Handelshøyskolen BI

GRA 19703 Master Thesis

Thesis Master of Science 100% - W

Predefinert informasjon

Startdato:	16-01-2022 09:00	**Termin:**	202210
Sluttdato:	01-07-2022 12:00	**Vurderingsform:**	Norsk 6-trinns skala (A-F)
Eksamensform:	T		
Flowkode:	202210||10936||IN00||W||T		
Intern sensor:	(Anonymisert)		

Deltaker

Navn:	Audun Foyn og Mattias Emil Haughom

Informasjon fra deltaker

Tittel *:	Stock Return Prediction with Sentiment Analysis of Twitter Data
Navn på veileder *:	Jonas Moss

Inneholder besvarelsen konfidensielt materiale?:	Nei	**Kan besvarelsen offentliggjøres?:**	Ja

Gruppe

Gruppenavn:	(Anonymisert)
Gruppenummer:	235
Andre medlemmer i gruppen:	

Master's Thesis

Stock Return Prediction with Sentiment Analysis of Twitter Data

Mattias Haughom
Audun Foyn

Business Analytics
BI Norwegian Business School
Oslo
July 2022

Abstract

We attempt to make improvements to stock return prediction accuracy through sentiment analysis of Twitter data. Our hypothesis is that Twitter users mainly consists of retail investors, implying that the aggregation of sentiment will influence stocks with lower levels of institutional ownership.

Our analysis involves thee sentiment approaches. The first approach gives labels to tweets based on magnitude and direction of changes in the stocks price. The second is a manual labelling approach, where the authors went through tweets manually and determined whether the tweets had a positive, negative or neutral sentiment. The last is using a dictionary created from financial tweets. For the first two approaches, we utilised three text classification methods Naïve Bayes, Logistic Regression and SVM.

The Sentiment features were used in tandem with common financial features, momentum, liquidity and volatility, to compare predictive power through three supervised regression models, Random Forest, Gradient Boosting and a neural network model - LSTM.

We find that including sentiment in the models decrease accuracy slightly across all models, and that including the level of stock institutional ownership has limited effect on improving predictions, in our selected sample. We argue that larger data size may be beneficial create an accurate market sentiment proxy, and that sentiment analysis should be more useful when focusing on special cases, like peak volumes and the number of followers.

Preface

We would like to express our gratitude to our supervisor Jonas Moss, for valuable insights and always being available for discussions.

We would also like to thank the Twitter users for sharing their colourful enthusiasm for stocks, providing us with rich sentiment data. Last but not least, we would like to thank Igor from the Twitter developer forum for a quick response to our Twitter data issues. Without you, we would have wasted at least two more days banging our head against a brick wall.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

According to traditional financial theory, investors are rational and stock prices accurately reflect all relevant information about a firm's value. If prices deviate from their fundamental value, rational investors will quickly identify under- or overpricing, taking advantage of deviations in the price, subsequently causing a reversal back to the fundamental value.

Behavioural financial theory, on the other hand, states that market sentiment, and a range of other psychological factors, can cause prices to deviate from their fair market value for long periods of time. We have seen several examples of this over the past decades, like the dot com bubble of 2001[1] and the more recent GameStop short squeeze[2]. Many researchers therefore no longer question whether or not market sentiment may cause stock prices to deviate from their fundamental value, the question is rather how to optimally quantify its effects (Baker & Wurgler, 2007).

An important explanation why the GameStop short squeeze took place is the increase in active retail investing. For long, retail investors have been largely dominated by passive and systematic investment plans, making them irrelevant when framing market forecasts. However, retail investors now account for almost

[1]Harvard Business Review
[2]Business Insider

as much volume as mutual funds and hedge funds combined (US data)[3], making retail investors significant players in the stock market, especially when their behaviour is coordinated.

Stock return prediction is a common and challenging problem within finance. This is because of the erratic and dynamic changes that can occur to a stock's price during trading hours. Technological developments and increased availability of more diverse data have allowed new stock prediction methods to emerge. The analysis of textual data from social media is one of these trends.

This thesis looks at three different methods of estimating retail investor sentiment through the analysis of Twitter posts. Twitter is a social media platform where people from all over the world share their views and participate in discussions on countless topics, including what stocks to buy or sell. This makes Twitter a suitable data source for input into the modelling of market sentiment.

Sentiment analysis involves the process of accurately determining the direction of these opinions into positive, neutral or negative categories. The labels of one period are aggregated into a sentiment indicator that should, in theory, tell us something about the direction that the stock is trending or is going to trend. We try to combine sentiment indicators with traditional financial machine learning features to see what additional information the sentiment can bring to stock return prediction. We evaluate the models' performances by comparing their prediction accuracy to the accuracy of a baseline model.

We find that the created sentiment indicators do not improve stock return prediction when combining them with common financial features. We argue that the lack of useful information in the sentiment features, mainly comes from relatively small data size for the companies selected, problematic Twitter data and not focusing enough on special aspects of the Twitter data.

We assume that if a company is largely owned by institutional investors, there will be fewer discussions about this company's stock price online, and the extent

[3]Financial Times

to which they reflect the owners' views is limited. The aggregated sentiment of these discussions will therefore have less influence on stock prices. Instead, online stock discussions are dominated by retail investors, such that stocks primarily owned by retail investors are more influenced by aggregated Twitter sentiment.

To investigate whether our assumption holds water, we compare prediction accuracy between groups of companies with low institutional ownership and high institutional ownership. We find that the level of institutional ownership is not substantively important when comparing the accuracies across firms. We argue that the latter is more likely because of a lack of predictive power from our sentiment variables, rather than ownership being unimportant.

The thesis is structured as follows: Chapter 2 outlines previous research on market sentiment in the context of finance. Chapter 3 presents the data sources and the different methods that we have used in the modelling of market sentiment and stock returns. Chapter 4 and 5 summarise our results and discuss our findings. Chapter 6 concludes the thesis by emphasising key findings and main takeaways.

Chapter 2

Literature Review

Social media has an increasing presence in modern society. People from all over the world can interact with each other, share opinions and participate in discussions online. In other words, these platforms contain information on what the masses think, which can be useful in various topics of research. In finance, researchers believe that textual data may contain information about abnormal returns, sparking an interest in measuring the direction of social media sentiment through natural language processing (NLP) methods.

The notion of sentiment describes an established or future preference, in our case the preference of owning, purchasing or selling a specific stock. Sentiment analysis refers to methods that identify and extract this subjective information (Xing, Cambria & Welsch, 2018). Sentiment analysis is an approach where many different parts of the NLP method are used in tandem, for example, subjectivity detection (Chaturvedi, Ong, Tsang, Welsch & Cambria, 2016), user profiling (Mihalcea & Garimella, 2016), and aspect extraction (Poria, Cambria & Gelbukh, 2016). In this section, we provide a summary of empirical evidence on the relationship between sentiment and stock prices, as well as more recent developments in stock prediction with textual analysis.

2.1 Sentiment Analysis and Financial Forecasting

The study of sentiment, and its impact on asset returns, has a long history in financial literature. Cowles (1933) categorised an editorial column in the Wall Street Journal from 1902-1929, as "doubtful", "bullish" and "bearish", subsequently trying to predict future returns of the Dow Jones. Niederhoffer (1971) divided New York Times headlines into 19 different semantic categories. He found that markets have the tendency to overreact to bad news. Contemporary sentiment-based predictions are computationally driven, but they still apply conceptually similar methods as Crowles and Niederhoffer (Gentzkow, Kelly & Taddy, 2019).

2.1.0.1 Dictionary

Sentiment analysis in finance has typically gone in two main directions. The first is to use word lists or dictionaries to determine if a word falls within the negative- or positive dimension. They are either manually created, usually by linguistic experts, making them smaller in size but more accurate, or they are first constructed from some manually selected seed words and then automatically expanded by rules defined by application (Li, Xie, Chen, Wang & Deng, 2014).

Tetlock (2007) used the Harvard IV-4 manual dictionary to predict individual firm stock prices, using text from a popular Wall Street Journal (WSJ) column. He used the fraction of negative words in each article to measure the tone, condensing daily sentiment scores into a single pessimism factor. He found that the observed negative effect on stock returns and pessimism related to the WSJ column is not because it provides new information about company valuations, but rather that it is a proxy for investor sentiment.

Loughran and Mcdonald (2011) later identified that many of the negative words used in the Harvard dictionary were not applicable to finance. They solved this problem by creating their own finance specific list by analysing 10-K reports. The dictionary performed better relative to other existing dictionaries, indicating

that negative and positive words depend on which discipline you are studying.

Chen, De, Hu and Hwang (2014), used the negative words from the Loughran and Mcdonald dictionary to capture sentiment on a social media platform. Their research proved that posts are associated with stock returns and earnings up to three months ahead. Nevertheless, Loughran and Mcdonald (2016) states that using the 10-K report financial dictionary to analyse the sentiment of other media, including social media without modification would be problematic. They suggest using the 2011 dictionary as a common base with explicit modifications, such as identifying and reporting words that dominate the word counts, to reduce miscalculations.

Another disadvantage with dictionaries is that they usually give words a proportional value, implicitly assuming that the sentiment of all words is equally important (McGurk, Nowak & Hall, 2020). For example, the word *trivial* can have the same negative weight as *worthless*, even though the latter would typically indicate a more negative sentiment. Certain dictionaries account for this through several different categories of words and sentimental weight, but these are not specific to financial social media.

2.1.0.2 Vectorisation

The second main direction is to transform words into quantitative weights through vectorisation. One of the simpler language processing methods that have been used extensively in financial literature is the bag-of-words (BoW) approach. Text is split into words, removing the sequence, determining the "importance" of words by their frequency (Tetlock, 2007; Xing et al., 2018). This method is often used because of its good performance while being a simple model to implement (Renault, 2017).

The BoW approach relies heavily on being able to filter out common, unimportant words and phrases, distinguishing between frequency and importance. For example, words such as "if" and "then" will have a high frequency, but will say

little about the underlying sentiment. For social media, this might be difficult, as many posts rely on specific vocabulary (Loughran & Mcdonald, 2016).

There are a number of different possible weighting schemes available in linguistics literature, Term Frequency - Inverse Document Frequency, (TF-IDF) is another common method. TF-IDF consists of two terms, with the first term simply placing high weight on frequent words. The second term places less weight on words if they occur very frequently across all documents, and as a result, weighting less frequent words higher.

Multiplying the terms means that very frequent words and very rare words will have low weight. (Loughran & Mcdonald, 2011). This reduces noise from very common words that occur in many different tweets, which is why this method is often used for text filtering (Gentzkow et al., 2019).

Renault (2017) used both BoW and TF-IDF when vectorising social media message board texts for use in his supervised methods. He found that BoW performed slightly better than TF-IDF when classifying sentiment, a result that was consistent with previous research.

2.1.0.3 Alternative methods

An alternative method of giving weights to words is through manually labelling text to identify relevant words, subsequently using different machine learning methods to classify more text (McGurk et al., 2020; Renault, 2017; Taddy, 2013). For example, Bartov, Faurel and Mohanram (2018) uses both traditional dictionary approaches, and classification with the Naïve Bayes method, where the probability of being positive and negative is calculated based on manually labelled text. They also weight tweets by the number of followers the individual has.

They further classify tweets into those about company fundamentals and those about everything else. They find that the quarterly aggregated sentiment of the tweets can predict stock returns prior to an earnings call, but that it makes little

difference if you use dictionaries or classification, or if the tweet regards fundamentals.

Renault (2017) also performs analysis above dictionary approaches and compares several different sentiment measuring methods. Following previous work, he utilises a subset of 750 000 messages already tagged as positive or negative by online investors while also creating another lexicon by manually labelling words that appear above 75 in their sample. He compares the performance against the Loughran and Mcdonald (2011) finance dictionary, the Harvard dictionary and a maximum entropy classifier. The two traditional dictionaries are significantly outperformed by the other methods, supporting the conclusion that one needs to create field-specific dictionaries for social media and message boards to optimise accuracy.

Some researchers have also found that it is possible to label sentiment in the market by first noticing market prices and then giving labels to reactions. One known example is Jegadeesh and Wu (2013), who labelled term weights into more negative or more positive based on the range of market reactions to 10-K reports. Unlike previous literature, they found a relation between positive words and the market.

For social media, Jaggi, Mandal, Narang, Naseem and Khushi (2021), use stock price changes both above and below 2%, and simply positive and negative changes to label tweets positive on positive return days and negative on low return days, creating two separate datasets of labelled tweets. The labels were aggregated on a daily basis to create sentiment indicators used for prediction.

These methods are interesting because they remove much of the subjectivity from dictionary and manual approaches, while being extremely fast. The main issue is that there is a higher probability of giving the wrong label to tweets because you give labels based on other factors than the content in the individual tweet.

2.2 Stock Market Prediction and Social Media

As the frequency and popularity of social media sharing and discussions have increased, the prospect of using the platforms as proxies for investor sentiment has also improved. Since the 2000s, researchers and investors have paid attention to the efficiency and accuracy of information online.

Many studies have focused on testing the theory of the "The wisdom of crowds" (Surowiecki, 2004). The theory poses that information that is aggregated results in decisions that are more accurate than if one were to consider isolated pieces of information alone. In this view, professionals are more likely to herd, where they focus too much on one central piece of information, instead of the aggregated sentiment of the market. These studies often look at message boards, only they are mostly finance specific.

Early research of internet forums found no link between message board activity and abnormal stock returns (Tumarkin, 2002; Tumarkin & Whitelaw, 2001). Antweiler and Frank (2004) found a small but significant effect from pessimistic comments, and that message board activity can predict volatility. Furthermore, they also found that higher disagreement between the commenters was linked to higher subsequent trading volume.

Later studies have presented various new ways to measure the link between stock discussions on Twitter and movements in stocks, and have found mixed, but more promising results. For example, Hill and Ready-Campbell (2011) used data from Motley Fool, a stock prediction community, and showed that their strategies, based on predictions, outperformed the S&P 500.

