Magnus Fjærli
Joakim Espeseth Larsen

Sentiment Analysis of Nasdaq News

Trading Decisions Based on Dictionary-based Sentiment Analysis of Nasdaq News

Master's thesis in Economics and Business Administration
Supervisor: Arild Brandrud Næss
Co-supervisor: Denis M. Becker
May 2022

Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Economics and Management
NTNU Business School

NTNU
Kunnskap for en bedre verden

Magnus Fjærli
Joakim Espeseth Larsen

Sentiment Analysis of Nasdaq News

Trading Decisions Based on Dictionary-based Sentiment Analysis of Nasdaq News

Master's thesis in Economics and Business Administration
Supervisor: Arild Brandrud Næss
Co-supervisor: Denis M. Becker
May 2022

Norwegian University of Science and Technology
Faculty of Economics and Management
NTNU Business School

NTNU
Norwegian University of
Science and Technology

Abstract

Sentiment analysis has in the recent years seen improvement and promise in a variety of fields within finance. In this thesis, we have used a dictionary-based approach to analyse the sentiment of news articles published on Nasdaq News. The thesis is an attempt to see if the efficient market hypothesis (EMH) holds, or if there are opportunities to exploit inefficiencies in the market following news events. The dataset was made of 1.4 million articles by various news providers published in the period 2007–2019. Where needed, text processing like stemming and stop word removal were applied to the article content before the Sentiment Analysis. The dictionaries used and evaluated are: TextBlob, VADER and SentiWordNet. The sentiment scores were used to predict rise or fall in the stock price by the help of different machine learning classifiers. We compared the best performing models from the three sectors evaluated. The sectors we looked at was the Finance, Energy and Technology sector. Models with the best F1 score was implemented in a long/short trading strategy. The research found that the Finance and Energy sector had opportunities for good risk adjusted returns, represented by high Sharpe ratio values. The same methods proved unable to find good trading opportunities in the Technology sector. Generally, the K-Nearest Neighbour (KNN) classifier seemed to be the best performing classifier and SentiWordNet the best suited dictionary on our dataset. Our thesis does not contradict the EMH, but our findings could indicate that there are some sectors and time periods in which the strongest form of the EMH does not hold. Based on our results, we find it more likely than not that a purely dictionary-based approach with the dictionaries examined is a too simple approach. Obtaining abnormal investments returns applying our models and trading strategy alone is difficult. It does, however, provide interesting opportunities in terms of using our findings as trading signals in a more comprehensive trading strategy.

Sammendrag

Sentimentanalyse har de siste årene sett stor utvikling og oppløftende resultater innen en rekke felt innenfor finans. I denne oppgaven har vi brukt en dictionary-basert tilnærming for å analysere sentimentet i nyhetsartikler publisert på Nasdaq News. Avhandlingen er et forsøk på å se hvorvidt hypotesen om markeds effisiens holder, eller om det er muligheter for å utnytte ineffektivitet i markedet etter nyhetshendelser. Datasettet består av 1.4 millioner artikler fra forskjellige nyhetsleverandører, publisert i tidsperioden 2007–2019. Hvor nødvendig, ble tekstprosessering som “stemming” og “stop word removal” brukt på artikkelinnholdet før sentimentanalysen. Ordbøkene som er brukt og vurdert er: TextBlob, VADER og SentiWordNet. Sentimentanalysen gjennomført av de ulike ordbøkene ble brukt til å forutsi økning eller fall i aksjekursen ved hjelp av forskjellige klassifikatorer. Vi undersøkte finans, energi og teknologi sektorene. Videre sammenlignet vi modellene som presterte best for de forskjellige sektorene. Våre modeller med best F1 score ble implementert i en long/short handelsstrategi. Strategien viste at finans- og energisektoren hadde muligheter for god risikojustert avkastning, representert ved høy Sharpe ratio. De samme metodene viste seg å ikke være i stand til å finne gode handelsmuligheter for teknologisektoren. Generelt genererte K-Nearest Neighbour (KNN) best resultater blant klassifikatorene. SentiWordNet presterte best totalt sett blant ordbøkene. Avhandlingen vår motsier ikke hypotesen om markedseffisiens. Likevel, kan funnene våre indikere at det er noen sektorer og tidsperioder der den sterkeste formen for markedseffisiens ikke holder. På bakgrunn av våre resultater finner vi det mer sannsynlig at en utelukkende dictionary-basert tilnærming med de brukte ordbøkene er en for enkel. Å oppnå unormal avkastning ved hjelp av våre modeller og handelsstrategier alene er vanskelig. Det gir imidlertid interessante muligheter til å bruke funnene våre som et handelssignal i en mer avansert handelsstrategi.

Preface

This Master's thesis concludes the authors' Master of Science degrees in Economics and Business Administration at the Norwegian University of Science and Technology (NTNU), in the Spring of 2022. The thesis is written by Joakim Espeseth Larsen and Magnus Fjærli under supervision by Associate Professor Arild Brandrud Næss. His knowledge of the field and guidance in our work has been of great use. We also want to thank Associate Professor Denis M. Becker for providing us with the dataset of the news articles. We are convinced that the learning outcome of writing this thesis has enhanced our skill set to meet future challenges after our studies at NTNU. Any potential mistakes in this work are our own and should not in any way reflect poorly on our supervisors or the university.

Contents

Abstract **i**

Sammendrag **ii**

Preface

1 Introduction **1**

2 Background **4**

2.1 Market Efficiency 4

2.2 Text Processing 6

2.3 Dictionaries 6

2.4 Previous Work 8

2.4.1 Dictionary Based News Sentiment Analysis 8

2.4.2 Dictionary Based Sentiment Analysis on Social Media 10

2.4.3 Sentiment Analysis without Dictionaries 11

3 Research Questions **16**

4 Research Design **17**

4.1 Dataset 18

4.2 Text Processing 20

4.3 Sentiment Analysis - Dictionaries 21

4.3.1 TextBlob 21

4.3.2 VADER 22

4.3.3 SentiWordNet 23

4.3.4 Dictionary Comparison 25

4.4 Data Preparation 25

4.5 Classifiers 26

4.5.1 Linear Discriminant Analysis 26

4.5.2 Naive Bayes 27

4.5.3 K-Nearest Neighbour 28

4.5.4 Support Vector Machine 28

4.5.5 Decision Tree 30

4.6 Trading Design 30

5 Results **32**
5.1 Dictionary Performance Comparison 32
5.2 Model Evaluation 33
5.3 Trading 37

6 Discussion **43**
6.1 Market Efficiency Implication 43
6.2 Dictionaries Implication 44
6.3 Classifiers Implication 46
6.4 Trading Implication 47

7 Conclusion **50**
7.1 Future Work 51

List of Figures

1 Flow Chart of Research Design 17
2 Energy trades comparison 39
3 Finanace trades comparison 40
4 Technology trades comparison 40

List of Tables

1 A feature-level comparison of the dictionaries used in the thesis 7
2 Characteristics of included research with sentiment dictionaries 14
3 Characteristics of included research with machine learning approach 15
4 Descriptive statistics of sentiment score & stock returns-Technology 19
5 Descriptive statistics of sentiment score & stock returns-Finance 19
6 Descriptive statistics of sentiment score & stock returns-Energy 20
7 TextBlob-Dictionary sample word score 22
8 VADER-Dictionary sample word score 23
9 SentiWordNet-Dictionary sample word score 24
10 Dictionary-Comparison 25
11 Dictionary Regression where Return is the dependent variable 32
12 Classification report metrics Energy 34
13 Classification report metrics Finance 35
14 Classification report metrics Technology 36
15 Trading metrics for the Technology companies 37
16 Trading metrics for the Finance companies 37
17 Trading metrics for the Energy companies 38
18 Sharpe ratio for trading strategies 41

1 Introduction

With the emergence of new communication technology, the way people consume news has changed. The old physical newspaper has been replaced by a virtual online version, and a variety of new sources of information like weblogs and social media has surfaced. Readers are more inclined to use online sources of news mainly due to two key features: interactivity and immediacy (Karlsson, 2011). Interactivity refers to the tendency shown by the masses to consume more news of their interest. Simply put, interactive media allows users to connect with others and make them active participants in the media. Immediacy is a feature that represents peoples need to be informed about news with no delay in time (Kohut et al., 2010). As a result of this change in consumption behaviour, ways of analysing the emotion expressed in the media has become a field of major interest for researchers in many fields, especially within finance.

Sentiment Analysis, sometimes referred to as Opinion Mining, is a method to find the polarity or strength of the opinion (positive or negative) that is expressed in written text. In our case this is the news articles from the Nasdaq News website. Two popular ways to automate the Sentiment Analysis process exists; the Dictionary method and the Machine Learning method. We have chosen a dictionary approach, utilizing pre-programmed dictionaries. There exists a large pool of dictionaries to choose from. As our articles contain financial news, the dictionaries chosen to analyse the sentiment have all previously showed promising results on similar text corpuses. The dictionaries compared are VADER, SentiWordNet and TextBlob.

