not the case for the moods Alert and Kind. This is an interesting approach of analysis and should help improve our understanding of stock movement and moods. It should be stated that their understanding of sentiment could equally be considered as cumbersome towards understanding human emotion. While this questionnaire does expand the human emotion spectrum to four additional moods, this is still a limited view of the human brain. Human emotion is most assuredly more than simple classes of classification.

Table 2: DJIA 5-SCV Accuracy Using 4 Different Algorithms

Algorithm	Evaluation	I_D	I_{CD}	I_{CHD}	I_{CAD}	I_{CKD}	I_{CHAD}	I_{CHKD}
Linear Regression	MAPE	7.28%	7.26%	7.66%	7.05%	7.43%	7.57%	7.78%
	Direction	64.44%	64.44%	71.11%	64.44%	64.44%	68.89%	71.11%
Logistic Regression	Direction	60%	60%	60%	60%	60%	60%	60%
SVM	Direction	59.75%	59.75%	59.75%	59.75%	59.75%	59.75%	59.75%
SOFNN	MAPE	9.71%	9.66%	11.03%	9.22%	11%	10.52%	11.78%
	Direction	64.44%	71.11%	75.56%	68.89%	73.33%	73.33%	73.33%

Figure 2.5 Machine learning algorithm results from Mittal and Goel analysis, 2012

Bollen *et al.* (2011) had a similar idea in relation to sentiment analysis in general being quite simplistic. They measured six different moods, which are calm, alert, sure, vital, kind, and happy.

Bollen *et al.* (2011) similarly found issue with the reduction of human emotion to a mere four features in their attempt to gain an understanding of the human brain. From this we can see that both authors were able to find a more descriptive analysis of a given days mood. They made their own dictionary and called it Google Profile of the Moods (G-POMS). G-POMS specifically does this by analyzing word co-occurrences from a collection of 1 trillion-word tokens collected from public websites, enabling the tool to have a larger selection of mood terms apart from positive and negative (Bollen *et al.*, 2011). We found this area of sentiment research and stock prediction to be particularly enlightening as it did indeed indicate the need for the expansion of human understanding. They extend the original POMs question to a lexicon of 964 associated terms by analyzing co-occurrences in a collection of 2.5 billion 4 and 5 grams. Their lexicon can then be applied to data. Their methodological approach can be seen in the figure below.

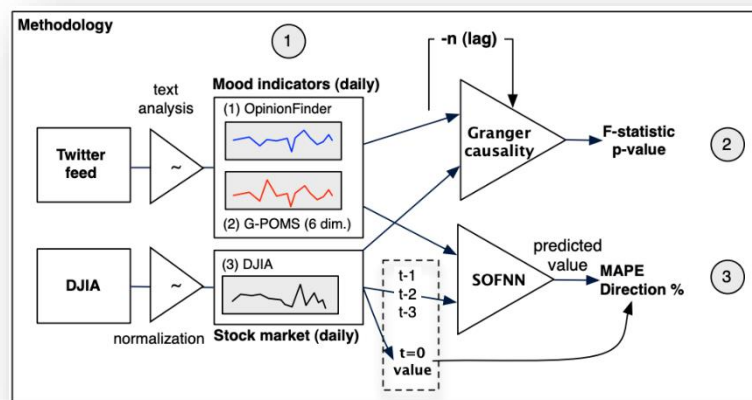


Figure 2.6. Methodology of Bollen et al. (2011)

Beyond the correlation of a given mood and the Dow Jones Industrial Average (DJIA), Bollen *et al.* (2011) also mentioned the linearity of granger causality and the perspective that the relationship between mood and stock price can be considered non-linear. To assess whether public mood can predict stock price, they used a self-organizing fuzzy neural network (SOFNN). A SOFNN is newer algorithm which attempts to automate structure and parameter identification simultaneously based on input target samples. A self-organizing cluster approach is used to create the structure and initial parameters, which are then fine-tuned with a supervised machine learning method (Qiao & Wang, 2008). In this case the SOFNN assessed previous three days of DJIA and various permutations of the mood timeseries done in the previous analysis. The results of this SOFNN showed that including calm tweets in the model significantly improved its performance. Additionally, it should be mentioned that sentiment from the first tool **OpinionFinder** did not increase the performance of SOFNN, meaning that it is not a valuable predictor. This means that

the original lexicon-based approach was ineffective in generating meaning. Their combined approach of labelling their own data and creating a lexicon was more effective.

Bollen *et al.*'s (2011) overall concluding results were that they believed there to be an indication of correlation between the measurements of the mood states found from Twitter feeds and the DJIA. Alas, they do stress that this does not necessarily indicate causality, but they do indicate that the public interest exists in relation to the DJIA, and this illustrates the possible impact of their investment decisions.

These researchers sought to combine the best of machine learning with lexicon approaches. The task that they undertook was not one that sought to avoid manually labelling the data, but it was one that sought to combine both approaches and leverage their strengths. In addition, the added complexity of extending human emotion beyond the three typical categories added further value to their studies.

Machine learning approach

According to Dhaoui *et al.* (2017), “the objective of using a machine learning technique is to train classifiers from examples to perform the category assignments automatically”. This brings us back to what input data we feed the models. Machine learning has flourished quite significantly within sentiment analysis due to the model's ability to automate and handle substantial amounts of data. Moving through previous literature we noticed that most of the work lay within the bounds of supervised learning. Before we can progress to outlining the areas of research, it could make sense to state how it is different from the lexical based approaches. With lexical, there is a set dictionary of words. With machine learning, a researcher will typically go through each line of text and will label this as positive or negative. It is then specific to the individual data set. It can be quite confusing for the reader as both approaches are similar and can easily be combined. We saw this with Mittal and Goel (2012) and Bollen *et al.* (2011).

As sentiment falls within the realm of classification-based tasks it makes sense that many of the previous areas of research utilized algorithms such as a Naïve Bayes, Support Vector Machine (SVM), and neural networks. We will begin by explaining some of work within the field of machine learning sentiment analysis. The key works that resonated with us was Sprenger *et al.* (2014), Ranco *et al.* (2015), and Nti *et al.* (2020). Each respective work utilizes specific machine learning models which give us some insight into the nature of the model and its performance with specific data types.

Sprenger *et al.* (2014) presented a methodology for a broad range of news events based on microblogging messages. Their study was specifically event driven as they wanted to examine a series of events and their impact upon stock price. The challenge they specified that resonated with us was that often choosing the accurate timing of events can be difficult as time stamps online often do not accurately reflect the event in time. Moreover, they also specified the challenges that emerge with merely calculating “good versus bad” news. Lastly, they had multitude of conflicting news sources and their capturing of the news can be a difficult fact to account for. In other words, different news channels may report events differently. Therefore, sentiment can vary quite drastically online.

Their use of Naïve Bayesian text classification aimed to extract the sentiment and event type and/or news category from the text. Their approach examined the probability of the message belonging to a particular class is calculated through the conditional probability of the words occurring in a document. They manually coded the conditional probability of over 2500 tweets themselves and classified it according to sentiment and event type (Sprenger *et al.*, 2014). Ranco *et al.* (2015) later criticized this work and found the model to be too simplistic in terms of understanding text data.

Ranco *et al.* (2015) approached the sentiment analysis from the perspective of supervised learning. They stipulated that they calculated the sentiment through using a supervising learning method. Specifically, they used around 10 financial experts to label over 100,000 tweets. This was then used to build a Support Vector Machine. Their model discriminated between the different labels we discussed earlier (positive, negative, and neutral). The small set of prelabelled tweets was then parred against a dataset of over 1.5 million tweets. The polarity results were later utilized in conjunction with grangers causality and Pearson’s correlation. They deemed their study to be rather accurate in its calculation of model accuracy utilizing the SVM, and that further research into the model itself would merely result in overfitting. More importantly, they stipulated that it surpassed other previous works done by that of Sprenger *et al.* (2014) where their Naïve Bayes model for sentiment classification on Twitter data resulted in an accuracy of around 64.2%.

Upon further investigation, we found Nti *et al.* (2020) utilized a more complex model. They assess the correlation between the public sentiment and the future stock price using an Artificial Neural Network. Their initial assumption states that they believe when sentiments or the emotions of investors are low, or there is a general distrust in the media this can cause stock price to drop. As we have seen in the other lexical approaches and their criticisms of opinion mining, this is a one-dimensional understanding of the complexities of human behaviour.

Nti *et al.* (2020) retrieve their results through a process of using various sources of text data, and from there they use these sources to predict the price movement on the Ghana stock exchange. They utilized positive and negative sentiment as classes for predicting change in future stock price. Prior to doing so, they break up their dataset into 80% training data and 20% testing data. From there, they utilized the Multi-Layer Perception Artificial Neural Network and provided the best working tuning parameters that they found. The best accuracy that they were able to yield was 77.12%. This is high given some of the other areas of research. However, we should stress that these results should be taken lightly as there are numerous limitations that are identifiable. The first being that this research pertains solely to the Ghana stock exchange. The model itself is not generalizable to other geographical areas. Nonetheless, their use of multiple social media sources indicates that they did validate their results.

In the paper, there was a difference in results when considering the days ahead. It was quite interesting to note that the further they attempted to predict, the more accurate their results appeared to be. This contradicts most of theory regarding both behavioral and the EMH. As behavioral finance seems to indicate that emotions can vary quite significantly, thus the ability to assume a constant mood seems to be a weak argument. This is interesting as it appears to contradict with some of the research regarding strong form predictability. Satchell (2007) argues that using information about previous stock prices may be considered a fragile methodological approach. Therefore, Nti *et al.* (2020) approach contains its limitations. Nevertheless, their approach does possess merit in using of all publicly available information. Considering that Satchell (2007) made this assumption over a decade ago the initial understanding of data behavior could be limited.

Now we will present the results of the different sections as a means of comparing them. We have first examined how they approach the task. Now we shall display the accuracy of their tools. The results overall are quite interesting. From an initial glance, we can see that the combined approach appears to have yielded the best results. Nonetheless, we can see that the machine learning methods have also improved with time. Each of the respective researchers that utilized machine learning, all seemed to improve upon their predecessors works in terms of model accuracy. More importantly, an understanding of the nature of human behavior as well as the models themselves seems to have taken hold.

Source	Method Type	Best Model	Accuracy
Mittal and Goel (2012)	Combined	Fuzzy Neural Network	75%
Bollen <i>et al.</i> (2011)	Combined	Fuzzy Neural Network	87%
Sprenger <i>et al.</i> (2014)	Machine Learning	Naïve Bayesian	64.2%

Ranco <i>et al.</i> (2015)	Machine Learning	Linear SVM	77%
Nti <i>et al.</i> (2020)	Machine Learning	Neural Network	77.12%

Table 2.1. Summary of previous works results.

Final Overall thoughts on machine learning

In terms of the above literature in the machine learning section, a limitation that they all have in common is the lack of presenting the coefficient score of the positive and negative words. A common tool utilized within machine learning is the ability to print which words are associated with positive, negative, and neutral. Typically, the models themselves will weigh the words differently and will attempt to assign them a score. These coefficients have distinct levels of importance and can thus hinder the model's predictability. It could have been useful to examine what the models view as important.

The literature covered several different model types as well as various data sources. We were able to gain quite a broad overview of what models may perform the best with our data. The only limitation in terms of model selection was the mentioning of the tuning of parameters. We felt as though none of the theorists really delved into detail regarding how they treated the models. More importantly, in terms of text analytics the insights themselves were lacking. We believe that they could have benefited more from really deep diving into the social media data and understanding the nuances of the data itself.

Drawing Parallels to Theory

When drawing parallels to behavioral finance, it illustrates the desired need for an improved method. While all these methods are an attempt to interpret or mimic human emotion, we believe that future modeling requires a more diverse understanding of mood through text, which is not a simple feat. We can also see this need for improvement when comparing to the adaptive market hypothesis. As we attempt to understand the learning cycles that market participants and the market itself experience, we will need to better interpret the data from social channels. GPOMs was the closest to this notion, however it is not to say that we believe that the GPOMS method is the sought-after solution. The GPOMS approach is limited in focusing upon a singular psychometric tool. As there are a cornucopia of tools that could easily be as useful, this could itself contain bias. However, it could be worth considering in terms of the next step towards understanding mood of the public and by extension behavioral biases.

One researcher using a moods test is not a sufficient answer to a complex problem of understanding stock movement. There ought to be a more complex way of achieving this. It does

seem counterintuitive for a singular researcher or team to define the moods. Whilst we all can understand basic emotions, not all of us having the complexity to understand behavior to its fullest. Correspondingly, this makes sense in the context of machine learning approaches within sentiment. We found that most of the previous works regarding sentiment machine learning also chose to categorize their text data based upon the binary positive and negative classification labels. We found that the researchers within sentiment machine learning tend to label the tweets positive or negative. In some of the cases, this panned over thousands of works. It bares the same singular approach to that of G-POMS where a group of researchers are subjectively deciding what constitutes these different moods. It makes sense in many ways that EMH tends to take precedence over other theories in terms of favorability, when considering how limited the existing methodology is. This now brings us to the following section where we will attempt to analyze text and stock data utilizing machine learning methods. After reviewing the previous works regarding sentiment, we shall now dive deeper into the previous works regarding machine learning methods, as a means of later driving our methodological process.

Summary of Literature

This section contains an overview of the literature contained in this previous section. It additionally sorts this literature into themes contained in this research then further describes the literature in detail of what the specific piece of literature contains. This can be found in our appendix F.

Main Hypothesis

A sizable portion of the literature regarding social media data and stock price movement in general focuses upon text data and what impact it has upon the stock price movement. We have thus, after reviewing all the previous literature, arrived at a predisposition in relation to what our results may indicate. The H_a refers to the alternative hypothesis, which is the hypothesis we will test if true (Illowsky & Dean, 2013). The null hypothesis is the hypothesis we will hope to reject, this is labelled with H_0 . The aim is to check which is most likely.

- H_a = Tweet data will have a large impact on detecting price movement
- H_0 = *Tweet data will not have a large impact on detecting price movement*

After we have determined from our results which of the hypothesis, we will then decide. There are several options that we can choose from. We can reject the reject H_0 if the data favours the alternative, or we can choose to “not reject” and lastly, we can choose to “decline to reject H_0 ” (Illowsky & Dean, 2013).

3 Methodology

As we have adopted a concurrent embedded design within the mixed method approach, we shall progress with a single phase of data collection, thereby collecting both qualitative and qualitative data. A common phrase that we need discuss before we begin the data collection process is the Application Programming Interface (API). This is an application that allows two software applications to speak to one another. So, in the case of our work we will establish an API connection in our code to obtain our data. In the case of our paper, we shall use both an Application Programming Interface (API), and existing datasets. Based upon our literature review, it occurred to us that the criticisms in this area of research seem to revolve around the validity of the results. The primary criticism stems from the singular use of one media source, and one subject. To negate these assumptions, we focus upon more than one and use it to gain a better understanding of the data and its relationship to stock movement. As part of this process, we shall conduct an initial exploratory analysis. This will involve examining the data prior to modeling. From there we shall move towards the processing and data transformation. Afterwards, we shall model our data using machine learning models, these results shall be visualized.

3.1 Data & Data Collection

Before we can model anything or even visualize, we need to collect data. As our topic is quite robust, it requires several sources of data to obtain the most trustworthy and accurate results. We have decided for the purpose of efficiency to assign our data groupings. This will be used throughout the data collection, description, and exploration sections. The key groups of data we have identified are as follows:

→ Group 1: Tweets of Elon Musk

- *Group 1.1 Elon Musk tweets from 2010 to 2017*
- *Group 1.2 Elon Musk tweets from 2015 to 2020*

→ Group 2: US financial news data set

→ Group 3: Yahoo Finance stock data for Nasdaq and Tesla

- *Group 3.1 Nasdaq index data (IXIC)*
- *Group 3.2 Tesla stock data (TSLA)*

Group 1

We originally attempted to utilize Twitter's API to capture tweets over the last decade. However, we ran into several limitations when we attempted to call the API using the **get** function. After researching we discovered that due to the sheer volume of developers making requests to the Twitter API, limits were placed on the number of requests that can be made in one instance. The

limits help Twitter to create a scalable API that the developer community can rely upon. If we were to utilize the basic package, the number of requests would be based upon a time interval of fifteen minutes with a rate limit of 900 requests. In addition to this, the period was limited to the past 7 days. This caused us to reevaluate the ways in which we collect our data. As we are examining the changes of stock movement over time, it is incredibly limiting if we only focus upon 7 days.

We thus decided to attempt to find a preexisting dataset with tweets. We decided to use google data sets. This is a comprehensive search engine designated for data set retrieval. It directed us to Kaggle, where there were two separate data sets. The first from 2010 to 2017 and the other from 2015 to 2020. We decided to combine them as we could obtain a more compressive overview of Elon's tweeting history. When using the data in Power BI it is quite easy to append the data to one another by column name; however, this is not the case when modelling this in python. Just to clarify, the exploratory section and modelling section will describe two different processes for joining this data.

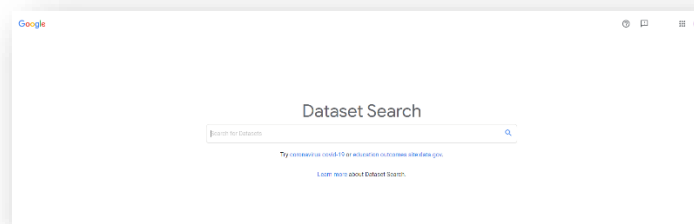


Figure 3.1 Google data search

Below is the description of group 1's data. It contains feature labels, type, and description. We have noticed that there are over 10357 lines of tweets, with over 18 columns. Feature description and type can be seen in the below table.

Feature Label	Type	Description
Id	Int64	This is the tweets id.
Created_at	Int64	This is the date in time in which the tweet has been created
Conversation_id	Int64	This is the id of the tweet.
Date	Object	This is the time stamp.
time	Object	This is the time in which the user tweeted.
Timezone	Object	This is the time zone in which the user tweeted.
User_id	Int64	This is the user id of the person tweeting.
Username	Object	This is the persons username.

Name	Object	The person's name.
Place	Object	The place in which they are tweeting from.
Tweet	Object	The actual tweet.
mentions	Object	This is if the person tweeting is commenting on a post made by someone else or is tagging them in something.
Urls	Object	This is if the person tweeting includes a link.
Replies_count	Int64	This is the number of times someone has replied to a tweet.
Retweets_count	Int64	This is the number of times the tweet is retweeted by others.
Likes_count	Int64	This is the number of individuals who have liked the tweet.
Hashtags	Object	This is the tagging symbols that the individuals may have used in their tweet.
Cashtags	Object	Twitter uses cashtags to track tweets on stock tickers but does not give any context about the stocks themselves.

Table 3.1. Group 1 description of features and rows.

Group 2

We followed the same process as seen in figure 3.1 above. This data was also obtained from Kaggle. It is a US Financial stock news data set collected from 2008 to 2020. It consists of many different news providers and pertains to many different stock tickers. The data set consists of 9 columns, and over 221513 rows.

Feature Label	Type	Description
Id	Int64	The is the unique id of the days post
Ticker	object	Each article categorized based upon a specific stock ticker. For example, one post can be about Tesla.
Title	Object	The is the news articles title
Category	Object	Binary category that distinguishes between news or an opinion.
content	Object	Actual content of the article.
Release_date	Object	Date article was published
Provider	Object	Publisher of article. i.e., Washington Post
URL	Object	Link to the article.
Article_id	Int64	This is the unique id with the article

Table 3.2. Group 2 description of features and rows.

Group 3

This section will thus dive into how we will attempt to obtain our price data. Upon further research, we found that the Yahoo Finance API is the most used to obtain this information. We examined

```
[2]: TSLA= get_data("tsla", start_date="30/01/2010", end_date="14/07/2020", index_as_date = True, interval="1d")
[3]: TSLA['date'] = TSLA.index

[44]: IXIC= get_data("^IXIC", start_date="30/01/2010", end_date="14/07/2020", index_as_date = True, interval="1d")
[45]: IXIC['date'] = IXIC.index
```

the documentation and then created our own code. We originally tried to call both TSLA and IXIC in one attempt. However, in our initial calling of the API using the **get_data** function, we noticed that TSLA and IXIC were called to us in a dictionary format. Unfortunately, this caused us to call the data separately and store it within a data frame afterwards. This can be seen in the following:

Figure 3.2 API get function

First, we attempted to specify the parameters in the API. The first elements that we wanted to examine were the start and end date. As our tweet data set was in a specific period, it was important that we attempted to ensure the same consistency with regards to our stock data. The second parameter was specified was the interval. This was the “1d”. This indicated that we wished to see the daily fluctuations in price. After we obtained the stock data from Yahoo Finance, we examined it initially and noticed several elements. First, we had to initially remove the date time from the index, as we required this data for an initial exploratory analysis. This was noticed when we attempted to export our data to a CSV.

In our initial stock data set, prior to pre-processing, we found that there were 2744 rows and 7 columns. The data description and feature types can be seen in the table below.

Feature Label	Type	Description
Open	Float64	This is the price that the stocks open at.
High	Float64	This is the stocks highest value of the day.
Close	Float64	This is the stock’s closing price.
AdjClose	Float64	The adjusted closing price changes a stock's closing price to reflect that stock's value after accounting for any corporate actions
Volume	Float64	This is a measure of how much of a given asset has traded in a period.

Ticker	Object	This is the stock symbol. i.e., “IXIC, TSLA”
Date	Ticker	This is the date in which the data was obtained.

Table 3.3. Feature Description for Group 3.

3.2 Pre-Exploration Processing

To prepare our data for exploration there were a few minor steps that we wanted to take in advance to be able to optimize our visualizations. We will follow the same logical grouping of data that we spoke about above. To reiterate, group 1 refers to Elon Musk’s tweets, group 2 refers to the financial news data set, and group 3 refers to the stock data. Some of our pre exploration processing involved python. Even though the other tools that we use possess similar functionality, it made sense to use python as we used to capture the data originally, and we were going to use it to model our data later. It therefore made sense to make all these consequential changes in the same area.