Ranco, Aleksovski, Caldarelli, Grčar and Mozetič (2015) found that the effect from discussions on prices is low when looking at whole periods, however, when looking at the peak of twitter activity they find a significant dependence. The amount of cumulative abnormal returns is low, 1-2%, but significant for several days after the peaks.

McGurk et al. (2020) also found some evidence that investor sentiment influences stock returns. They estimate a firm-specific and daily sentiment measure through manually labelling negative and positive words in 3000 tweets, creating a Twitter finance specific dictionary. They find that both increases in negative and positive sentiment are related to abnormal returns, with the relationship being slightly stronger for smaller companies. In terms of being useful for prediction, they found that gains to forecast accuracy were limited to around 1% over a constant only model.

Nguyen, Shirai and Velcin (2015) looks at topic sentiment, in contrast to the overall sentiment or mood of a market, and compares their method to a baseline on historical prices. They find that the sentiment of topics discussed on a company (product, service, divided etc.) improved their model by 2%.

In contrast, looking at general sentiment towards companies at an aggregate level, Nofer and Hinz (2015), found no effect. Although, when including more influential people, they found that their sentiment measure created a strategy that beat market benchmarks by double digits.

Lastly, Khan et al. (2020) looked at both textual data from Twitter and news to create two sentiment features to predict the stock market. They used a classifier to predict the directional movement of particular stocks and showed that the social media measure gained a higher accuracy of 80% relative to 75% for news.

Overall, previous research find that improvements to investment strategies, and to the general relationship between sentiment and future stock return are limited without checking for special cases. Looking closer at these cases, for example, the number of followers and peak Twitter volume seem like an essential part of extracting the value of this type of data. With that said, most of the above cases focus on regression methods rather than machine learning, which have different goals. We will focus on the value of sentiment for predictions in this thesis.

2.3 Return Prediction Feature, and Model Selection

One of the most studied phenomena in financial literature is prediction of expected returns. Machine learning methods are largely specialised for prediction problems, and because expected returns is the conditional expectation of a future realised excess return, it makes machine learning ideal for this estimation (Gu, Kelly & Xiu, 2020).

Traditional time series models typically assume historical data and future data as predictors and target variables, respectively, directing to establish a relationship between them. Many also require valid ranges for various parameters and their connections, as well as assumptions about underlying distributions. These will be unrealistic to fulfil when attempting to take advantage of large volumes of information and predicting the nonlinear characteristics of the stock market (Jin, Yang & Liu, 2020).

A diversity of models able to approximate complex nonlinear relationships between predictors, make machine learning well suited for such challenging prediction problems (Gu et al., 2020)

The literature has identified a large list of features that are argued to influence expected return, with hundreds of features on the stock level and dozens of features on the market-level (Green, Hand & Zhang, 2013; Harvey, Liu & Zhu, 2016; Welch & Goyal, 2008). This also makes machine learning more attractive because you can easily consider a large number of features at the same time, and subsequently reduce them to the most important.

Gu et al. (2020) does a comprehensive review of the most important features for the most common and tested models in financial machine learning, tree-based and neural networks. They test over 900 different features, identifying the most important as momentum, liquidity and volatility-related.

These dominant predictive signals are consistently important over all the methods tested, including simple, penalized and generalised linear models and a number of different machine learning models. Their prediction horizon was monthly predictions over 60 years, on almost 30 000 different stocks.

Typically, studies attempt to predict the closing price of a stock on a one-day horizon (Jiang, 2021). The closing price is the last price recorded before the market closes. Rather, we chose return prediction because this feature is less autocorrelated, this will in turn give us a better indication if sentiment features are improving predictions of our models. We go further into detail about this in the method section.

2.4 Our Contribution and Hypothesis

Previous research focused attention on either Twitter, news or finance related message boards to calculate market sentiment. We propose a similar setup, where sentiment is quantified by comparing manual labelling, a field-specific dictionary and an automatic labelling approach based on price changes, using it to create a model that predicts several specific stocks for a specified period.

This research will deviate from previous work by looking at ownership structures of the stocks in our sample, to understand how the effect from market sentiment affects companies differently. A possible hypothesis could be that stocks where retail investors own the majority of stocks are most affected by social media discussions. We assume this relationship because most Twitter users are regular retail investors, indicating that the aggregation of Twitter sentiment mostly influences stocks with low levels of institutional ownership.

Chapter 3

Method and Data

This section outlines the methods used, why they were chosen, and how we performed our analysis and evaluation. Figure 3.1 illustrates the process from data collection to stock prediction.

At the base level, we describe the stock market and Twitter data that were preprocessed and cleaned such that we could perform feature extraction and analysis. After this, we describe the processes of sentiment analysis and how the output was used in the machine learning process.

Machine learning is involved in both sentiment analysis and in the stock return prediction. One difference to keep in mind is that classification was used to give categorical labels to text, while regression models were used to predict stock return. Lastly, we outline the evaluation methods used to determine the performance of the models.

The main goal of this research is to investigate how, and if, aggregated Twitter sentiment and differences in institutional ownership can improve stock return prediction. We theorise that the effect of social media sentiment largely improves predictions when the focus is on companies with low levels of institutional ownership.

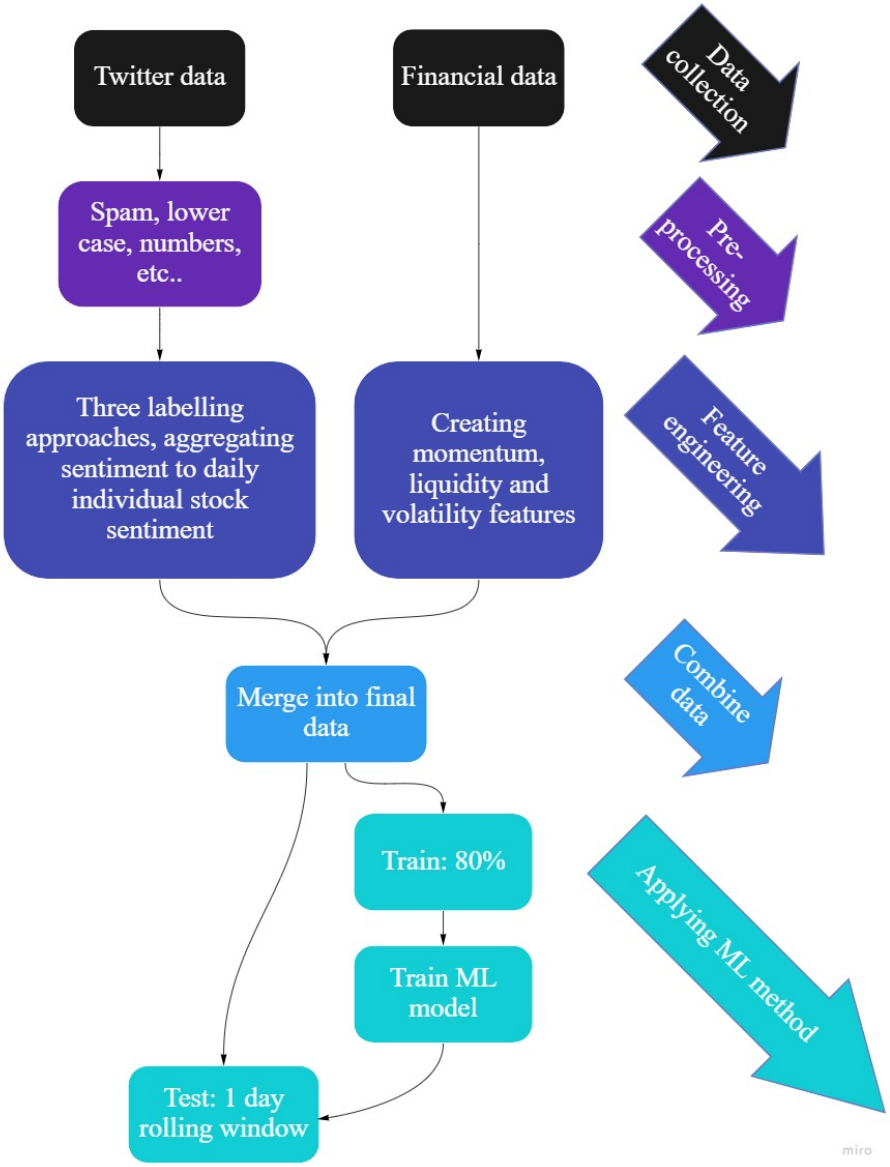

Figure 3.1: Method Process Chart

3.1 Data

3.1.1 Stock Data

The analysis involved 19 stocks from the S&P 500 index, presented in table 3.1. The index consists of the 500 biggest US companies based on market value of outstanding shares[1]. Tickers are an arrangement of letters, representing publicly

[1]The total number of shares already owned by shareholders

traded securities. For example, Apple Inc will be listed as AAPL on the NASDAQ stock exchange.

Ticker	Company
UHS	Universal Health Services
CTRA	Coterra Energy
GPS	Gap Inc.
AAL	American Airlines Group
LYV	Live Nation Entertainment
O	Realty Income Corporation
NLSN	Nielsen Holdings
F	Ford Motor Company
HST	Host Hotels and Resorts
XOM	Exxon Mobil
DISCA	Discovery Communications
BA	Boeing Company
REG	Regency Centers Corporation
BF-B	Brown–Forman Corporation
UDR	UDR Inc
NCLH	Norwegian Cruise Line
CDAY	Ceridian HCM Holding
AMCR	Amcor plc
WU	The Western Union Company

Table 3.1: Stock Ticker and Company Name

For this analysis, we were only interested in the companies with either the highest or lowest fractions of float[2] owned by institutional owners. Institutional owners can mean mutual and pension funds, asset managers, university endowments etc. Institutional ownership data for most listed companies is available on Yahoo Finance, which was used to filter for the relevant tickers in this analysis.

An overview of the different company tickers, the total share of institutional ownership and the number of tweets per company can be found in Table 3.2. A company can have over 100% of its shares owned by institutional investors, because of delays in reporting ownership between institutions and short selling (Lewellen, 2011).

[2]Shares that are available for purchase by the public

High institutional ownership			Low institutional ownership		
Ticker	% of Float	# Tweets	Ticker	% of Float	# Tweets
UHS	115.03%	4554	CTRA	57.21%	1066
GPS	112.51%	17082	AAL	56.87%	37163
LYV	112.17%	6951	O	54.89%	19526
NLSN	108.70%	5541	F	53.94%	96212
HST	107.16%	11945	XOM	53.46%	71015
DISCA	106.38%	9132	BA	53.33%	219228
REG	105.99%	5854	BF-B	51.60%	4092
UDR	105.37%	3797	NCLH	50.12%	9600
CDAY	105.31%	4347	AMCR	41.47%	992
WU	104.94%	6737			

Table 3.2: Level of Institutional Ownership and Number of Tweets

The stocks that are the most popular on Twitter was excluded from the analysis. For example, Apple and Tesla have low institutional ownership, but they would capture a large portion of the Tweet cap (limited to a certain number of tweets downloadable each month), limiting the horizon and number of companies we would be able to investigate.

Daily and hourly data was downloaded for the selected stock tickers, in addition to daily data about the VIX[3] index, the S&P 500 index, and different industry indexes (e.g. Health or Industrials). The VIX index is calculated from implied volatility of option contracts on S&P 500 companies, and is therefore one of the best measures of expectation of volatility for the next month for the stocks we are investigating. These downloads were done to calculate features and to backtest the return prediction model. Other sources were utilised when Yahoo Finance could not supply historic levels, like for example, outstanding shares, which needs to be downloaded from paid services.

Typical market data consists of closing price, high and low price for that interval, open price and volume. We mainly used closing price and volume in combination with the data mentioned above for our analysis. Closing price is the last price reported in a trading interval (e.g. a day), while volume is the number

[3]Chicago Board Options Exchange

of stocks traded that interval.

Table 3.3 illustrates the format of the data, where for example, "AAL" represents a ticker, and the numbers below represent closing prices.

Date	AAL	AMCR	BA	BF-B	CDAY
03/01/2022	18.75	11.76	207.86	71.39	104.94
04/01/2022	44611	11.94	213.63	71.23	100.57
05/01/2022	18.68	11.96	213.07	71.65	92.94

Table 3.3: Closing Price Data

3.1.2 Twitter Data

After pre-processing, the analysis consisted of a total of 1.5 million tweets, posted during a four-year period from February 12 2018 to February 22 2022.

A search query consists of the relevant stock tickers and the "Cashtag" ($). Typically, tweets posted with financial applicability use "$" + "ticker" when referring to a specific company. Hence, this method relies on the notion that many discussions on financial aspects of companies are done using cashtags. To the best of our knowledge, all relevant tweets available are collected.

As figure 3.2 illustrates, there is a large variation in the number of tweets when comparing tickers. This may influence the final results such that we see less effect on tickers with low volume, or it may generally lead to lower confidence in the final results. However, we argue that it may still be interesting to investigate whether there are any sentiment effects from stocks with low volume on Twitter.

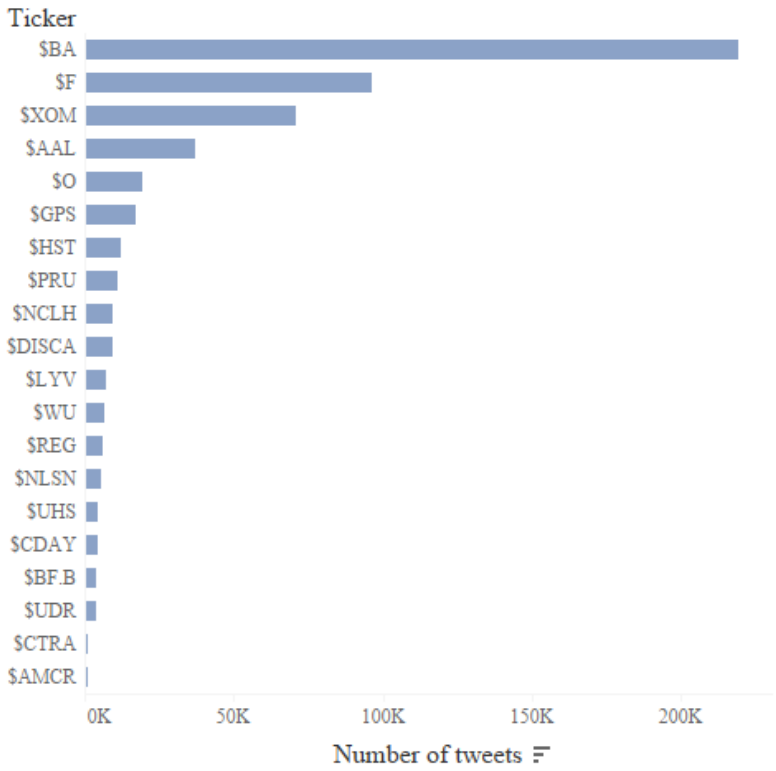

Figure 3.2: Number of Tweets per Company

3.2 Time Series Data

Stock prices tend to be non-stationary, displaying significant autocorrelation. The current closing price is normally very similar to the previous day's closing price. This aspect is useful when the goal is to forecast trends in prices. A model would then choose to focus most its attention to the previous closing prices to predict the next periods.