As the quality of dictionaries and Sentiment Analysis is improving, opportunities arose for investors to use Sentiment Analysis in their investment decisions. Much research has been done on different dictionaries and stock price prediction. Jagdale et al. (2018) provides a comprehensive review of the most common dictionaries for Sentiment Analysis. However, little research has been conducted on comparing the different dictionaries' stock prediction abilities with each other. Even less research has been carried out on investigating potential differences within the different sectors in the market. Dodevska et al. (2019) explored financial and technology companies. Nevertheless, there was no attention directed towards comparing the two sectors. Makrehchi et al. (2013) predicted price movement of the 30 biggest companies on the Dow Jones Index and found company specific differences. Even so, no clear conclusion regarding specific sectors were drawn. The focal point in our research is the Finance, Energy and Technology sector.

The motivation for investors to utilize dictionaries for investing decisions can come from numerous sources. There are two main arguments for this technique. The speed of which one can get the sentiment is way faster than any human could possibly read an article. The other argument is that the bias is low or non-existent when performing the Sentiment Analysis. If present, the bias is descendent from potential biases inherent in the creation process of the dictionary. One can argue that dictionaries are easier for humans to intuitively understand compared to most machine learning techniques. Using dictionaries, one would avoid any potential black box problems.

The main research question of this thesis is: “Can dictionary-based Sentiment Analysis of Nasdaq News improve stock price prediction and trading decisions?”

To answer the question, dictionaries have been combined with natural language processing (NLP) on a dataset from Nasdaq News containing 1 410 535 articles. The content in these articles has been evaluated by three different sentiment dictionaries. Five machine learning classifiers have also been used to test if they are able to predict rise or fall on stock prices, this is based on the sentiment derived from the dictionaries. The efficient market hypothesis (EMH) states that it is impossible to achieve excess return on the market over a longer period. Our models report mixed results. The best models are profitable, with a higher risk adjusted return compared to comparable indexes over the test period. This contradicts the EMH. Nonetheless, most of the models presented in this thesis agree with the theory of EMH, when looking at how they are not achieving excess returns. To produce the results, text filtering and text processing of the news article has been executed. Both stop word removal and Porter stemming has been applied to some of the content. Additionally the results have been generated by applying Linear Discriminant Analysis (LDA), Naive Bayes, K-Nearest Neighbours (KNN), Support Vector Machine (SVM), and Decision Tree.

Following the main research question, other research questions arise. “How do our findings effect the EMH?” Also; “Which dictionaries provides most useful Sentiment Analysis?” and lastly “What classifiers provides best stock price prediction?”

This thesis contributes to the research on dictionary-based Sentiment Analysis on news articles by analysing some of the most used dictionaries. Our results compare models using different dictionaries to predict stock price movements within various sectors. Where most papers in this field only use the classification report metrics to evaluate the stock movement predictions, our thesis also proposes a trading strategy. The strategy is then applied on out-of-sample data, making the results re presentable for real life trading.

The results from this thesis found promising risk adjusted returns for the Energy and Finance sector, but was unable to do so for the Technology sector. Our trading models achieved Sharpe ratio value up to 2.929 for the Finance sector and 2.030 for the Energy sector. The best suited dictionary and classifier seemed to be SentiWordNet and KNN respectively. SentiWordNet performed well on both the Finance and Energy sector while TextBlob only performed well on Energy. Interestingly, VADER performed the best on Technology. Despite this, the model did not provide good risk adjusted returns. The main conclusion to draw from the thesis, is that it is hard to achieve abnormal annualized returns using the rather simple dictionary-based approach. Findings indicate, however, that there could be evidence that our models perform well both in a bear and bull market. This affirms that our models could serve as trading signals in the Energy and Finance sector, especially in down trending markets. Traders could be interested in implementing this to their trading strategies.

The structure of the thesis is built as follows. The thesis starts with a background section. The background chapter includes economic theory, key features of dictionaries, and classifiers applied. The next section elaborates on previous work in the field of Sentiment Analysis focusing especially on research with emphasis on EMH and stock price prediction. Subsequently, the research design is elaborated and discussed. Then, the results and findings from the research is presented and this is followed by a discussion of the EMH, the classifiers, the dictionaries, and the trading implications. Lastly, all our findings are summarized in a conclusion where possible future work is presented.

2 Background

This background chapter is included to provide the reader with a general understanding of the field of dictionary-based Sentiment Analysis, focusing on applications on stock price prediction. To answer our thesis' research question, this chapter provides a surface level description of the most important aspects in the field, as well as the key research methods.

Efficient market theory is included because of its undeniable connection with market information and market performance. Text processing is covered as the reader should be aware of the importance of preparing a text corpus for Sentiment Analysis purposes. The dictionaries are presented as the way they operate are a key factor to be able to answer our research question. Previous work on this field is presented as a tool for the reader to quickly get an insight in what has been done earlier on this topic. All research methods will also be thoroughly explained in chapter 4.

2.1 Market Efficiency

The efficient market hypothesis is a theory that has had a significant impact on market prediction. According to this theory, the price of a security reflects complete market information and as new information is introduced, the market instantly adjusts the stock price to reflect it (Fama, 1965). Fama categorized three degrees of market efficiency: *Weak form*, *Semi-strong-form*, and *Strong form*.

The weak form for market efficiency assumes that previous price movements, earnings and volume cannot be used to predict future stock prices. Weak form for market efficiency does, however, suggest that it is possible for investors with extraordinary expertise or material non public information (MNPI) also known as inside information, to achieve abnormal returns. Abnormal returns refer to the risk adjusted return exceeding the comparable expected return on an asset or portfolio. Often represented by an index like the S&P 500 (Modigliani & Modigliani, 1997).

The semi-strong form for market efficiency refers to markets where all publicly available and past information is absorbed and fully reflected in the stock prices. This contests the weak form´s belief in fundamental analysis and expertise. The semi-strong form only explains abnormal returns by MNPI.

Strong form for market efficiency is the most stringent form of the three. It states that all information, both public and non-public is accounted for in the stock price. This includes MNPI. From this follows that even an investor with insider information will not be able to achieve abnormal returns.

EMH stems from Eugene Famas work in his PHD distortion in the 1960s. It was, however, popularized by Burton Malkiels book "A Random Walk Down Wall Street" (Burton, 2018). The book states that short term future stock prices are random and not influenced by past events. In his book, Malkiel is critical to the role of people like portfolio managers and financial advisors. He is quoted claiming: *"A blindfolded monkey throwing darts at a newspaper's financial pages could select a portfolio that would do just as well as one carefully selected by experts".* Random walk claims that stock prices move in random directions and that it is not possible to predict these movements using historical prices (Technical Analysis) or financial information (Fundamental Analysis).

Today there is no real consensus on whether stock prices follow a "random path" or if they are possible to predict. For each article that supports EMH another one denies it. Even testing for market efficiency is difficult according to large parts of the literature. As market and economic conditions changes continuously, tests are hard to develop and replicate (Ţiţan, 2015). One of the most researched reasons for possible inefficiencies in stock prices, are that investors are inattentive and not able to absorb and assess information fast enough for markets to always be efficient. Shleifer et al. (1990) and Hirshleifer et al. (2012) both found that prices can diverge significantly from fundamental values even in the absence of fundamental risk. Studies like Hamid et al. (2010) arguments that there is hold for the weak form of EMH. The article looks at 14 countries in the Asia-pacific region and finds that in these markets the stock prices does not follow the random walk, and that traders would be able to benefit from this. Anomalies are empirical results that seem to be inconsistent with the EMH. When these are present, it would indicate that there are profit opportunities or inadequacies in the way asset pricing-models work. However, after being documented these anomalies disappear, reverses or are reduced. This raises the important question on whether these opportunities existed in the past and were arbitraged away or if they were just statistical aberrations (Schwert, 2003). Some well-known examples of such anomalies are "the weekend effect" and "the dividend-yield effect". This question is used as an argument in favour of the weak form of market efficiency.

2.2 Text Processing

Prior to the Sentiment Analysis, filtering of the text corpuses to make the corpuses ready for the sentiment analysing techniques is important. When applied purposely, the text processing increase accuracy of the sentiment classification. One of the text processing functions are stop word removal.

When the text analysing programs iterates through a text corpus, many words will have none or small impact on the overall sentiment. *Stop word removal* removes words that are listed in a lexicon of words that would not have an impact on the overall score (Geekofgeeks, 2021). This is done to increase the accuracy of the sentiment scores and to achieve faster running time for the algorithms.