Group 1

As stated above, we have two separate Elon data sets. Unfortunately, they overlap within a certain period. This led us to map the relevant dates that they both did not have in common, and from there we decided to filter the data. This can be seen in the figure 3.3 below.

```
[10]: G12 = pd.read_excel(r'C:\Users\jamie\Desktop\elon_new.xlsx')
      G11 = pd.read_excel(r'C:\Users\jamie\Downloads\elonmusk_tweets2010-17.xlsx')

[12]: start_date = "2010-01-01"
      end_date = "2015-1-29"
      after_start_date = G11["date"] >= start_date
      before_end_date = G11["date"] <= end_date
      between_two_dates = after_start_date & before_end_date
      G11 = G11.loc[between_two_dates]
```

Figure 3.3 Data filter function

The first step was to load the two files using a the **pd.read_excel()** function. The data was called into a data frame labelled **G11** and **G12**. **G11** contains data from 2010 to 2017 and **G12** contains the data from 2015 to 2020. The next step was to define the start and end date of the **G11** file. Data frame **G11** possesses 2016 and 2017 data which overlaps the **G12** file which contains data from 2015 to 2020. The reason we chose to keep the newest files 2016 and 2017 data was because the file itself contained more features. The next step was to then create a date boundary. In doing so we instantiated **after_start_date**, and **before_end_date**. In this figure 3.3 above, you can see the use of the greater than or equal to signs that indicate the span of the dates. The last step was

to use the `loc[]` statement. This will retrieve the data stipulated between a function. It takes the `between_two_dates`, and from there filters the file and stores it in the newly processed **G11** data frame. This will now allow us to visualize the data set in the exploratory section. These data sets are then exported and will be displayed in Power BI. Of course, the merge at this point is not as extensive as needed for data modelling in python. We have chosen a basic solution at this point as it is merely exploratory as we must see if the data are viable.

Group 2

Group 2 required no preprocessing prior to the exploration section. However, this does not mean it does not require any preprocessing prior to the modelling section. This will be discussed in the subsequent sections.

Group 3

We realized that there were a few math calculations that would be valuable for the visualizations section. Understandably, Power BI has the capabilities to make these calculations. Nonetheless, it is not as straightforward. It occurred to us that cumulative returns would be interesting to examine as it can show the progress of a stock price over time. This is viewed as the aggregate amount that the investment has gained or lost over time (Investopedia, 2021). It is often expressed as a percentage and the mathematical return of the following steps.

Step 1. Calculate returns

$$r = \frac{(\text{Current Price of Security}) - (\text{Original Price of Security})}{\text{Original Price of Security}}$$

Step 2: Cumulative returns

$$(1 + r_1)(1 + r_2) \cdots (1 + r_n) - 1 = \prod_i (1 + r_i) - 1$$

To do this in python, you complete it in a similar fashion using the following steps. The first being to calculate the daily percentage change. This takes the adjusted closing price and uses the function `pct_change()`. From there we then create a new column called **cumulativereturns** and use the `cumprod()` function. This copies step two from above.

```
[75]: Tesla['daily_pc'] = Tesla['adjclose'].pct_change(axis=0, fill_method='bfill')  
[76]: Tesla['cumulativereturns'] = (1+ Tesla['daily_pc']).cumprod()
```

Figure 3.3 Cumulative returns code calculation

3.3 Data Exploration

There are several ways we can visualize our data and results. We could either progress to utilize code within python to create graphs and tables, or we can utilize one of the many available business intelligence tools to create a dashboard. Often the purpose of business intelligence tools is to gain insight into a company and their data. We shall focus upon Microsoft's Power BI, which is a tool that can be hosted on a portal or server. We wanted to visualize all our datasets in order to deep dive into the data, and to examine any inconsistencies or patterns that may emerge. We also wanted to use it to guide us further with regards to the processing, analysis, and modeling of our data. As we have several data sets, it makes sense that we incorporate all of them into a complete dashboard, that can be interacted with. The overall dashboard itself contains seven different pages following the three main groups outlined above. We shall now both outline and visualize how we created the entire dashboard through a series of screen shots. All the figures in this section will have a series of blue squares indicating the topic of the specific section. Before we begin this section, we do feel it is important to stress that we will create additional calculations using Data Analysis Expressions (DAX). DAX is the name of the coding language utilized in Microsoft Power BI. Often it is utilized to create custom calculated columns. More importantly, DAX is used to create measures. These will allow us to create common aggregates of the data such as sum, count, averages, etc.

Step 1. Building the data model

As outlined previously in our data collection process, we obtained several data sheets from multiple sources. At this point in the methodology, we have several excel sheets that ought to be uploaded into Power BI. Typically, with Power BI, you import your data using Power Query. Power Query is often described as a data connection technology that can enable individuals to connect, combine, and refine data sources. The first step that we undertook was to upload our data into Power Query. This is done through the get data button seen in figure 3.4 below.

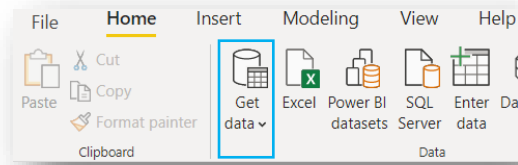


Figure 3.4. Power BI get data button

Step 2. Data Modelling Tab

This next step consisted of us going to the data modelling tab in Power BI. Here we can see if any of our data sets can be connected based on a unique identifier. This is useful because we could connect the data sets based off existing keys. Otherwise, you would typically have to create your own unique identifiers based on keys. A unique identifier can typically consist of an id, or anything that does not repeat itself. In the case, of stock and tweets we would have to create a unique date that we could then join the data on. As we can see that the data does not possess this, it tells us that we need to create this when we go to model our data in the data processing portion of the methodology. The process behind this can be seen in the figure below. The light blue square indicates where you find the data modelling.

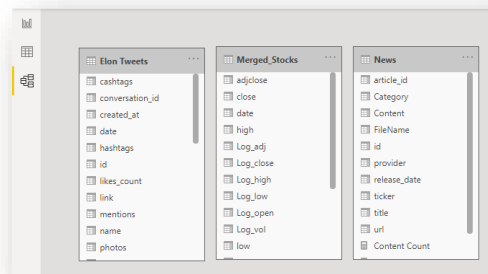


Figure 3.5. Data modelling tab

Step 3. Data Type Rectification

The next step we undertook was to examine the data types. In the figure below, you can see we have highlighted the data type button. We checked each column to ensure that each column was the correct type. We do this as it could potentially be a problem when visualizing the data in graphs and utilizing DAX. Typically, if the data types are incorrect then the data will not appear in the visualizations. This can be seen in Figure 3.6 below.

id	conversation_id	created_at	date	time	timezone	user_id	username	name	place	tweet
24	1.281725e+18	10. juni 2020	21:28:14	UTC	44136397	elonmusk	Elon Musk			I spoke with Elon's family today
38	1.281525e+18	10. juni 2020	09:31:25	UTC	44136397	elonmusk	Elon Musk			Wow, HOP & GIMM are close
39	1.281525e+18	10. juni 2020	09:26:26	UTC	44136397	elonmusk	Elon Musk			Best use of the term "Full Stack"?
45	1.281125e+18	10. juni 2020	04:25:45	UTC	44136397	elonmusk	Elon Musk			At symbols while u wait
51	1.281125e+18	9. juni 2020	07:08:51	UTC	44136397	elonmusk	Elon Musk			Progress update August 28
53	1.281125e+18	9. juni 2020	07:51:44	UTC	44136397	elonmusk	Elon Musk			If you can't beat em, join em New
75	1.279885e+18	5. juni 2020	20:39:10	UTC	44136397	elonmusk	Elon Musk			Read The Story of Civilization by v
77	1.279874e+18	5. juni 2020	20:04:27	UTC	44136397	elonmusk	Elon Musk			Dang, we broke the website
78	1.279874e+18	5. juni 2020	20:02:44	UTC	44136397	elonmusk	Elon Musk			Only \$68 420!!
87	1.279868e+18	5. juni 2020	06:16:45	UTC	44136397	elonmusk	Elon Musk			Beautiful fireworks in LA tonight
89	1.279479e+18	4. juni 2020	17:47:29	UTC	44136397	elonmusk	Elon Musk			Please take a moment to report a
96	1.279479e+18	4. juni 2020	17:46:43	UTC	44136397	elonmusk	Elon Musk			Miss Happy 4th of July!! Love
137	1.278764e+18	2. juni 2020	18:57:26	UTC	44136397	elonmusk	Elon Musk			SEC, three letter acronym, middle
158	1.278764e+18	2. juni 2020	18:50:12	UTC	44136397	elonmusk	Elon Musk			Will send some to the Shortstail
163	1.278764e+18	2. juni 2020	18:40:47	UTC	44136397	elonmusk	Elon Musk			Tesla will make the fabulous short sh

Figure 3.6 Date time conversion in Power BI.

The first issue we came across was the date time format. The second issue we noticed was the format of the tweet data. This had to be converted into a text type format.

Step 3. Creating Group 1 – Twitter Dashboard

We then proceeded to create group 1. When thinking about what we needed to display, we primarily wanted to see if there were any trends, or any significant dates or tweets within Elon's data. As you can see in the figure 3.7 below. There are several "sections" on the first page of the dashboard. We have called this the twitter user page, so that we can look at the stats of the user's twitter data.

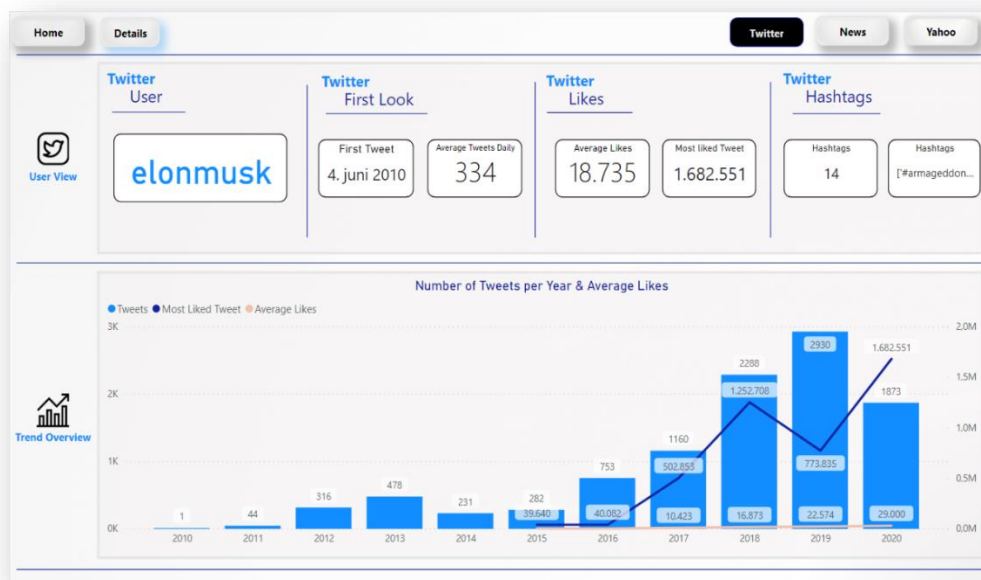


Figure 3.7 Twitter user page in Power BI

From the Twitter User section in the upper left-hand corner, there is a visual called card. The card displays text in a field. It can display the first and last category in a field, and as we only have one user, it is Elon. In the second box in the “First Look” section, we can see that his first tweet was June 4, 2010. This was achieved through using the CALCULATE and MIN function within DAX.

```
1 First Tweet = CALCULATE(  
2   MIN('Elon Tweets'[date]),  
3   filter( 'Elon Tweets', 'Elon Tweets'[name]="Elon Musk")  
4 )
```

Figure 3.8 First tweet DAX code

The other metric in this section is the average tweets that Elon produces daily. This was achieved through using the AVERAGEX and SUMMARIZE functions in DAX.

```
1 Average Tweets Daily = AVERAGEX (  
2   SUMMARIZE (  
3     'Elon Tweets',  
4     'Elon Tweets'[date].[Day],  
5     'Elon Tweets'[name],  
6     "Count", COUNT ( 'Elon Tweets'[tweet] )  
7   ),  
8   [Count]  
9 )  
10
```

Figure 3.9 Average Tweets Daily DAX code

This logic was replicated for many of the formulas within our visualization dashboard. In the far-right corner, for the Hashtag count and identifier, we used the CALCULATE and DISTINCTCOUNT functions in DAX. This can be seen in figure 3.10 below.

```
1 Hashtag count = CALCULATE(  
2   DISTINCTCOUNT('Elon Tweets'[hashtags]),  
3   SUMMARIZE (  
4     'Elon Tweets',  
5     'Elon Tweets'[name])  
6 )
```

Figure 3.10 Hashtag count DAX code

As seen in the overall view of the dashboard in figure 3.7 above. We can see that the hashtag's count was lower than we anticipated. Once visualized, we noticed that there were only 14 displayed. This could be due to two reasons. The first being that Elon may not use hashtags often. The second could illustrate that the data requires further processing. At the bottom of the page, we

used a line and stacked column chart that exhibits the number of tweets per year and average likes. The dark line on the graph indicates the most liked tweet of that year, and the peach-colored (very bottom of the graph) line indicates the average likes of that year. The functionality allows us to drill down into the months and individual days. This exhibits the same overall monthly and daily trends. This is seen in the figure 3.11 below.

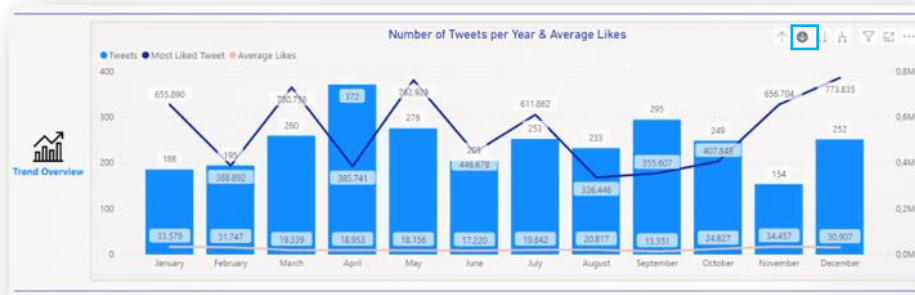


Figure 3.11 Number of Tweets per year

In terms the analysis, we can see that he became more popular in terms of tweet likes as time has progressed. It has exponentially grown over the years. However, his presence seems to have taken precedence after 2016. It could potentially be due to him using the platform more often. We decided there was a need to create a second page for group 1. This can be seen in the following figure 3.12.

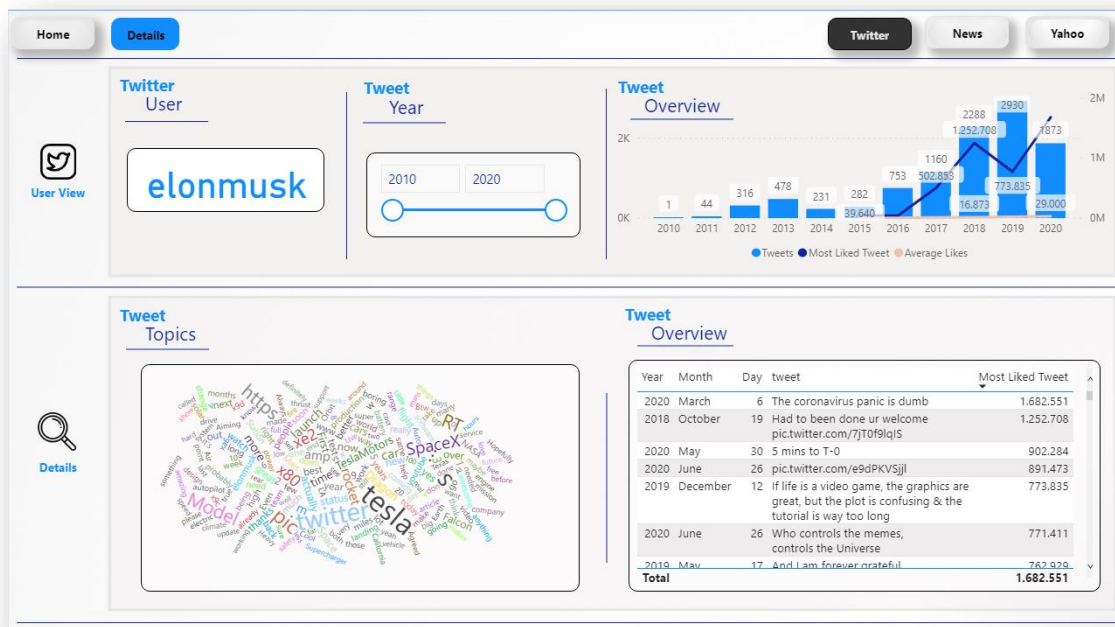


Figure 3.12 Twitter details page in Power BI

Correspondingly, the tweet overview graph in the top right corner will adjust as well. In this case, after we have clicked on Tesla, we can see which years it was mentioned the most in the tweets. We can see that 2013 stood out quite significantly with over 78 tweets. This can be seen in the figure below. This was interesting as we found upon further research that in this particular year, Elon deemed the company to be profitable. We scrolled through the tweet overview and found that Elon had tweeted significantly about the company's performance. This is beneficial to note when we go to examine the stock data. The image below highlights the increase in tweets regarding Tesla in 2013 and displays the relevant tweets. It provides some context in relation to the data.

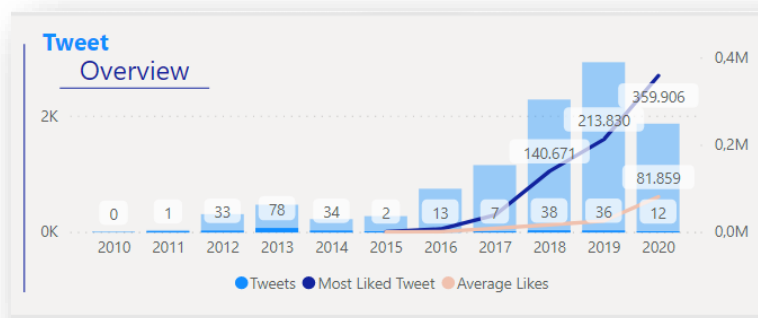


Figure 3.14 Tweet Overview in Power BI over the years

Step 4. Creating Group 2 – News Dashboard

On the main page for the news dashboard, we utilized the same visuals and code to replicate the figures. In the figure below, you can see there are two streams. At the top, we have the same the same slicer, category, and filter. The primary difference on this page is the count of publishers and the count of titles. Count of publishers refers to the unique individuals who have posted about US financial data. This was done with a quite simple distinct count function. We can also see that there are 215270 titles that we can view. We can see that the dataset is quite large. Therefore, this could potentially hinder the performance of both our dashboard and our machine learning models.

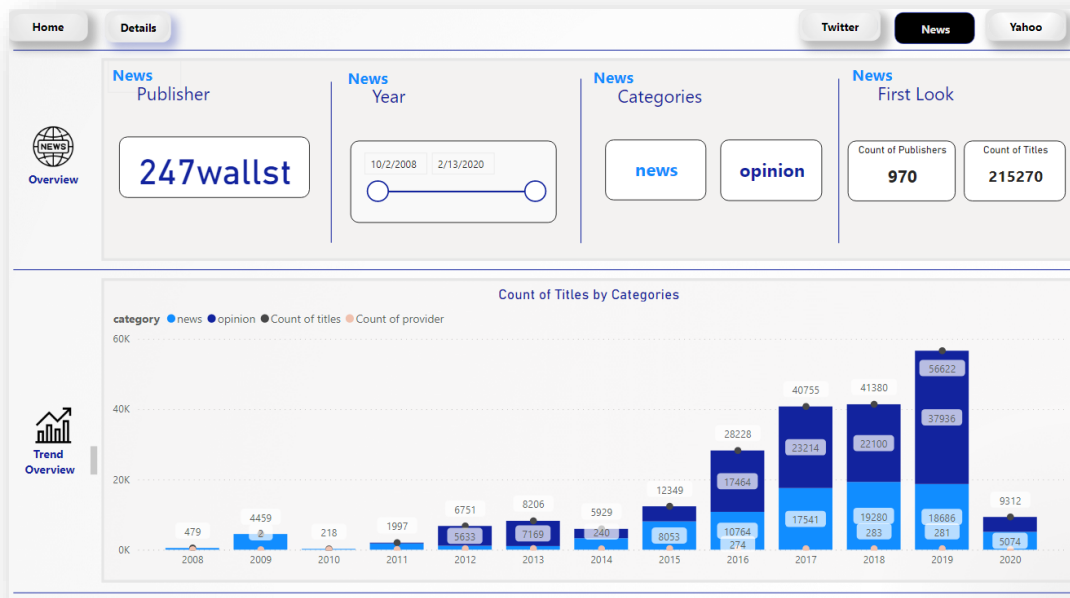


Figure 3.15 News overview in Power BI

We can also see that the sheer volume of posting only seemed to take precedence after 2015. This data set exhibits the same posting patterns as the Elon tweet data. Unfortunately, the year 2020 appears to be lacking in data. This is because the dataset itself was only collected and published in early 2020.

This now brings us to the second page called “details” for group 2. There are a few differences on this page. The first is the news provider section. We have added a Ticker/Subject slicer. It is just formatted differently from our year slider in the news year section. It allows us to narrow down the ticker data that we are looking at. There are a plethora of tickers and data pertaining to each respective stock. Thus, we deemed it prudent to be able to filter and examine each ticker individually. For relevance, we filtered for Tesla (TSLA). The slicer then adjusts the entire dashboard accordingly. This dashboard differs significantly from Elon’s tweets as it deems 2018 to be the most prominent year for Tesla. We decided to deep dive into this in the news details section at the bottom right of the figure 3.16 below. Whilst some titles do indeed reference Tesla. We can see that a substantial portion of them seem to mention facts that could be related to Tesla. For example, when we filtered for Tesla some of the article's mention “green cars.”