In contrast, our goal was to understand what *drives changes in the closing price*. With this objective, it is no longer as useful to focus on the previous day closing price because previous changes may not inform future changes. Rather, it comes down to choosing the data that have influence on driving these every day changes.

Changes in the stock price is calculated by taking the current closing price, subtracting the previous day's closing price, dividing by the previous day's closing price. This will force the models to focus on other features than the previous time steps, a much stronger test of the models' predictive powers.

$$Return_t = \frac{Close_t - Close_{t-1}}{Close_{t-1}} \tag{3.1}$$

Logarithmic returns are optimal to use because they are closer to stationary, however, we did not use log returns in this thesis. Nevertheless, returns are smaller over shorter periods, and because log returns are approximately equal to returns if x is small. We can expect the daily returns to be equal to the log returns (Ruppert & Matterson, 2019). In this regard it will not have a large impact that we have not used log returns because we only perform one day predictions.

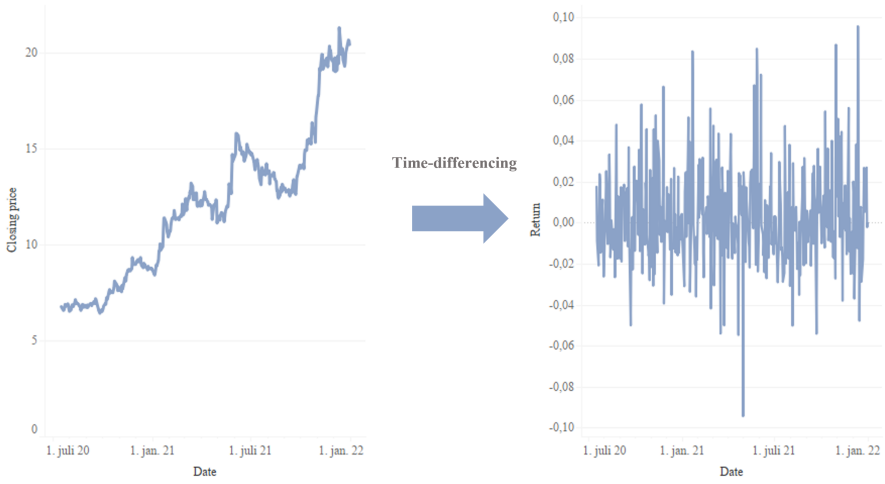

Figure 3.3: Closing Price Time Differencing

3.3 Pre-Processing

Pre-processing is the process of cleaning and structuring the such that features and results can be extracted. Stock data only needs simple data transformations

to be ready for feature extraction, hence we will not describe this in detail here.

Twitter data on the other hand, needs several steps of pre-processing to be ready for sentiment analysis. The first step, is to filter the json API output, selecting the data of interest. It is unfortunately not possible to filter out non-English tweets in the query, hence, these are removed as the first step in the pre-processing.

The next step is to remove what we like to call spam tweets. These are tweets that tend to follow the same patterns, often linking to content outside Twitter and referencing a group or community where one can earn money investing. A typical example of a spam tweet is presented in the box below.

> *"i'm part of a server where top traders actually get paid based on results (over $70k last month). all based on merit. that's how you know you're getting the best alerts https://t.co/lpiheiob99 $nclh $nkla $baba $aaps $bbd $tlry $pltr $itub $didi $fb $m $edu $uber $vale $navx"*

We subsequently create day, date and time columns which are commonly used in machine learning and time series forecasting, and in our case, to use these variables to merge different data and aggregating sentiment features.

The following pre-processing steps are performed on the text before analysing the tweet content.

- Transform all text to lowercase
- Remove stop words[4], punctuation, URLs, mentions, hashtags and emojis
- Replace repeating characters in words with the stem; hurryyyyyyyyy -> hurry
- Compressing elongated words using a word segmenter; for words longer than 100 characters

[4]Stop words that may speak to a stock price's direction were excluded from the list of stopwords (up, down, above, below, not)

- Fix contractions; words that are shortened, omitting certain letters
- Removing numbers
- Changing all tickers of interest to “COMPANY” and all other stock tickers to “OTHERCOMPANY”

We change stock tickers to COMPANY/OTHERCOMPANY in attempt to train a model that can generalise better. For example, if Exxon Mobile (XOM) has a very good period of returns in the chosen training period, the model may mistakenly associate "XOM" with positive sentiment. However, if we substitute "XOM" to "COMPANY", the model will look at the positive sentiment associated with XOM as general statements, as illustrated below.

The same tweet has been pre-processed to become completely general. A text classifier can thence predict similar statements as positive/negative for other companies.

$xom plans to maintain its capital spending through 2027

⇓

COMPANY plans maintain capital spending

3.4 Sentiment Analysis

The goal of sentiment analysis is to accurately label a given body of text. This thesis includes three main labelling methods. The first is looking at labelling tweets based on the size of the percentage change in price that day. The second is a supervised machine learning approach, trained on manually labelled tweets and the last is based on the simple finance Twitter sentiment dictionary of McGurk et al. (2020).

Table 3.4 presents an overview of the three methods. The percentage change and manual labelling approaches are similar because they require both text vec-

	Approach to tweet labeling		
	Percentage change	Manual	Dictionary
Manual/automatic	Automatic	Manual	Automatic
Requires vectorisation	Yes	Yes	No
Requires ML	Yes	Yes	No

Table 3.4: Sentiment Approaches

torisation and machine learning. Thus, the first part of this section will explain the idea and process of the two approaches, before we explain the vectorisation and classification models chosen. We then describe the dictionary approach, and present other features that we created based on sentiment.

3.4.1 Percentage Change Method (PCM)

This method follows the same logic as in Jaggi et al. (2021) and Jegadeesh and Wu (2013), where market changes are used to label text, which is opposite to other sentiment methods. In this thesis, we label twitter posts based on positive and negative market returns. The method can be broken down into several steps outlined in Figure 3.4. We will hereafter refer to the Percentage Change Method as "PCM".

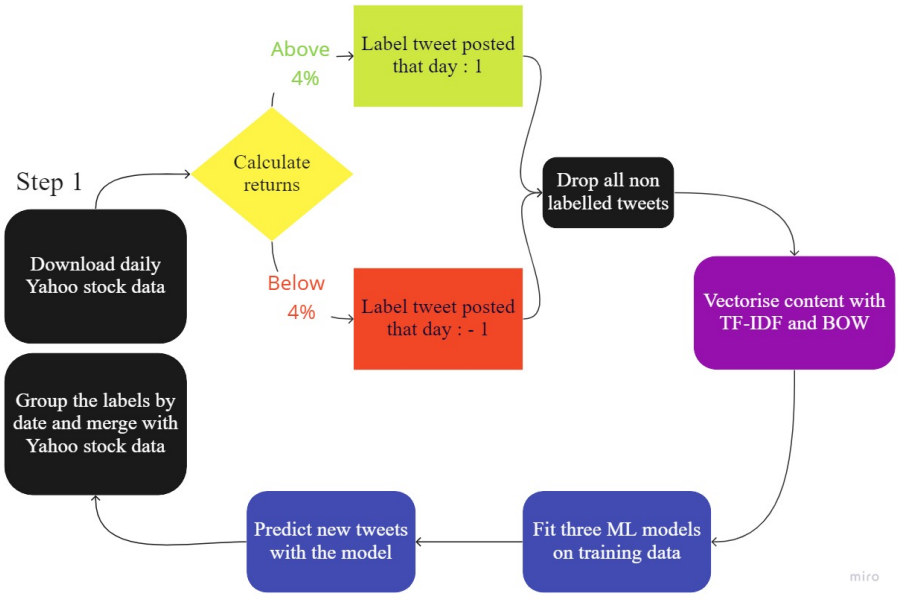

Figure 3.4: PCM Process Chart

The first step is to download daily prices of all tickers for the same period where tweets are available. The data was split into several periods over the years from which we had Twitter data available, as presented in Figure 3.5. Social media is changing constantly, hence, we split the data into the three recent trends we could interpret from the data. These periods are (1) the pre-covid period, (2) the strong bull market after the crash of 2020, and (3) the bear market in early 2022. The model should train on data that is closer to what it is trying to predict, which we believe is more likely after splitting the data. The two other methods have other ways to adjust for different time periods which we will describe later in this section.

Figure 3.5: S&P 500 Index Chart Periods

The initial step is to calculate daily simple returns. Then, as seen in equation 3.2, in a new column, we label individual rows 1 if the return for that stock has been above 4% that day, and -1 if the stock has returned less than -4%. All other rows, where the stock has returned between 4% and -4%, we label 0.

$$Label = \begin{cases} 1, \text{ if return} > 0.04 \\ \text{-1, if return} < 0.04 \\ 0, \text{ if return} \in [-0.04, 0.04] \end{cases} \tag{3.2}$$

We subsequently merge the stock data into the tweet data on date, such that all tweets that mention a particular stock are categorised with the label associated with the stock return of that day.

We are interested in giving the correct labels to tweets, which means accurately labelling a tweet positive or negative. Ideally, a model would train on texts that have very different words, to easily notice the differences between the categories. The model would then easily identify the tweets that are positive or negative when predicting out of sample text. We hypothesise that this is more likely when the price changes are larger because social media users are more likely to voice their opinion in line with the changes in the market.

Table 3.5 illustrates the logic behind the method. For example, when stock return is above 4% the first day, all tweets posted that day will be labelled 1, indicating positive sentiment. Next day, the return is 1% and the same tweet that was labelled 1 the day before is now labelled 0. In other words, the tweets are labelled regardless of the actual tone of the text, the labelling is rather determined by the return in the market.

Daily stock returns		Labelling of tweets			
Date	Return	Time	Label	Tweet	Date
07/01/2022	0.04	09:35:00 a.m.	1	Elon dusk think it will go to moon, buy	07/01/2022
		10:35:00 a.m.	1	Real value stock, paying dividens from 2100	07/01/2022
		11:35:00 a.m.	1	Best products ever, soon to the moon.	07/01/2022
08/01/2022	0.01	09:20:10 a.m.	0	Elon dusk think it will go to moon, buy	08/01/2022
		10:25:00 a.m.	0	Insane bull on this stock	08/01/2022
		02:55:00 p.m.	0	To much debt, needs capital raising	08/01/2022
09/01/2022	-0.06	03:10:00 p.m.	-1	Uncertain demand in future	09/01/2022
		05:15:00 p.m.	-1	Elon dusk think it will go to moon, buy	09/01/2022
		05:15:00 p.m.	-1	Higer interest harder to service debt, good luck	09/01/2022

Table 3.5: Illustration of PCM labelling

After tweets are labelled, we filter out all neutral tweets, and split the remaining labelled tweets into train and test datasets, using 80 per cent of the dataset for training and 20 per cent of the dataset for testing. The steps for text vectorisation and machine learning are identical to the manual labelling approach. Therefore, before we continue with explaining these next steps, we will first describe how the manual labelling approach differs from PCM.

3.4.2 Manual Labelling Approach

The main philosophy of the manual labelling approach is to filter out much of the noise (spam, unrelated posts etc.) that is posted and to identify important words that are used frequently to express opinions. We found that this is necessary because much of the tweets posted are nonsensical and may not express any explicit sentiment.

We initially pre-process the data before sampling, such that spam is reduced. The reduction of spam is a repeating process because one continuously notices patterns of spam in the data when looking through it. Then we remove duplicates and save a random sample from the whole data-period. After this, we were left with a dataset of 3500 Tweets.

The main issue with doing our own labelling is that it may introduce subjectivity, that our own categories and techniques are arbitrary.

To reduce these concerns, the tweets were labelled manually by both authors in a similar technique as in McGurk et al. (2020). Both participants took at least 30 seconds on each tweet to ensure enough time to analyse the sentiment of the tweet. In cases where tweets were labelled inconsistently by the authors they were discarded because it was uncertain what the true sentiment was. For example, the same tweet was labelled 1 and -1 by the two authors, would result in the tweet being removed. This resulted in a sample of 3283 tweets, which means that the error rate of the manual labelling was approximately 6%.

3.4.2.1 Challenges with Manual Labelling

Labelling tweets that voice financial sentiment is difficult because financial vocabulary is complicated, especially when including options contracts. It is therefore not likely that a person without a finance background can label all tweets correctly.

For example, one tweet may be positive for the company of interest, but neg-

ative for another.

> *COMPANY plans to double production, might heavily dent OTHERCOMPANY*

Lacking the context of the tweets sometimes makes it hard to actually say if a tweet is positive, neutral or negative for the stock. For example, a post referencing an specific USD in sales for a month says little when no more context is provided.

Ultimately, it comes down to some individual words that provides some indication of the actual sentiment. Nevertheless, even with difficulties and individual cases of uncertainty, the authors agreed on most tweets, and the aggregated result may still improve the validation of other methods and increase accuracy of predictions of new text.

The main benefit from the manual approach is that it is easier to validate the subsequent steps, because it is more likely that the target variable is correctly labelled. Although, our challenges in determining the correct sentiment for a large part of the tweets does question a model's ability to determine the sentiment correctly and the overall value that these posts will have in improving price prediction.