Another filtering tool to prepare the news article content, is *stemming*. Stemming is a tool which shortens words down to the base or root of the words involved (Jivani, 2011). It creates fewer words in total and the meaning of the word could have the same impact on the score, even if it is fully written or on base/root form. Porter stemming is the most used stemming version, it is a fast and effective tool. An example of how stemming works: “Fishing”, “Driving”, “Washing”, is transformed to “Fish”, “Drive”, “Wash”. Stemming is especially useful to apply for dictionaries with a limited number of scored words.

2.3 Dictionaries

A sentiment dictionary is a list of lexical features which are generally labelled according to their semantic orientation as either positive or negative. Manually creating–much less, validating a comprehensive sentiment dictionary is a labour intensive and sometimes error prone process, so it is no wonder that many opinion mining researchers and practitioners rely so heavily on existing dictionaries as primary resources. The dictionary-based approach consists of three steps: feature extraction, feature scoring and score aggregation (Hassan Yousef et al., 2014).

Feature extraction is the extraction of the components in a text that is used to measure sentiment. Dictionary-based approaches differ in that some of them only extract the words in the text while other, more advanced dictionaries also extract negations and degree modifiers. Negation refers to the act of denying or contradicting what follows in a sentence. Words like “not”, “nobody” and “none” are examples of words that triggers the negation functions in these dictionaries. During the text analysis, the polarity of the dictionary item that stands next to the negator in a sentence reverse. Degree modifiers also referred to as intensifiers refer to words like “very”, “most” or “slightly”. “Very” and “most” are called amplifiers and increase the polarity. “Slightly” is an example of a downtoner and decrease the polarity. The degree to which the polarity is increased or decreased depends on the modifiers.

Feature scoring is the process of scoring the extracted features. The scores are built upon the predefined rules and scores in the dictionary in question. Dictionaries to this in various ways. The simplest ones use binary scoring, where they either label the words as positive or negative. However, most of the dictionaries in use today are using a scale for implying the extracted value within a range. VADER for instance has a scale in the range $[-4,4]$. To illustrate the effect of negations and degree modifiers, let’s say we have a positive word like “good”. It is given a 1.8 in the scale. If a negator is used, “not good” would have a score of -1.8. If you put a modifier in front of it and get “mostly good”, the score would be 0.9. The scores for words and weighting of modifiers and negations are classified by an author or a group of people. It is common to use the average score when a group of people are classifying.

The final step in dictionary-based approaches is score aggregation. This step deals with how the extracted scores of a text should be combined. As mentioned, the simpler dictionaries aggregate the scores feature by feature, while other dictionaries aggregate based on more complex relations.

In our work, we have used three different dictionaries: VADER, TextBlob and SentiWordNet. Differences between these dictionaries are found in all steps and are attempted illustrated in table 1.

Table 1: A feature-level comparison of the dictionaries used in the thesis

Dictionary	Feature extraction	Feature scoring	Score aggregation
VADER	Words, emojis, punctuation, negations, degree modifiers	scale:$[-4,4]$	sentences
TextBlob	Words	scale: $[0,1]$ & $[-1,1]$	Word
SentiWordNet	synsets	scale: $[0,1]$	synsets, context

2.4 Previous Work

Sentiment Analysis refers to the use of natural language processing, text analysis, and computational linguistics to identify and extract subjective information in source materials (Haff, 2010). Research confirming the relationship between news and market performance can be dated back to at least Niederhoffer (1971). He examined NY Times headlines and found that world events were more likely to be followed by large market movements than random days. A world event was defined as headlines over a certain size in the newspaper. It was also discovered that there was a strong tendency for large movements in price on the first and second day after a world event. Day 2–5 tended to revert to the level before the world event. This indicates that markets appear to overreact, especially when facing negative news. Since Niederhoffers early efforts, the field of news Sentiment Analysis has been a popular area of research. Recent improvements of computational capabilities, and advancements in machine-learning applications for textual analysis, has caused an exponential growth in the amount of research on this field in the last decade (Ligthart et al., 2021).

Sentiment Analysis is in the literature typically divided into two separate categories. The *dictionary-based approach* and *the machine learning approach*. The dictionary-based approach in some cases use machine learning techniques to make sentiment scoring during the development of the dictionary. These have been previously trained, often as a combination of supervised and unsupervised learning. Machine learning approaches applies machine learning algorithms to evaluate the sentiment. They can vary in nature, but usually consists of a combination of Neural Network, Semantic Analysis, and a variety of classifiers like Naive Bayes, or Support Vector Machine (SVM). In our presentation of previous work on this field, we have narrowed our scope down to presenting research concerning predictive strategies which use sentiment from either the dictionary method or the machine learning method.

Further it is possible to navigate through the rest of the previous work, by looking at table 2 and 3. These two tables represent two different aspects of research on the topic. This could provide the reader with a good overview of the key characteristics of the included literature.

2.4.1 Dictionary Based News Sentiment Analysis

In much the same fashion as Niederhoffer in 1971, Nemes and Kiss (2021) used news headlines to predict the behaviour of financial assets. Their method, however, was not as dependent on manual labelling as was normal in the 1970s. They collected headlines from various news outlets on finviz.com for the four companies: AMD, Google, Amazon, and Facebook. They used two dictionaries: TextBlob and VADER, in addition to a Recurrent Neural Network (RNN). The study found that all three methods showed correlation between the sentiment found and stock variables like close, open, high, low and volume. TextBlob and VADER showed similar results, in the correlation between close, open, high, and low. The RNN model was also able to do this, and it was also able to predict volume better.

More work in this field has been executed on article content. Dodevska et al. (2019) used news articles from Reuters, CNN, and CNBC to predict the stock movements in a 30-minute period after a news drop. They gathered news on four companies: Goldman Sachs, Wells Fargo, Microsoft, and Qualcomm over a three-year period from 2016–2019. They made use of the Loughran and McDonald dictionary which was made specifically for financial information. They initially found no significant correlation between the sentiment and the stock price movement. Eventually, they decided to make models for each individual company, arguing it would be necessary as they all have their own characteristics. They saw prediction accuracy improve significantly when changing to company specific models.

X. Li et al. (2014) used news provided by FiNet News and Yahoo Finance to predict the stock movements for Hong Kong listed stocks. Experiments were conducted over a five-year historical horizon from 2003–2008. Results show that at individual stock, sector and index levels, the models that used Sentiment Analysis outperformed a bag-of-words model. Their tests also concluded that focusing solely on the polarity (negative or positive) of the news did not result in useful predictions. This alludes to the conclusion that subjectivity and objectivity also should be considered. There were only slight differences between the two dictionaries in question, the Harvard IV-4 dictionary and the Loughran and McDonald dictionary.

Deng et al. (2011) researched another genre in which Sentiment Analysis on news articles could be helpful. They combined Sentiment Analysis using the SentiWordNet dictionary with technical analysis of the stock prices. They collected news articles dated from the beginning of 2006 to mid 2008 for the large Japanese US listed companies Sony, Panasonic and Sharp. They used a multiple kernel learning model (MKL) and found that their model outperformed the baseline model. The results showed that depending on which stock was examined, the influence of Sentiment Analysis and technical analysis varied. For Sony and Panasonic, Sentiment Analysis had more influence on the model. For Sharp, pure technical analysis had more influence. Interestingly, they were able to provide better predictions for all three stocks when combining sentiment and technical analysis.

The dictionary-based approach has also seen encouraging results in languages other than English. Day and Lee (2016) made use of two Chinese dictionaries: CSKI and NTSUD. They collected news articles from some of the most popular Chinese news outlets in 2013 and 2014. The news covered 18 publicly traded Chinese companies. The period they examined was the 3-minute period after the news dropped. Interestingly, they found that the accuracy of their models was significantly different depending on the news provider. This indicates that one should pay closer attention to some news providers rather than others.

A. Khedr et al. (2017) used a modified version of the Harvard IV-4 dictionary in their research. They retrieved news from Yahoo Finance, Reuters, and Nasdaq News to examine the behaviour of three of the largest market cap companies traded on the Nasdaq stock exchange: Yahoo, Microsoft, and Facebook. They made a model combining numeric attributes in the prices for the stocks with sentiment from the news articles. The model predicted whether the stock was up or down for each day. They achieved their best results using KNN as a classifier for the stock price direction. In fact, their research showed the best accuracy of all the articles we examined, at 89.9%. This is much higher than most other papers on the field, with other comparable studies presenting accuracy around 60%. The closest to their accuracy were Shynkevich et al. (2015) with 79% accuracy and Y. Kim et al. (2014) with 65%.

2.4.2 Dictionary Based Sentiment Analysis on Social Media

In addition to the literature on financial news and headlines, there are also many researchers who have tried applying the same techniques on social media, usually from Twitter or Reddit. These researchers often have a more trading-minded approach, and we find it relevant to include some of the more prominent work in this direction.