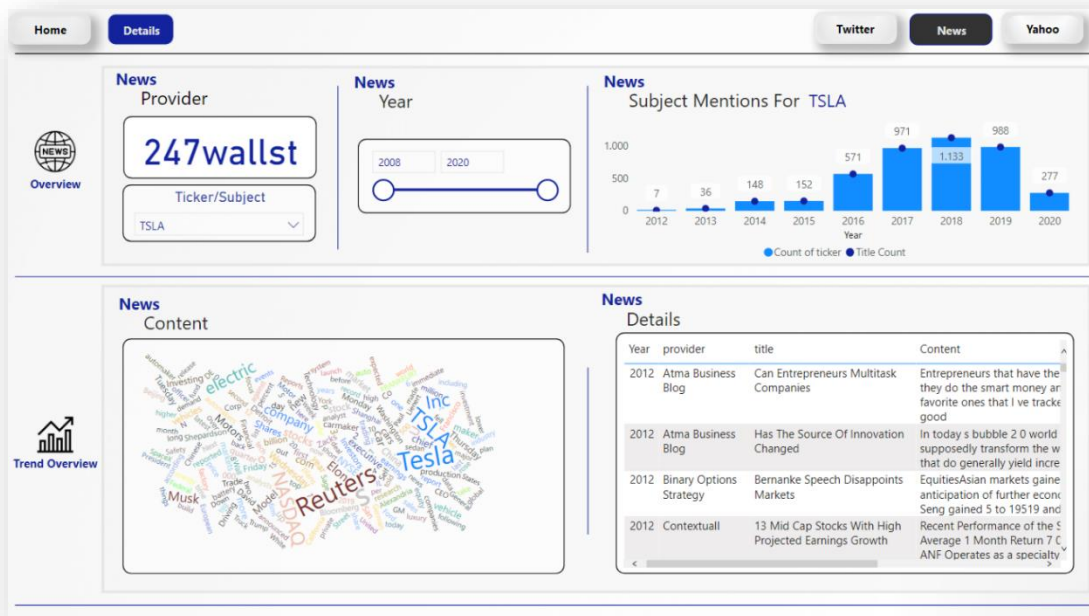


Figure 3.16 News details overview in Power BI

We also use the word cloud to narrow down the articles further. This is completed in a similar fashion to the group 1 tweet word cloud. By selecting Tesla, or TSLA we can see what the news articles display as well as how many of them mention Tesla by name. The is exhibited in the upper right-hand corner in the category news – subject mentions for TSLA.

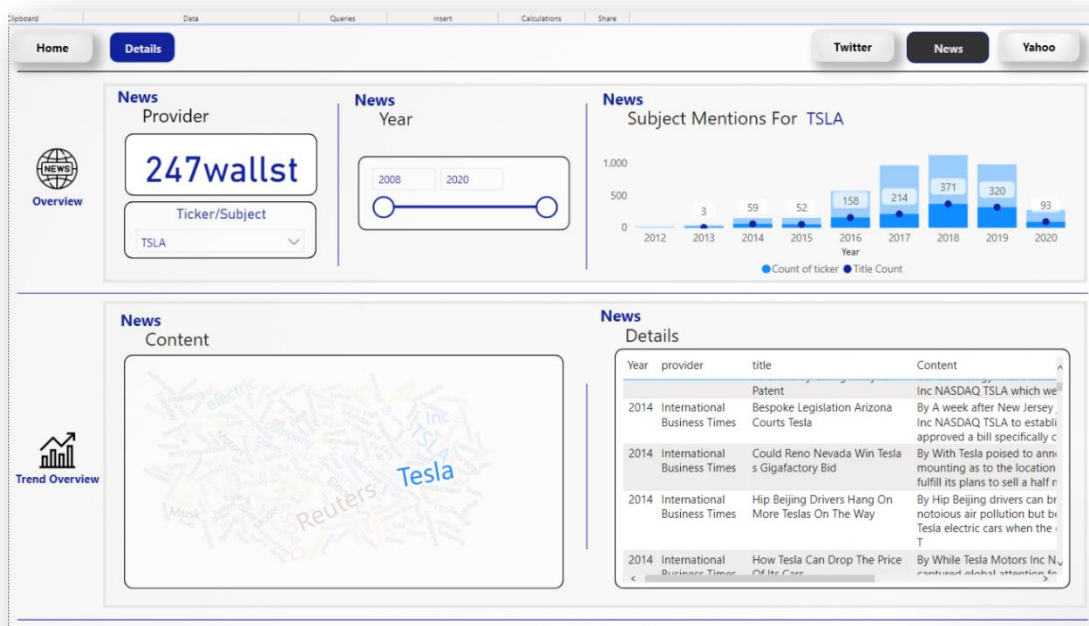


Figure 3.17 News details overview in Power BI filtered for Tesla

Step 5. Creating Group 3 – Yahoo Finance Dashboard

Again, we have what can be described as two streams on the main page of the Yahoo Financial data dashboard. At the top of the page, much like the tweets and news dashboard we have the card that indicates the ticker available. Similarly, we also have the section Yahoo year, which contains a year slicer. This can also be used to filter the year on the page. In the upper right-hand side of the first stream, we have the cumulative returns.



Figure 3.18 Yahoo Finance overview of data

We decided to visualize the cumulative returns on investment. As stated in our section 4.2 above, we performed some calculations using python.

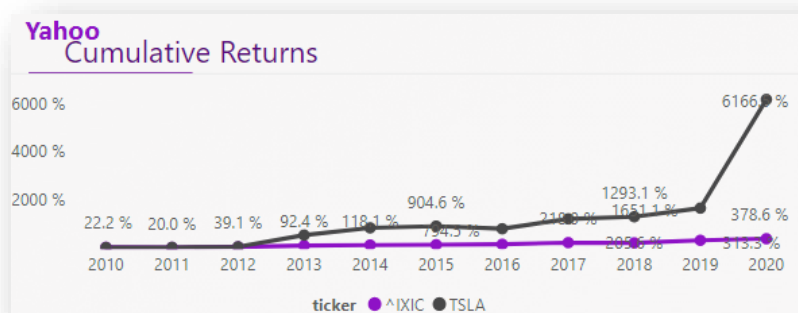


Figure 3.18 Cumulative returns of IXIC & TSLA

We can see that the return on investment has clearly risen for both stocks since 2020. It should be stated that our extract of 2020 is not a complete picture as we filtered it based on our tweet and

news data timeline. However, it is quite fascinating to note that between 2012 and 2014 there appears to be a significant increase for TSLA. When comparing this to Elon's tweets in group 1, we found two insightful points. The first being that 2013 marked the year in which his sheer volume of tweets increased. The second being the choice of topic in each of his tweets. This refers to his accelerated discussion of Tesla. More importantly, 2013 was the year that he spoke the most about Tesla. The count of which was 78 times. Overall, it does indicate to an extent that there is some sort of a relationship between the tweets and the stock data.

In terms of how both tickers fluctuate together, there are a few nuances within the data where they behave quite differently to one another. This is interesting as part of our paper aims to examine the possible impact or relationship that both may have in relation to one another. The NASDAQ composite seems to be growing quite steadily during this period. Whereas Tesla appears to exhibit more volatile growth. This is specifically seen from 2019 to 2020. It is a fascinating anomaly in the data to inspect, as this could impact the way in which our models respond to the data. This will be interesting when looking at the overall impact that Tesla has on the NASDAQ Composite in the subsequent model section. Although if we can recall, Tesla is one of the top 10 largest companies on the NASDAQ composite today, but still not above those such as Apple Inc. Meaning that the index may not reflect the losses Tesla experienced.

Moreover, we thought it could be quite illuminating to examine the tickers individually to look closer at their trends, as it can be quite "busy" to look at them on the same graph. In following figure 3.19 below, we have decided to visualize some of the additional columns that our data set possessed. In the figure 3.19 below, we created a similar dashboard. The primary difference being the emphasis upon the NASDAQ composite. On the main page of Yahoo Finance, as seen in figure 3.18 above, it was difficult to examine trading volume. The figure 3.19 below exhibits a clearer picture of this, where we can see that trading volume for the NASDAQ composite has fluctuated over the years while maintaining steady growth in price. In addition, the cumulative returns have increased. This is of course expected as the NASDAQ composite contains many valuable stocks.

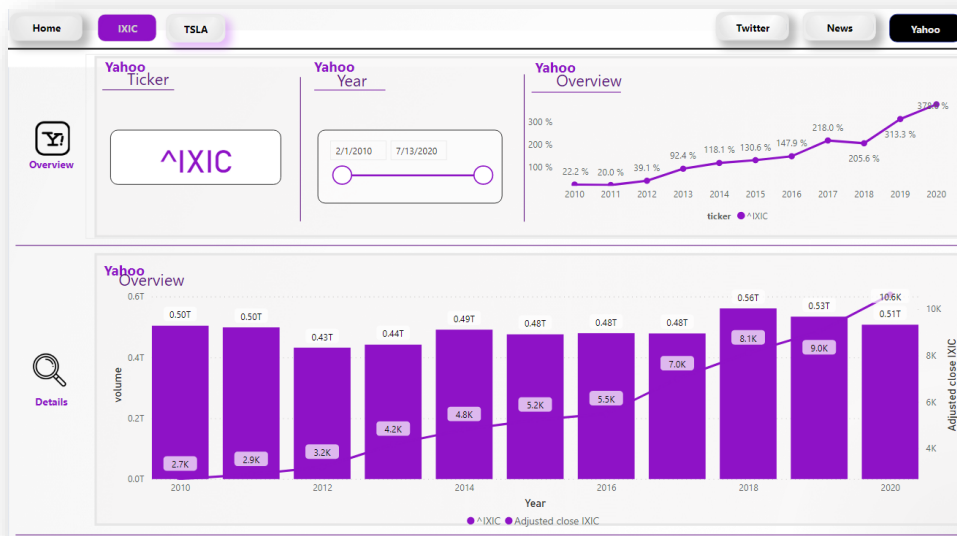


Figure 3.19 IXIC Overview in Power BI

Tesla's volume in contrast has increased quite substantially over time. This movement is directly proportional with the overall price of the stock itself. When considering volume as an indicator, there are a few things to remember. Increase in volume can indicate enthusiasm in a rising market, which is necessary to push prices higher. Moreover, a price increase while there is large volume can indicate a fundamental change in a company. The last three years exhibit a steady increase in both volume and stock price. As mentioned above, from 2019 to 2020 there appears to be a sharp increase in price, possibly a fundamental change for Tesla. It could be positive for Tesla that they are experiencing an increase in price, while maintaining volume, as compared to earlier years with varying volume, but moderately steady price increase. According to Investopedia (2021) a sharp increase in both volume and price can indicate that traders are jumping in after an opportunity has presented itself. This typically will indicate the end of a trend. In 2013 more people seem to be interested in Tesla as the price and volume both have increased simultaneously. Today's volume should not necessarily be compared to 2010; however, it should be seen loosely as a surge in Tesla's popularity over the years. Again, this can also be seen in 2013, as it is quite an outlier in comparison to the previous years. Additionally, we can deep dive into the volume month to month within these graphs to see patterns that led to the 2013 price & volume increase.

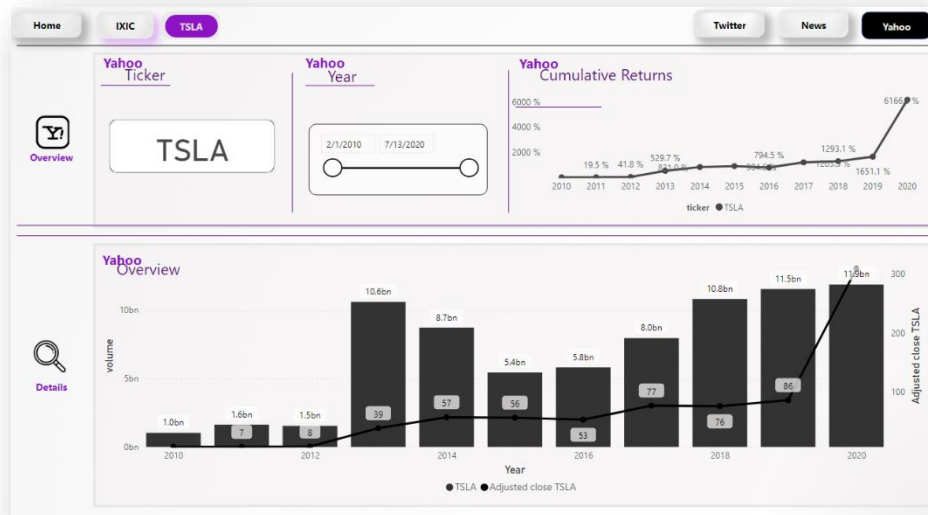


Figure 3.20 TSLA Overview in Power BI

Concluding thoughts on Group 1, 2 & 3

Overall visualizing our data in this format proved to be fruitful. We were able to familiarize ourselves with our data in ways that are not necessarily possible in both python and excel. By having the dynamic interactive elements of Power BI, we were able to retrieve valuable insights in real time and in a comprehensive manner. Of course, we could have potentially created graphs in python or excel; however, we believe there is significant value to be obtained from visualizing data before you take any action with. During our exploratory visualizing process, we were able to identify anomalies within the data groups. This has now led us to a series of actionable insights that we will use within our modelling and model pre-processing sections. The largest take away from group 1 and 2 was the pre-processing of the text data. Our word cloud showed us that we need to pay further attention to the cleansing of unnecessary characters in our text data as this can hinder model performance. It also occurred to us that the smiley faces, although insightful in a BI context, could prove to be limiting within the models themselves. Punctuation and symbols are difficult to classify.

In addition, our group 2 demonstrated that the size of the dataset will play a significant role on the performance of our models. When we tried to send the Power BI file, it was an arduous task due to the size of the news data file. To be efficient, in our processing of the news data it could be necessary to narrow the dataset even further. This is something that we will keep in mind when processing in python.

Lastly, the key deduction from group 3 was the behaviour of the stocks themselves. We examined Tesla individually in order to see elements that mirrored the tweet data set in group 1. This has provided some indication that there could potentially be a relationship. The question is at this point is whether it is causal, correlative, or neither. Nonetheless, we are quite ecstatic to progress to the next section where we can deep dive into the further processing and manipulation of the data to prepare it for the models.

3.4 Pre-model processing

After the exploratory analysis, we realized that our data required a significant amount of preprocessing prior to the modeling. We cannot use the data in its original form as data can vary quite significantly in terms of scale. Therefore, it requires manipulation. If we were to leave the data as is, some of the data inconsistencies could lead to varied results in our models. Therefore, we need to evaluate the data types, transform, calculate, and create custom columns.

To reiterate, we will describe the processing in line with our previous grouping of data. To recall, the groups refer to the following:

Group	Data set	Features	Rows
Group 1	Tweets of Elon Musk	18	10,357
Group 2	US financial news	9	221,513
Group 3	Yahoo stock data (IXIC & TSLA)	7	2,744

Table 3.4. Grouping description of features and rows.

By grouping the data into these three categories, it makes it a bit easier to follow the steps that we have taken throughout the research. More importantly, the groups are all quite different from one another. Therefore, it is pertinent to separate them from one another in the initial processing stages. Lastly, in the subsequent sections regarding groups 1 and 2, we handled the features in seven consecutive steps. For efficiency, we will illustrate this in depth for group 1, and will mention which of the steps have been used in group 2.

Group 1

To make use of our group 1 dataset, first we will need to transform the data to fit our models. As social media data in general has a tendency to be quite unstructured, we must take particular care to remove unwanted characters in the data. There are a few steps for this data set, beyond the ones that we took to adjust the data for visualization. Even though we have previously exported the data sets individually for visualization, the way in which it was unified in Power BI was not

sufficient for data modelling. This was initially discovered when we initially attempted to merge the data sets. Therefore, we will take the following steps below and show how we merged the two data sets in python. As we have stated prior, there are two data sets that we found. This was the Elon Musk Data set from 2015 to 2020, and 2010 to 2017. For clarity purposes, they will henceforth be referred to as the following:

→ **Group 1.1: Elon tweets from 2010 to 2017**

○ *Referred to as data frame G11*

→ **Group 1.2: Elon tweets from 2015 to 2020**

○ *Referred to as data frame G12*

Step 1 Feature Dropping

The first step in this process was to decide which features to keep. After the data exploration section, we had a clear picture of what data would be useful to us prior to modelling. As we initially had quite a few features, the first logical step to us was the removal of the unnecessary features. These can be seen in the figure 3.21 below:

```
[82]: G12.drop(['id', 'conversation_id', 'timezone', 'name', 'urls', 'photos', 'replies_count', 'retweets_count', 'likes_count',  
            'cashtags', 'link', 'hashtags', 'place', 'user_id', 'time', 'created_at', 'mentions', 'Unnamed: 0'], axis = 1, inplace = True)  
  
[ ]: G11.drop(['id'], axis = 1, inplace = True)
```

Figure 3.21 Feature dropping in G12 & G11

Step 2 Data Type conversion

Our data exploration within Power BI indicated that there were a few features that required manipulation regarding their data type. The first that we believe is the most important is the date type, as this is how we will join the other groups together.

```
[83]: G12['date'] = pd.to_datetime(G12['date'], infer_datetime_format=True)  
  
[81]: G11['date'] = pd.to_datetime(G11['date'], infer_datetime_format=True)
```

Figure 3.22 Date type conversion in G12 & G11

Step 3: Hashtag Handling

When we initially examined the datasets, group 1.2 contained a hashtag feature and group 1.1 did not. As iterated above in our exploratory analysis, there were less hashtags than expected in G 1.2.

Our visualizations indicated that there were 14 hashtags to be found in the feature. Therefore, we decided to further explore the tweet feature to see if there were more to be found from both groups.

```
[84]: G12['hashtag'] = G12['tweet'].apply(lambda x: re.findall(r"#(\w+)", x))
      G11['hashtag'] = G11['tweet'].apply(lambda x: re.findall(r"#(\w+)", x))
```

Figure 3.30 Hashtag removal in G12 & G11

As we can see from the results in the figure 3.31 below, there were clearly more hashtags located in the tweet feature. This was a good step as it could have possible skewed our models.

```
[20]: G11['hashtag'].value_counts()
[20]: [] 1059
      [Dragon] 11
      [Dragon, ISS] 4
      [Tesla] 3
      [DragonLaunch] 3
      [dragonlaunch] 3
      [OnionInnovation] 2
      [SKSW] 2
      [climatechange, sealevelrise] 1
      [NASASocial] 1
      [SB47] 1
      [PlutoStamp] 1
      [Fathersday] 1
      [TeslaRoadTrip] 1
      [antarctica] 1
      [TeslaNC] 1
      [TIME100, 60Minutes] 1
      [whatcouldpossiblygowrong] 1
      [Dragon, Hawthorne] 1
      [OnionReview] 1
      [CharlieHebdo] 1
      [OccupyMars, APspaceChat] 1
      [RacingExtinction] 1
      [nhgttg] 1
      [ChefMovie] 1
      [1] 1
      [OccupyMars] 1
      [FF] 1
      [memory, glass] 1
      [x] 1
      [Mars] 1
      [Models, HS] 1
      [KatieWoodenclark] 1
      [AkoV, ModelS] 1
      [Spacegiving] 1
      [GrasshopperProject] 1
      [hawthorne, coasttocoasttocoast, modelS60] 1
      [Climate] 1
      [Yutu] 1
      [AwesomeXmasGifts] 1
      [tank, amazing] 1
      [Haiti, cholera] 1
      [HB2524, SB6272] 1
      [TeslaTX] 1
      [Zeitgeist2012] 1
      Name: hashtag, dtype: int64
```

```
[19]: G12['hashtag'].value_counts()
[19]: [] 9257
      [battery] 4
      [FalconHeavy] 3
      [donotpanic] 1
      [CancelNewsNetwork] 1
      [1] 1
      [DeleteFacebook] 1
      [Armageddon69] 1
      [Von_Neumann_probes] 1
      [moneygang] 1
      [Market_manipulation] 1
      [NewProfilePic] 1
      [OccupyMars] 1
      [ThrowFlamesResponsibly] 1
      [tmhmdj] 1
      [62b6267b4465] 1
      [Pravduh] 1
      [FalconHeavy, SpaceX] 1
      [6a31c3f54ceb] 1
      [2, 17] 1
      [pt0] 1
      [329d7ef64c32] 1
      [736393734035] 1
      [7] 1
      [JusticeForGeorge] 1
      Name: hashtag, dtype: int64
```

Figure 3.31 G12 & G11 hashtag value counts after extraction

Step 4: Appending Group 1.1 & Group 1.2

Now that both data sets have separately been cleaned and have the same number of features, the next logical step is to append them to one another. This can be seen in figure 3.32 below.

```
[86]: G1 = G12.append(G11)
```

Figure 3.32 G12 appending data

Step 5: Aggregation of data

Our next issue when considering grouping group 1 and group 3 by date, was that each data set needs unique date values to join properly, so that the date may be used to provide us with the most accurate comparison for each day. To do this we decided to aggregate tweets for each date into one single row. Logically, Elon will tweet more than once a day. For our model to be able to process all his words in one day, we need to make all the tweets one row for one specific day. This is done through a process of aggregating. This is achieved using the **groupby()** and **agg()** function in python.

```
[88]: G1 = G1.groupby(['date'], as_index = False).agg({'tweet': ' '.join})
```

Figure 3.33 G1 group by date

To confirm that this has happened we decided to check the unique value count when summarized by date in the original dataset to gain insight into how many unique dates there were in the tweet data set. We can see in the figure xx, on the left side that there are quite a few counts per unique value on some of the days. For example, for 2020-04-16 there were 57 tweets per that day. We did this to ensure that once we aggregated each tweet to its given day, that we did not lose any days in the data set during this process. As we can see below after our aggregation, we have 2475 unique dates both before and after we aggregated the tweets to a date.

```
[87]: G1['date'].value_counts()
```

```
[87]: 2020-04-16 00:00:00    57
      2017-03-24 00:00:00    49
      2018-06-17 00:00:00    48
      2019-12-30 00:00:00    46
      2020-05-15 00:00:00    44
      ..
      2016-10-22 00:00:00     1
      2012-12-14 14:21:00     1
      2013-05-25 01:45:00     1
      2013-11-28 23:53:00     1
      2013-02-26 07:29:00     1
      Name: date, Length: 2475, dtype: int64
```

```
[89]: G1['date'].value_counts()
```

```
[89]: 2019-07-06 00:00:00     1
      2013-07-24 06:35:00     1
      2018-09-15 00:00:00     1
      2019-12-09 00:00:00     1
      2016-07-06 00:00:00     1
      ..
      2017-10-15 00:00:00     1
      2014-08-05 00:22:00     1
      2016-11-22 00:00:00     1
      2012-11-23 03:16:00     1
      2013-04-02 17:49:00     1
      Name: date, Length: 2475, dtype: int64
```

Figure 3.34 Value counts of G1 date

We were skeptical of this result, so we decided to search the data set for a specific date. This is the 2020-04-16 date. We printed the tweets just to be certain that we had aggregated it properly. We did this using the following code. We searched the data for a specific date that we had previously mentioned from the figure above. We wanted to see if 2020-04-16 still contained 57 tweets, and if we successfully managed to aggregate them into a single row.

```
[22]: G1[G1['date']=='2020-04-16']
```

Figure 3.35 Searching for a particular date

Prior to aggregation, we printed the tweets for this specific date. We can see that there are multiple tweets for 2020-04-16. This is exhibited in the figure 3.36 below.

date	hashtag	tweet	username
2020-04-16	[]	Reviewing overall system with vehicle engineer...	elonmusk
2020-04-16	[]	Reduced size by ~3%, center line is more level...	elonmusk
2020-04-16	[]	🤔🤔	elonmusk
2020-04-16	[]	All new	elonmusk
2020-04-16	[]	It is about 10% too small, but lots of fun 🤔	elonmusk
2020-04-16	[]	We're working on increasing dynamic air suspen...	elonmusk
2020-04-16	[]	Karma is real	elonmusk
2020-04-16	[]	Yeah, super messed up	elonmusk
2020-04-16	[]	These were based on direct requests from their...	elonmusk

Figure 3.36 printed results for 2020-04-16 prior to aggregation

Then we proceeded to search that date again after our aggregation formula. In the figure 3.37 below, we can see that it was successful as it shows there is now one row for this date.