3.4.3 Text Vectorisation

Once the tweets are labelled, either by using the PCM or the manual labelling approach, the next step is to turn the text into vectors. We have used two alternative methods for this exercise, the term frequency-inverse document frequency method and bag-of-words method. We describe these methods in the following sections. In 4, we compare the results from the two methods.

The different vectors will contain information about the text, in a format that is easy to process for computers.

3.4.3.1 Bag-of-Words

Bag-of-words (BoW) is a very common method in financial textual analysis (Loughran & Mcdonald, 2016). The order words is ignored, the focus is instead on how many examples there are of each word.

c is a matrix, where element c_{ij} has a value equal to the number of times each word j is present in tweet i (Gentzkow et al., 2019). This means that BoW places larger weight on words that occur more frequently in the training data.

We apply a common adjustment to the method by log normalising the pure word-counts, as this reduces the influence of document length to the impact of certain words. Log normalisation results in smaller impact from very large differences. For example, if the word *loss* occurs 10000 times more often than the word *destitution*, it will reduce the impact of *loss* relative to *destitution* (Loughran & Mcdonald, 2011).

Without log transformation, word importance will be correlated with dataset length, making common words significantly more important than rare words that still may be important to the semantics.

BoW is not good at capturing the effect of low frequency words that may still be important for the context or sentiment of the sentence. However, it is a simple method that has proved efficient for predicting Tweet sentiment in previous research (Renault, 2017; Tetlock, 2007).

3.4.3.2 Term Frequency-Inverse Document Frequency

Term frequency-inverse document frequency (TF-IDF) can be used as an alternative to BoW. It is comprised of two main aspects.

The first aspect is term frequency, (tf_{ji}) refers to the count c_{ij} of occurrences of each word j in document i. This is exactly the same as BoW, increasing the weight of frequent words.

Inverse document frequency on the other hand, (idf_j) is the logarithm of the total number of documents n over the number of documents containing j : $log(n/d_j)$ where $d_j = \sum_i 1[c_{ij} > 0]$. The score and object of interest, is the product $tf_{ij} \times idf_j$ (Gentzkow et al., 2019).

This means that both very rare and very frequent words will have low weight when trying to predict sentiment of a text. Very rare words will have a low weight because the term frequency tf_{ij} is low. Similarly, very common words will have low weight because idf_j will be low. This means that TF-IDF attempts to give larger relative weight to less common words compared to BoW.

3.4.4 Text Classification Models

In previous sections, we describe how we have used both PCM and the manual labelling approach to determine the sentiment of a selection of tweets. We also describe how we weighted the importance of tweets using Bag of Words and Term Frequency-Inverse Document Frequency.

These vectors of weighted words can be further processed as a input in supervised learning methods for prediction of text sentiment.

Three supervised models, Naïve Bayes, Logistic Regression and Support Vector Machine were used for classifying sentiment, ultimately selecting the best model for use in further analysis.

Modelling procedure:

1. Use training data set as input
2. Specify supervised learning method and relevant parameters
3. Training: Identify patterns and relationships in the training data set
4. Get the objective function capable of classifying out of sample tweets
5. Test: Use the test data to check how well the model is generalising

Common for all three models is that they use features (the vectorised words of a tweet) to classify an out of sample target (sentiment).

A well functioning classifier can accurately identify important patterns in the training data to determine the class of an out of sample tweet.

3.4.4.1 Logistic Regression

Logistic Regression is a non-linear probability model that describes how a range of observable features influence the probability that a target variable belongs to a certain category (Provost & Fawcett, 2013). In the context of this thesis, the aim of the Logistic Regression model is to classify the sentiment of tweets (target variable) based on the words that the tweets contain (the features).

We have two categories, positive sentiment and negative sentiment. The words are vectorised with either BoW or TF-IDF to give each word a weight. We can, therefore, refer to one tweet as a vector of features x_i, while all tweets can be represented as x.

The separation of the groups is called the decision boundary, which for logistic regression can be represented by equation 3.3.

$$f(x) = \sum_{i=1}^{n} w_i \cdot x_i + b \tag{3.3}$$

The decision boundary equals a weighted sum of all vectorised tweets, where w_i represents weights and x_i represents tweets.

The model fits on the training data by estimating the optimal weight of each feature in its relevance in discriminating between the classes. The optimal weights are those that maximise the likelihood of making the observations that we have made. (Provost & Fawcett, 2013).

The linear function $f(x)$ in equation 3.3 is transformed to the logistic function 3.4.

$$P_{positive}(x) = \frac{1}{1 + e^{-f(x)}} \tag{3.4}$$

$P_{positive}(x)$ represents the probability that a tweet with features x is positive. The probability that the tweet is negative is $1 - P_{positive}(x)$.

The fitted Logistic Regression model takes an out of sample vectorised tweet as input and outputs the probability that it has a positive sentiment.

$$\text{Classification of } x_i = \begin{cases} 1 & \text{, if } x_i > \text{decision boundary} \\ 0 & \text{, otherwise} \end{cases} \tag{3.5}$$

From 3.5, if a new tweet has the probability of 0.6, the tweet would have been classified as positive if the decision boundary was 0.5. Section 6.2.1 in the appendix contains information on the parameters we used.

3.4.4.2 Naïve Bayes classifier

The Naïve Bayes approach derives an estimated probability that an event $C = c$ occurs conditional on an observed event E. In the context of textual analysis, event c is a sentiment category and E is a vector of independent words (Jurafsky & Martin, 2021).

The model differs from Bayes theorem in that it assumes conditional independence between the observed words. This is a strong assumption, but it simplifies the estimation substantially. The alternative would require estimating the probability of observing a specific combination of words, which is a much more challenging task than estimating the probability of observing a single word independent of

the other words in the text (Provost & Fawcett, 2013).

Equation 3.6 shows how Bayes' Rule is transformed after applying the "naive" assumption of conditional independence between the words in the dataset.

$$P(C = c \mid \mathbf{E}) = \frac{P(\mathbf{E} \mid C = c) \cdot P(C = c)}{P(\mathbf{E})} = \frac{P(e_1 \mid C = c) \cdot P(e_2 \mid C = c) \cdots P(e_k \mid C = c) \cdot P(C = c)}{P(\mathbf{E})} \quad (3.6)$$

On the left hand side of equation 3.6, $P(C = c \mid \mathbf{E})$ represents the probability that words in vector $\mathbf{E}$ belongs to sentiment category c. On the right hand side, $P(\mathbf{E} \mid C = c)$ represents the weight of features (individual words) that we can observe in a certain class. The weight of these words depends on the vectorisation method used. $P(C = c)$ is the percentage of examples in training data that has class c.

The denominator is usually left out of the equation in classification problems because it is unnecessary for determining if the probability a positive class being smaller or larger than the probability of negative class (Provost & Fawcett, 2013). Section 6.2.2 in the appendix contains information on the parameters we used.

3.4.4.3 Support Vector Machine Linear Classifier

Simplified, the Support Vector Machine (SVM) defines a loss function, referred to as "hinge loss", and the optimisation method Stochastic Gradient Descent (SGD) minimises it. This separates the algorithm from Logistic Regression, in that Logistic Regression uses likelihood in its optimisation rather than hinge loss (Provost & Fawcett, 2013).

Figure 3.6 is an illustration of how SVM separates between two classes. The model trains on a set of features which are labelled into two different classes and represents them as points in an n-dimensional space. The algorithm then separates

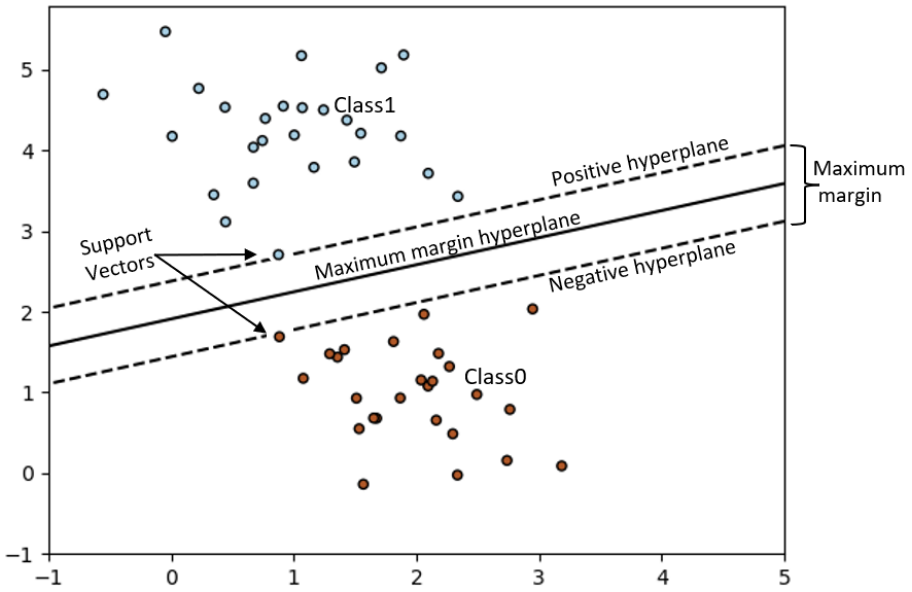

Figure 3.6: General illustration of linear SVM classifier

the points by a hyperplane[5], with the widest possible gap between the classes. The position of the hyperplane margins is determined by the points closest to it, the "support vectors" (Pimpalkar & Raj, 2020).

Out of sample features are then mapped to their locations and classified to their respective class. SVM is good for text analysis because of its relative robustness for overfitting and ability to handle high dimensional data (Pimpalkar & Raj, 2020).

Stochastic Gradient Descent is the optimizing algorithm(SGD) using the derivative of the function minimized in direction of the steepest ascent to find the optimal stationary point. The algorithm starts optimizing with larger steps, reducing the size of change when changes of results becomes smaller."Stochastic" is the description of the models ability to optimize globally by randomly using new data on each iteration, not getting a false local optimum (Bottou, 2010).

Section 6.2.3 in the appendix contains information on the parameters we used.

[5]A decision boundary that is one dimension lower than the number of features

3.4.4.4 Choosing the Best Model

The best model is determined by comparing the different models' classification accuracy, which is reflected in the ratio between the number of accurate predictions and the total number of predictions. The ratio is presented in equation 3.7 below.

$$Accuracy\ score = \frac{\#\ Accurate\ predictions}{\#\ Total\ predictions} \tag{3.7}$$

The predictions/labels were aggregated to daily sentiment measures, that either indicate positive or negative daily sentiment for each stock.

PCM predictions were validated against text that was also labelled using PCM. This means that we are not sure if the tweets used in the test-set actually contains positive or negative sentiment, rather they are posted on a day with high or low return.

In other words, the results of the classification does not necessarily measure the accuracy of the model's ability to determine negative or positive sentiment, but rather the model's consistency and ability to determine tweets that are posted on high or low return days (Right or wrong prediction). The summarised results can be found in Chapter 4.

3.4.5 The Dictionary Approach

Dictionaries are positive and negative word lists, which are used to give negative or positive scores to text. They are very common within financial textual analysis, where initial work used non-domain specific dictionaries like the Harvard IV dictionary, and gradually developed into being created specifically for the domain that it is being used for. This is due to the large difference in textual content and form between domains.

A tweet can typically be very informal, filled with sarcasm and slang, and because of the low character limit, be straight to the point. Hence, for Twitter, researchers have argued that short and simple lists can be as effective as long and complicated ones (McGurk et al., 2020).

The dictionary we created in this thesis was created from positive and negative unigram coefficients from manually labelled tweets. Unigrams are single words from a sequence of words. All words in the tweet are divided into several unigrams through different vecotrisation methods described above. When a classification model is fitted on these vectors, unigram coefficients are outputted, and range from positive to negative. The positive and negative coefficients are associated with positive and negative sentiment respectively.

We hand-picked some unique unigrams with both very high positive value and very high negative value, that we believe clearly express sentiment. These words were combined with the dictionary created for the same purpose in McGurk et al. (2020). This reduces the chance that the dictionary was not biased towards our subjective labelling techniques, and that the use remained specific to financial tweets. The list of words is available in Table 3.6.

Formally, the positive word list (PWL) and negative word list (NWL) can be presented as in equations 3.8 and 3.9 respectively.

$$PWL = (PWL_1, PWL_2, ...) = (buy, bought, bull, ...) \tag{3.8}$$

$$NWL = (NWL_1, NWL_2, ...) = (pathetic, downer, sold, ...) \tag{3.9}$$

Using these lists, we can calculate the total positive sentiment for each tweet x_n as the total number of positive words that are contained in x_n. Each positive word adds 1 to Ps_n:

$$Ps_n = \#(x_n \cap PWL) \tag{3.10}$$

Positive words		Negative words		
buy	buying	pathetic	sell	selling
bought	long	downer	gare	gross
bull	bullish	sold	short	bear
green	play	imperfect	inferior	junky
excellent	exceptional	bearish	bad	atrocious
favourable	great	abominable	amiss	crappy
positive	awesome	awful	crummy	dreadful
call	like	cruddy	dissatisfactory	erroneous
rocket	watchlist	lousy	poor	rough
hold	volume	fallacious	faulty	godawful
increasing	love	sad	unacceptable	blah
growth	dividend	inadequate	substandard	unsatisfactory
profit	break	inflated	dumping	drained
		bummer	restricted	tanking
		shitty	disappointed	drag
		tainted	downward	

Table 3.6: List of Positive and Negative Words

Similar to negative sentiment, we take the number of negative words that intersect with the tweet. Each negative word adds -1 to Nsn:

$$Ns_n = \#(x_n \cap NWL) \tag{3.11}$$

To get the overall sentiment, we simply take the sum of positive and negative sentiment:

$$s_n = Ps_n + Ns_n \tag{3.12}$$

If there are more positive words than negative the sum becomes positive and vice versa. If a tweet contains one positive word, and four negative, the tweet will receive a score of -1.