Dictionary application on Twitter has been very popular over the last few years. The VADER dictionary, due to its ability to pick up the nuances in common language on such platforms, is a popular choice for social media Sentiment Analysis. Elbagir and Yang (2019) made a point of proving that VADER works well in analysing sentiment in adjacent fields to stock predictions. They demonstrated that by using VADER on tweets regarding the US election in 2016, one could make a more precise prediction of the outcome of the election than most traditional polls.

Bing et al. (2014) targeted tweets of 30 US companies in various sectors, using SentiWordNet they developed a trading algorithm. Their algorithm was better at predicting stock prices in a 3–day time-horizon. The TextBlob dictionary has also been used on tweets. I. Gupta et al. (2022) combined TextBlob and a Long Short Term Memory (LSTM) network; They looked at companies listed on the National Stock Exchange of India (NSE). The time-period of the tweets collected was not clearly stated, but they tried to predict the daily closing price of the listed companies using the sentiment from the tweets with their combined model. The group found a strong correlation between the movement of stock prices and the public perception of news articles.

Other than Twitter, Reddit is also a key source for researchers to mine for sentiment on social media. Reddit's architecture is designed with focus on interaction in various subreddits, serving the role of forums for discussing a wide range of topics including stocks. This could potentially increase the relevance of the data collected as compared to Twitter. Biswas et al. (2020) found that there was a firm correlation between news on the pandemic and the spread of Covid-19 with the movement of share prices in stock markets. They compared the sentiment gathered by the TextBlob dictionary using the prices of the Bombay Stock Exchange index (SENSEX). They collected news and user-interaction within the timeframe 02.05.2019–20.03.2020. Their conclusion indicates that not only did the sentiment correlate with the spread of the virus, the sentiment could also improve the prediction of the SENSEX's closing price.

C. Wang and Luo (2021) attempted to see if it was possible to use the subreddit r/wallstreetbets to predict the stock price of the GameStop stock during the time-period from 04.01.2021–31.03.2021. They tried to combine the VADER dictionary with a variety of other classifier models. The study was unable to definitively demonstrate a strong relationship between sentiment and price movement.

2.4.3 Sentiment Analysis without Dictionaries

Much of the literature on *prediction motivated* Sentiment Analysis does not use pre-trained dictionaries to assess sentiment in text. They opt for the machine-learning alternative. Machine-learning approaches are recognized using machine-learning techniques like Naive Bayes, Neural Networks, or even semantic analysis to classify text into categories (Guo et al., 2017).

Kazemian et al. (2016) trained a sentiment analyser using the SVM technique. They trained their sentiment analyzer on Reuters news for 22 New York Stock Exchange listed companies. The news articles were from 1997–2013. Multiple time periods for a trade strategy were examined. 1-, 3-, 5-, and 30-day intervals. They achieved high accuracy when tested on the financial news data. When compared to a momentum model and the S&P 500 Index, their model outperformed both in terms of absolute return and risk adjusted return, as measured by Sharpe ratio.

Reuters was also the news source used by (Uhl, 2014). He used 3.6 Million Reuters articles aiming to predict the Dow Jones Industrial Average Index (DJIA) monthly returns. He tested a trading strategy where he went long or short the index for the month, depending on the model's prediction. He used a sentiment algorithm called Reuters Sentiment to classify each article into positive, negative, or neutral sentiment. Compared to the market index, the trading strategies with Reuters Sentiment achieve significant outperformance with high success rates, as well as high Sharpe ratios.

Jishag et al. (2020) demonstrated how one could combine machine-learning approaches with Sentiment Analysis and technical analysis. *Autoregressive integrated moving average* (ARIMA) was their method of choice for technical analysis. News was gathered from Yahoo News, Nasdaq News and The Wall Street Journal. The news concerns ten of the companies with the largest market capitalization on Nasdaq. The research also compared their model with a dictionary-based model, using the Loughran and McDonald dictionary. Combining the sentiment scores and the ARIMA data, they created a model that performed better than the dictionary model.

Rather than trying to predict the binary outcome of increase or decrease, W. Wang et al. (2013) tried to predict jumps in stock prices. A stock jump was defined as a 2% or larger change in the stock price in one trading day. Companies listed on DJIA were explored. Articles were collected from the Dow Jones News Wire. Findings were promising for predicting jumps in stock prices. Interestingly, they also found conclusive evidence that news sentiment rather than news volume is useful for predicting stock price jumps. For sentiment classification they used a Probabilistic Neural Network (PNN).

Chantona et al. (2020) proved that machine learning methods for Sentiment Analysis can work well in adjacent fields like FOREX-trading. News was gathered from DailyForex.com in a ten-year time-period from 2009–2019. They predicted multiple currency exchange rates like EURUSD and GBPUSD. They looked at various intraday time frames from 15 min to 4 hours. Both a RNN and a Convolutional Neural Network (CNN) were used to find sentiment. Findings show that their Neural Network performs better and yields higher returns on most currency pairs for smaller timeframes. The addition of semantic information from news headlines helps to manage risk with faster withdrawals, resulting in a better maximal drawdown.

Cryptocurrency trading application share much of the same principles as trading stocks. This is also a popular field of research. Lamon et al. (2017) measured the interaction between media sentiment and tweets with the prices of Bitcoin, Litecoin and Ethereum. They tried multiple classifiers in order to classify the news and tweets. Logistic Regression and Bernoulli Naive Bayes were both tested, but Logistic Regression produced the best results. A functioning model that could make cryptocurrency price predictions using non-technical data was developed, and the model was generally able to predict the largest (magnitude) price increases and decreases correctly.

Where Lamon et al. (2017) combined news and tweets, many researchers have used only tweets to try to make trading decisions. Pagolu et al. (2016) collected in a year over 250 000 tweets about Microsoft, using the Twitter API to filter for tweets containing Microsoft, #MSFT or Windows. Best results were obtained using a Random Forrest classifier to classify tweets into negative, positive, or neutral. This was achieved by using the Word2Vec or N-gram representation of the text corpuses. The finds in the research support their initial hypothesis that positive emotions or sentiment towards a stock on social media, reflects in its stock price. The validation of Twitter as a source for useful sentiment mining, is backed up by both Makrehchi et al. (2013) and Colianni et al. (2015) who found well-preforming trading strategies for companies listed on the DJIA and Bitcoin respectively.

Table 2: Characteristics of included research with sentiment dictionaries

Authors	Name	Dep. Var	Dictionary	Source	Examined	Time period	Frequency	Comment
Khedr, A. E., & Yaseen, N. (2017)	Predicting Stock Market Behavior using Data Mining Technique and News Sentiment Analysis	Price	Harvard and Kalman filter	Yahoo Finance, Reuters, Nasdaq.com	Yahoo, Microsoft, Facebook	Not stated	Daily	Improvement of predicting accuracy of three different companies for the future trend of the stock market
Shihab Elbagir & Jing Yang (2019)	Twitter Sentiment Analysis Using Natural Language Toolkit and VADER Sentiment	Sentiment	VADER	Twitter	US election	22.11.2016–24.11.2016	Daily	VADER was an effective choice for sentiment analysis classification using Twitter data
Charlie Wang & Ben Luo (2021)	Predicting $GME Stock Price Movement Using Sentiment from Reddit r/wallstreetbets	Price	VADER	Reddit	Gamestop	04.01.2021–31.03.2021	Intraday	Unable to definitively demonstrate a strong relationship between sentiment and price movement.
Li, X., Xie, H., Chen, L., Wang, J., & Deng, X. (2014)	News impact on stock price return via sentiment analysis	Price	Harvard IV-4, Loughran and McDonald	FINET, Yahoo Finance	Multiple stocks on Hong Kong Stock Exchange	January 2003–March 2008	Daily	Sentiment analysis helps increase precision accuracy. The models with sentiment analysis performs better than bag-of-words and polarity analysis.
Biswas, S., Sarkar, I., Das, P., Bose, R., & Roy, S. (2020)	Examining the Effects of Pandemics on Stock Market Trends through Sentiment Analysis	Price	Textblob	Reddit	SENSEX, The Bombay Stock Exchange Index	02.05.2019–20.03.2020	Daily	The proposed model can predict market trends as there is a firm co-relation between news on pandemic and spread of Covid with that of movement of share prices in stock markets.
Bing, L., Chan, K. C., & Ou, C. (2014)	Public Sentiment Analysis in Twitter Data for Prediction of A Company's Stock Price Movements	Price	SentiWordNet	Twitter, Yahoo Finance	30 companies in different industries	October 2011–March 2012	3–day interval	Their algorithm is better at predicting stock prices when there is a 3 days interval.
Deng, S., Mitsubuchi, T., Shioda, K., Shimada, T., & Sakurai, A. (2011)	Combining Technical Analysis with Sentiment Analysis for Stock Price Prediction	Price	SentiWordNet	Technology Social Network Source: Engadget, Google Finance Website	Sony, Panasonic, Sharp	01.01.2006–17.08.2008	Daily	The Multiple Kernel Learning (MKL) performs better than other baseline methods.
Gupta, I., Madan, T. K., Singh, S., & Singh, A. K. (2022)	Historical and Sentiment Analysis based Stock Market Forcasting Model	Price	Textblob	Twitter	NSE, National Stock Exchange India	1 year period. Unknown date	Daily	Combined Textblob with LSTM. Found a strong correlation between the movement of stock prices and the public perception of news. of news articles.
Day, M. Y., & Lee, C. C (2016)	Deep learning for financial sentiment analysis on finance news providers	Price	CSKI and NTSUD (Chinese dictionaries)	NowNews, Apple Daily, Liberty Time News, MoneyDJ	18 public Chinese companies	01.01.2013–31.12.2014	Intraday (3 min before and after news)	Deep learning helped improve accuracy of predictions. Found significant differences in news providers.
Dodevska et al. (2019)	Predicting companies stock price direction by using sentiment analysis of news articles	Price	Loughran and Mc Donald	Reuters, CNN CNBC	Goldman Sachs, Wells Fargo, Microsoft, Qualcomm	01.01.2015–31.12.2018	Intraday (30 min before and after news)	No significant correlation between the news sentiment and the change of the stock prices. When creating models for just one company however, the accuracy and the prediction were higher.
Nemes, L., & Kiss, A (2020)	Prediction of stock values changes using sentiment analysis of stock news headlines	Price	VADER, Textblob, also a Recurrent Neural network (RNN)	Headlines from various financal news sites	Advanced Micro Devices, Google, Facebook and Amazon	27.10 2020–14.11.2020	Daily	Finds that news headlines sentiment can indicate stock price movement. RNN outperforms the other sentiment tools.