[24]:	date	tweet
2390	2020-04-16	Reviewing overall system with vehicle engineer...

Figure 3.37 printed aggregated results for 2020-04-16

Step 6: Merging the data

Now that we have ensured unique date values exist in the tweet dataset, we can now proceed with merging the dataset. This is done using the date column and a left join.

```
[28]: Dataframe1 = pd.merge(TSLA, G1, how="left", on=["date"])
```

Figure 3.38 Dataframe1 Merged

Step 7: Custom functions

The next step we did was to create a custom **text_process** function. The sole purpose of this is to create a function that removes any unwanted punctuation and stop words. Punctuation will not really tell us anything about the words in question. It is also quite difficult to tokenize punctuation in general.

```
[26]: ## This is a function that is placed inside the pipeline.  
def text_process(tweet):  
    nopunc = [i for i in tweet if i not in string.punctuation]  
    nopunc_text = ''.join(nopunc)  
    return [i for i in nopunc_text.split() if i.lower() not in stopwords.words('english')]
```

Figure 3.39 text_process function

Group 2

As mentioned above, we will replicate the steps from group 1, within relation to this specific data set. As described prior, group 2 consists of the below:

→ **Group 2: Financial news tweets from 2010 to 2020.**

○ *Referred to as data frame G2*

Step 1 Feature Dropping

As we had features that were not of use to us in this analysis, the following features were dripped from the dataset.

```
[8]: G2.drop(['provider', 'category', 'article_id', 'url'], axis = 1, inplace = True)  
    ## dropping unnecessary columns
```

Figure 3.40 dropping unnecessary columns

Step 2 Data Type conversion

Similarly, we needed to transform the date feature to reflect datetime, as the date will be our point of comparison to other data sets.

```
[9]: G2['date'] = pd.to_datetime(G2['release_date'], infer_datetime_format=True)
```

Figure 3.41 G2 date type conversion

Step 3: Hashtag Handling

We deduced that as our focus was upon title of the news article, there is a low chance that they contain hashtags. This would also take an extensive amount of computing power as the dataset is enormous. Accordingly, this step was skipped.

Step 4: Appending data

This step was also skipped as we have one singular massive data set.

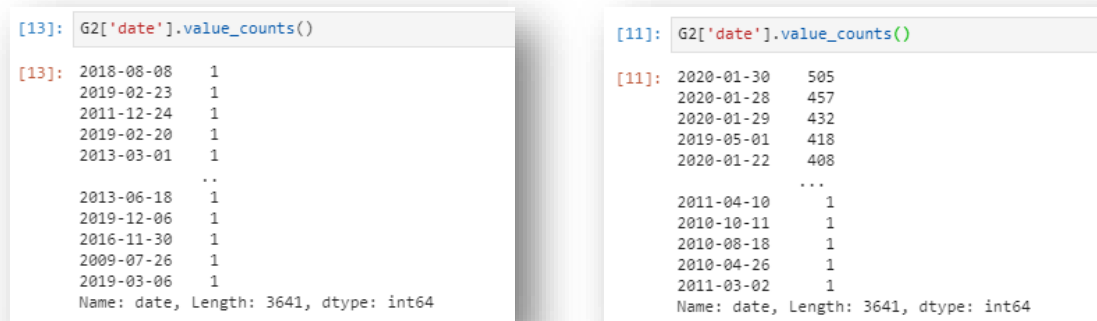
Step 5: Aggregation of data

It was also necessary to aggregate the data based on new article date published, as many articles were published on the same date.

```
[12]: G2= G2.groupby(['date'], as_index = False).agg({'title': ' '.join})
```

Figure 3.42 G2 groupby date

Once again, we have searched for unique data value counts, which decreases our final data set size from the original count of 221,513 to 3,641. This is interesting as we can see that there were clearly many articles for each date.



```
[13]: G2['date'].value_counts()

[13]: 2018-08-08    1
      2019-02-23    1
      2011-12-24    1
      2019-02-20    1
      2013-03-01    1
      ..
      2013-06-18    1
      2019-12-06    1
      2016-11-30    1
      2009-07-26    1
      2019-03-06    1
      Name: date, Length: 3641, dtype: int64
```

```
[11]: G2['date'].value_counts()

[11]: 2020-01-30    505
      2020-01-28    457
      2020-01-29    432
      2019-05-01    418
      2020-01-22    408
      ...
      2011-04-10     1
      2010-10-11     1
      2010-08-18     1
      2010-04-26     1
      2011-03-02     1
      Name: date, Length: 3641, dtype: int64
```

Figure 3.43 value counts of G2

As we tested this code in group 1, we will not display the same filter by date search here.

Step 6: Merging the data sets

As there is just one gigantic dataset, we do not have to complete this step.

Step 7: Custom functions

Once again, we create a custom text cleaning package. As described above, this is needed for the pipelines.

```
[26]: ## This is a function that is placed inside the pipeline.
      def text_process(tweet):
          nopunc = [i for i in tweet if i not in string.punctuation]
          nopunc_text = ''.join(nopunc)
          return [i for i in nopunc_text.split() if i.lower() not in stopwords.words('english')]
```

Figure 3.44 text_process function for group 2

Group 3

This section will now examine group 3. As it is numerical, it is processed differently from groups 1 and 2. As mentioned in the data collection section, this information was obtained via the Yahoo Finance API. This section is a continuation of the pre-exploration processing section. We will not keep the same columns that were created for the visualization of the data.

→ Group 3.1: Nasdaq Stocks from 2010 to 2020

- *Referred to as data frame IXIC*
- **Group 3.2: Tesla Stock from 2010 to 2020**
- *Referred to as data frame TSLA*

Step 1: Logarithmic Transformation

There are other transformations that can often prove useful for transforming certain features. Specifically, we can choose to apply mathematical functions such as exp, log, or sin (Müller & Guido, 2016). As certain models are tied quite significantly to the scale and distribution of each feature it can become quite hard to model when there are nonlinear relations. Therefore, these functions can help us in situations where we made need to readjust the scale.

Logarithmic Transformation: Log considers the scale of the data. As price can vary quite significantly, the log transformation can be quite a powerful tool to utilize. We found that based on the previous literature, calculating log returns is the best practice. Nti *et al.* (2020) discuss the calculation of log returns as a means of determining the change in stock price between two days. The formula is as such.

$$\ln\left(\frac{\text{Current Price}}{\text{Original Price}}\right)$$

We did some further investigation and found that the panda's package can account for this with a simple line of code that harnesses the **.shift()** function. This will allow us to consider the lags and time frequency. Therefore, we will attempt to use the mathematical formula within python to effectively transform our data. It could be conducive towards our models and data if we were to further transform the data. As price can vary quite drastically, we thought that the best conceivable way of ensuring the scale of our data remain consistent would be to log transform the price data. When doing so in python there is one customary practice that is often undertaken. We shall import the **NumPy** package as np. The **NumPy** package shall be instantiated as it possesses the **log()** function. This calculates the natural log of the value inside of it.

Moreover, as part of this process we require utilizing the **.shift()** function. According to the **Pandas** documentation within python, this shifts index by the desired number of periods within an optional time frequency (Pandas, 2021). In the context of our code, the shift function allows us to take the row just above the present row. We have set the lags to 1. Log returns is calculated through taking the log of the ending value divided by the beginning value. Therefore, the final code can be seen in the figure below.

```
[5]: TSLA['Log_adj'] = np.log(TSLA['adjclose']/TSLA['adjclose'].shift(1))
TSLA['Log_close'] = np.log(TSLA['close']/TSLA['close'].shift(1))
TSLA['Log_low'] = np.log(TSLA['low']/TSLA['low'].shift(1))
TSLA['Log_high'] = np.log(TSLA['high']/TSLA['high'].shift(1))
TSLA['Log_open'] = np.log(TSLA['open']/TSLA['open'].shift(1))
TSLA['Log_vol'] = np.log(TSLA['volume']/TSLA['volume'].shift(1))
```

```
[4]: NDAQ['Log_adj'] = np.log(NDAQ['adjclose']/NDAQ['adjclose'].shift(1))
NDAQ['Log_close'] = np.log(NDAQ['close']/NDAQ['close'].shift(1))
NDAQ['Log_low'] = np.log(NDAQ['low']/NDAQ['low'].shift(1))
NDAQ['Log_high'] = np.log(NDAQ['high']/NDAQ['high'].shift(1))
NDAQ['Log_open'] = np.log(NDAQ['open']/NDAQ['open'].shift(1))
NDAQ['Log_vol'] = np.log(NDAQ['volume']/NDAQ['volume'].shift(1))
```

Figure 3.45 Log Transformation

Our initial thoughts were to test all the different price features against text. Therefore, this step displays all the price features being transformed. This does not necessarily mean we will use every single feature in the model processing section.

Step 2 Movement column creation

After we completed the calculations, we thought it could be interesting to create another conditional column. This column will be a binary categorical feature consisting of “increase” and “decrease” of price based off day-to-day movement. To do this, we shall use the newly created log columns and create a new column called “Movement.”. We thought that this data could potentially prove useful when fitting our models. In this case, we would treat it as a binary classification task. The logic of the custom column involves utilizing an if statement. The calculated log column indicates whether the value has increased or decreased. We then take this value and create an if statement. So, if the value in the log column is above 0, then we categorize it as “Increase”. If the value is below zero, then it is categorized as “Decrease.”

```
[9]: TSLA['Movement'] = ['Increase' if x > 0 else 'Decrease' for x in TSLA['Log_adj']]
```

```
[5]: NDAQ['Movement'] = ['Increase' if x > 0 else 'Decrease' for x in NDAQ['Log_adj']]
```

Figure 3.46 Creation of movement column

Summary of pre-model processing

The last three sections focused upon column and feature cleansing. These steps were necessary for us to complete before we were able to group our data and run our models. It should be stated

there are some additional model processing steps. However, we have decided to detail them in the model processing section as these steps include the combined data sets in different variations. Also, some of the text processing is interwoven within the models themselves within a data pipeline. It thus made sense to include these steps in the ensuing section.

3.5 Data frame Grouping

As stated above we have previously organized the data into three groups. However, these groups now ought to be merged. To recap the original groups created are:

→ **Group 1: Tweets of Elon Musk**

- *Group 1.1 Elon Musk tweets from 2010 to 2017*
- *Group 1.2 Elon Musk tweets from 2015 to 2020*

→ **Group 2: US financial news data set**

→ **Group 3: Yahoo Financial stock data for Nasdaq and Tesla**

- *Group 3.1 Nasdaq stock data (IXIC)*
- *Group 3.2 Tesla stock data (TSLA)*

Our next steps require us to group the data in different ways as the aim of our methodology it is to ensure trustworthy results. The premise of this is to see how the different data sets perform on the models. This can indicate where relationships exist. This will give us the most unbiased and accurate results.

Name	Combination	Features	Rows
DateFrame1	Group1 vs TSLA	18	947
DateFrame2	Group 2 Vs IXIC	16	2395
DateFrame3	Group 1 vs IXIC	16	947
DateFrame4	Group 2 vs TSLA	18	2395

Table 3.5. Data frame description of features and rows.

In the following sections we will go over all four data frames. We will start by outlining how we processed Dataframe1 and DataFrame3. After we have outlined our processes for Dataframe1 and DataFrame3, we will merely mention how the subsequent data frames differ.

Dataframe1 and DataFrame3

As part of handling Dataframe1, we needed to take multiple steps once again prior to modelling. This involves merging the data sets and the removal of empty rows.

Step 1: Merging of the Group 1 and TSLA

As you can see in the figure 3.47 below, we have merged the group 1 and TSLA on a left join using the date column. As our data was previously cleansed, this is now plausible.

```
[25]: Dataframe1 = pd.merge(TSLA, G1, how="left", on=["date"])
```

Figure 3.47 Dataframe1 merging of TSLA and G1

Step 2: Removal of NaNs

After we merged the data, we could see that there were NaNs in the dataset. This is problematic because we need our movement feature to match the tweet feature. Otherwise, the data frame will not run through the models. This is achieved through simple **notna()** function in python.

```
[ ]: Dataframe1 = Dataframe1[Dataframe1['tweet'].notna()]
```

Figure 3.48 Dataframe1 removing the na

Step 3: Applying a Twitter preprocessing cleansing package

After we examined the merged data, we found that there were still inconsistencies in the text processing. Moreover, we also realized that a sizeable portion of our text still contained references to symbols and pictures. We thus proceeded to find a custom tweet processor package online. This was imported as a library and instantiated as the letter p. As you can see in figure 3.49 below, we take the feature Dataframe1 ['tweet'] and apply the cleansing package to it. The package was utilized for efficiency purposes, as it cleans the entire feature instantaneously. We could have created a series of our own functions to achieve the same result. However, this package was all encompassing and less time consuming when used to clean the URLs, mentions, reserved words, emojis, and smileys.

```
[ ]: Dataframe1['text'] = Dataframe1['tweet'].apply(p.clean)
```

Figure 3.49 applying the tweet cleaning package

Step 4: Bag-of-Words

Now that we have summarized our tweet data set as well as merged it together with our stock data set, we can now transform our tweet feature using bag of words. This is a form of representing text in alternative ways. Using bag of words will take each word in the text feature and build a vocabulary (Müller & Guido, 2016). This will disregard the structure of the original text and move on to count the instances or occurrences of the word in the corpus. This will then represent the text as a bag. The bag of words process consists of three steps tokenization, vocabulary building and encoding. The steps can be seen in the figure below.

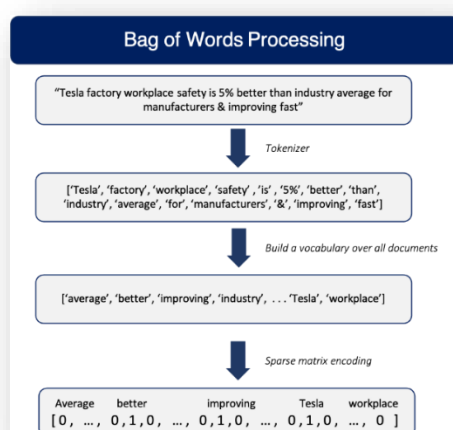


Figure 3.50 Bag of words processing diagram

Below we import **CountVectorizer()**, which handles the tokenization and vocabulary building of the tweet data. Once we fit this to our tweet feature it will split each unique word into tokens, taking into consideration stop words and the case of the text (lowercase). The **min_df()** is a parameter that tells the function to ignore terms that have a document frequency lower than the threshold.

```
[ ]: vect = CountVectorizer(min_df = 1, lowercase=False).fit(Dataframe1.tweet)
```

Figure 3.51 CountVectorizer

To create the bag of words representation we call the transform method. The bag of words representation is then stored in a SciPy sparse matrix. Then we can see below the size of the resulting matrix 947x9935 which implies we have 9935 words in our data set.


```
[34]: bag_of_words = vect.transform(Dataframe1.text)
      print("bag of words: {}".format(repr(bag_of_words)))

      bag of words: <947x9935 sparse matrix of type '<class 'numpy.int64''>'
                    with 55867 stored elements in Compressed Sparse Row format>
```

Figure 3.52 vect.transform function

Step 5: Rescaling the data using term frequency-inverse document frequency (tf-idf)

After we utilized the approach above, we thought that it could be prudent to approach it differently and rescale the features. Often with the strategy above, it can be quite easy to drop features without examining them. Another approach is through using term frequency–inverse document frequency (tf-idf). This tool will allow us to give a high weight on any term that appears frequently within a specific document but not within many other documents within the corpus (Müller & Guido, 2016). If a word appears quite common in one document but not another it is plausible that it pertains heavily to the document that it exists within. This can be applied in scikit-learn through two specific classes. The first is the **TfidfTransformer**, this takes the sparse matrix output from the bag-of-words count vectorizer and will transform it. The **TfidfVectorizer** does both the bag-of-words feature extraction and tf-idf transformation. The code for this is used in conjunction with the model in a data pipeline. This will effectively combine all of the code into a single effective workflow. The way in which it is instantiated can be seen below. We will delve into further details regarding the pipeline in the modelling section.

```
pipe2 = make_pipeline(TfidfVectorizer(min_df=1, analyzer=text_process),
                      MLPClassifier())
```

Figure 3.53 make_pipeline figure

Dataframe2 and Dataframe4

Step 1: Merging of the Group 2 and IXIC

This incorporates the same steps that were found above in Dataframe1. The only difference is the specifying of files.

```
[62]: Dataframe2 = pd.merge(G2, IXIC, how="left", on=["date"])
```

Figure 3.54 Dataframe2 merging G2 & IXIC

Step 2: Removal of NaNs

After the merging, we completed a similar step. However, we removed the NaNs in the movement column instead. There are appeared be to no differences in doing so, as we wanted to ensure that they matched the text column. In other words, there needs to be a movement category type associated with the title of the article, just as there needed to be a price.

```
[16]: Dataframe2 = Dataframe2[Dataframe2['Movement'].notna()]
```

Figure 3.55 Removing na from Dataframe2

Step 3: Applying a Twitter preprocessing cleansing package

Even though we were using the article titles. We decided to cleanse this using the tweet preprocessor package as well. We were uncertain about the general content of the news article titles. In a data set of over 200,000 titles, it is a bit difficult as well as cumbersome to search and identify if it contains any unusual characters. Despite our efforts in the exploration portion of this research, we were unable to confirm or deny the existence of these characters. Consequently, we made the decision to apply the same package to the data.

Step 4: Bag-of-Words

We chose to not use the same approach from Dataframe1 in these data frames. As our initial results, did indicate that it was ineffective on its own. This will be further elucidated upon the ensuing modelling section.

Step 5: Rescaling the data using tf-idf

This has become the primary methodological choice for text, as it provided to be the most efficient in terms of results and the handling of text data. The same approach was utilized in these data frames.

3.6 Model processing

In this section we will be using two models, both of which have general steps in which we follow for the creation and processing of them. These steps are as follows:

- Step 1. Pipeline creation
- Step 2. Parameter tuning
- Step 3. X and y instantiation
- Step 4. Test-train-split
- Step 5. Grid Search Cross validation
- Step 6. Evaluating the Classification Algorithms
- Step 7. Coefficient Visualization

Model choice and selection can be quite a perplexing task to undertake. In the case of our paper, we found that previous works recommended a series of models that can be used in the prediction of price. We chose our models with our own understanding of the data as well as how they have approached within previous works. As we may have alluded to in the previous pre-processing section, we had the original ambition of creating a regression-based task. This would have involved the actual prediction of price.

Upon reflection, we decided to create a classification type task with the aim of predicting a class label from a predefined list. In this case, it is a binary classification task. Therefore, our original calculations and processing of the data using log have now been made redundant. However, we will still keep them as they were pivotal in relation to our processes. We shall thus use classification supervised machine learning algorithms. We will first describe a logistic regression model, and from there will progress to describe a neural network. We find that each model in many ways builds upon the next, and by going through both we will ensure our understanding of the mathematics as well as the functional logic of the models themselves.