3.4.6 Additional Sentiment Features

Previous studies have included analysis on the number of followers into their analysis to increase accuracy of their sentiment indicators. Followers is a measure of a person's influence in the social media community. It may provide insight to market sentiment to determine the sentiment of influential people.

To include the effect from more influential twitter users, we used the sentiment calculated for each tweet, and multiplied each line with the number of followers that each user had. Hence, if a tweet contained negative sentiment, and the author had a large following, a large negative score would be attributed to that day. We also created a similar feature for the number of likes for each tweet.

This means that after the sentiment analysis, we ended up with 3 features for each approach. This includes sentiment, aggregated on a daily basis, and the two additional features mentioned in the previous two paragraphs.

3.5 Return Prediction

The return prediction part of the thesis is focused on return as a target variable, and sentiment, and other , that may or may not influence changes to it. To check if the features we have created have any value in a machine learning context, we needed to establish a baseline model to compare against.

The following subsection will outline 20 financial features that are used as input in the baseline model for stock return prediction. These are highlighted in the literature for having predictability on future stock returns. We wanted to investigate if the sentiment indicators that we created, using PCM, the manual labelling approach and the dictionary approach will improve predictions when comparing to the prediction power of the baseline features only.

We created four datasets, one to test each approach. One dataset only con-

sisted of the 20 baseline features, while the three others had an additional three variables, $sentiment$, $followers \cdot sentiment$ and $likes \cdot sentiment$.

This will indicate the additional value of including sentiment features on prediction accuracy, relative to only using traditional financial features.

3.5.1 Baseline Features

The baseline stock prediction features was developed with inspiration from the paper written by Gu et al. (2020). The dataset consists of features from three categories, momentum, liquidity, and volatility. As mentioned earlier, these have been chosen because previous research has identified these as relevant variables when forecasting future stock returns[6].

Momentum can describe a collection of features that all refer to the velocity of the price movements of a stock.

Within this category, we included annual and monthly stock momentum, and rate of change (RoC). In addition to the 12 day RoC, recent maximum return and industry RoC on an monthly and annual basis. We also included simple annual and monthly moving averages.

Stock momentum is calculated simply by taking closing price differences for a fixed time interval, in our case one month and one year. The rate of change is similar, only one takes the current closing price divided by the previous closing price, on different time intervals. We perform rate of change calculations for both industry prices and individual stocks.

The annual rate of change and stock momentum features exclude the last month returns because of reversal effects. This means that stocks that have had high or low momentum in the last month have an tendency to reverse back to

[6]See Ang, Hodrick, Xing and Zhang (2006); Asness, Moskowitz and Pedersen (2013); Baker and Stein (2004); Brennan and Subrahmanyam (1996); Hurst, Ooi and Pedersen (2017); Lochstoer and Muir (2022) for more information about how and where these variables are used

their mean (Asness et al., 2013; Hurst et al., 2017)

Liquidity features consist of volume, turnover, turnover volatility and dollar volume. These features describe the efficiency in which a security can be sold or bought without affecting its market price. The dollar volume is the closing price multiplied with the volume.

Turnover is calculated by dividing the volume of shares traded during a day by the total number of outstanding shares that day, for each individual company. While turnover volatility measures the variation in turnover over a 3-month period. These two features are an indication of how many stocks are being purchased, relative to how many are available for purchase (Brennan & Subrahmanyam, 1996)

Lastly, volatility features include annual and monthly return volatility, daily VIX and the three-year stock beta. Volatility is calculated by taking the standard deviation of daily returns, multiplying with the square root of the horizon (21 trading days for monthly and 250 for annual). The VIX index price can be downloaded directly from Yahoo Finance.

A stocks beta is a measure of a stocks volatility relative to the market volatility[7], over various time horizons (from 12 months to 8 years). Research have indicated that betas calculated over shorter time horizons are better suited to tackle dynamic changes to the volatility of a stock price, that can occur in a fast phased business environment (Daves, Ehrhardt & Kunkel, 2000). We therefore chose a three year horizon. We calculate the market volatility by downloading the S&P 500 index price and measuring the volatility over the same horizon.

Previous percentage change and closing price was also included in the model.

In Table 3.7 , we see an extraction of the dataset used for training and testing. The setup is a traditional machine learning setup with a date index and features as the columns. The ticker column contains the different tickers used in the analysis.

[7]Harvard Business Review

The tickers are used to split the dataset into several sections when inputting features and, later, performing machine learning on the features.

date	ticker	close	volume	annualMom	indAnnualRoC
29.12.2021	XOM	60.48	12733600	22.07	0.48
29.12.2021	DISCA	23.87	4990800	-6.22	0.22
29.12.2021	GPS	17.35	4992100	-2.17	0.24
29.12.2021	HST	17.45	3826900	2.85	0.42
29.12.2021	UHS	130.82	256600	-5.71	0.24
29.12.2021	LYV	119.83	1457300	46.35	0.22
29.12.2021	F	20.45	37883000	11.75	0.24
29.12.2021	WU	17.63	3341100	-3.1	0.35

Table 3.7: ML Data Structure

3.5.2 Stock Return Prediction Models and Evaluation

Machine learning enables the use of a wide list of explanatory variables and specifications of functional form relative to traditional empirical methods of asset pricing. The three methods we use for comparison are a random forest regressor, gradient boosting regressor, and an artificial neural network - long short-term-memory (LSTM) model. These models were chosen because previous research identify them as good choices for stock market data and prediction (Gu et al., 2020).

3.5.2.1 Tree Structured Models

At the basic level, trees create a segmentation of the data, finding and grouping observations that behave similarly. The models predict a value by finding a corresponding segment, calculating the average value at that "leaf".

The process moves in a sequence of steps, starting at the root, through nonleaf nodes. These are referred to "decision nodes" because they make a decision of which branch to follow dependent on the value or category of the features used for training or testing. These decisions slice the n-dimensional space (n represents

number of features) into rectangular partitions. When the process reaches the leaf, the prediction becomes the average value within this space of observations (Gu et al., 2020; Provost & Fawcett, 2013).

In our analysis, we consider two “ensemble” tree regularizers that combine forecasts from many different trees into a single forecast.

3.5.2.1.1 Random Forest

The baseline random forest algorithm is a variation on a more general procedure called bagging (Breiman, 2001). Bagging takes several samples, with replacement, from the data, fitting a separate regression tree on each sample, and averaging their forecasts. As simple regression trees are very flexible, making them very prone to overfitting, bagging reduces variability in the forecasts and therefore tend to stabilise the trees predictive power (Gu et al., 2020).

One of the issues with this procedure is that the trees tend to be very similar when there are dominant features in the data. For example, if stock momentum reduces the measure of error the most, all the bagged trees have the same splits on momentum, resulting in highly correlated predictions. Random forests reduces correlation between the samples with a method called "dropout" (Breiman, 2001).

This method only chooses a subset of features for splits at each potential branch. This alteration causes all bagged trees to have different randomly selected features when going through the decisions in the tree. Therefore, the trees and predictions will be less correlated. Section 6.3.1 in the appendix contains information on the parameters we used.

3.5.2.1.2 Gradient Boosted Regression Tree

These models tries to add combinations of predictions from many different over-simplified trees, also known as "weak learners". A weak learner is a model that only barely performs above chance, where the typical for tree based models is the

one-level decision tree.

The baseline gradient boosted regression tree model initially fits on a weak learner, for example a one one-level decision tree. The fit is sure to have large bias and be a weak predictor. The algorithm subsequently fits another weak learner on the prediction residuals from the first learner. The two different forecasts are added together to an ensemble prediction of the outcome variable, where again the residuals are calculated. The theory is that many weak learners ensemble into a "strong learner" with more stability than a single complex decision tree (Gu et al., 2020).

One may tune the model by shrinking the residual forecast of each, previous step weak learner, by a factor between 0 and 1. The iteration of the number of learners can also be restricted, in addition to the depth of the weak learners used. Section 6.3.2 in the appendix contains information on the parameters we used.

3.5.2.2 Neural Network - Long Short-Term-Memory

Artificial neural network (ANN) models are in some ways similar to the two ensemble models outlined above. It is possible to think of ANN's as a stack of different models. The bottom layer of the stack are the input data, and the following layers are simple models that train, parts on output of the preceding layer and parts on new data.

The stack of models can be represented by a numeric function that contain the parameters of all the models. The neural network algorithm optimises an objective function, in our case, minimising mean squared error. This will identify the optimal parameters for all layers and how to combine them, simultaneously (Provost & Fawcett, 2013).

Recurrent neural network (RNN) is a class of ANN where connections between nodes form a graph along a temporal sequence. Normal RNN's are bothered by an issue of placing to little importance on new input if used on a large dataset

(Jiang, 2021).

The Long Short-Term-Memory (LSTM) model is a special case of RNN. The model was developed as a solution to the problem with low weights on new input data if the network unfolded too many times. A notable problem if a model is trained over longer periods time where large variations in data may occur.

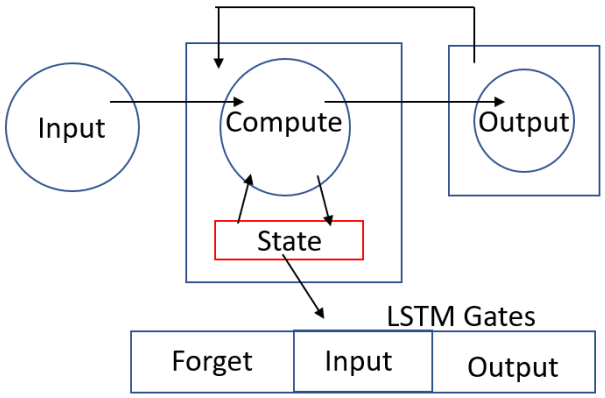

Figure 3.7: General Illustration of LSTM

Figure 3.7 illustrates the LSTM model process. "Input" describes the new data entering model, "compute" is where the computations of next prediction is done. "Output" is the prediction of the next period.

The figure illustrates that the model uses input data, in combination with predictions from previous periods, as input for next period predictions.

The "state" refers to the different LSTM "gates" that one can see in the lower part of the figure. These add the ability to selectively memorise important sequences of data and forgetting those that are not useful for predictions, allowing the model to adapt to structural changes in the input data (Moghar & Hamiche, 2020).

Section 6.3.3 in the appendix contains information on the parameters we used in the LSTM model.

3.5.2.3 Regression Accuracy Metrics

3.5.2.3.1 Measuring magnitude of forecast errors

Mean squared error (MSE) is a measure of forecast accuracy. It does not say anything about the direction of errors due to taking the square of the sum, but indicates how the model performs in terms of magnitude of errors. The measure penalises high forecast errors because each residual is squared. It is common to take the root MSE because this transforms the measure to be the same unit as the target variable.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2} \tag{3.13}$$

3.5.2.3.2 Measuring bias in forecast

Mean absolute error (MAE), measures the average error of the model. The will tell us the magnitude of the average prediction error in absolute value. This will make it easy to understand how much bias the model on average predicts in the target value's units.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|(Y_i - \hat{Y}_i)| \tag{3.14}$$

3.5.2.4 Time-series Cross-Validation Process

The principle of time series cross-validation is that we cannot use values from the future to forecast the past values.

The initial step is to select the training and testing periods. We perform an

80/20 split on the data, which runs for a total of 977 days, resulting in 782 training days and 195 testing days.

There is a trade-off in choosing the data length for financial data, in addition to the short data availability of Twitter data. For stock data, a short time-period runs the risk of being overfitted, while a long time period risks traversing different market trends and present out-of-dated results. As a result, many studies look at historic data between 5-10+ years, for daily stock returns (Jiang, 2021).

We only collected Twitter data from the beginning of 2018, meaning if we were to compare the baseline model to the sentiment models, we would need to keep the training period to the start of 2018 to compare on common ground. Hence, the training period was kept static at the beginning of 2018, to capture the full data length.

Figure 3.6, displays how we set up the training and testing datasets. This is referred to as successive training sets, where each loop increases the training set to include one more row of observations, testing on the next day. The training period starts at 80% and increases by one day after each iteration. The testing period is the last 20% of the data, where the model predicts one day at a time.

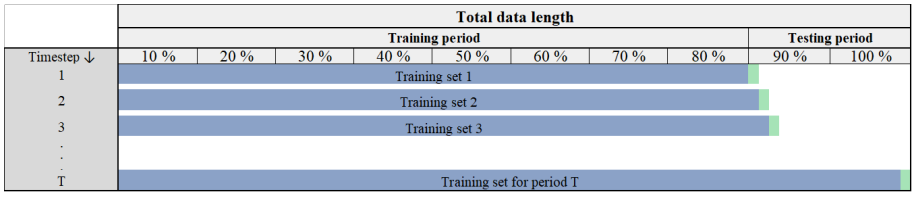

Figure 3.8: Time Series Successive Training Illustration

Sizes in the figure only for illustrating time selection. The actual model set up is described in the bullet points below:

- Observations from the training set occur "in time" before their corresponding test set, respecting the dependency of time.

    1. Initially, the model is fitted on 80% of the data.

2. The model predicts return (y variable) for the next day based on out of sample X variables, this includes the same day return.
3. The model performs a one day forecast.
4. The predicted return and the actual return are saved.
5. The loop increases the training period with one day, the model trains again, now including 80% of the data + 1 day.
6. When the prediction for each observation has been saved, evaluation metrics RMSE, MAE, and residuals are calculated and saved for that company.
7. The loop starts training and predicting on another company's data. Going back to step 1. until predictions for all companies in the sample have been saved.

- When predictions for all companies have been saved, calculate RMSE and MAE metrics to get final results.

3.6 Summary

This analysis involved a number of steps before coming to a final result. The method does not utilise any new individual types of analysis, but rather combines several different existing research methods on the area of sentiment analysis and machine learning stock market prediction. This combination is a novel approach to stock market prediction with sentiment analysis. Furthermore, including the separation of companies with high and low levels of institutional ownership, introduces another area of exploration within sentiment analysis and its effect on stock prices.