Table 3: Characteristics of included research with machine learning approach

Authors	Name	Dep. Var	Source	Examined	Time period	Frequency	Comment
Jishag, A. C., Athira, A. P., Shailaja, M., & Thara, S. (2020)	Predicting the Stock Market Behavior Using Historic Data Analysis and News Sentiment Analysis in R	Price	Yahoo Finance, Wall Street Journal, Nasdaq.com	Over 10 companies on NASDAQ	Not stated	Daily	Combined sentiment analysis with historical stock prices to make prediction for future stock prices.
Wang, W., Ho, K. Y., Liu, W. M. R., & Wang, K. T. (2013)	The relation between news events and stock price jump: an analysis based on neural network	Price	Dow Jones news wire articles	Dow Jones listed companies	2004–2012	Daily	Founds are promising for predicting jumps in stock prices. Also finds conclusive evidence that news sentiment not news volume is the useful for predicting stock price jumps.
Pagolu, V. S., Reddy, K. N., Panda, G., & Majhi, B. (2016)	Sentiment Analysis of Twitter Data for Predicting Stock Market Movements	Price	Twitter	Microsoft	31.08.2015–25.08.2016	Daily	Attempt to find correlation between Twitter Sentiment and stock price.
Cakra, Y. E., & Trisedya, B. D. (2015)	Stock Price Prediction using Linear Regression based on Sentiment Analysis	Price	Twitter REST API, Yahoo Finance CSV API	13 companies from Indonesia	14.04.2015–30.04.2015	Daily	Good score from sentiment analysis on both classifying of tweet data and stock price prediction. Retrieved R-squared value of almost 1 in price prediction, meaning the model fitted lots of data.
Wenbin Zhang & Steven Skiena. (2010)	Trading Strategies To Exploit Blog and News Sentiment	Price	Dailies, Twitter, Spinn3r RSS Feeds and LiveJournal	All stocks on New York Stock Exchange	2005–2009	Daily	Results are significant in confirming the performance of general blog and news sentiment analysis methods over broad domains and sources.
Colianni, S., Rosales, S., & Signorotti, M. (2015)	Algorithmic trading of cryptocurrency based on Twitter sentiment analysis	Price	Tweepy (Twitter API) and cryptonator.com API	Bitcoin	15.11.2015–04.12.2015	Daily and hourly	Used tweets and sentiment as feature vectors. Both had day-to-day outperform hour-to-hour, and tweets outperformed sentiment as feature vectors.
Kazemian, S., Zhao, S., & Penn, G. (2016)	Evaluating Sentiment Analysis in the Context of Securities Trading	Price	Reuters	22 companies listed on New York Stock Exchange	10.03.1997–10.03.2013	Multiple periods. 1-, 3-, 5- and 30-day frequency	Built a binary sentiment classifier that achieves high accuracy when tested on movie data and financial news data from Reuters.
M. Makrehchi, S. Shah & W. Liao. (2013)	Stock prediction using event-based sentiment analysis	Price	Twitter Search API	30 biggest companies on the Dow Jones Index	27.03.2012–13.07.2012	Daily	Applies lexicon and supervised learning approach to try to outperform S&P 500, and manages this successfully. Supervised learning is the approach that generates the best return.
Matthias W. Uhl (2014)	Reuters Sentiment and Stock Returns	Price	Reuters	Dow Jones Industrial Average Index	2003–2010	Monthly	Find that negative Reuters sentiment has more predictive power than positive Reuters sentiment. Trading strategies with Reuters sentiment achieve significant outperformance with high success rates as well as high Sharpe ratios.
Kevin Chantona, Ronsen Purba & Arwin Halim. (2020)	News Sentiment Analysis in Forex Trading Using R-CNN on Deep Recurrent Q-Network	Currency	DailyFX.com	EURUSD, GBPUSD, AUDUSD, NZDUSD, USDCAD, and USDJPY	22.10.2009–22.09.2019	Various intraday times. 15, 30, 60 and 240 min	Finds that their Neural Network performs better and yields higher returns on most currency pairs for smaller timeframes. The addition of semantic information from news headlines helps manage risk with faster withdrawals resulting in a better maximal drawdown.
Lamon, C., Nielsen, E., & Redondo, E. (2017).	Cryptocurrency Price Prediction Using News and Social Media Sentiment	Price	Cryptocoinnews.com and Twitter	Bitcoin, Litecoin and Ethereum	01.01.2017–30.11.2017	Interday	The model was able to correctly predict, on average, the days with the largest percent increases and percent decreases in price for Bitcoin and Ethereum over the 67 days encompassing the test set.

3 Research Questions

In this section, we define our main research question as well as the sub-questions. Below is the main research question listed:

Research question: "Can dictionary-based Sentiment Analysis of Nasdaq News improve stock price prediction and trading decisions?"

Following the main research question, the following sub-research questions are formulated. Our main research question is admittedly broad. Therefore, we considered it necessary to break it down into smaller questions to answer it sufficiently.

Sub-research question 1: Does the EMH make it unviable to achieve abnormal returns for dictionary-based Sentiment Analysis?

The EMH states that it is not possible in efficient markets to make use of Sentiment Analysis to achieve excess returns. As portrayed in the previous work section 2.4, the literature is not conclusive on this. Hence, we do not hypothesise that the EMH does not apply for this way of extracting sentiment from news articles.

Sub-research question 2: How does the choice of sentiment dictionary affect prediction of stock price movement?

As sentiment dictionaries come in various forms, there are possibilities that there will be discrepancy in the predictive applicability for the different dictionaries. The sentiment derived on the same text are usually somewhat different depending on dictionary. For this reason, examining how and why this occurs is important. Especially when implementing them in a predictive strategy.

Sub-research question 3: How does the choice of classifier affect dictionary-based stock movement prediction?

There are different classifiers suited for Sentiment Analysis and stock movement prediction, making it viable to test if some classifiers outperform the others. In our thesis classifiers predict the likelihood of price rise or fall in stock price given by a percentage. Therefore, it's viable to examine different classifiers to achieve the best suited models for our main research question.

4 Research Design

The research design in this thesis is illustrated in Figure 1.