We decided to test our models in a preliminary fashion. Before, we begin to tune and adjust the parameters, we wanted to explore them a little bit first. As we have alluded to prior, our initial sole of the bag-of-words in conjunction with **CountVectorizer()** was not as effective as we would have hoped it would be. It also was too simplistic in terms of its scaling and understanding of how text data functions. Therefore, the following section will progress with a thorough explanation of the steps taking to obtain the results from the models. Each model section will first begin with an in-depth description of the model and the mathematical logic behind its usage. From there we will follow a seven-step process that will allow us to obtain our results. In the subsequent steps, there is additional logic such as pipelines, and methods that can be used in the evaluation of models.

Moreover, the modelling section does not contain steps specific to each data frame. Therefore, we do require individual descriptions of the data frames being imported into the models.

Model 1. Logistic Regression

We shall first progress with using a class of models that are widely used in practice. This would be linear models for classification. The formula for a linear model for classification is the following:

$$\hat{y} = w[0] * x[0] + w[1] * x[1] + \dots + w[p] * x[p] + b > 0$$

The formula is quite like a linear regression model. The primary difference is that instead of returning the weighted sums, we shall threshold the predicted value at zero (Müller & Guido, 2016). So, if the function itself is smaller than zero then we will predict the class as -1, and if the function is larger than zero then it will predict the class as +1. With classification models, the decision boundary is a linear function of the input. Müller & Guido (2016) state that the boundary will separate two classes using a line, a plane, or a hyper plane. There are various kinds of classification algorithms that we can use to obtain our results. The two most common models used are logistic regression, that is implemented within **linear_model.LogisticRegression**, and linear support vectors that are implemented in **svm.LinearSVC**. Both models have similar decision boundaries and apply L2 regularization. L2 pushes the weights toward zero. In addition, both models use the tradeoff C parameter. We will delve deeper into how we will tune for C and the importance of this in the subsequent section. For this paper, we will instantiate a logistic regression model and see how our data reacts to it.

Step 1. Pipeline creation

The pipeline class is used to express a workflow. In simpler terms, it is a means of compacting several steps into a singular estimator (Müller & Guido, 2016). The pipeline class itself has a **fit**, **predict** and **score** method much like the others within scikit-learn. The most common way in which it is used is to scale the data together in a classifier. To express the workflow, we construct the pipeline from a base of estimators. The estimators in this context are base objects that one can implement a fit method to learn data from (Pedregosa et al., 2011). When attempting to code this, we call the **make_pipeline**. Within it we call the estimators and specify some of the additional parameters. In this context, we are calling the previously created **text_process** function within the **TfidfVectorizer**. We also set the **min_df** to 1. The logistic regression was also added within the pipeline.

```
pipe = make_pipeline(TfidfVectorizer(min_df =1, analyzer=text_process),
                    LogisticRegression())
```

Figure 3.56 make_pipeline using the logistic regression

It should be stated that in the next step regarding parameter tuning, we do not have to use the upper-case formatting of the **TfidfVectorizer** and the logistic regression as the constructor does not require their names to be in this format (Pedregosa *et al.*, 2011).

Step 2. Parameter tuning

The next step in this process is to create an object that contains the parameters that we want to check for in our **GridSearchCV**. The syntax for this utilizes the lower case of the model and the **TfidfVectorizer**. The double underscore indicates that it shall specify a parameter. In the case of parameters for the following **param_grid**. We have specified two types for the **GridSearchCV** to search for.

The first is the C parameter for the logistic regression. This is a way of tuning for the strength of model regularization. With a logistic regression model, it will often try and fit the training set as best as possible. In this instance of the **param_grid**, we chose five different C values for the model to choose from. The higher the value of C the less regularization.

```
param_grid = {"logisticregression__C": [0.001,0.01,0.1,1,10],
              "tfidfvectorizer__ngram_range": [(1,1),(1,2),(1,3)]}
```

Figure 3.57 parameter tuning for C and ngram

Using a low C will cause the algorithm to adjust to most of the data points, the higher the value, the more likely the model to stress the importance of the individual points. When you have low values of the C parameter it will put more emphasis upon finding a coefficient vector (w) that is as close to zero as possible (Müller & Guido, 2016). This could be interesting with text data as unique words could pertain heavily to price fluctuations.

The second parameter we focused upon was the n-gram range. This groups the words together and counts how many times they appear next to one another. Often in the bag-of-words approach, word order is disregarded, so this is a way of combating it and seeing if words together generate

meaning. We choose to represent n-grams in three ways. The first being unigrams represented by (1,1), then bigrams which is represented with (1,2), and finally trigrams which is represented by (1,3). We can of course have up to 5 grams, but this will lead to an explosion of the number of features and could lead to the overfitting of our models. As there will be many specific features within the model itself.

Step 3. X and y instantiation

The next step in this process is to instantiate the X and Y values prior to the test train split. In this step we specify our X and Y axis. The X in this case is the text data, and the Y is the Movement feature found in the data frame. It should be stated that we have labelled the following X and y as we have several models.

```
X = DataFrame1['text']  
y = DataFrame1['Movement']
```

Figure 3.58 instantiating X & y

Step 4. Test-train-split

As we want to build a machine learning model from the data above, we need to determine if the model is accurate. More importantly, we need to know whether we can trust our prediction. A common method in this process is to perform a test train split. We do this by splitting the data into two parts. One part of the data is used to build the model. This is the training data, and the other part is used to assess the model. The function we call will split the data set. Around 75% will go into the training set, and the remaining 25% will go into the test set. Moreover, the data is also shuffled prior to the split. The random state is a pseudorandom number generator with a fixed seed using the random state parameter.

```
x_train2, x_test2, y_train2, y_test2 = train_test_split(X2, y2, random_state = 42)
```

Figure 3.59 test train split

Step 5. Grid Search with Cross validation

Before we describe how a grid search cross validation works, we shall display an overview in figure 3.60 below.

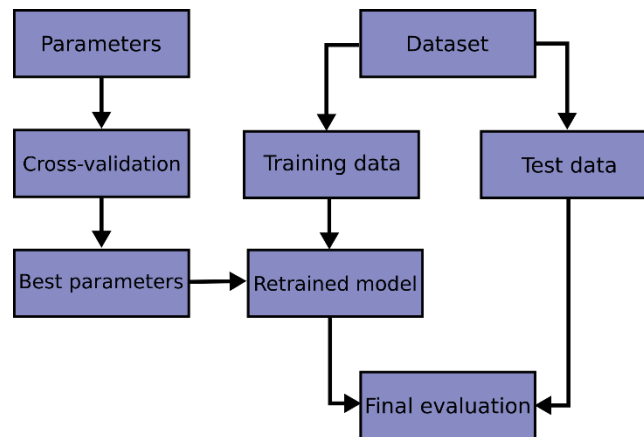


Figure 3.60 cross validation take from (Pedregosa *et al.*, 2011)

For a better estimate of the generalization performance of the data, we can go beyond the single split of the data in the test train split. We can use a cross validation approach that takes the parameters and splits it across several folds. Often it is considered a methodological mistake to repeat using the same labels of samples, or to not split it into folds. In many cases by not choosing to do this, the models can easily be overfit. For example, this can be seen in figure 3.61 below.

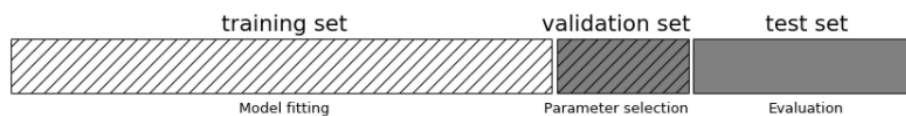


Figure 3.61 training split taken from Müller & Guido (2016)

This is represented in the code using cross validation (cv). We have chosen to set it to 5. This then splits the data, fitting a model and computes the score 5 consecutive times, thereby using different splits each time.

As we have parameters that we would like to tune, it makes sense that we use the **GridSearchCV**. This is an estimator and can thus be fit to the **x_train** and **y_train**. By approaching the code in this manner, it will allow the **GridSearchCV** to fit the feature selection within the cross validation. This is done to avoid data being leaked between the test and training sets. By leaking, we refer to a process in which the model will simply look across both sets and choose the features that are most correlated. This is information leakage. In this case, it can lead to unrealistic results. By using the pipeline in conjunction with **GridSearchCV** we eliminate this process, and the feature selection shall occur inside the cross-validation loop (Müller & Guido, 2016).

In figure 3.62 below, we can see that we have called an instance of grid. Inside grid, we have specified the pipeline, and the **param_grid** and then finally the cross validation. The cross validation is represented by **cv** in the code. It should be stated in the code below that we did not

use the test set to choose the parameter. This is stored in the **best_params_** attribute and the best cross validation accuracy. The cross validation best score and the best parameters were subsequently printed. The best score should not be confused with model performance used in the score method in the test set. The best score looks at the mean cross validation on the training set.

```
grid = GridSearchCV(pipe, param_grid, cv=5)
grid.fit(x_train, y_train)

print("Best Cross val score = {:.2f}".format(grid.best_score_))
print("best parameters: ", grid.best_params_)
```

Figure 3.62 GridSearchCV code with the param_grid and pipeline

Step 6. Evaluating the Classification Algorithms

After we cross validate with the **GridSearchCV**, we wanted to evaluate the model itself. One of the more common ways to evaluate binary classification is using a confusion matrix (Müller & Guido, 2016). This typically will create a plot that indicates the number of times the algorithm has not predicted accurately. This often refers to the number of false positives and false negatives within the algorithm. This is viewed as false positive (FP), false negative (FN), true positive (TP) and true negative (TN). However, viewing an entire confusion matrix will take an extensive amount of time. We therefore shall endeavor to compute the accuracy. Accuracy is typically calculated as such:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

There are other ways in which we can summarize the confusion matrix. This is readily achieved through examining the precision, recall and f-score (Müller & Guido, 2016). The precision calculates how many samples were accurately predicted as positive. The precision is often used when the goal is to limit the number of false positives. The recall is a measure of how many false positive samples are captured within the predictions. It is often used when the requirement is to identify all positive samples. The last measure that we mentioned is the f-score. This summarizes the above two. It is viewed as the “harmonic mean” of precision and recall (Müller & Guido, 2016). It is also calculated through the following:

$$F1 = 2 * \frac{precision * recall}{precision + recall}$$

The way in which we have chosen to visualize all three is using the **classification_report** function. In this function, we take the previous pipeline, and predict using the **x_test**. From there we print the classification report with the newly instantiated **pip_pred1**.

```
[74]: pipe.fit(x_train, y_train)
      pip_pred1 = pipe.predict(x_test)
      print(metrics.classification_report(y_test, pip_pred1))
```

Figure 3.63 fitting the pipe to the **x_train** and **y_train**

The results of this will later be discussed in the results section.

Step 7 Coefficient Visualization

As part of understanding how the model weights the different words, we wanted to visualize them in a graph. This will show it in two categories, as this is a classification problem. It will show the words associated with a price decrease and the words associated with a price increase. On the x axis it will show the features, which will show them either in unigram, bigram, or trigrams, and it will show the top 40 features for both categories. The code for this can be seen in the following figure 3.64. This will be displayed in the results section of the paper.

```
vectorizer = grid.best_estimator_.named_steps["tfidfvectorizer"]
feature_names = np.array(vectorizer.get_feature_names())

import mglearn
mglearn.tools.visualize_coefficients(grid.best_estimator_.named_steps["logisticregression"].coef_, feature_names, n_top_features =40)
```

Figure 3.64 code for the coefficient visualization

Model 2. Multi-Layer Perceptron Classifier (Neural Network)

Neural Networks

We pondered the likelihood that a singular linear model could be considered too simplistic for a task as extensive as this. Thus, we deliberated the use of a neural network. Our progression to utilizing an artificial neural network (ANN) stems from the fact that it in many ways can be viewed as generalizations of linear models that perform multiple stages of processing to arise at a decision (Müller & Guido, 2016). Neural networks are depicted as a family of algorithms that are often under the umbrella term deep learning.

Neural networks were created with the purpose of emulating the neurons in a person's body. Thus, the logic of it ought to account for the ways in which text could in turn impact the stock market. They similarly view a neuron as a central processing unit that performs a mathematical operation to generate an output from a set of inputs (Ciaburro & Venkateswaren, 2017). The output of the neuron is the weighted sum of the inputs plus the bias. The function itself is merely the computation of the outputs. Thus, a neural network is a set of mathematical function approximations. There are a few elements we should pay attention when approaching a neural network and these are as follows: input layers, hidden layers, output layer, weights, biases, and activation functions. This is called the layered approach.

Weights & Bias

The weights in an ANN are the most important in terms of converting the input to impact the output. The ANN conversely to a linear model for regression uses a matrix multiplication to arrive at a weighted sum (Ciaburro & Venkateswaren, 2017). The bias notion mentioned above behaves like the intercept in our linear model for regression. It is merely another parameter that can be used to adjust the output along with the weighted sum of the inputs to the neuron. Therefore, the processing can be seen as such:

$$\text{output} = \text{sum}(\text{weights} * \text{input}) + \text{bias}$$

Activation Functions

A function is then applied to the result. This is called an activation function. Therefore, the new input is now the old output. This is a series of mathematical equations repeated. This can be seen in Ciaburro & Venkateswaren's (2017) description of the equations.

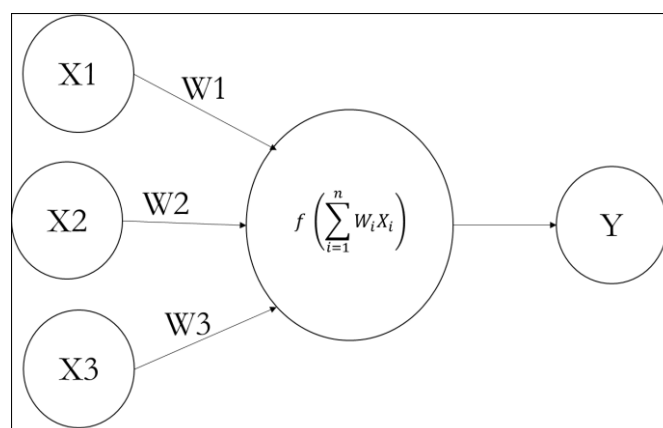


Figure 3.65 ANN structure as described by Ciaburro & Venkateswaren's (2017)

The next question we ought to ponder is how we are going to train the model. This can vary quite greatly in comparison to say that of a linear model for regression. ANN can be both supervised

and unsupervised and can achieve its results through a series of activation functions. The activation function is what makes a neural network as unique as it is. Neural networks are typically used in cases where it may be looking to solve nonlinear and complex problems. So, in the case of choosing an activation function, it gives the neural network the relevant nonlinear function. According to Müller & Guido (2016) usually the rectifying nonlinear unit (relu) or the tangens hyperbolicus (tanh) are applied as the activation functions. These functions are typically applied to the weighted sum that shall then progress to compute the y output. This can be seen in the figure above. The two functions approach the values differently. Relu for example cuts values below zero, whereas tanh saturates to -1 for low input values and +1 for high input values.

Nodes, Hidden layers, & penalty parameters

Moreover, the next element we need have careful consideration of when coding the neural network is the number of nodes. Depending on the size of our data, we can set this from 10 to even 10,000. Nevertheless, the computational power ought to be quite extensive. We can similarly add further hidden layers to add more complexity to the data. The logic behind this is that the smaller the number of nodes, the less complex the model is. Now, it is wise for us to consider how many when we go to model the data as the size of our data set will determine how we approach this.

In addition, we may also attempt to control the model's complexity by using an l2 penalty, the same way we could control some of the other linear models for regression (Müller & Guido, 2016). The penalty parameter in this context would be alpha. The default for neural networks is little regularization. This could be something that we should take into consideration when it comes to the modelling of our data. Lastly, the neural networks weights are set to random, therefore, the initialization will impact the model choice. We can thus obtain vastly different models through utilizing different random seeds.

Perceptron and Multilayer Architecture

The easiest to use interfaces for neural networks, as a starting point would be the MLPClassifier and MLPRegressor. This stands for multilayer perceptron. A perceptron is a single neuron that classifies a set of inputs into one of two categories (Ciaburro & Venkateswaren, 2017). We can find the relevant information within scikit-learn libraries. We can of course progress outside of the basic models and attempt to use the more complex libraries of tensor-flow, lasagna, and keras.

Now that we have at length discussed the various aspects of the neural network. It is time to apply this to our data. The next section shall commence with a similar step wise process to that of the logistic regression.

Step 1. Pipeline creation

This process was conducted in a similar fashion to the logistic regression. It involved the creating an instance of **pipe2**. This then consists of the **make_pipeline()**. We call the **TfidfVectorizer()** and set the **min_df()** to 1. From there we call the custom function **text_process**, that we created earlier. The last step that differs to the logistic regression model is the calling of the **MLPClassifier()**. The steps for this can be seen in the figure 3.66 below.

```
pipe2 = make_pipeline(TfidfVectorizer(min_df =1, analyzer=text_process),  
                      MLPClassifier())
```

Figure 3.66 make_pipeline code for the MLP Classifier

Step 2. Parameter tuning

Tuning a neural network is considered an art unto itself (Müller & Guido, 2016). The most common way in which we can do so is adjusting for the hidden layer sizes as well as alpha. As we mentioned prior, there are many ways in which we can tune the parameters. However, our computational power is limited so we will choose the two more common methods for tuning a neural network. This can be seen in the following code:

Step 3. X and y instantiation

We created the X2 and y2. The variables have different names because we wanted to be able to separate and identify the different models and where the different variables are located. When we go to compute the accuracy, it will be quite helpful to have them labelled differently, as this could potentially confuse our code.

```
X2 = Dataframe2['text']  
y2 = Dataframe2['Movement']
```

Figure 3.67 Instantiating X2 and y2

Step 4. Test-train-split

We then instantiated the test train split. This is similar to the process we previously followed.

```
x_train2, x_test2, y_train2, y_test2 = train_test_split(X2, y2, random_state = 42)
```

Figure 3.68 test train split for the MLP Classifier

Step 5. Grid Search Cross validation

We conducted a **GridSearchCV** as a means of choosing the best parameters and cross validating this upon the training data. The aim is still to avoid leakage of the data, as explained in the previous modeling section.

```
grid2 = GridSearchCV(pipe2, param_grid2, cv=5)
grid2.fit(x_train2, y_train2)
print("Best Cross val score = {:.2f}".format(grid2.best_score_))
print("best parameters: ", grid2.best_params_)
```

Figure 3.69 GridSearchCV with pipe2 & param_grid2

Step 6. Evaluating the Classification Algorithms

This step was also conducted in a similar fashion to the logistic regression model section. We wanted to see how the model performed in terms of the precision, recall and f1 score within the data. As this is a binary classification task, it will interesting to view how the model has predicted in relation to the increase and decrease in the price of the stock. This can be seen in figure 3.70.

```
pipe2.fit(x_train2, y_train2)
pip_pred2 = pipe2.predict(x_test2)
print(metrics.classification_report(y_test2, pip_pred2))
```

Figure 3.70 fitting x_train2 and y_train 2 to pipe2

Step 7. Coefficient Visualization

As the MLPClassifier is clearly different to the logistic regression, it is significantly more complicated to visualize the coefficient weights. The complexity of the model alone makes this difficult to visualize. In addition to this, the sheer computational power alone is tremendous when processing the MLPClassifier. To elaborate, if you have a binary classification dataset with 100 features and 100 hidden units then there are 10,000 weights between the input and the first hidden layer. There are also 100 hidden weights between the hidden layer and the output layers. Typically, you would match the number of the nodes to the number of features. In most cases, this should

seldom be higher than the low to mid thousands. Thus, it is the number of nodes per hidden layer. Therefore, if you add another hidden layer with 100 hidden units, then there are another 10,000 hidden units. The number of weights increases every time there is a new hidden layer. Therefore, it makes it incredibly difficult to visualize in the case of this model. It is not an impossible task but computationally it has been cumbersome for us to obtain in this section of the methodology (Müller & Guido, 2016).

Reliability, validity, credibility, and limitations

This section will assess our methodology and will mention the reliable, valid, and credible elements as well as the inherent limitations that it possesses. The validity and credibility of our research stems from the use of multiple data sets and the combinations of them. This refers to our creation of a grouping system. To recap, this was in the form of the aforementioned Dataframe1, Dataframe2, Dataframe3, and Dataframe4. The aim of utilizing them was to obtain a well-rounded approach to the problem. By having diverse sources of data, it enables us to add further reliability to our findings.

Credibility & Reliability

The findings themselves possessed credibility as we at every given point tried to avoid leakage and bias within our modelling. As we previously mentioned, data leakage ensures that no information has been leaked between the training and testing data. We guarantee that our models do not cheat and try and look at what the desired outcome ought to be. In addition, through the process of cross validation and testing on new models we maintained the reliability/dependability of the findings. Cross validation enabled us to use the best parameters of our models in a way that results in the lowest possible testing error.

Model and Pipeline limitations

This now brings us to the limitations of our models and our pipeline. Whilst we did aim to cross validate for the best parameters, there were some restrictions when parameter tuning for the neural network. Overall, the MLPClassifier was incredibly difficult to tune for and we were quite ambitious initially with our goal to leverage the weights, biases, activation functions, nodes etc. Unfortunately, we only chose to tune for two parameters due to our computational restraints. In other words, it took simply too long to account for two parameters. It would have taken even longer if we had chosen more, although we could have obtained better results using the different parameters. With this being said, the run time on the models was cumbersome. Pipelines typically require some power, and so too do neural networks. In addition, our data sets were also large. The

combination of all of these caused our models to take an exorbitant amount of time to run. The computational power to undertake a task such as this was quite vast, and it could be argued that this is easily replicable if you were another data scientist with a larger CPU power.

Text data limitations

Another limitation we found in the preprocessing was exhibited within the text processing. The initial attempt to use a bag-of-words model in the **CountVectorizer** was difficult. After we had originally processed the data in this fashion and fitted it to our model. This heavily overfit the models. Therefore, this portion could have been omitted. Nonetheless, we feel it is important to keep this information and process as it displays our journey towards understanding text mining.