Chapter 4

Findings

This section summarises our results on the relationship between Twitter sentiment and financial features on stock return prediction accuracy. The method is divided into two main parts, the first being the sentiment analysis, vectorisation and machine learning classification. The second is the stock return prediction, which utilises machine learning regression. The results from the classification is displayed with the accuracy score, calculated as in equation 3.7, while regression accuracy is displayed through RMSE (equation 3.13) and MAE (equation 3.14).

We first outline the classification results, first for the different vectorisation methods and then the supervised classification models. We go on to show the correlation between different sentiment indicators to each other, and to stock return. Then, we determine the effect of institutional ownership and sentiment features, and lastly investigate the stock return prediction model and its accuracy.

4.1 Text Vectorisation and Classification

There were two methods used, Bag-of-words (BoW) and TF-IDF. The main purpose of these methods is transform textual data into vectors and get them ready for the classification models. To show the effect of the two methods, we train each

classification model two times, and display the results as an average. This section also displays the results from the classification models themselves, and compares the two different sentiment methods that utilised these steps in their analysis.

Table 4.1 & Table 4.2 shows that TF-IDF received overall better results.

	Precovid	Bull period	Bear period
TF-IDF mean accuracy	0.67	0.55	0.7
Bag-of-words mean accuracy	0.67	0.56	0.67

Table 4.1: Vectorisation Results of PCM Labelled Tweets

	Mean accuracy
TF-IDF	0.8
Bag-of-Words	0.76

Table 4.2: Vectorisation Results of Manually Labelled Tweets

In Table 4.1 you can see a summary of the results for PCM for the three periods. It looks from this table that the difference between TF-IDF and BoW are not that momentous, however, the difference between the two methods grows when using the best classification model, as one can see from Table 4.3. Here you can see that the accuracy of the SVM classifier increases from 66% to 69% percent on average by using TF-IDF.

Similarly, the numbers in Table 4.2 shows that TF-IDF performs better than BoW with manually labelled tweets. The difference increases by 4%, which is a large difference in a classification context.

	TF-IDF	Bag-of-Words	Mean
SVM	0.69	0.66	0.67
Naïve Bayes	0.62	0.62	0.62
Logistic Regression	0.62	0.61	0.61

Table 4.3: PCM Classification Accuracy

This is also apparent from Table 4.4, showing the results from the supervised models trained on manually labelled text. TF-IDF increases the accuracy of both

Naïve Bayes and SVM by several percentage points.

	TF-IDF	Bag-of-Words	Mean
SVM	0.83	0.77	0.80
Naïve Bayes	0.80	0.75	0.78
Logistic Regression	0.76	0.76	0.76

Table 4.4: Manual Labelling Classification Accuracy

Adding to this, we noticed that with PCM, increasing the size of the dataset will reduce the accuracy of the model. The accuracy decreases to approximately 50% when increasing the training period. Improvements were made by splitting the data into several market periods.

Lastly, we can see in Table 4.3 and 4.4, that SVM is the best performing classification method. In combination with TF-IDL, we acheive the all over best results. For this combination, the accuracy increases from 62% to 69% using PCM, and from 76% to 83% with the manual approach. Hence, we continued to use this combination when creating the final sentiment features.

4.2 Sentiment Indicators

Table 4.5 contains the correlation between return and the sentiment indicators, and between the different approaches. The correlation between return (percentage change) and the different indicators are close to zero. This does not automatically discard the features in their ability to predict return, as non-linear models may still capture more complex relationships between the feature and the target variable.

	Percentage change	PCM	Dictionary	Manual
Percentage change	1.0000			
PCM	0.0530	1.0000		
Dictionary	0.0095	0.1630	1.0000	
Manual	0.0300	0.2000	0.3610	1.0000

Table 4.5: Correlation between Sentiment Indicators and Return

The correlation between the sentiment indicators are low (from 0.163 to 0.361) demonstrating that the different methods used to derive numerical features from tweets content can widely differ in results. However, the correlation between the dictionary and manual approaches indicates that there is some consistency between these two, likely because the dictionary was influenced by words extracted from the model trained on manually labelled tweets.

4.3 Return Prediction and Sentiment Influence

Three different sentiment indicators, with followers and likes, are aggregated on a daily frequency and combined with the baseline dataset, to a total of four datasets, on which we trained three machine learning models. The results from these models are shown in Table 4.6, which contains mean absolute error (MAE) and root mean squared error (RMSE) for all the different machine learning models and datasets used. Each metric is calculated from 194 one day predictions compared to actual values, for 20 different companies. Each metric includes both high and low levels of institutional ownership companies.

	Neural Network		Random Forest		Gradient Boosting	
	MAE	RMSE	MAE	RMSE	MAE	RMSE
Baseline	0.0155	0.0212	0.0155	0.0209	0.0157	0.0212
PCM	0.0163	0.0223	0.0155	0.0209	0.0156	0.0212
Dictionary approach	0.0157	0.0218	0.0156	0.0211	0.0157	0.0214
Manual approach	0.0157	0.0218	0.0155	0.021	0.0159	0.0218
Mean	0.0158	0.0218	0.0155	0.021	0.0157	0.0214

Table 4.6: Machine Learning Accuracy, Baseline vs. Sentiment Models

The key finding is that sentiment features do not improve predictions. In fact, in a majority of results, predictions are further away from actual values, resulting in higher MAE and RMSE.

Secondly, we can see that the performance of the models are very similar.

With the random forest outperforming the neural network and gradient boosting models slightly. There may be qualities of the data that influences the accuracy of the neural network, like data length. As, these models typically thrive on large datasets. We did not collect tweet data older than 2018, likely resulting in a favourable environment for simpler models.

For the rest of the results, we only display the random forest model results, as this model performed marginally better.

4.4 Institutional Ownership

Companies with high and low fractions institutional ownership were divided into two separate datasets, where predictions and evaluation metrics were saved separately. Because MAE and RMSE can vary significantly between companies, we estimated the influence of institutional ownership on sentiment indicators by calculating the percentage decrease/increase in MAE compared to the baseline. A positive percentage represents a decrease in the MAE, which is an improvement, as you can see in Table 4.7 and 4.8.

It is difficult to see any clear trends from these percentages. None of the sentiment indicators show improvements across all companies, which corresponds with the previous results. Furthermore, there seems to be no clear difference between high and low institutional ownership. One can see that the mean improvement in MAE is larger in absolute value for companies with high institutional ownership, which is contrary to our hypothesis, however, this is too small to be certain of a clear trend.

There are certain companies that benefit a great deal from certain indicators, like CDAY in Table 4.7, where MAE is improved by 50% only by including manual labelling features. On the other hand, predictions for BF.B reduce in accuracy from all sentiment features included. Because there are so few clear trends, choosing to include a specific sentiment indicator for a selected number of

companies with a positive percentage would quickly lead to an overfitted model. With tweets, we only have a small time period to test the validity of the indicators for that stock, and it may be a temporary positive effect. Hence, over time, one could further test if these measures improve predictions for these companies and implement them to better improve investment decisions.

Ticker	Improvement PCM (%)	Improvement Dictionary (%)	Improvement Manual (%)
UHS	2.19	0.56	3.28
GPS	2.60	-0.56	0.41
LYV	1.96	-0.90	2.86
NLSN	0.88	-0.42	-1.22
HST	-0.55	-1.54	1.10
DISCA	-1.21	-0.79	0.28
REG	0.28	-3.72	-0.48
UDR	-0.97	-0.07	-1.43
CDAY	-1.45	-2.25	49.73
WU	3.08	0.40	2.68
Mean	0.68	-0.93	5.72

Table 4.7: High Institutional Ownership - MAE

Ticker	Improvement PCM (%)	Improvement Dictionary (%)	Improvement Manual (%)
F	-1.21	1.82	-2.82
O	1.29	-1.6	-0.87
BA	1.0	0.28	2.49
AMCR	2.76	0.83	1.37
NCLH	1.16	-4.55	1.39
BF.B	-6.94	-0.24	-6.16
XOM	0.28	1.11	0.98
AAL	-0.01	0.78	-6.07
CTRA	-0.24	0.83	-0.97
PRU	-0.09	2.64	-0.31
Mean	-0.2	0.19	-1.1

Table 4.8: Low Institutional Ownership - MAE

4.5 Return Prediction Permutation Importance

Permutation importance is a model inspection technique that can calculate the importance of estimators for a give tabular data set. The method adds noise to one feature, which randomly breaks up the relationship between the features and target. From the breaking up process, a error score is calculated as an indication of the features importance[1].

Increase in error-score from shuffling the column, indicates that the model is depending on the feature. Little change in error score and the column is likely not important.

The baseline model uses 20 input variables when predicting the next day's return. When including sentiment variables, this increases to 23. In Figure 4.1 you can see the average feature importance for the 8 different random forest models that constitute the above results. Random forest was chosen for this illustration because it achieved the best RMSE and MAE.

We can see that the features with either high or low correlation with return have high importance for the model predictions. The 12-day rate of change, previous return, and the VIX index all display the highest model importance. Other features that are important include industry and monthly momentum and turnover volatility.

[1]Scikit-learn documentation

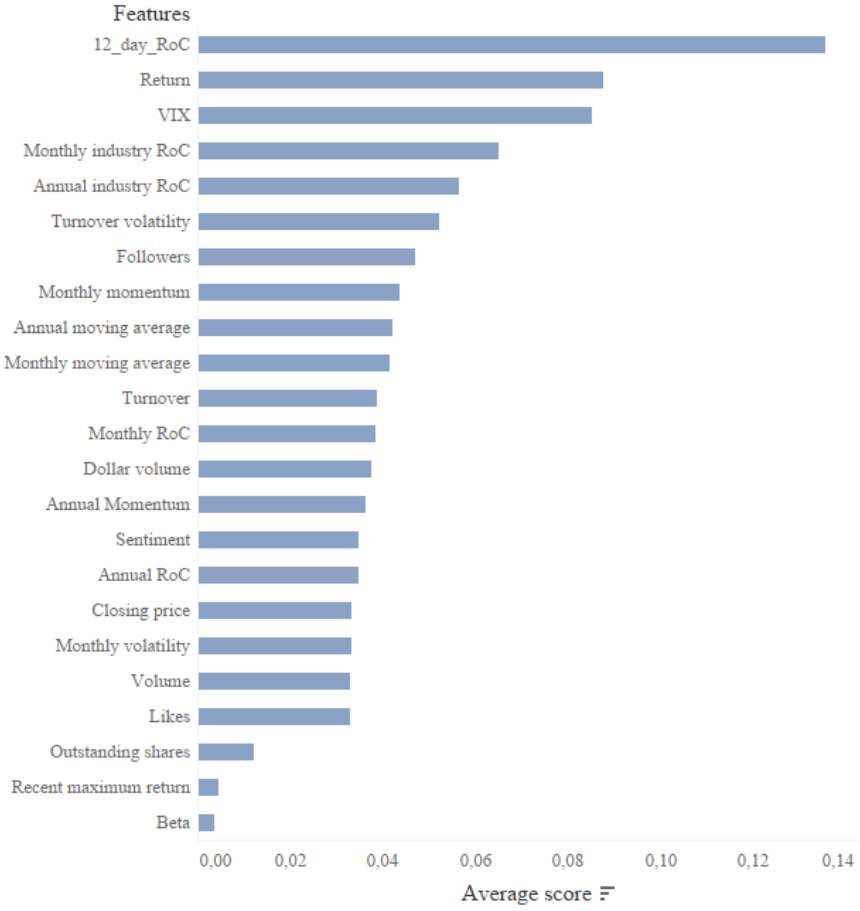

Figure 4.1: Random Forest Model Importance

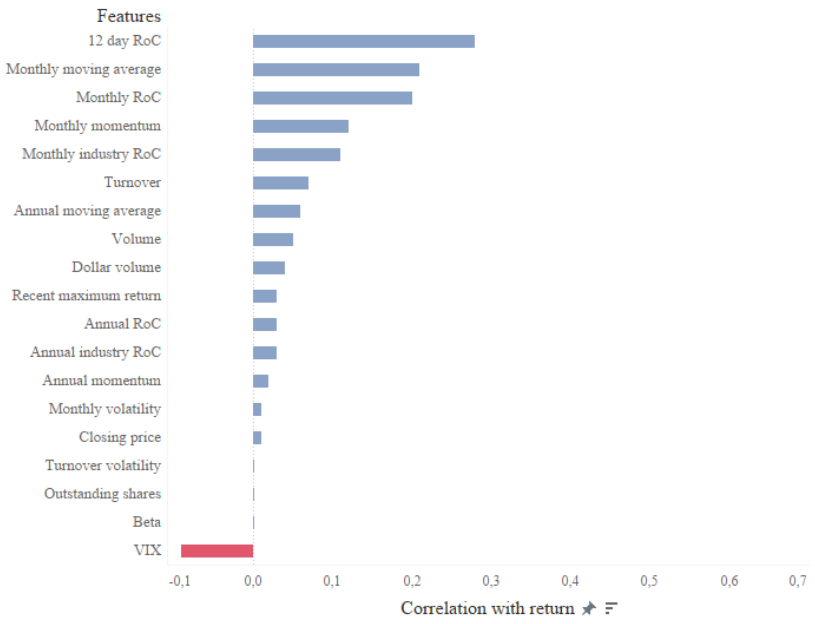

Figure 4.2: Feature Correlation with Return

The indicator built on the number of followers also has some weight, and polarity and likes are not completely insignificant either. However, the models are using these features in their predictions without increasing accuracy. Even if the sentiment indicators are showing importance, it is not showing improvements in the validation, reducing the validity of these features. Further iterations to the model building process would likely exclude these features because it would seem that they are not adding any value to the accuracy.

Other features also show themselves to be redundant, likely because we are predicting single company returns, like for example beta and recent maximum return, and should therefore be removed. A global model where the model is predicting several stock returns might find these features more important because it could easier compare differences between stocks. Momentum features achieve higher importance overall relative to volatility and liquidity variables, but the model seems to depend on a good mix of the three categories for its predictions.

It is difficult to determine if the model is useful from comparing MAE and RMSE. These vary significantly depending on the period and company in question.