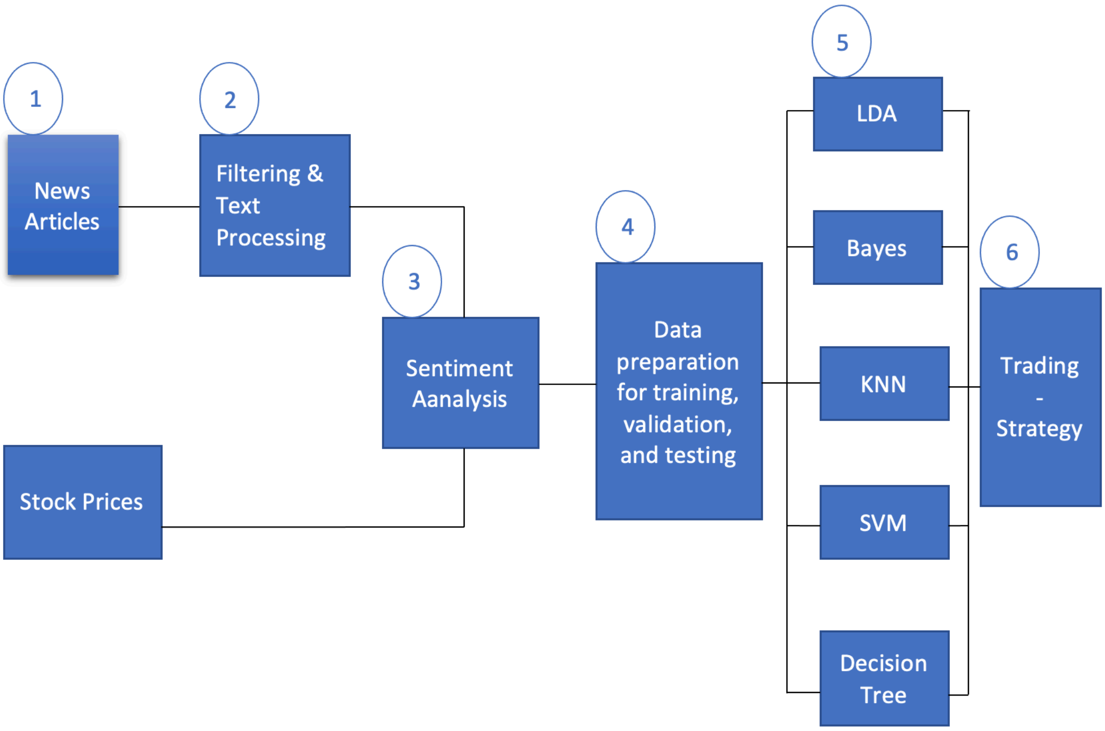

Figure 1: Flow Chart of Research Design, showing each stage of the research process

Figure 1 illustrates the research designs' proposed model for implementing a trading strategy based on news sentiment. Step one includes the raw unfiltered articles from Nasdaq News and stock prices retrieved from Yahoo Finance. Step two introduces a few changes to the news articles. The articles are made subjects to filtering and text processing before they are merged with the stock prices on the corresponding date. Further on, step three is introduced, where the Sentiment Analysis is performed on the article content. After the Sentiment Analysis is performed, the different sectors of news articles are split into test, training, and validation sets. We used the holdout method where the tail (20% newest articles) of the observations are set aside for testing. The next step in the process are the classifiers. The classifiers were applied on the training set, before using their output to create implied probabilities for price rises or falls for the testing sets. At last, the results have been applied for a long/short trading strategy.

As the computation of the models were complex, the iterations in Python were executed on one of the student's desktop computer. The computer contains a Ryzen 5 CPU with speed of 3.6 GHz and DDR4 SDRAM. Most of the iterations were relative fast. When the datasets were larger, containing more and longer articles, some of the iterations could take up to 10 minutes. It is also worth mentioning that the size of the initial news dataset on 1.4 million news articles that was filtered, spent up to 30 minutes importing to R.

4.1 Dataset

In this research, we used a dataset consisting of 1 410 535 news articles from the Nasdaq News page. The articles largely include content on company news, analyst ratings and macro news. Articles from 02.01.2007 to 26.10.2019 were included. In addition to the article content, our dataset shows 11 variables, including the time the articles were published, author, headline, news site and the stock ticker of the companies discussed and mentioned.

In addition to the dataset containing news articles, historical stock price data for all relevant companies were gathered using Yahoo Finance's API in Python. The returns were calculated using the intraday opening and closing price the same day.

Furthermore, articles from the original dataset posted during the markets opening hours were not included in our analysis. This was done to ensure the sentiment of these articles were not already absorbed and priced into the market. Articles tagged with multiple stock tickers were also removed from the data. Articles with multiple tags, were more general in nature and was often tagged with all companies within a given geographic area or sector. To have our data be as company specific as possible and keep the run time for our models manageable, only articles with a single ticker tag were included.

The publication period of news articles was limited to outside the stock exchange opening and closing hours (08:00–16:00). The news articles analysed were used to predict rise or fall on the closing price of the next day on the stock exchange. The choices in limiting the time period, were made in an effort to pick up on sentiment from news the market had not yet a chance to react on. The negative aspects of these choices could be the instant impact of news articles and day trading. By only working on news articles published outside the opening hours, these types of elements would not be picked up on. Also, pre-market and after-hours trades could weaken this approach. The positive aspect of the approach is a smaller, but still adequate sample size which is more manageable.

To investigate possible differences between the three different sectors, further filtering was done to include the six companies within each sector with the most amount of news articles. It was done to have as large a data foundation as possible. The categorization of stocks into each sector is in line with the Industry Classification Benchmark (ICB). For the Finance sector the following companies were included: Bank of America, BlackRock, Citigroup, Goldman Sachs, JP Morgan Chase & Co, and Wells Fargo. The Technology sector consisted of these companies: Alphabet, Amazon, Apple, Meta, Microsoft, and Tesla. And for the last sector, Energy, the listed companies were chosen: British Petroleum, Chevron, Conoco Phillips, Enbridge, Exxon Mobil, and Schlumberger.

The test set is chosen to be the tail of the dataset. This is because we want the models to generalize into the future, and make sure they did not merely learn temporal or other between sample dependencies in the data set. 20% were set aside for testing.

Table 4, 5 and 6 shows descriptive statistics for the three different sectors: Technology, Finance and Energy. Some adjustments were made in terms of removing outliers to remove articles with an illogical high or low sentiment score. These articles were removed on the basis of the length of the article and exclusively positive or negative sentiment.

Table 4: Descriptive statistic of sentiment scores & stock returns Technology companies

	Mean	**Median**	**Max**	**Min**	**Std**	**Skewness**	**Kurtosis**
SentiWordNet	2.363	1.250	30	−25.250	5.570	0.940	1.860
TextBlob Polarity	0.096	0.104	0.500	−0.373	0.084	−0.279	0.558
TextBlob Subjectivity	0.465	0.463	1	0	0.088	1.653	9.290
Vader Negative	0.034	0.032	0.279	0	0.026	1.006	2.377
Vader Neutral	0.862	0.862	1	0.662	0.047	0.159	0.353
Vader Positive	0.103	0.102	0.338	0	0.038	0.053	0.805
Return	−3.64e−04	3.05e−06	0.129	−0.132	0.016	−0.351	5.368
Volatility	0.022	0.018	0.234	0.004	0.014	2.805	17.838

Table 5: Descriptive statistic of sentiment scores & stock returns Finance companies

	Mean	**Median**	**Max**	**Min**	**Std**	**Skewness**	**Kurtosis**
Senti Word Net	2.206	0.500	19.625	−9.625	3.845	1.167	2.460
TextBlob Polarity	0.071	0.068	0.366	−0.387	0.085	0.050	0.753
TextBlob Subjectivity	0.411	0.417	0.818	0	0.099	−0.405	1.104
Vader Negative	0.043	0.038	0.168	0	0.032	0.796	0.298
Vader Neutral	0.860	0.867	1	0.708	0.048	−0.089	0.108
Vader Positive	0.096	0.091	0.224	0	0.038	0.342	0.228
Return	−5.72e−07	6.9e − 04	0.064	−0.071	0.013	−0.438	2.453
Volatility	0.019	0.017	0.085	0.005	0.009	1.593	3.883

Table 6: Descriptive statistic of sentiment scores & stock returns Energy companies

	Mean	**Median**	**Max**	**Min**	**Std**	**Skewness**	**Kurtosis**
Senti Word Net	−0.490	−0.750	13.125	−10	2.767	0.986	4.192
TextBlob Polarity	0.045	0.027	0.381	−0.393	0.088	0.182	4.192
TextBlob Subjectivity	0.436	0.442	1	0.100	0.031	1.157	1.814
Vader Negative	0.040	0.034	0.173	0	0.031	1.157	1.814
Vader Neutral	0.875	0.876	1	0.731	0.043	−0.050	0.852
Vader Positive	0.085	0.081	0.262	0	0.036	0.441	1.561
Return	−0.007	9.2e−04	0.046	−0.056	0.013	−0.275	2.020
Volatility	0.019	0.016	0.069	0.004	0.011	1.536	2.850

As seen in tables 4, 5 and 6 there is not much difference between the different sectors. A notable difference can be noted in SentiWordNet for the Energy sector. The mean and median are negative values when the values are positive in both Technology and Finance. The information presented from Return gives an indicator of a small negative expected return. On all three sectors the mean and median of Return are negative. This may at first seem strange, given that the markets in the same period are largely up. The reason, however, seems to be that as our returns are only calculated for days where news articles were dropped, there is a tendency for news articles to be published more often in bad times for the companies. The volatility of all three sectors is low, almost at zero, meaning the risk of investment is low. Overall, there is a negative expected return associated with publication of news article from Nasdaq News.