Replicability to other tasks

This methodology is applicable to any social media task. It could be paralleled to cases of determining say clothing value based off the description of the content on a web page. Some retailers for example have conducted similar tasks within natural language processing where they examine reviews and ascertain value and profitability of a particular product. Regardless of the data relationship in the case of our models, the logic is quite generalizable. Throughout the process of this research, there appeared to be no ethical concerns as both our data sets were publicly available. None of the tweets possessed sensitive or personal data that could cause concern. Moreover, the Yahoo Finance API is also a publicly available tool that any person can use at any given point. There is also no limit to the number of extractions per hour. This was also beneficial and allowed us to avoid both search costs and confidential data.

Overview of Methodology

Below is an overview of this methodology section. Highlighted are the various steps that were taken to prepare our data as well as obtain the results for the proceeding section.

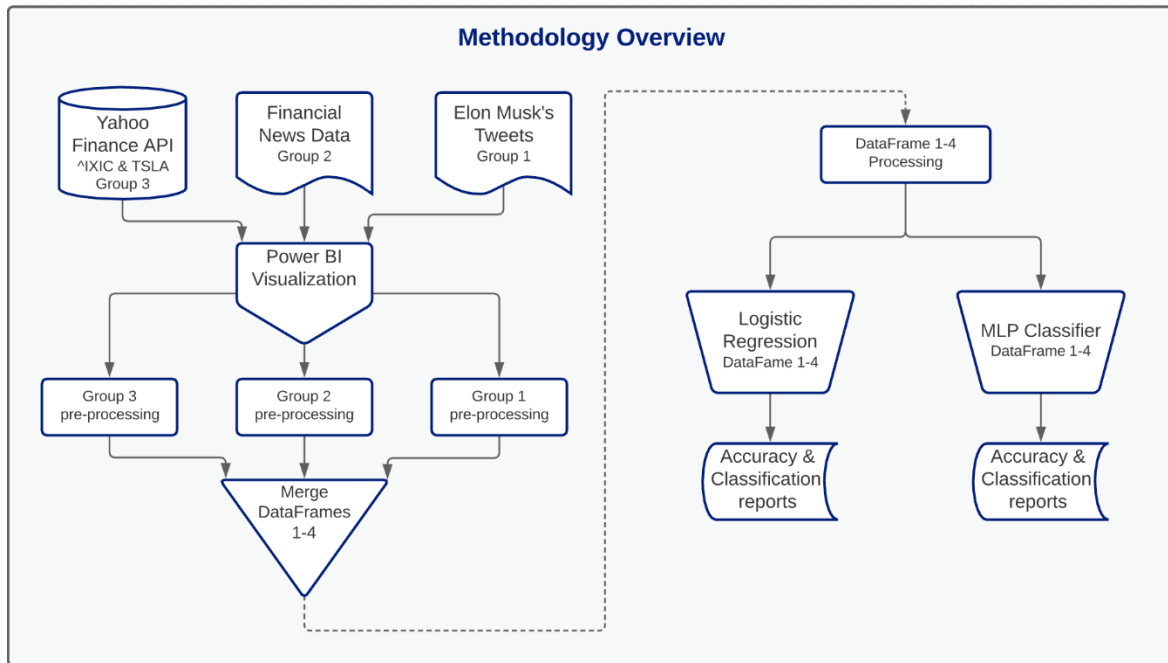


Figure 3.71 Methodology Overview

4 Results

This section will take all the results from the modelling section above and will discuss in depth the meaning behind the results and what this means from the perspective of using social media to gain insight into price movement. To recap, we initially created several instances of grouping that were named DataFrame1, DataFrame2, DataFrame3, and DataFrame4. Below is a table to recall what each data frame consists of, where Group1 refers to Elon tweets, and Group2 refers to financial news data.

Name	Combination	Features	Rows
DateFrame1	Group1 vs TSLA	18	947
DateFrame2	Group2 vs IXIC	16	2395
DateFrame3	Group1 vs IXIC	16	947
DateFrame4	Group2 vs TSLA	18	2395

Table 4.1. Data frame description of features and rows.

In the proceeding sections, we will first discuss our initial exploration of the models and what we found interesting. We should preface, that the preliminary exploration was only utilized for DataFrame1. From there we will progress to discuss the results obtained from the data pipeline. To reiterate, the pipeline included the use of the logistic regression and MLPClassifier. Inside the pipeline we tuned for parameters and cross validated using **GridSearchCV**. This section will

describe what these results mean from a data perspective in terms of model performance. What the results means for our topic regarding price movement and social media can be found in the proceeding section.

Initial Exploration of Models Results

As mentioned prior in our methodology, we attempted to create the bag-of-words representation and from there fit it to a logistic regression and a MLPClassifier. The results can be seen in the table below. To be clear, this was primarily an exploratory portion of our research. The results were not used to generate any conclusion other than the encouragement to utilize a more sophisticated approach towards model tuning, and cross validation of our results. The accuracy on the logistic regression was close to 100 percent. This indicates that the models were heavily overfitting. Intuitively speaking, this occurs when the data fits the model too well. The large gap in the MLPClassifier between the training and testing accuracy indicates that we could get better results with more regularization within the model. As the aim is to have our models generalize from the training set to the test set, this was not achieved. These initial results from the exploration of the models were in part the reason we progressed to use the **TfidfVectorizer** in conjunction with the pipeline.

Data	Model	Training Accuracy	Testing Accuracy
DateFrame1	Logistic Regression	1	0.99
DataFrame1	MLPClassifier	0.83	0.53

Table 4.2. Accuracy Results of Initial Exploration.

The aim with accuracy should ideally have been to find the “sweet” spot between underfitting and overfitting. This was clearly not achieved in our initial exploration of the models using the bag-of-words approach. We obtained a figure from scikit-learn which accurately depicts the alleged “sweet” spot of model accuracy (Pedregosa *et al.*, 2011). This can be seen in the following figure 4.1.

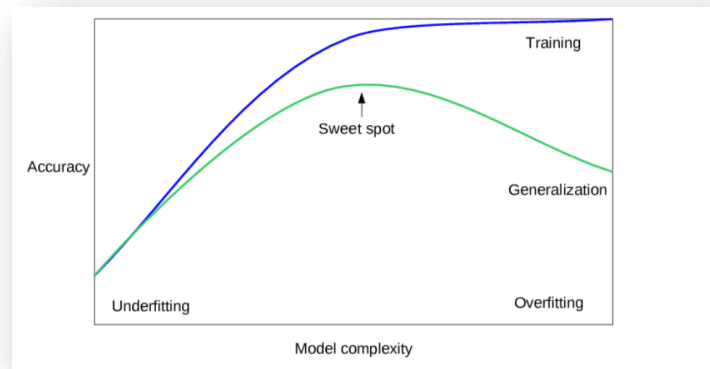


Figure 4.1. Accuracy Results and Model complexity taken from (Pedregosa et al., 2011)

Logistic Regression

This section shall first start with displaying the results obtained from the logistic regression. The sub-sections shall provide an overview of what these results mean in terms of model performance and the mathematics. The overall connection to our research question shall be discussed at lengths in the discussion of this research.

Data	Best CV Score	Test set CV score	C Parameter	N-grams
DataFrame1	0.54	0.52	1	(1,1)
DataFrame2	0.66	0.67	10	(1,1)
DataFrame3	0.56	0.52	1	(1,1)
DataFrame4	0.55	0.56	1	(1,1)

Table 4.3. Accuracy Results of the Logistic Regression Model

Cross Validation Performance

This section will focus upon the results obtained from utilizing the pipeline in conjunction with the **TfidfVectorizer**. In the results above, the data frame that performed the best was DataFrame2. This was the IXIC data used in conjunction with group 2 (the news data set). For DataFrame2, the best CV score, which should not be seen as the generalization performance of the model was around 0.66 with the test set CV score being around 0.67. The best score is merely the overall mean cross validated accuracy that was performed on the training set of the data. Where the testing cv accuracy is how well the model performed on the testing set. The size and nature of the data set alone could have been the reasoning behind the accuracy of the score.

Moreover, the grouping that yielded the worst results was DataFrame1. The overall accuracy was 0.54 for the best score, and 0.52 for the test set. This was quite interesting when considering that this data frame is comparing Elon Musk's tweets to Tesla's stock. We originally assumed that his results would have yielded the best results. This was however not the case. Lastly, it is important to note that the accuracy overall is still quite low across the entirety of the data frame groupings.

The n-gram parameter

Moreover, it is important to discuss the optimal n-gram parameter for the logistic regression model. The preferred parameter for this model was unigrams. This indicates that the logistic regression model weighs its accuracy based upon singular word usage. When comparing this to the price movement of the data, the model computed the most accurately when utilizing a singular sequence of words. This would be in rebuttal to viewing two words together or three words together. We could postulate that the reasoning behind the parameter choice of unigrams stems from the lack of model dimensionality. It indicates an inability for the model to find adequate patterns. If the model were to choose bigrams or trigrams, it would suggest that it can find other consistent patterns. Meaning, the model is struggling to find patterns amongst the words. For example, the model could notice the phrasing "bad car," or "large financial trouble," and from there associated this with a price decrease. However, the model is struggling, in a manner of speaking to leverage the use of these bigrams or trigrams. Thus, we are left with the singular word choice in this regard. The results here speak to the complexity of the problem itself as well as the difficulty of handling text data and stock price.

The Best C Parameter

As we have mentioned prior, we chose to tune for the C parameter with the logistic regression. The higher the C the more the model will try to fit the training set as best as possible. The lower the value of C the more emphasis they will place on finding a coefficient that is as close to zero as possible. For example, the best way to interpret the results of the C parameter would be using the following figure.

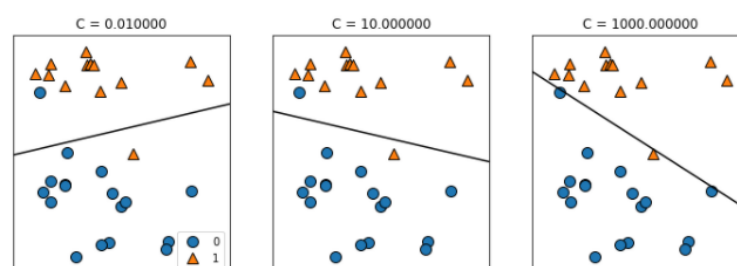


Figure 4.2. C parameter interpretation taken from Müller & Guido (2016)

When looking at the results it is quite curious that DataFrame1, DataFrame3, and DataFrame4 all had an optimal C parameter of 1. DataFrame2, which performed the best, had an optimal C parameter of 10. The lower the C indicates a lot of model regularization. The stronger model chooses a line that is relatively horizontal. This is exhibited in the Figure 4.2 above. The middle image above, which corresponds to a tilted decision boundary, shows only a couple of the points that are misclassified. In addition, the higher C parameter indicates that within DataFrame2, the C stressed the importance of more individual data points. From a general perspective, this means that the model is looking for patterns. It assesses the overall situation and tries to deduce meaning and adjust for broader trends. Thus, the model that performed the best is the one that sought to examine broader trends and put lower weights on more extreme cases. By extreme cases, we mean the adjustment to outliers in the data as can be seen in the far left of figure 4.2 when $C = 1000$.

Coefficient Results

We thought that as part of the results section, it could be interesting to examine which of the coefficients the logistic regression deemed important when classifying the movement of price. As DataFrame2 appeared to perform the best, we thought it could be interesting to display the results. The data pipeline using **tf-idf** clearly valued certain coefficients more than others. We visualized these results using the **best_estimator_.named_steps**. The figure 4.3 below indicates the 25 largest and the 25 smallest coefficients of the logistic regression model that were utilized. The negative coefficients highlight words that are associated with a price decrease. The positive coefficients indicate words that are associated with an increase in price movement.

Upon an initial inspection, it occurred to us that there were remaining stop words that had failed to be removed. However, it was quite interesting to see that on the left side of the chart, the words that were associated with a price decrease were: tumble, lower, plunge, fail, loses, etc. On the right side, the coefficients that were associated with a price increase were rates, rises, higher, surge, etc. This makes a lot of sense when you think of the decision-making process of investors themselves and what could potentially drive this process of investment.

However, there are clearly many words that we could argue possess no meaning from a very human perspective. If an investor were to see the word “exporters” on its own it really would not generate anything. The reason we mention this is because it highlights the complexity of the problem. When dealing with an issue as complex as stock price prediction, it is difficult to ascertain value from models that leverage the use of single words.

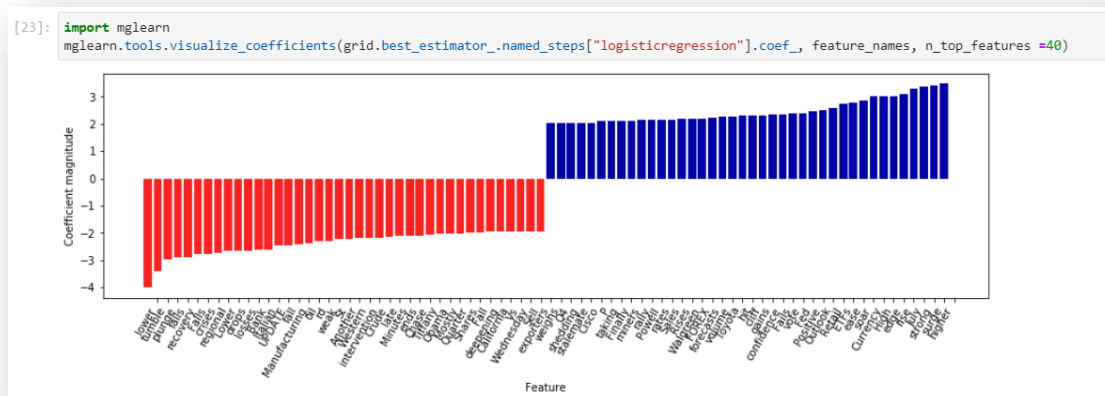


Figure 4.3 Coefficient visualization of logistic regression

Classification Report

We wanted to initially report upon the evaluation of the binary classification model itself. Typically, the way in which this is done is through a confusion matrix. Previously in the methodology, we chose to print a classification report of the logistic regression. To recap, the **classification_report** function prints the *precision*, *recall*, *f1 score*, and *support* of the model. These metrics are tools that enable us to evaluate the model. It is typically used in the case of binary classification. The results are displayed in the tables below. We believe that it is imperative to examine how each of the combinations behaved in their respective models. As there were different data set combinations it would hinder our analysis if we were to negate this section entirely.

To reiterate, the *precision* is a performance metric that aims to limit the number of false positives. Therefore, we ask the question what percentage of the predictions were accurate. In the case of table 4.4 below, we can see for precision, the models were more likely to predict accurately with an increase than a decrease in price movement. The *recall* difference in some of the reports were also quite massive in terms of the disparity between the decrease and increase. To recap, the recall asks what percentage of the positive cases were caught in the modelling process. Both metrics can be visualized nicely in the following:

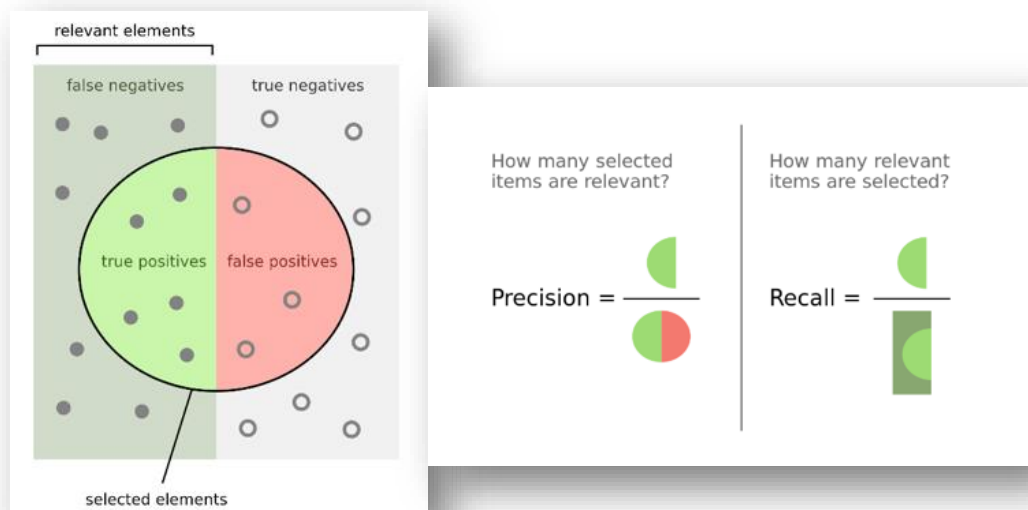


Figure 4.4. Outline of Precision and Recall taken from (Brownlee, 2020)

The *f1 score*, or the harmonic mean of the precision and recall also favored the increase as opposed to the decrease in price. Lastly, the *support* is the number of actual occurrences in the dataset. In the case of the below tables, it does look like it is imbalanced in some of the groupings, where it tends to favor the decrease as opposed to increase. This could indicate a structural weakness. Overall, the classification reports illuminated upon something that we had not initially considered. We already knew from the validation of our accuracy that the results were lower than expected. We can see this fact in the tables below. However, the disparity between increase and decrease was thought-provoking. In this case it could mean that bad news is an easier metric to account for as opposed to good news. Good news, or news that corroborates a price increase, appears to be a lot more complex in terms of the nuances found in the words themselves. We shall now provide some additional commentary regarding the individual classification reports and our assumptions regarding the results.

	Precision	recall	F1-score	Support
Increase	0.47	0.38	0.42	109
Decrease	0.55	0.64	0.59	128
Accuracy	n/a	n/a	0.52	237
Macro avg	0.51	0.51	0.50	237
Weighted avg	0.51	0.52	0.51	237

Table 4.4. Classification Report of Logistic Regression DataFrame1

We can see in the table above that the recall, precision and F1 score all favor the price decrease. An example can be seen in the recall, where we see 0.38 for price increase, and 0.64 for price decrease. When reflecting upon the grouping combination, it is Elon's impact on the price of Tesla. This contradicted our previous conceptions as we assumed that Elon would have had a substantial impact on the increase in Tesla's price. This then begs the question: is the hype around Elon as viable as we once thought? We will elaborate upon this in the discussion portion of our research.

	Precision	recall	F1-score	Support
Increase	0.55	0.42	0.48	269
Decrease	0.61	0.72	0.66	330
Accuracy	n/a	n/a	0.59	599
Macro avg	0.58	0.57	0.57	599
Weighted avg	0.58	0.59	0.58	599

Table 4.5. Classification Report of Logistic Regression DataFrame2

DataFrame2 was the combination of IXIC price data against the financial news data set. As this was the data set that performed the best in terms of the models, it is interesting to examine. Again, we have a similar situation in which there is quite a large gap between the increase and decrease as well as the number of elements that were not relevant. We can see that with the decrease the number of elements that were relevant was around 0.61. Nonetheless, we can see an overall improvement in the F1 score as far as the model is concerned. However, it is a marginal improvement.

	Precision	recall	F1-score	Support
Increase	0.27	0.03	0.05	116
Decrease	0.50	0.93	0.65	121
Accuracy	n/a	n/a	0.49	237
Macro avg	0.39	0.48	0.35	237
Weighted avg	0.39	0.49	0.36	237

Table 4.6. Classification Report of Logistic Regression DataFrame3

This is the classification report for the tweets of Elon Musk against the IXIC. Unsurprisingly, the scores are incredibly low overall and indicate low correlation and model viability. Upon deep diving into the individual respective columns, we noticed that the divide between the increase and decrease results for the precision, recall and F1 score were also significant. More importantly, the results for price increase were abysmal. This is exhibited quite clearly in the recall column where we see 0.03 for price increase, and 0.93 for price decrease.

As we mentioned earlier, our first intuition was to examine the support. From there we can see that structural differences are still not large enough to create this deep a chasm between the two of scores. The total observations for price increase were 116 and for price decrease they were 121. Also, the F1 score was also rather low with a result of around 0.49 for accuracy.

Lastly, when considering the overall data combination, it is fairly bizarre to think that the weight of Elon Musk's words appears to be directly correlated with price decrease. Often you would assume based on the hype of Elon, he would have a greater impact on the models in general.

	Precision	recall	F1-score	Support
Increase	0.56	0.50	0.53	288
Decrease	0.57	0.62	0.60	305
Accuracy	n/a	n/a	0.56	593
Macro avg	0.56	0.56	0.56	593
Weighted avg	0.56	0.56	0.56	593

Table 4.7 Classification Report of Logistic Regression DataFrame4

The results and distribution between the increase and decrease were better in this data frame combination. The combination we are looking at is Tesla's stock price against the financial news data set. Aside from the fact the scores are obviously low, there are a couple of other elements that stand out. These results were to be expected due to the sheer size of the dataset alone. In addition, as Tesla is spoken about often in the news it does not make sense that the data frame performed this poorly. The F1 score was also quite low with an overall score of 0.56.

MLPClassifier

Delving into the second model, we shall first present the MLP Classifier results. From there we shall deep dive into the respective sections and note the key takeaways and what they mean in terms of model performance.

Data	Best CV Score	Test CV score	Alpha	Hidden Layers	N-grams
DateFrame1	0.54	0.55	0.05	(10,10)	(1,2)
DateFrame2	0.63	0.63	0.05	(10,10)	(1,1)
DateFrame3	0.55	0.51	0.05	(10,10)	(1,2)
DateFrame4	0.54	0.53	0.05	(10,10)	(1,2)

Table 4.8. Accuracy Results and Optimal Parameters of the MLPClassifier.