Hence, to evaluate the model usefulness we calculated the mean actual return for different bins and compared this to the mean predictions on this range of values.

This resulted in Figure 4.3, which displays the return on the y-axis, and bins created on the distribution of returns we observed on the x-axis. Within each bin, we take the mean of actual returns, and compare that to the mean predictions for actual values in that bin. It should be noted that most values lie between -0.01 and 0.01.

The figure displays an inability of the model to predict returns accurately, that it is difficult to see any similar trend in predictions as in real values. When return becomes extremely negative or positive, predictions also tend to be further away from zero. This is likely because the model predicts high or low values when it sees spikes in volatility or liquidity. This means that it will be infeasible to interpret predictions that deviate from zero.

Figure 4.3: Mean Predictions vs. Mean Actual Returns

Another aspect to model evaluation is looking how the wider financial market and economy influenced the model performance. In Figure 4.4, one can see the average model prediction error over the period. The US financial markets have been experiencing higher volatility since 2018, but especially since the corona pandemic. One can see that the model performs worse in the end of 2021 and beginning of 2022, likely because of a trend change in the markets at that time,

followed by increased volatility from the Russian invasion of Ukraine.

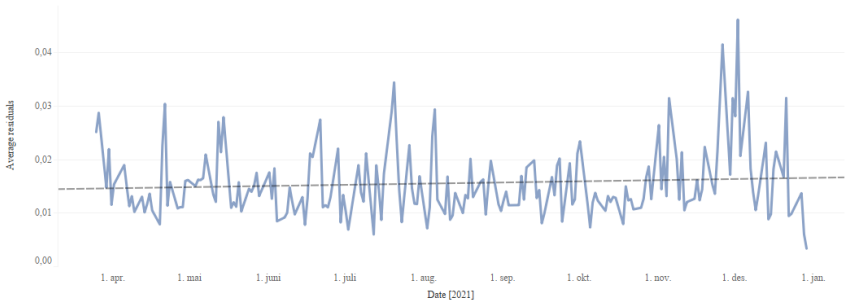

Figure 4.4: Random Forest Mean Residuals

Chapter 5

Discussion

Chapter 4 outlined the results from the sentiment methods, the relationship between sentiment features and the stock return prediction results. This section provides discussion and interpretation of the results. We discuss and evaluate the sentiment methods used, and why classification accuracies differed. We also examine return prediction accuracy, limitations of the method, data and potential future study.

5.1 Text Vectorisation and Classification

There are three aspects of the sentiment analysis part of the thesis. The sentiment methods, the vectorisation models and the classification models. Two sentiment approaches, PCM and the manual, use vectorisation and classification. The vectorisation methods display their impact on accuracy when the classifcation models output their results. The classification models themselves are of interest because they have an influence on the accuracy of predictions on out of sample text.

We start at the end, where we found that the Support Vector Machine (SVM) achieved the best results out of the three classification models. We will therefore refer to this model when discussing the results of the vectorisation and the different

sentiment approaches in the next few sections.

SVM's trained with the use of stochastic gradient descent (SGD) algorithm is in recent years reported to outperform most other classifying methods (Pimpalkar & Raj, 2020; Pujari, Aiswarya & Shetty, 2018). We found similar results, a clear advantage its the models ability to handle the dimensional characteristics of natural text.

We found that Term Frequency - Inverse Document Frequency (TF-IDF) is the most accurate vectorisation method to predict which sentiment a tweet should have. The Support Vector Machine (SVM) classification accuracy increased by 3% with PCM and 6% with the manual approach, only from switching from Bag-of-Words (BoW) to TF-IDF.

If a word is rare or common across all tweets it will have a low TF-IDF score. Hence, as TF-IDF was more effective than BoW, it means that frequently used words in all tweets are as not indicative of sentiment as those with high TF-IDF scores. High TF-IDF score words could be words that are used several times in one specific tweet, with those words rarely used in other tweets. Additionally, a tweet with mostly common words across all tweets generates a low TF-IDF score.

According to Loughran and Mcdonald (2016), it is likely that because there are so many different users, resulting in a diverse language, it causes BoW to struggle. Alternatively, it could mean that the data is not sufficiently pre-processed to remove common unimportant phrases. TF-IDF would give low scores to words that are common across all documents, reducing this kind of noise.

The text prediction results from Table 4.2 and 4.3, showed differences between the accuracy of PCM and the manual approach. Meaning, the SVM classifier was able to notice more distinct patterns in the text labelled manually than that labelled automatically from price changes.

This was expected, because PCM labels all tweets positive or negative regardless of whether or not they are actually containing positive or negative sentiment.

However, we find that the accuracy of PCM increases when splitting the training testing period up into several parts, as you can see in Figure 3.5. When increasing the horizon, not splitting the data, the accuracy decreases to around 50% or lower.

This indicates that there are troubles generalising the model for multiple periods. The reason may be social media language, vocabulary and content, changes significantly over longer time periods. The model fails to establish clear patterns of consistent text to produce accurate predictions.

Still, PCM is less consistent overall in its predictions even after splitting the time period into three parts, likely because it is more influenced by spam, and a mix of positive and negative sentiment. In a worst-case scenario, the PCM amplifies the "wrong signal" from bots retweeting other posts.

Hence, the ability to mark non-relevant messages with the score of zero (neither positive or negative), or remove spam tweets, improves the manual over the PCM. While this process is highly time consuming and may introduce subjective opinions into the process, it does lead to the most consistent model of the two methods, with an accuracy of 83%.

The dictionary approach benefits from its simple set up and automatic execution. However, the dictionary benefits from going through the content that you are studying manually, which can make the method dependent on some manual labour. This was the reason why the manual approach had higher correlation with the dictionary approach, as seen in Table 4.5. Still, after a dictionary is created, other researchers can easily utilise it for similar purposes and add or remove words easily. The dictionary approach automatically labels tweets without any evaluation hence, one has to measure the usefulness of this approach from the improvement on machine learning return accuracy.

5.2 Sentiment Indicators and Prediction Accuracy

Predictions do not improve by including the created sentiment indicators, in fact, the accuracy seems to decrease across all datasets compared to baseline. The models' predictions are influenced by these features, which could be seen in the feature importance plot, Figure 4.1. This contributes to decreasing the accuracy, likely because the sentiment indicators are not containing information that improves out of sample predictions. This indicates that progress can be made by removing these features from the model, or alternatively, changing the model parameters like prediction horizon.

While some of the sentiment features, for specific companies do show an improvement on predictions, this needs to be further tested and validated to see if these findings are consistent to limit over fitting. We believe that because the different sentiment indicators are showing such erratic improvements on different stocks, it is less likely that they will have a generalise and consistent effect in the future.

We also find that differences in low and high institutional ownership has a low impact on the effectiveness of sentiment indicators. This is surprising because we expected that companies with a higher level of retail investor owners would be more influenced by the aggregated social media opinions of the stock. There are also generally more tweets posted about stocks with low levels of institutional ownership, indicating that the sentiment would reflect a larger part of investors in that stock. However, this finding alone does not discount differences in institutional ownership and the effect of retail investor sentiment, it may simply mean that our sentiment indicators were not reflecting the true retail sentiment.

There may be several reasons why there was no significant improvement from the created sentiment indicators on stock return prediction accuracy. Below, we outline the four main reasons why the created sentiment features were not containing useful information, in addition to traditional financial features.

The first is that the size of the dataset was not large enough or containing too few tweets per company to accurately reflect market sentiment. One way to mitigate this could be to focus on companies with a larger number of tweets. Our data was highly skewed, many companies had a low number of tweets, which should mean that the aggregate sentiment is a non-sufficient proxy for the total retail investor sentiment for that particular stock. We also found no relationship between number of tweets and improvement of predictions for those stocks. Still, the most discussed tickers in our sample have a low number of tweets compared to more popular tickers, which may be better proxies.

The second reason is that it is difficult to automatically, as well as manually, extract accurate financial sentiment from tweets. Even after filtering out all tweets without cashtags, only possible with Twitter API academic access, a large portion of tweets contains unconnected information to the search. It is clear that developing accurate sentiment indicators are very dependent on data cleaning, processing and spam reduction methods. Unrelated noise will negatively influence the accuracy of these labels. Hence, it is essentially about developing methods to reduce this type of content to the minimum. Our pitfall may have been that we found no established and automatic methods to reduce spam. or importantly, filter out tweets that are expressing sentiment for a ticker that we are not interested in.

The third could be that, as previous research have pointed out, extracting value from the positive and negative effects from Twitter sentiment have come from looking at special cases, like peak Twitter volumes and followers. Though we included a feature on the number of followers, in addition to the sentiment features indirectly including Twitter volumes in the aggregation. The thesis was not focused on developing these parts of the models and data to any great length. However, special cases should be a significant part of any sentiment analysis project looking at social media data.

Lastly, the model parameters were not set up in the most optimal manner for the features used. There are a number of ways to set up the models to take advantage of the features we created. We chose to predict the next day returns,

which is common in financial literature. For example, the effect of overwhelming positive sentiment may have longer term effects, which will not be reflected in the accuracy metrics.

How information is spreads and what determines the popularity of topics is important to understand when researching the influence of media on the behavior of individuals. If the sentiment reflects what is the users true opinions, the sentiment could be a good indicator for future stock price changes. If the information "shared" mostly is based on what algorithms promotes, retail investors should have less influence on the sentiment and price changes.

5.3 Stock Return Prediction Models

We chose a neural network model and two different three based machine learning models to predict stocks in this paper. The accuracies are very comparable, and none greatly outperform the others. However, the relatively simple random forest model outperforms the others slightly. This makes the choice to utilise this model for other analysis simple, as one clear con with the neural network model is that it takes a number of times longer to run, in addition to not increasing accuracy. One reason can be that a simple model deals with a short training period better, while the neural network needs more data to achieve its full potential.

Prediction of stock movements is extremely challenging because of its erratic, dynamic and non-stationary nature. Generally, when evaluating the model's performance, one needs to look at the training and testing periods chosen. Historically, the stock market in the US has had clearly defined periods where stock indexes have generally gone upward, downward or sideways. A period with high volatility will increase RMSE and MAE significantly.

This can indicate that the models will perform better if it is trained on periods that is similar to the one that it is predicting. We can see how this plays out in the residual plot that we presented in the findings, Figure 4.4. The models are

not understand what returns are going to be in the end of 2021/beginning of 2022 because of high volatility, and a change in trend at this time. Hence, one mitigation could be to train models on recent data and subsequently retire the model if volatility increases significantly.

5.4 Future Work and Limitations

Methods for increasing the quality of data are in high demand, with the need of several problems to be handled in social media data. Tweet content unfortunately contains a lot of spam and nonsensical texts, and information that is not relevant for determining sentiment for the relevant stock. A brief summary on problems needed to be handle:

- Bot's created for retweeting spam tweets or creating tweets them self, Nonsensical texts are usually a sign of this.
- Wrong and confusing "tags" of posts not providing relevant information.

Providing researchers with more details and effect of using different data collection and pre-processing methods would provide time saving and possibly improve the quality of future research. Information on the volume of data is needed to increase the probability that the aggregated social media sentiment reflects the true retail investor market sentiment.

Over time, it will be possible to increase the training and validation periods for these types of features, such that one can identify stock specific trends and take advantage of these. The continued development and popularity of social media platforms should motivate continued study of the platforms.

Section 5.2 went into detail about what we view as the main limitations of the sentiment analysis performed in this thesis.

Other limitations include not using log returns for predictions, as they ensure an approximately stationary time series.

We should also have used random states for all machine learning models, as this ensures replicability. Lastly, we should have cross-validated model parameters more thoroughly before choosing. This ensures that the models are parameterised for the data and predictions horizon chosen.

Chapter 6

Conclusion

Behavioral financial theory puts forward that market sentiment can explain longer term stock mispricing. Several studies supports the hypothesis that stock prices are not completely efficient, which implies that sentiment contains information that could be useful in machine learning.

We tested predictive power by comparing our sentiment indicators to financial variables typically viewed as important for stock prediction. Overall, we find no evidence that our sentiment features improve machine learning predictions above traditional financial features. We find it likely that this result is related to data issues, including the volume and quality of the data downloaded and the superficial analysis of special cases like peak volumes and the number of followers.

We further investigate a hypothesis that social media has value primarily by indicating retail investor sentiment, rather than the total market sentiment. Retail investors represent the majority of social media users, but we found no clear relationship between the level of institutional ownership and improvements in predictions. Nevertheless, this might speak to the lack of predictive power from our sentiment variables, rather than ownership being unimportant.

The process from data to return predictions involves a number of steps. The problem with downloading Twitter data is a majority of information online directs

you to time consuming and complicated methods, which only work with specific API access levels. Even after finding easy to use tools, the API limitations and access levels cause Twitter data to loose much value for comparable analysis. This thesis is an example of this problem, because even after using the highest access level, we were not able to find any useful information in the data.

From our sentiment analysis, we found that TF-IDF performs better than BoW. This showed frequent words tell us less about sentiment than those with high TF-IDF scores. This may result from tweets having diverse and unconventional language, or from our inability of filtering out all unimportant common words. We also believe that manual labelling and the dictionary approaches present the most favourable results. The dictionary approach is simple to understand, easy to code and relatively fast. The main requirement is a domain specific dictionary, that can be created from some manual work, or looking at previous research.

The manual approach is the most labour intensive, but in return achieved much higher accuracy than PCM. The main issue with PCM is that there is still negative or neutral sentiment expressed on days with high return, and the other way around, resulting in inconsistent results. We believe that in order to achieve improvements on our results, better automatic methods of identifying spam or irrelevant information is needed. The manual labelling is useful in this manner because you can carefully filter out such tweets, but large amounts of data discourages this approach.

Finally, we find that the SDGC optimiser with support vector machine classifier is the best performing method for text prediction, while random forest regression performs marginally better for stock return prediction. The latter likely performs better because of the lower data availability that comes from working with Twitter data. Future work should consider companies that are discussed more frequently to increase the chance that the aggregation is enough to explain a mispricing in the stock, better representing the true retail investor sentiment.