4.2 Text Processing

As mentioned in section 2.3, three different types of dictionaries have been used to solve the research problem. Given the complexity in how the dictionaries are built, different methods of data filtering had to be applied.

For VADER to be as accurate as possible, there is no other filtering on the article content except the Sentiment Analysis itself. VADER has a requirement of separating by sentences, a filtering could create a problem going forward with the Sentiment Analysis. Therefore, the Sentiment Analysis from the VADER dictionary was directly used on the article content from each news article before it was trained on five different classifiers. VADER is a popular dictionary because of its easy application and the fact that it does not require much data filtering.

The same applies for SentiWordNet, no text processing was used before the Sentiment Analysis were applied on the article content. SentiWordNet has a big variety of words in its library, by filtering either with stemming or stop word removal, there could be a chance of removing some of the words contributing to the sentiment score.

With TextBlob on the other hand, filtering was executed with both stop word removal and stemming. The Sentiment Analysis from the TextBlob dictionary is more accurate after filtering. Stop word removal was implemented since most of the words in the list of stop words, are nonessential for the sentiment score of the article content. Stop word removal also reduces run time. Stemming was also applied to the article content to shorten down the number of words and to make sure the words inspected are found in the dictionary.

As appears from our review of the previous work, well illustrated in table 2 other dictionaries like Harvard IV-4, and Loughran and McDonald exists. The dictionaries chosen all showed promising results in comparable studies. After a thorough review VADER, TextBlob and SentiWordNet were found fitting for our content.

4.3 Sentiment Analysis - Dictionaries

For a dictionary-based Sentiment Analysis to be performed, dictionaries are needed. These are selected on background of the research problem and their compatibility to it. This next section will provide an insight of how these three dictionaries, TextBlob, VADER and SentiWordNet, operate and calculate the sentiment score on news article content.

4.3.1 TextBlob

TextBlob is a Python library for processing textual data. It provides a simple API for common natural language processing (NLP) tasks such as part-of-speech tagging, noun phrase extraction, Sentiment Analysis, classification, translation, and more (Loria, 2020). In this section we refer to the built in dictionary in the library.

When assessing sentiment, TextBlob uses WordNet, a lexical database of semantic relations between words to measure polarity and subjectivity in a text. The polarity score is a float within the range $[-1, 1]$. The subjectivity is a float within the range [0, 1] where 0.0 is very objective and 1.0 is very subjective. To illustrate, we take an example of the word “calm” with a polarity score of 0.3 and a subjectivity score of 0.75. The polarity score indicates a very positive sentiment in the text. The subjectivity score indicates that this word contains more personal opinion, emotion, or judgement rather than factual information. TextBlob attributes these scores by iterating through a corpus, looking for words and phrases it can assign subjectivity and polarity to. When it’s done iterating, it states the average of all the scores found within the corpus. TextBlobs sentiment module offers two Sentiment Analysis implementations. The pattern analyzer described above, which is the default option, and the Naive Bayes Analyzer. The Naive Bayes Analyzer was trained on movie reviews. The analyzer learned what words and in which context to consider as positive or negative by the connection with the review corpus and the numerical rating. Movie reviews are in general more subjective than the news articles in our dataset. As a result, the default pattern analyzer was used. The TextBlob dictionary is based on the WordNet dictionary. There are 2 917 scored words in the dictionary. Some words like "great" have more than one score as it has more than one sense. When assessing scores, the average of all the senses is used. Table 7 illustrates how TextBlob score some words.

Table 7: TextBlob-Dictionary sample word score

Word	Polarity	Subjectivity
Positive	0.227	0.545
Calm	0.30	0.75
Overwhelming	1	1
Difficult	-0.5	1
Dreadfully	-1	1

It is also worth noting that TextBlob takes into consideration negation and degree amplifiers. TextBlob handles negation by multiplying the polarity by -0.5. If something is described as "not difficult", that would mean multiplying difficult's polarity score by -0.5. The subjectivity is ignored when introducing a negation. TextBlob implements amplifiers by assigning amplifying words like "very" with an intensity score. "Very" has an intensity score of 1.3. "very difficult" means multiplying the polarity and subjectivity score with 1.3. It's important to note that both the polarity and subjectivity still does not exceed their ranges. Negation combined with modifiers is an interesting edge case: in addition to multiplying by -0.5 for the polarity, the inverse intensity of the modifier enters for both polarity and subjectivity (Loria, 2018).

4.3.2 VADER

The VADER (Valence Aware Dictionary and sEntiment Reasoner) dictionary was developed in 2014 to measure sentiment (Hutto & Gilbert, 2015). The dictionary was originally intended to analyse social media, but has shown promising results in other textual applications as well. The dictionary contains of 7 520 scored words, emoticons, initialism, and acronyms. During the development of the dictionary, each word (or acronym/emoticon) in the dictionary was scored by ten individuals, on a scale from -4 to 4. A word was only included in the dictionary if the standard deviation of the responses was less than 2.5, in which the word was given the average score of the individuals' scoring. To provide a form of intuition for the dictionary, some words are listed with their inherent score in table 8.

Table 8: VADER-Dictionary sample word score

Positive	2.50
Calm	1.30
Overwhelm	-0.70
Difficult	-1.50
Dreadfully	-2.90

The method also extracts punctuation, negation and degree modifiers in the sentence before scoring it. Each sentence is given a sentiment score, calculated by compounding the sentiment score of each word in the dictionary, adjusted according to the rules, and then normalized to be between -1 (most negative) and $+1$ (most positive). The sentiment of each news article is found by taking the average of the normalized compound scores of all the sentences in an article. VADER's Sentiment Analysis also provide the "pos", "neu", and "neg" scores. These are ratios describing how much of the text that fall in each category. They add up to 1 in total. These are the most useful metrics if you want to analyse the context and presentation of how sentiment is conveyed or embedded in rhetoric for a given sentence. These proportions do not make use of the rules that compound includes, like word order sensitivity or negotion polarity switches (risky to not risky). Promising results have been shown in recent years using the dictionary. Sazzed (2020), Y. B. Kim et al. (2016) and Stenqvist and Lönnö (2017) have shown that VADER can work well on news articles, social media, and online discussion forums.

4.3.3 SentiWordNet

SentiWordNet version 3.0 is the version used and described in our work. SentiWordNet is a lexical resource which uses sets of synsets instead of individual terms (Esuli & Sebastiani, 2006). Synsets are a set of synonyms that share a common meaning. The reasoning for doing this in favour of the individual term approach used by VADER and TextBlob is that different senses of the same term may have different opinion-related properties. SentiWordNet is believed to be a dictionary of general application. The scores assigned have shown to be useful regardless of the domain the text is retrieved from (Husnain et al., 2019).

SentiWordNet is built on the WordNet dictionary. WordNet is a large lexical resource with information about nouns, verbs, adjectives, and adverbs. Each of the unique words in WordNet has its own entry in WordNet. The entry contains one or more senses or meanings of the word. For example, the word mask is both a noun and a verb. As a result, mask has different meaning depending on the context. SentiWordNet finds sentiment by assigning three numerical scores: objectivity, positivity, and negativity – to each synset of WordNet. These scores describe how Objective, Positive, and Negative the terms contained in the synset are.

SentiWordNet works based on the quantitative analysis of the glosses associated to synsets. Glosses being the explanation of a word or expression. SentiWordNet then uses these term representations for semi-supervised synset classification. Some were manually labelled, and the classifier learned from these labels and classified the remaining terms. The three scores are derived by combining the results produced by a committee of eight ternary classifiers, all characterized by similar accuracy levels but different classification behaviour (Esuli & Sebastiani, 2007).

SentiWordNet will give each of the three notions: positivity, negativity and objectivity a score ranging from 0 to 1. The sum of these scores is always 1. How these scores and the synsets works is illustrated in table 9.

Table 9: SentiWordNet-Dictionary sample word score

Positivity	Objectivity	Negativity	Text
0.75	0.25	0	good#1
0	1	0	trade_good#1 good#4 commodity#1
0.50	0.25	0.25	Hopeful#1
0	0.65	0.35	promising#2 hopeful#2 bright#10

The word “good” is one of the most ambiguous words in the dictionary. It has 21 different scoring categories. Good#1 in the table is the adjective form of the word and refers to having desirable or positive qualities especially those suitable for a thing specified. For example, “She has a good report card”. Good#4 on the other hand is the noun version of the word and is a synset of trade good or and commodity and refers to articles of commerce. As illustrated, these two categories of good gives two completely different scores. Good#1 scores 0.75 positivity and 0.25 objectivity, while good#4 is completely objective. Also, hopeful#1 and good#1 does not have any synsets. This means that when SentiWordNet identifies these words in their respective category (#1), there are no other synonyms that could give the same understanding of the sentence.