Cross Validation Performance

This model performed marginally worse in terms of the best CV score and test score. A neural network should in theory perform better than the logistic regression, as each layer of the neural network possesses a linear relationship to the next layer. However, the non-linear activation function is what makes this different. Thus, from a model perspective it is the best. The versatility of the model alone is in part the reasoning behind the testing score of 0.63 for DataFrame2. In theory, a neural network can grasp intricacies within the data that the n-dimensional logistic regression models are incapable of grasping. As it is a multidimensional model, that aims to replicate human thought, the results were thus incredibly surprising. We were optimistic for the neural network and expected it to perform the best. However, this was not the case when comparing it to the logistic regression. The data frame that performed the worst in terms of testing accuracy was the Elon Musk dataset combined with the Nasdaq stock exchange price data. These results were immensely surprising.

The n-gram parameter

The tuning based on n-gram performed differently with the MLPClassifier. There was one instance in which it performed better with unigrams, and other examples where its preferred choice was bigrams. What is interesting is it indicates that in some cases the neural network does try to capture the entire picture of the data.

With DataFrame2, and the results it contained, we thought that bigrams or trigrams would be the preferred parameter choice. Alas, this remained the opposite of what we had expected. Unigrams prevailed as the optimal parameter choice within the data pipeline. This was different to the other data frame examples where they favored bigrams. Once again, like with the logistic regression, the parameter choice for DataFrame2 was considerably different. It leaves us wondering why this has occurred. Is the data set imbalanced? Is the data set merely large enough to warrant these results? It is difficult to have a definitive answer in relation to this due to the complexity of text data as well as the model choice.

In terms of the other data frames and their results. We could venture an answer in relation to why they performed better using bigrams. Combinations of words had a greater impact on model performance in the other data frames. It is slightly difficult to deduce why these models chose different n-gram preferences. It could also be in part due to the multidimensional nature of the model itself, or merely the data set choice.

The Best Parameter

As we have alluded to previously, it was a rather intricate task to tune the parameters for the neural network. Many have claimed that is unto an art itself to be able to choose the correct parameters. In the case of the pipeline and grid search combined, this was a tremendous task. Therefore, the results are quite varied in terms of their choice of tuning parameter. In the table 4.8, you can see that we have attempted to tune for alpha and the number of hidden layers. The combination of the two plays a role in controlling the complexity of the model. The alpha corresponds to the regularization of the model itself. According to Pedregosa *et al.* (2011), it is a means of combatting overfitting. By decreasing the alpha, we may fix the high bias, which is a sign of underfitting. By increasing the alpha, it could perhaps fix instances of high variance, which is a sign of overfitting. This encourages smaller weights and results in a decision boundary that possesses less curves. Our results contained quite a low alpha, and quite a high number of hidden layers. We have quite a low alpha of 0.05, which means it is curved, this indicates that we have a complicated decision boundary. So, the model is attempting to absorb as much information as possible from the complicated decision boundary. This comes at the cost of increased variance.

Our top parameter choice was around [10,10] for the hidden layer parameter. A second hidden layer is typically added to try and create a smoother decision boundary. The 10 refers to the units within a hidden layer. For this type of model, one would typically expect a couple of hidden layers considering the complexity of text data. To refer to the n-gram section, this makes sense in terms of the model choosing bigrams in some of the instances.

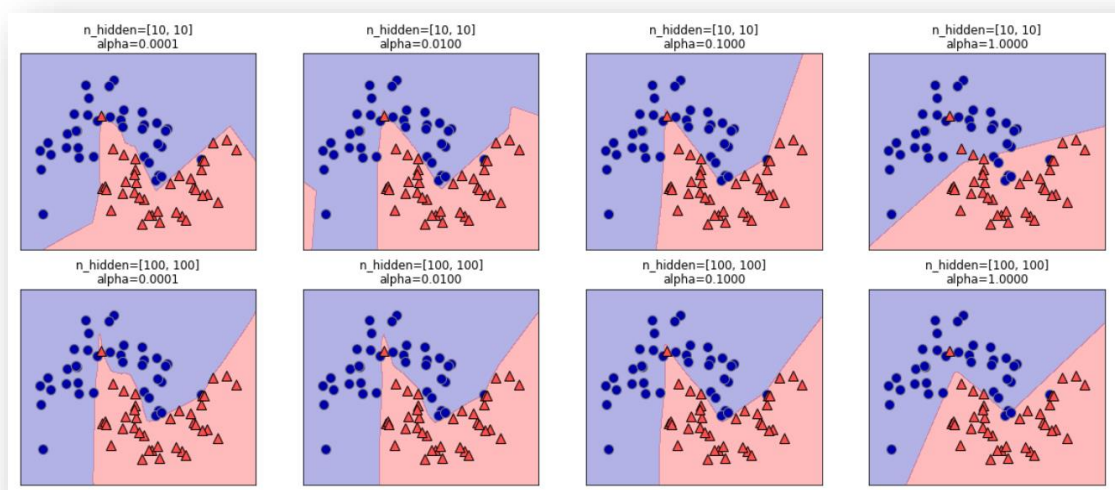


Figure 4.5 Hidden layers and alpha as understood by Müller & Guido (2016)

Coefficient Results

As it difficult to display the MLP coefficient results. This section shall unfortunately remain empty. A neural network is quite a complex and large model. Therefore, it is difficult to see the weight of the coefficients. In addition, the processing power itself is quite intense.

Classification Report

This portion was conducted similarly to that of the logistic regression. The classification reports were also printed. Again, we can see comparable results to the logistic regression. We can see that the models yield greater accuracy in terms of decrease as opposed to an increase in price movement.

	Precision	recall	F1-score	Support
Increase	0.49	0.45	0.47	109
Decrease	0.56	0.60	0.58	128
Accuracy	n/a	n/a	0.53	237
Macro avg	0.53	0.53	0.53	237
Weighted avg	0.53	0.53	0.53	237

Table 4.9. Classification Report of MLP DataFrame1

The first classification report we printed exhibited the evaluation of Elon Musk's tweets against Tesla's stock price. The results were interesting in terms of recall. It shows that Elon's words have a larger weight on the decrease in Tesla's price. These results were confusing to us as we would have expected the opposite. We can see that there are minor structural differences in terms of the support; however, this also seems too miniscule to matter overall.

	Precision	recall	F1-score	Support
Increase	0.55	0.42	0.48	269
Decrease	0.61	0.72	0.66	330
Accuracy	n/a	n/a	0.59	599
Macro avg	0.58	0.57	0.57	599
Weighted avg	0.58	0.59	0.58	599

Table 4.10. Classification Report of MLP DataFrame2

As this was the data frame combination that performed the best for the neural network, it is illuminating to see the intricacies behind the model performance itself. Again, we have another situation where the data does appear to favor the decrease as opposed to the increase in terms of classification of the data itself. As this is IXIC price data against the financial news data set it makes sense that it would perform this well. What is quite fascinating is that the F1 score for DataFrame2 is the same for both the neural network and the logistic regression.

	Precision	recall	F1-score	Support
Increase	0.51	0.30	0.38	116
Decrease	0.52	0.72	0.60	121
Accuracy	n/a	n/a	0.51	237
Macro avg	0.51	0.51	0.49	237
Weighted avg	0.51	0.51	0.49	237

Table 4.11. Classification Report of MLP DataFrame3

The results of DataFrame3 were quite curious. It exhibits the same theme as the others in terms of better results in relation to the category decrease as opposed to the category increase. The gap itself between the two is vast. The result of 0.72 in the recall for decrease as opposed to 0.30 for the increase is what originally perplexed us.

	Precision	recall	F1-score	Support
Increase	0.53	0.50	0.51	288
Decrease	0.55	0.58	0.56	305
Accuracy	n/a	n/a	0.54	593
Macro avg	0.54	0.54	0.54	593
Weighted avg	0.54	0.54	0.54	593

Table 4.12. Classification Report of MLP DataFrame4

Lastly, these were the results from Tesla's stock price in conjunction with the news data set. Although this data set did not contain the worst results, it also did not perform the best. When thinking about the combination, it makes us wonder about Elon. A financial news data set, which discusses all the major tech companies, was unable in the same regard to predict a price change with Tesla. Could this be due to Elon's power? Does Elon then have a greater impact on the stock price of us own company? In this capacity, it forces us to think about the very nature of a CEO's role in the overall price of their own company. It would have made more sense for several major news outlets to have better insight into Tesla's investor behavior. Yet, we are left puzzled and questioning this very notion.

Results Summary

For reference we will once again mention the various groupings of our data.

Name	Combination	Features	Rows
DateFrame1	Group1 vs TSLA	18	947
DateFrame2	Group 2 Vs IXIC	16	2395
DateFrame3	Group 1 vs IXIC	16	947

DateFrame4	Group 2 vs TSLA	18	2395
-------------------	-----------------	----	------

Table 4.13. Data Grouping

As we close the results section, let's summarize the main findings before we continue to discuss their relevance in the proceeding section. Overall, DataFrame2 performed the best of all data frames. Below are the various results from our two models:

Logistic Regression

Data	Best CV Score	Test set CV score	C Parameter	N-grams
DateFrame1	0.54	0.52	1	(1,1)
DateFrame2	0.66	0.67	10	(1,1)
DateFrame3	0.56	0.52	1	(1,1)
DateFrame4	0.55	0.56	1	(1,1)

Table 4.14. Results for Logistic Regression

MLP Classifier

Data	Best CV Score	Test CV score	Alpha	Hidden Layers	N-grams
DateFrame1	0.54	0.55	0.05	(10,10)	(1,2)
DateFrame2	0.63	0.63	0.05	(10,10)	(1,1)
DateFrame3	0.55	0.51	0.05	(10,10)	(1,2)
DateFrame4	0.54	0.53	0.05	(10,10)	(1,2)

Table 4.15. Results for the MLPClassifier

5 Understanding stock movement using social media

In the concluding portion of this research, we wish to discuss the meaning which we gathered throughout the process of this exploration. To state how we have gone about discussing these various points, please see the structure of our discussion and how it will progress.

We have outlined it as such:

- **Results Interpretation**
- **Hypothesis Discussion**
- **A comparison with previous works**
- **Theory in practice**

- **Organizational implications**
- **Study implications**
- **Future work**

As we begin the discussion, we shall aim to answer and ascertain whether we were in fact able to inform our main research objective, which was to gain insight into price movement through our specified methodological approach. In this process we shall also endeavor to link our results to our findings within the literature and theory discussed in this paper. The hope is that our work can both corroborate and negate some of the previously formulated theory. What should be iterated is that we were in no means attempting to revolutionize the field itself. We as researchers merely wanted to examine a highly contested area which we also had personal interest in exploring. As we subscribe to the pragmatist philosophy with an exploratory research strategy, it makes sense that we recognize that there are numerous approaches towards understanding problems and undertaking research. We maintained this philosophy throughout our paper and shall now continue with the same sentiment within our discussion. As pragmatists, we believe that no single view can give us an accurate picture of a situation, but a culmination of views always being more accurate. This way of thinking also holds true in lieu of understanding stock price movement. Before we progress, we shall restate our research questions once more. These will be referred to loosely throughout the course of the discussion part of our paper.

Research Question:

How can social media be used to gain insight into price movement?

→ ***How can we predict stock price movement in relation to public sentiment?***

Results Interpretation

Overall, our results left us with some conflicting conclusions about our research question when compared with our initial theories coming into this research. As we have stated, we initially believed that we would see a strong connection between Elon Musk's tweets and the stock price of Tesla or the NASDAQ Composite index. Although these results were not as good as we would have hoped, we were pleasantly surprised by the results when comparing the tickers to the financial news, specifically ^IXIC. We are content that our use of additional data sets indeed provided a more solid solution to our problem while also relieving us of bias. Now we would like to dive into our results and their meaning pertaining to this research.

Based upon our results, it is difficult to give a definitive answer regarding the plausibility of price movement prediction. On the one hand with our most impressive data set, DataFrame2, our

logistic regression model performed with an accuracy of around 67%. This indicates that around a little over half the time we can predict what the price movement is going to be on a given day. With the same data set however, the neural network performed with an accuracy over 63%. These results are still significantly better than we would have initially hoped for, as well as comparable to previous works. It does indicate that in some instances there is a relationship between stock price movement and social media text. Showing us that diverse text data that has a better gauge of the overall subject at hand will be a better predictor. This was seen when we compared the text performance of Elon's tweets versus the extensive financial news data. From this we gather that in order to obtain better and better results in this context, one must collect more and more social text data. However, it is difficult for us to say without any hesitation that stock market prediction is an easy task to undertake. The results themselves are not sufficient for one to develop a generalized approach towards trading.

However, in general we found some interesting nuances in our results which left us questioning the meaning behind them. The first element we noticed was the model's tendency to favor predicting a price decrease as opposed to a price increase. This was a common theme throughout each of the models as seen in the classification reports. In some cases, the chasmic divide between predicting increase and decrease deepened based upon the combination we had. This left us asking why? There are numerous reasons why the models could perhaps be better at predicting a price decrease. The first being that the data set is merely imbalanced. We could have simply had more rows of data that favored decrease. The second could be that the words merely have a larger impact on the price decrease. In the case of Elon himself, he is renowned for using the internet to his advantage. In some cases, he has taken it upon himself to express disdain for other companies. Thereby, causing their stock price to decrease. A third reason could be related to the public's vulnerability to negative media. It is quite commonly known that media companies benefit more from publishing dramatic and dooming headlines over nice ones. As we are constantly notified about these unnerving events, we could be more susceptible to reacting to negative media. We believe to find this answer it would require further research.

Moreover, could the above be contributing factors to the poor results obtained from the data combination of Elon's tweets and TSLA as well as Elon's tweets and ^IXIC. As stated, the results came as a surprise, as Elon consistently harnesses the internet through substantial amounts of engagement. We see this through his presence in the media as well as his popularity on social media. His success and uniqueness have warranted a large following with continuous support, specifically on Twitter where he is known for starting conversations about various companies. The low results of Dataframe1 left us questioning. This could have been due to many reasons. The first

being that the companies Elon mentions in his tweets either are not listed on the IXIC, or simply do not hold much weight. Additionally, we could have had too broad of a scope in relation to Elon's tweets and Tesla. In 2013 when Tesla became increasingly popular, it would have been interesting to limit the data closer to this period for that data frame. It could simply be that more negative headlines are able to capture potential decreases in price. In hindsight, we can see Elon's tweets as leaving something to be desired, as they are quite difficult to interpret and could be seen as random. The low volume of text that can be procured from his tweets could have hindered the results. In fact, we saw that having a more diverse text dataset would lead to the models performing better. However, it remains that a proper data pairing is the key to obtaining the better results.

To reinforce our results obtained DataFrame2, we would like to point out a happy mistake that made us see the value in this data pairing. Due to an early running of the models, before we utilized the ^IXIC index, we had taken the ticker NDAQ and compared it to the financial news data set. As described earlier in this paper, this is the corporation which runs NASDAQ stock exchange, not the index which follows many companies stock prices. This was a temporary mistake, but we saw significant result differences when comparing the financial news dataset to ^IXIC vs NDAQ. Regardless of the error, we stored the results and saw when using NDAQ the logistic regression yielded 58% and the neural network yielded 59%. For comparison, this pairing still performed better than data frames 1,3 and 4. This to us meant that the models had an easier time predicting price movements when relating financial news to an index (^IXIC) over a company that runs stock exchanges (NDAQ). Yet it is interesting that the financial news data produced better results when compared to NDAQ over TSLA. The diversity of our results further enforced the importance of choosing proper data pairings and the model's ability to detect these relationships.

When focusing on DataFrame1, DataFrame3 and DataFrame4 the significance of proper data pairing is further exhibited. Whilst the results themselves were overall too miniscule to deduce any concrete conclusions, it was interesting to note the lack of success these data frames possessed. To reiterate, these data frames displayed the impact a financial news data set had on the price of Tesla, as well as the effect Elon's tweets on both tickers. These results are in no way significant or revealing enough, as they do not differ from each other more than 1% in accuracy. To us this indicates that there is a need to further explore the relationship between general news and indices and to not focus as heavily upon individual stocks. However, we do believe that DataFrame2 is showing a path forward, while also displaying the importance of choosing text data that pertains to the subject at hand.

Nevertheless, we do not have faith in the generalizability of our *results* as they are at this moment, however interesting we find them. The reasoning for this stems from the lack of computational power that we possess, as well as access to all pools of data at once. Our models took an exceeding amount of time to process, and in doing so it made it difficult to conclude in the moment whether a certain price would go up or down. The increased processing time also limited us to trying new tuning parameters on our models. We could postulate that under certain circumstances if we were able to account for all information, or at least as much information as possible, we could in that instance give a better prediction of a price movement. We are not in any position where we can definitively say we can predict the price of stocks five years from this point. What our unstructured data merely shows is that short term prediction could be plausible. However, considering our data quality, this leaves us uncertain. As researchers, we feel it ethically irresponsible based upon our results to give a definitive yes.

Hypothesis

For efficiency, we shall restate our null and alternative hypothesis.

H_a = Tweet data will have a large impact on detecting price movement

H_0 = Tweet data will not have a large impact on detecting price movement

After our results section, we remain adamant that we have unknown knowledge that does not permit us to either accept or reject the null hypothesis. By unknown knowledge, we are once again referring to the cornucopia of wildly available information online. We therefore progress with the assertion that we choose to decline to reject the null hypothesis.

A comparison with previous works

As many previous works about stock market prediction in relation to text data inspired our research, we would like to revisit some of the works that provoked this study and compare our work. We would like to overall observe what we did differently, what may have worked better, or what could have yielded better results. To provide an overview, below is a table of the previous works we examined with their methods, models and obtained accuracy. For comparison we added our results to this overview.

Source	Method Type	Best Model	Accuracy
Mittal and Goel (2012)	Combined	Fuzzy Neural Network	75%
Bollen <i>et al.</i> (2011)	Combined	Fuzzy Neural Network	87%
Sprenger <i>et al.</i> (2014)	Machine Learning	Naïve Bayesian	64.2%
Ranco <i>et al.</i> (2015)	Machine Learning	Linear SVM	77%
Nti <i>et al.</i> (2020)	Machine Learning	Neural Network	77.12%

Jackson and McGraw (2021)	Machine Learning	Logistic Regression	67%
---------------------------	------------------	---------------------	-----

Table 5 A comparison of previous works

To summarize the previous works, we found research that primarily focused on either using lexicon-based sentiment analysis, machine learning sentiment analysis or a combination of both. While we thought these approaches sounded like the logical way to approach text data, as well as to derive meaning from the mass amounts of text online, we ourselves used neither of these techniques. The main goal of this paper was to utilize a machine learning approach to text analysis. As discussed in the literature, to classify sentiment using machine learning classification models, our data would need to have been pre-labelled with sentiment subjectivity. Alternatively, we could have used a lexicon approach to achieve a polarity score. Our data did not contain this, nor did we think it sufficient to simply classify our data into only three categories of sentiment (positive, negative, or neutral). The next step for us to use sentiment with machine learning would be to then label all data points, like what was done with the POMS (profile of the moods state) method, which was used by both Mittal and Goel (2012), as well as Bollen *et al.* (2011). This technique is exciting to us, and it also shows much improvement on traditional sentiment scoring by diving deeper into how words can be accurately categorized into human emotions. However, we simply did not have the time or workforce to manually label over 250,000 rows of data points to accurately reflect the sentiment of our text data.

When comparing the results achieved by some of the most intriguing previous works analyzed in our literature section, which were objectively better than the results achieved by our modeling, it seems that there is a superior method. To recall we employed similar models, as both logistic regression and neural networks were used in previous works and our research. In Mittal and Goel's (2012) analysis, they utilized a profile of the moods state to find mood classification of their text data. They then used these classifications to train their models. Whereas our approach focused upon the use of tokenization of the words in our data and then proceeding to utilize n-grams to find the optimal parameter choice. N-grams added a higher level of dimensionality that Mittal and Goel (2012) did not possess.

Mittal and Goel's (2012) technique being quite different, utilized various classification combinations when training models. This was done to capture how the separate classes would affect the accuracy of the model. This technique yielded a score of around 60% accuracy for the logistic regression algorithm for all variants of input, whereas the scores achieved by the neural network varied depended on the mood classification included. For example, the lowest score was given by a combination of all moods (Calm, Happy, Alert, Kind) at 64.4% accuracy, and the highest accuracy was attained through the combination of calm and happy at 75.5% accuracy. When

comparing outcomes, our results were more relatable to the results obtained by the combination of all moods in Mittal and Goel's (2012) analysis. With Dataframe2 (Financial news dataset and ^IXIC) achieving the highest score for both models, logistic regression with 67% accuracy and our MLPClassifier with 63% accuracy. Clearly the comparison of both models brings forth the method that works the best, but what was interesting to us was the combination of data which comprised Dataframe2 and its superior results when comparing to the other data frames.

Much like how we have combined our data in diverse ways, we could have added one more dimension to our analysis, by simply combining all text data (news tweets & Elon's tweets) and comparing this to our stock data specifically ^IXIC. Nti *et al.* (2020) saw their results significantly improve when combining their text data sources, although they utilized more than two, this could still produce higher accuracy within our results. By comparison, Nti *et al.*'s (2020) results were overall lower than our outcomes and slightly contradictory when comparing text datasets results. Their twitter data yielded the higher results of 55.5% accuracy. Compared to their online financial news data, which produced 50.43% accuracy. The interesting results emerged when they combined all their text data sources (Google trends, twitter, forum posts, and web financial news) to yield results of 70.66%.

Overall, it would be fascinating to see our combined results, but we simply do not possess the computational power. In combination with utilizing Nti *et al.*'s (2020) text data amalgamation, and that of Mittal & Goel's (2012) labeling of the text in detail as opposed to utilizing n-grams, we do see the possibility of improving our score when utilizing these additional techniques. Again, our only hesitation with personally labeling data or words to pertain to a mood classification is that one researcher's classification will not be broad enough to accurately depict human emotions. With the proper measures, it is our belief that this process can be fine-tuned into accurately categorizing emotions through text.