References

Ang, A., Hodrick, R. J., Xing, Y. & Zhang, X. (2006). The cross-section of volatility and expected returns. *Journal of Finance*, *61*. doi: 10.1111/j.1540-6261.2006.00836.x

Antweiler, W. & Frank, M. Z. (2004). Is all that talk just noise? the information content of internet stock message boards. *Journal of Finance*, *59*. doi: 10.1111/j.1540-6261.2004.00662.x

Asness, C. S., Moskowitz, T. J. & Pedersen, L. H. (2013). Value and momentum everywhere. *Journal of Finance*, *68*. doi: 10.1111/jofi.12021

Baker, M. & Stein, J. C. (2004). Market liquidity as a sentiment indicator. *Journal of Financial Markets*, *7*. doi: 10.1016/j.finmar.2003.11.005

Baker, M. & Wurgler, J. (2007, 4). Investor sentiment in the stock market. *Journal of Economic Perspectives*, *21*, 129-151. Retrieved from `https://pubs.aeaweb.org/doi/10.1257/jep.21.2.129` doi: 10.1257/jep.21.2.129

Bartov, E., Faurel, L. & Mohanram, P. S. (2018). Can twitter help predict firm-level earnings and stock returns? *Accounting Review*, *93*. doi: 10.2308/accr-51865

Bottou, L. (2010, 01). Large-scale machine learning with stochastic gradient descent. *Proc. of COMPSTAT*. doi: 10.1007/978-3-7908-2604-3_16

Breiman, L. (2001). Random forests. *Machine Learning*, *45*. doi: 10.1023/A:1010933404324

Brennan, M. J. & Subrahmanyam, A. (1996). Market microstructure and asset pricing: On the compensation for illiquidity in stock returns. *Journal of*

Financial Economics, *41*. doi: 10.1016/0304-405X(95)00870-K

Chaturvedi, I., Ong, Y. S., Tsang, I. W., Welsch, R. E. & Cambria, E. (2016). Learning word dependencies in text by means of a deep recurrent belief network. *Knowledge-Based Systems*, *108*. doi: 10.1016/j.knosys.2016.07.019

Chen, H., De, P., Hu, Y. & Hwang, B. H. (2014). Wisdom of crowds: The value of stock opinions transmitted through social media. *Review of Financial Studies*, *27*. doi: 10.1093/rfs/hhu001

Cowles, A. (1933). Can stock market forecasters forecast? *Econometrica*, *1*. doi: 10.2307/1907042

Dadakas, D., Karpetis, C., Fassas, A. & Varelas, E. (2016). Sectoral differences in the choice of the time horizon during estimation of the unconditional stock beta. *International Journal of Financial Studies*, *4*. doi: 10.3390/ijfs4040025

Daves, P., Ehrhardt, M. & Kunkel, R. (2000). Estimating systematic risk: the choice of return interval and estimation period. *Journal of Financial and Strategic Decisions*, *13*.

Gentzkow, M., Kelly, B. & Taddy, M. (2019). Text as data. *Journal of Economic Literature*, *57*. doi: 10.1257/jel.20181020

Green, J., Hand, J. R. & Zhang, X. F. (2013). The supraview of return predictive signals. *Review of Accounting Studies*, *18*. doi: 10.1007/s11142-013-9231-1

Gu, S., Kelly, B. & Xiu, D. (2020). Empirical asset pricing via machine learning. *Review of Financial Studies*, *33*. doi: 10.1093/rfs/hhaa009

Harvey, C. R., Liu, Y. & Zhu, H. (2016). ... and the cross-section of expected returns. *Review of Financial Studies*, *29*. doi: 10.1093/rfs/hhv059

Hill, S. & Ready-Campbell, N. (2011). Expert stock picker: The wisdom of (experts in) crowds. *International Journal of Electronic Commerce*, *15*. doi: 10.2753/JEC1086-4415150304

Hurst, B., Ooi, Y. H. & Pedersen, L. H. (2017). A century of evidence on trend-following investing. *Journal of Portfolio Management*, *44*. doi: 10.3905/jpm.2017.44.1.015

Jaggi, M., Mandal, P., Narang, S., Naseem, U. & Khushi, M. (2021). Text mining of stocktwits data for predicting stock prices. *Applied System Innovation*, *4*. doi: 10.3390/asi4010013

Jegadeesh, N. & Wu, D. (2013). Word power: A new approach for content analysis. *Journal of Financial Economics*, *110*. doi: 10.1016/j.jfineco.2013.08.018

Jiang, W. (2021, 12). Applications of deep learning in stock market prediction: Recent progress. *Expert Systems with Applications*, *184*, 115537. Retrieved from `https://linkinghub.elsevier.com/retrieve/pii/S0957417421009441` doi: 10.1016/j.eswa.2021.115537

Jin, Z., Yang, Y. & Liu, Y. (2020). Stock closing price prediction based on sentiment analysis and lstm. *Neural Computing and Applications*, *32*. doi: 10.1007/s00521-019-04504-2

Jurafsky, D. & Martin, J. H. (2021). *Speech and language processing* (3rd ed. draft ed.).

Khan, W., Mustansar, Ghazanfar, A., Muhammad, Azam, A., Karami, A., ... Alfakeeh, A. S. (2020). Stock market prediction using machine learning classifiers and social media, news. *Journal of Ambient Intelligence and Humanized Computing*, *1*, 3. Retrieved from `https://doi.org/10.1007/s12652-020-01839-w` doi: 10.1007/s12652-020-01839-w

Kumar, B. S. & Ravi, V. (2016). A survey of the applications of text mining in financial domain. *Knowledge-Based Systems*, *114*. doi: 10.1016/j.knosys.2016.10.003

Lewellen, J. (2011). Institutional investors and the limits of arbitrage. *Journal of Financial Economics*, *102*. doi: 10.1016/j.jfineco.2011.05.012

Li, X., Xie, H., Chen, L., Wang, J. & Deng, X. (2014). News impact on stock price return via sentiment analysis. *Knowledge-Based Systems*, *69*. doi: 10.1016/j.knosys.2014.04.022

Lochstoer, L. A. & Muir, T. (2022). Volatility expectations and returns. *Journal of Finance*, *77*. doi: 10.1111/jofi.13120

Loughran, T. & Mcdonald, B. (2011). When is a liability not a liability? textual analysis, dictionaries, and 10-ks. *Journal of Finance*, *66*. doi: 10.1111/

j.1540-6261.2010.01625.x

Loughran, T. & Mcdonald, B. (2016). Textual analysis in accounting and finance: A survey. *Journal of Accounting Research*, *54*. doi: 10.1111/1475-679X .12123

McGurk, Z., Nowak, A. & Hall, J. C. (2020). Stock returns and investor sentiment: textual analysis and social media. *Journal of Economics and Finance*, *44*, 458-485. doi: 10.1007/s12197-019-09494-4

Mihalcea, R. & Garimella, A. (2016). What men say, what women hear: Finding gender-specific meaning shades. *IEEE Intelligent Systems*, *31*. doi: 10.1109/MIS.2016.71

Moghar, A. & Hamiche, M. (2020). Stock market prediction using lstm recurrent neural network. *Procedia Computer Science*, *170*, 1168-1173. Retrieved from `https://www.sciencedirect.com/science/article/pii/S1877050920304865` (The 11th International Conference on Ambient Systems, Networks and Technologies (ANT) / The 3rd International Conference on Emerging Data and Industry 4.0 (EDI40) / Affiliated Workshops) doi: https://doi.org/10.1016/j.procs.2020.03.049

Nguyen, T. H., Shirai, K. & Velcin, J. (2015). Sentiment analysis on social media for stock movement prediction. *Expert Systems with Applications*, *42*. doi: 10.1016/j.eswa.2015.07.052

Niederhoffer, V. (1971). The analysis of world events and stock prices. *The Journal of Business*, *44*. doi: 10.1086/295352

Nofer, M. & Hinz, O. (2015). Using twitter to predict the stock market. *Business and Information Systems Engineering*, *57*. doi: 10.1007/s12599-015-0390-4

Pimpalkar, A. P. & Raj, R. J. R. (2020). Influence of pre-processing strategies on the performance of ml classifiers exploiting tf-idf and bow features. *ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal*, *9*. doi: 10.14201/adcaij2020924968

Poria, S., Cambria, E. & Gelbukh, A. (2016). Aspect extraction for opinion mining with a deep convolutional neural network. *Knowledge-Based Systems*, *108*. doi: 10.1016/j.knosys.2016.06.009

Provost, F. & Fawcett, T. (2013). *Data science for business*. Beijing: O'Reilly. Retrieved from https://www.safaribooksonline.com/library/view/data-science-for/9781449374273/

Pujari, C., Aiswarya & Shetty, N. P. (2018). Comparison of classification techniques for feature oriented sentiment analysis of product review data. In (Vol. 542). doi: 10.1007/978-981-10-3223-3_14

Ranco, G., Aleksovski, D., Caldarelli, G., Grčar, M. & Mozetič, I. (2015). The effects of twitter sentiment on stock price returns. *PLoS ONE*, *10*. doi: 10.1371/journal.pone.0138441

Renault, T. (2017). Intraday online investor sentiment and return patterns in the u.s. stock market. *Journal of Banking and Finance*, *84*. doi: 10.1016/j.jbankfin.2017.07.002

Ruppert, D. & Matterson, D. S. (2019). *Statistics and data analysis for financial engineering with r*.

Surowiecki, J. (2004). *The wisdom of crowds. 2004*.

Taddy, M. (2013). Multinomial inverse regression for text analysis. *Journal of the American Statistical Association*, *108*. doi: 10.1080/01621459.2012.734168

Tetlock, P. C. (2007). Giving content to investor sentiment: The role of media in the stock market. *Journal of Finance*, *62*. doi: 10.1111/j.1540-6261.2007.01232.x

Tumarkin, R. (2002). Internet message board activity and market efficiency: A case study of the internet service sector using ragingbull.com. *Financial Markets, Institutions and Instruments*, *11*. doi: 10.1111/1468-0416.11403

Tumarkin, R. & Whitelaw, R. F. (2001). News or noise? internet postings and stock prices. *Financial Analysts Journal*, *57*. doi: 10.2469/faj.v57.n3.2449

Welch, I. & Goyal, A. (2008). A comprehensive look at the empirical performance of equity premium prediction. *Review of Financial Studies*, *21*. doi: 10.1093/rfs/hhm014

Xing, F. Z., Cambria, E. & Welsch, R. E. (2018). Natural language based financial forecasting: a survey. *Artificial Intelligence Review*, *50*. doi: 10.1007/s10462-017-9588-9

Appendix

6.1 Data - Technical Details

6.1.1 Stock Data

We downloaded data through the yfinance package in python.

6.1.2 Twitter Data

We used the Twitter APIv2, with academic access to download Twitter data, which is the most efficient way to collect data in large scale for sentiment analysis about stocks. This is because the Twitter API limits the ability to extract data with the ticker "Cashtag", $, and to get Tweets older than 7 days in the past, to those with the academic access level. With this access, one can extract up to 10 million Tweets per month of stock-specific information, all the way back to when the first Tweet was created in 2006.

We used "twarc" and "twarc csv" to download the data and to convert it into csv format. This method is convenient because it requires little coding and outputs all tweet specific information without needing to specify it in the query. This means that data on other characteristics of the tweets like follower count, verification status and likes are outputted in the JSON file automatically.

6.2 Classification Parameters

6.2.1 Logistic Regression

Parameters used (all other are set to default)[1]:

Logistic regression parameters = n_jobs : 1, C : 1e5

- n_jobs : The total number of CPU cores used, we use 1 core.
- C : Default is set to 1 our value is 1e5 (100000) Inverse of regularization strength, we have a weak regularization.

6.2.2 Naïve Bayes

Parameters used (all other are set to default)[2]:

Naïve bayes parameters = $alpha$: 0.1

- $alpha$: smoothing technique parameter set to 0.1, where interval is 0 to 1. Moderate grade of smoothing.

6.2.3 Support Vector Machine

Parameters used (all other are set to default)[3]:

Parameters SVM = $loss$: 'hinge', $penalty$: 'l2',$alpha$: 1e-3, $random\_state$: 42, max_iter : 5, tol : None

- $loss$: Loss function used "hinge" gives a linear support vector

[1]Logistic Regression
[2]Naïve Bayes
[3]SVM

- *penalty*: Regularization term,used value "l2" default for standard regularised for linear models.
- *alpha* : Constant multiplies the regularization term value set "1e-3" (0.01) weak regularization
- *random_state* : Set integer for reproducible output across multiple functions, value set to "42".
- *max_iter* : The maximum number of passes over the training data (aka epochs). Impacts only behavior of fitting method, value set to "5".
- *tol* : Stopping criterion of training, stop when errors is not decreasing. Can be used to set early stopping avoiding over fitting of model.

6.3 Regression Parameters

6.3.1 Random Forest

Parameters used (all other are set to default)[4]:

Random Forest parameters = $random\_state$: 0, n_jobs : 6

- n_jobs : The total number of CPU cores used, we use 6 core.
- $random\_state$: Set integer for reproducible output across multiple functions, value set to "0".

6.3.2 Gradient Boosted Regression Tree

All parameters other are set to default[5].

[4]Random Forest
[5]Gradient Booster

6.3.3 LSTM

LSTM parameters = *dropout* : 0.2 , *units* : 9

Dropout : The fraction of inputed units to drop before the linear transformation. The dropout value is specified a percentage between 0 (full connetion) and 1 (no connection).

LSTMs can easily overfit training data, reducing the models predictive skill. Using dropout to regularise the model, input and recurrent connections are excluded from activation and weight updates while training. This has the effect of reducing overfitting and improving model performance[6].

Unit : Dimension of the inner cells in LSTM and the output. Dimension of hidden and output layer should have the same dimension, In Keras only one LSTM block is defined, we use return_sequence=True, using timestamp as a output feature.

More units makes model more complex. Generally speaking, if the period for training your model is longer, a more complex model would be better suited for learning your data[7].

[6]Keras documentation
[7]Keras documentation