Theory in practice

Being able to accurately follow moods through a culmination of online data would lead to a greater understanding of cycles and trends existing in the market, as humans are likely the ones influencing them. Referring back the discussion of whether markets are a predictable space or if we are operating in a completely efficient market can still not be conclusive from this early work. Although from the initial work done here, it is more likely to us that there is an existing ability to predict movement to some capacity, if only the methods were perfected, and the data foundation was stronger.

While our results do leave us questioning the efficient market hypothesis, we do concur with the notion that it is exceedingly hard to accurately predict price movement, but we can see the case for predictability, with only a bit more access to data and processing power. As we recall from the theory section, the EMH, changes in stock price have the same distribution and are random. Thus, all price changes represent random departures from the previous price. However, social media could create and garner a collective group mindset that is both cultivated and analyzed using machine learning methods. This does somewhat refute the notion that we have a constant and uninterpretable flow of information. We find that we are certain about the notion that gathering information from the internet will assist in interpreting the constant flow of information. We do have the tools that can decipher the difficult flow of information but the ability, time, and effort to create them effectively remains to be the question.

Relating our results, specifically pertaining to results seen when utilizing the news dataset, we can see a relationship between the use of this data and the adaptive market hypothesis discussed earlier in this thesis. Particularly, when thinking of the market as having participants with bounded rationality. As this dataset was by far our largest source, as well as our most broad source, this data was our best gauge on the overall human discussions that were happening online, pertaining to financial subjects. The adaptive market hypothesis suggests that like humans experiencing evolution, the markets evolve, and the fittest survives. To us, online media seems like an invisible force, that is all encompassing and simply too confusing for any one individual to correctly interpret.

Therefore, this hypothesis is probable in this instance, reflecting that market participants do not act completely rationally but will consider the pieces of news (or media) around them to make the best decisions, or bounded decisions. To have the ability to capture the data written by the market participants, media influencers and noisemakers, to then interpret their meanings into clean piles of data would be an immensely powerful tool in pursuit of explaining the changes of financial markets. As a mass distributed tool, this would even make market participants more rational as they are able to make more informed decisions. Of course, we should consider the EMH in this instance, as once this is a tool available to the masses, there is no longer an advantage, only to those who constantly improve their methods. However, as the adaptive market hypothesis theory suggests the reason the market experiences inefficiencies are due to the cyclical and adaptive nature of its participants and their bounded rationality. Suggesting that in the future we will always experience new events and then adapt to them.

While our results do technically validate the efficient market hypothesis theory, or at the very least fail to dispute it. We remain convinced in the power of data, and that at some point, a researcher may have all the information available to easily predict a stock price movement. As human beings are irrational creatures that are readily influenced by the happenings around them it would be incredibly difficult to deduce how their behavior will be in the future. This could be paralleled to Elon Musk as a social media giant. His tweets are uncontrolled and often contain both insightful as well as absurd information. It is quite difficult for an outsider at any given moment to know what he will publish on his Twitter account and how that will affect other actions.

What does this mean for organizations?

While this research pertains specifically to text analytics and stock movements, the methods and meaning can be applied to different organizations. Thus, we would be most interested in the application of this research to organizations who develop relationships online with their customers. From the current trends of digitalization of the business, this will inevitably include most organizations in the future. Where our research would become useful is tapping into online sentiment pertaining to the organization. Whether this is from customers, market competitors or speculators on the industry, this type of analysis may give leaders the advantage they need to remain competitive in their industry so they can forecast outcomes that could harm their business.

More importantly, as we have continued to mention, Elon did to an extent have an impact on the model performance. Malhotra & Malhotra (2016) in their research encouraged the involvement of CEOs in their usage of social media tools. Whilst we can see to an extent that Elon does have an impact, it is difficult to state that it should bear any weight on organizations. However, once again, our results are not generalizable to CEOs across several industries. This brings us back to the words of Strauss & Smith (2019) and their assumption of Elon existing as a market anomaly. So, we are left with the question: should CEO's leverage their social media more? From a logical perspective, yes this would make sense. However, Elon does fall in a bit of a grey area. An area in which he exists as an industry influencer, as well as a risk taker that often pushes the boundaries of legality. In Strauss & Smith's (2019) paper they argued the risks of violating the United States Fair Trade act. Thus, as researchers we can see ethical dilemma of using Elon Musk as the figure head to look up to and follow in suit of. We do nonetheless note the potential in continuous exposure towards a company itself. By having a CEO utilize their social media, it can in many ways humanize the company and provide a personalized narrative that adds to the story telling journey of the company itself. However, it is difficult in the scope of this paper to examine the weight that CEOs possess. This would ultimately warrant its own paper.

There were other instances within our paper that we believed possessed significant value in relation to data analysis in general. This was seen in the use of business intelligence tools. In relation to our results, we certainly owe credit to our initial data exploration in Power BI. Due to this exploration of our data, we were not only able to see our data and view trends, but also pinpoint certain problem areas within the data. We believe these additional analyses led us to obtaining the best results for our data. Initially, we did not expect this to yield value as our main goal. Yet when visualizing our data, we gained an overall better understanding of what the data looked like and what the individual features contained. Power BI as a tool allowed us to visualize the intricacies of tweets and stock price data in many ways. The visualizations alone were pivotal in marking key moments in Tesla's social media journey in general. We were able to see moments in which Tesla had performed the best, and what tweets were published on specific days. For instance, using our word cloud we could click on the term "Battery" and see the tweet about the release of a new battery, or the declaration of perceived company success.

We were also extremely interested in trend analysis for both text data, as well as the price data, both were very forthcoming in the Power BI tool. In the process of our paper, we were able to utilize API's and several tools that were capable of being visualized in real time. For any business, this exhibits tremendous value. Using tools such as these makes a much more user-friendly data interaction experience. Whereas in the past it could take a whole team of analysts to make one report, now tools exist that can fetch data instantly, clean it and populate it into a report, much like we displayed here. Whilst it was not a tool that can be used for prediction directly. It was a tool that allowed us to generate and analyze social media insights as well as price trends. This was a question that we wanted answered and explained. However, we expected the python and machine learning portion of our paper to provide more value in terms of our data exploration.

Nonetheless, we found many practical business implications throughout our research. As we ourselves are business analysts, we are always working with data and tools to learn how to improve retrieval of our data as well as our interpretations of it. To us the key to a successful organization is one who knows its data and can then apply them.

E-business implications

The relevance of our work is laid within the overall perspective of our study program. E-business focuses upon opportunities information technology can provide private and public organizations alike. The three pillars of E-business are founded within IT, law, and business. In the course of our work, we have focused heavily upon the effective usage of specific types of technological tools, and how they can in fact be used in connection with business.

The tools we have utilized throughout the course of this research have alluded to a very specific subset of IT that seeks to predict the overall needs and desires of consumers. Machine learning, business intelligence tools, data pipelines and APIs are all essential tools that require effective strategizing and knowledge acquisition. Although the tools we used during this project do not contain the technology superiority that some of our predecessors have created, they do possess immense value and stress the importance of further investigation. During our research, we do touch heavily upon how an individual can step by step approach a problem, and solve it using the technology they have available. The code we have created is of course scalable and could potentially be implemented within a company.

From the legal perspective of technology, it has been interesting to examine the outlier case of Elon Musk. In his constant usage of social media, he has tweeted things that are on the line in terms of feeding investors illegal information about the company's success. Granted, our research did not focus upon this aspect of the E-Business program as it remained outside of the parameters of our projects research. We were legally in control of remaining compliant in terms of our extraction of tweets. However, we are not in control of the behavior of a social media personality. Thus, as an overall topic this seemed irrelevant in the grand scheme of prediction of price movement.

Throughout the course of this degree, we have upgraded our perspective of not only what a business should be, but what they are becoming. As we move into a technology focused world, we can see the value of learning and applying these tools. We see that the advanced business has begun to master the concept of "Big Data" and will soon begin to shift its focus to proactive data analytics along with predictive analytics or machine learning. Through this research we believe that we have challenged ourselves to learn more about data and prediction using technology that pushes boundaries. With this we hope to not only contribute our learnings to our program, but also research that intrigues other students to discover more about this subject. In summation, our research strove to embrace the program of E-business and take the practices that we learned throughout the program and apply them throughout the course of research.

Suggestions for future research

Throughout the research process we have been constantly learning since the conception of our idea. It was necessary to analyze existing theory, to find works that have been previously performed in this field, as well as finding data and producing results. The constant discovery process has put us in a state of perpetual learning which is still with us as we conclude this research. Due to this, we do see the future potential that this research can expand into.

As most recently mentioned, and most heavily mentioned, we could attempt to access more computational power than we had. With this we would like to explore how a news dataset and a major index such as the NASDAQ Composite would perform with machine learning sentiment analysis. Specifically, one like the POMS (profile of mood states) mentioned in this research. We believe that this analysis would hold immense value when done properly and thoroughly. As we could categorize our results as not bad, but also not great, we would be extremely interested in future research that took similar data and applied these methods.

Overall, we follow the premise that there is in fact predictability in the market, be it short term or cyclical. Further research into the predictability of various indices and individual stocks could provide great insights when used with the correct text data. Furthermore, the exploration of machine learning techniques is equally as important in this endeavor. Given the possibility to quickly process our models we feel as though there would be more room for improvement, any future researcher will need to explore this. Interestingly there are companies, or funds rather, who use these techniques. They process financial news data using natural language processing, which is then used within mathematical models that will accurately predict price movements in financial markets.

Whilst our aim did not extend to predict a price, we have geared our efforts towards understanding how insights can be generated from social media text data. We understand that the generalizability of our results remains solely within the domain of the Nasdaq Composite Index and Tesla. Granted, our results were in no way outstanding, yet we feel obliged to state the obvious. Further diving into the complicated neural networks, and deep layered kernel machines could bring about accuracy far better than we have achieved here. Additionally, we see the exploration of topic modeling being immensely valuable in this kind of research.

Even if we were to examine different areas, there is an additional level of complexity that we would have to take into consideration. Whilst geographically filtering for data would not be a problem, the primary issue that would emerge lies in the translation of different languages. Words bare different weights across different countries. Whilst our models merely look at coefficient weights, this does posit a question for the general sentiment analysis tools that focus upon predefined lists or dictionaries. Future work, at least in the branch of “sentiment analysis” should aim to expound their limitations for various geographical regions.

Whilst it did aid us in our journey in understanding text analytics and price movement, we do think that the models could have been significantly improved upon if we had an overview of all data pools. As we mentioned in our literature review, some individuals seek to combine the

fundamental and technical analysis tools. Even though we tried to replicate this, we believe that due to data availability we fell short. If we were to have knowledge of all areas and incorporate this into a machine learning model, then stock market prediction could in theory be possible. Fundamental analysts in their journey take into consideration all factors pertaining to a company. This could include everything from CEO messages to earnings, expenses, assets, and liabilities etc. We do remain convinced that social media is incredibly powerful as a modern leverageable tool, we do think that certain parts of society and the business world remain unpublished. A future case study could seek to incorporate all these features into a neural network.

6 Final Thoughts

As we have demonstrated throughout the course of this research, the potential and boundaries of social media are endless. Whilst the text data extracted from twitter was cumbersome to handle in its unstructured state, we were able to gain immense insight into price movement in general. Above all, our research aimed to in many ways use social media to gain insight into the human approach of investing. In doing so, we hoped that our insights may allude to the decision-making processes that may occur when a person inevitably looks at a social media post.

We have gained appreciable amounts of insight into not only the possibilities of social media data, but also the powers and prospects of machine learning. These are factors we expected to get to know through the course of this research, however the additional knowledge we gained was equal to this. Diving into theories around the subject of stock prediction in general opened our eyes to the many opposing views that exist today. We also were equally surprised by the plethora of dimensions which take careful consideration, such as data selection and methodological choice. Through our research we were able to formulate an opinion based on our methodology and results. While we don't think there is a perfect answer to be found we do believe that we have been able to greatly improve our knowledge of the subjects at hand.

Our primary research question was indeed answered throughout the course of this paper. We can use social media to gain insight into price movement. At several points within our methodology, we were able to prove this with our use of data analysis and visualization tools. As we can see the very beginning of this relationship, we believe that there is much more to be explored. Within this framework we see the possibility for vast improvements and exciting insights.

References

- Abirami, Abdullah, Askarunisa, Selvakumar, & Mahalakshmi. (2017). Sentiment Analysis. In Handbook of Research on Advanced Data Mining Techniques and Applications for Business Intelligence (pp. 162-174).
- Ali Hasan, Sana Moin, Ahmad Karim, & Shahaboddin Shamshirband. (2018). Machine Learning-Based Sentiment Analysis for Twitter Accounts. *Mathematical and Computational Applications*, 23(1), 11.
- Aggarwal, C. C., & Zhai, C. X. (2012). *Mining Text Data*. Springer New York.
- BLACK, F. I. S. C. H. E. R. (1986). Noise. *The Journal of Finance*, 41(3), 528–543.
<https://doi.org/10.1111/j.1540-6261.1986.tb04513.x>
- Bollen, J., Mao, H., & Zeng, X. (2011). Twitter mood predicts the stock market. *Journal of Computational Science*, 2(1), 1-8. doi: 10.1016/j.jocs.2010.12.007
- Brownlee, J. (2020, January 14). *A Gentle Introduction to the Fbeta-Measure for Machine Learning*. Machine Learning Mastery. <https://machinelearningmastery.com/fbeta-measure-for-machine-learning/>.
- Brownlee, J. (2020). *A Gentle Introduction to the Bag-of-Words Model*. Machine Learning Mastery. Retrieved from <https://machinelearningmastery.com/gentle-introduction-bag-words-model/>.
- Chartmill. (n.d.). chartmill.com. <https://www.chartmill.com/>.
- Ciaburro, G., & Venkateswaren, B. (2017). *Neural Networks with R*. Birmingham, UK: Packt Publishing.
- Dhankar, R., & Maheshwari, S. (2016). Behavioural Finance: A New Paradigm to Explain Momentum Effect. *SSRN Electronic Journal*. <https://doi.org/10.2139/ssrn.2785520>
- Electric Cars, Solar & Clean Energy*. Tesla. (n.d.). <https://www.tesla.com/>.
- Dhaoui, C., Webster, C., & Tan, L. (2017). Social media sentiment analysis: lexicon versus machine learning. *The Journal of Consumer Marketing*, 34(6), 480–488.
<https://doi.org/10.1108/JCM-03-2017-2141>

- Exchanges, T. W. F. of. (n.d.). *Welcome to the Future of Markets*. Welcome to the future of markets | The World Federation of Exchanges. <https://www.world-exchanges.org/>.
- Fama, E. F. (1970). Efficient Capital Markets: A Review of Theory and Empirical Work. *The Journal of Finance*, 25(2), 383. <https://doi.org/10.2307/2325486>
- Gabriele Ranco, Darko Aleksovski, Guido Caldarelli, Miha Grčar, & Igor Mozetič. (2015). The Effects of Twitter Sentiment on Stock Price Returns. *PloS One*, 10(9), E0138441.
- Hasan, A., Moin, S., Karim, A., & Shamshirband, S. (2018). Machine Learning-Based Sentiment Analysis for Twitter Accounts. *Mathematical and Computational Applications*, 23(1), 11. <https://doi.org/10.3390/mca23010011>
- Illowsky, B., & Dean, S. (2013, July 18). *Null and Alternative Hypotheses*. Introductory Statistics. <https://opentextbc.ca/introstatopenstax/chapter/null-and-alternative-hypotheses/>.
- Investopedia. (n.d.). Investopedia. <https://www.investopedia.com/>.
- Ji, X., Wang, J., & Yan, Z. (2021). A stock price prediction method based on deep learning technology. *International Journal of Crowd Science*, 5(1), 55–72. <https://doi.org/10.1108/ijcs-05-2020-0012>
- Kaplan, A. M., & Haenlein, M. (2010). Users of the world, unite! the challenges and opportunities of social media. *Business Horizons*, 53(1), 59-68. doi: 10.1016/j.bushor.2009.09.003
- Kochkodin, B. K. (n.d.). *GameStop's Reddit Believers Find Hope in Short-Squeeze Mention*. Bloomberg.com. <https://www.bloomberg.com/>.
- Lo, A. W. (2004). The adaptive markets hypothesis. *The Journal of Portfolio Management*, 30(5), 15-29. doi:10.3905/jpm.2004.442611
- Lo, A. W., & Repin, D. V. (2002). The Psychophysiology of Real-Time Financial Risk Processing. *Journal of Cognitive Neuroscience*, 14(3), 323–339. <https://doi.org/10.1162/089892902317361877>
- Li, N., & Wu, D. D. (2010). Using text mining and sentiment analysis for online forums hotspot detection and forecast. *Decision Support Systems*, 48(2), 354–368. <https://doi.org/10.1016/j.dss.2009.09.003>
- Machiraju, S., & Modi, R. (2018). Developing Bots with Microsoft Bots Framework: Create Intelligent Bots using MS Bot Framework and Azure Cognitive Services. Berkeley, CA: Apress.

- Madhoushi, Z., Hamdan, A., & Zainudin, S. (2015). Sentiment analysis techniques in recent works. 2015 Science and Information Conference (SAI), 288-291.
- Macy, M. W., Mejova, Y., & Weber, I. (2015). Twitter. a digital socioscope. In *Twitter. A digital socioscope*(Vol. 1 - Analyzing Twitter Data, pp. 21-51). Cambridge: Cambridge University Press.
- Macy, M. W., Mejova, Y., & Weber, I. (2015). Twitter. a digital socioscope. In *Twitter. A digital socioscope*(Vol. 6 - Disaster Monitoring, pp. 131-160). Cambridge: Cambridge University Press.
- Malkiel, B. G. (2003). The efficient market hypothesis and its critics. *Journal of Economic Perspectives*,17(1), 59-82. doi:10.1257/089533003321164958
- Malhotra, A., & Malhotra, C. (2016). How CEOs can leverage Twitter. *MIT Sloan Management Review*, 57(2), 73-29.
- Mittal, A., & Goel, A. (2011). Stock Prediction Using Twitter Sentiment Analysis.
<https://doi.org/http://cs229.stanford.edu/proj2011/GoelMittal-StockMarketPredictionUsingTwitterSentimentAnalysis.pdf>
- Müller, A. C. & Guido, S. (2016). Introduction to machine learning with python: A guide for data scientists. O'Reilly Media
- Nguyen, T., Shirai, K., & Velcin, J. (2015). Sentiment analysis on social media for stock movement prediction. *Expert Systems with Applications*, 42(24), 9603–9611.
<https://doi.org/10.1016/j.eswa.2015.07.052>
- Niederhoffer, V. 1997. *Education of a Speculator*. New York:
John Wiley & Sons
- Nofer, M. (2015). *The value of social media for predicting stock returns: preconditions, instruments and performance analysis*. Springer Vieweg.
- Nti, I. K., Adekoya, A. F., & Weyori, B. A. (2020). Predicting stock market price movement using sentiment Analysis: Evidence FROM GHANA. *Applied Computer Systems*,25(1), 33-42. doi:10.2478/acss-2020-0004

- Nti, I., Adekoya, A., & Weyori, B. (2020). A systematic review of fundamental and technical analysis of stock market predictions. *The Artificial Intelligence Review*, 53(4), 3007–3057. <https://doi.org/10.1007/s10462-019-09754-z>
- N. (2021, March 31). NASDAQ Composite. Retrieved 2021, from https://indexes.nasdaqomx.com/docs/FS_COMP.pdf
- Pai, P., & Liu, C. (2018). Predicting Vehicle Sales by Sentiment Analysis of Twitter Data and Stock Market Values. *IEEE Access*, 6, 57655-57662.
- Pandas. (n.d.). *pandas documentation*. pandas documentation - pandas 1.2.4 documentation. <https://pandas.pydata.org/docs/>.
- 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. doi: 10.1016/j.eswa.2014.07.040
- Pedregosa, F., Varoquax, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... Duchesnay, E. (2011). *Journal of Machine Learning Research*, 12.
- Qaisi, L., & Aljarah, I. (2016). A twitter sentiment analysis for cloud providers: A case study of Azure vs. AWS. 2016 7th International Conference on Computer Science and Information Technology (CSIT), 1-6.
- Qiao, J., & Wang, H. (2008). A self-organizing fuzzy neural network and its applications to function approximation and forecast modeling. *Neurocomputing*, 71(4-6), 564-569. doi: 10.1016/j.neucom.2007.07.026
- Ranco, G., Aleksovski, D., Caldarelli, G., Grčar, M., & Mozetič, I. (2015). The Effects of Twitter Sentiment on Stock Price Returns. *PloS One*, 10(9), e0138441–e0138441. <https://doi.org/10.1371/journal.pone.0138441>
- Roberts, H. 1967. Statistical versus clinical prediction of the stock market. Unpublished manuscript, CRSP, Chicago: University of Chicago, May.
- Saunders, M., Lewis, P., & Thornhill, A. (2019). *Research methods for business students*. (8. ed.). Pearson Education Limited.

- Satchell, S. (2007). In *Forecasting expected returns in the financial markets* (pp. 1–8). essay, Academic Press.
- Seskar, A., Milasinovic, B., & Fertalj, K. (2018). Workflow for image categorization using cognitive computing services. 2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), 1551-1556.
- Simon, H. A. (1997). *Models of Bounded Rationality*.
<https://doi.org/10.7551/mitpress/4711.001.0001>
- Sprenger, T., Sandner, P., Tumasjan, A., & Welpe, I. (2014). News or Noise? Using Twitter to Identify and Understand Company-specific News Flow. *Journal of Business Finance & Accounting*, 41(7-8), 791-830.
- Strauss, N., & Smith, C. (2019, June 4). Buying on rumors: How financial news flows affects the share price of Tesla. Retrieved from
<https://www.emerald.com/insight/content/doi/10.1108/CCIJ-09-2018-0091/full/pdf?title=buying-on-rumors-how-financial-news-flows-affect-the-share-price-of-tesla>
- Vijayakumar, T. (2018). *Practical API Architecture and Development with Azure and AWS: Design and Implementation of APIs for the Cloud*.