4.3.4 Dictionary Comparison

Table 10: Sample sentences and their score, based on dictionary

Sentences	VADER				TextBlob		SentiWordNet		
	Compound	Negative	Neutral	Positive	Subjectivity	Polarity	Negative	Positive	Senti_Score
Amazon recalls power banks due to fire, chemical burn hazards.	-0.178	0.198	0.661	0.140	0.375	-0.125	0.250	0	-0.250
Snap shares plunge on weak results.	-0.178	0.319	0.440	0.242	0.624	-0.375	0.613	0.375	-0.238
Rolls-Royce CEO cuts 4.600 jobs to boost profitability.	0.382	0.183	0.417	0.400	0	0	0.125	0.875	0.750
New York City suing BP for deepwater horizon oil spill shares slightly lower.	-0.204	0.247	0.617	0.136	0.311	-0.015	0.250	0	-0.250

Table 10 illustrates four headlines from our dataset and how the three chosen dictionaries score them. The headline "Rolls-Royce CEO cuts 4.600 jobs to boost profitability" is not given a score when using the TextBlob dictionary. This is because there are no words in the sentence that is found in the TextBlob dictionary. As there are only 2 917 scored words, there are bound to be sentences like this without a sentiment score. For the other sentences there are a pattern between the dictionaries, they are in the same range when classifying the sentence as more positive or negative.

4.4 Data Preparation

The classification of stock movement direction is trained differently in the three different dictionaries. The common factor for all three is that the response variable was the direction of the stock movements. A dummy variable was created to serve as the response variable, showing 1 if stock price increased and 0 if it fell. The TextBlob dictionary used training on the independent variables, subjectivity, and polarity. The SentiWordNet used only a single sentiment score as the independent variable for training. The VADER on the other hand, used four different independent variables: compound, negative, neutral, and positive.

After text processing and sentiment extraction, the datasets were split into training and test sets, where the training set contained 80% of the content and the test set 20%. This was applied as the content for all the classifiers in 4.5, except for KNN. For KNN to be optimized as a classifier, a validation set were necessary to include. The validation set contained 10% of the content, while the training and test set had 80% and 10% respectively. This was done to prevent overfitting the model.

4.5 Classifiers

Classifiers in machine learning is a type of algorithm used to create a class label based on data input. In this thesis, we have used the classifiers to train predictive models for stock price movement based on Sentiment Analysis on news articles. To best classify the stock price influence based on news articles, five different classifiers have been tested based on similar research: Linear Discriminant Analysis (LDA), Naive Bayes, K-Nearest Neighbours (KNN), Support Vector Machine (SVM), and Decision Tree. Pythons Scikit-learn library was used to perform the classifications.

4.5.1 Linear Discriminant Analysis

LDA or Linear Discriminant Analysis is a linear model for classification and is used as a tool for statistics and data visualization. The analysis also reduces dimensionality to make the result easier to interpret and present (Mehta, 2019).

There are mainly three steps to calculate and preform LDA. The first one is to calculate the between-class variance, which is the separability between classes (Mehta, 2019). Equation 1, shows how to calculate the between-class variance:

$$S_b = \sum_{i=1}^{g} N_i(\bar{X}_i - \bar{X})(\bar{X}_i - \bar{X})^T \tag{1}$$

where $\bar{X}i$ represents the mean of X through all observations, and $\bar{X}$ represents the mean of one observations of X in total. *g* represents the total number of classes, while *N* is the sample size of class *i*. The *T* at the end of the equation represents time.

The next step is to calculate the within-class variance, which is the space between the sample of all classes and the mean, and shown in equation 2 (Mehta, 2019).

$$S_w = \sum_{i=1}^{g} (N_i - 1)S_i = \sum_{i=1}^{g} \sum_{j=1}^{N_i} (X_{i,j} - \bar{X}_i)(X_{i,j} - \bar{X}_i)^T \tag{2}$$

The third and last step is to minimize step 2 and maximise step 1 by creating a lower dimensional space, this is called the Fisher criterion (Mehta, 2019). Equation 3 is illustrated, describing Fisher´s criterion.

$$P_{\text{LDA}} = \text{argmax}\frac{P^T S_b P}{P^T S_w P} \tag{3}$$

The Fisher criterion is a discriminant criterion and by maximizing it, there is a possibility to obtain an optimal discriminant projection axis (S. Z. Li & Jain, 2009). The P is the determinant for the two different S variables. The LDA computes the best prediction by looking at new sets of input probabilities to every class. The output generated gives the highest probability (Mehta, 2019). No changes to the default LDA algorithm in Python made by Pedregosa et al. (2011) was conducted. We wanted to include all features and the training set was sufficiently large, we did not see the need of shrinkage or feature selection.

4.5.2 Naive Bayes

Naive Bayes is a simple classification model that utilizes Bayes' rule. The classification bases the prediction on a target variable by using features. It also assumes no correlation between the features and that they are independent. In the Naive Bayes classifier all features are weighted the same and have the same contribution to the final outcome (Zhang, 2004).

Naive Bayes classifier uses Bayes theorem which is shown in equation 4:

$$P(y|X) = \frac{P(X|y)P(y)}{P(X)} \tag{4}$$

X is the input vector and y is the class variable. The theorem estimates probability of each class by the absence or presence of each value or feature at the input vector.

Gaussian Naïve Bayes is used when we assume all the continuous variables associated with each feature to be distributed according to Gaussian Distribution also called normal distribution (Zhang, 2004).

$$P(x_i|y) = \frac{1}{\sqrt{2\pi\sigma_y^2}}\text{exp}\left(-\frac{(x_i-\mu_y)^2}{2\sigma_y^2}\right) \tag{5}$$

Assuming the i^{th} attribute is continuous and its mean and variance are represented by $\mu_{y,i}$

and σ$^2_{y},_i$, respectively, given the class label y. Hence, the probability of observing the value xi in ith attribute given the class label y, is computed by equation 5. The implementation of the Gaussian Naive Bayes algorithm is simple and without options for parameter tuning in Python's Scikit-learn library.

4.5.3 K-Nearest Neighbour

K-Nearest Neighbour or KNN is a simple supervised machine learning algorithm which is easy to implement and is a useful tool to use when solving classification problems. (Harrison, 2018).

$$d(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{6}$$

Equation 6, showing the Euclidean distance, which is the most used metrics for KNN. The formula calculates the distance between two different observations, which is represented by x and y (Danielsson, 1980).

For every X, we find the K closest neighbours to X in the training set, and examine their corresponding Y. If the majority of the Y´s are X, X is predicted, otherwise Y is guessed. In our application of KNN, this is the stock price rise or fall variables categorized as 1 or 0. The disadvantage of KNN is that it can become slow if the numbers of variables or observations increases (Harrison, 2018).

Using KNN, a number of neighbours (K) needs to be selected. The best approach to achieve the most optimal value of K is to use a validation set. When choosing the K, one tests the value of K on the validation set until an increasing number of errors arise. At this point, you have pushed the value of K too far (Harrison, 2018). Several adjustments can be made to the KNN, weighting the distance for the neighbours are one example. Except from tuning the number of neighbours based on the validation set results, the default version from Pedregosa et al. (2011) in Python was applied.

4.5.4 Support Vector Machine

Support Vector Machine or SVM is a simple algorithm, mostly used with classification problems. The algorithm does not use much computing power but is still accurate (Gandhi, 2018).

SVM uses multiple hyperplanes, with the goal of finding the hyperplane with the biggest margin between the data points of both classes. This is the most optimized hyperplane. By maximizing the distance, the model gets a better prediction foundation for future data (Gandhi, 2018). The general equation form for a hyperplane is illustrated in equation 7:

$$\beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_p X_p = 0 \tag{7}$$

p describes the form of the hyperplane, for instance, when *p* = 2, the hyperplane is a line. The vectors β are called the normal vector and it points in an orthogonal direction towards the surface of a hyperplane. The only way for the hyperplane to go through the origin is when β_0 is equal to zero.

The way SVM maximizes the margin, is by using the support vectors. They are observations closer to the hyperplane and have a more direct impact on the position and orientation of the hyperplane. If the support vectors are removed, the hyperplane will change its position, creating a possibly issue for the model. The support vectors are the observations that helps create and build the Support Vector Machine (Gandhi, 2018). In real-life applications, linearly separable datasets are unlikely to come by. What we will find, is either an almost linearly separable dataset or a non-linearly separable dataset. To handle this problem, one needs an equation that allows a few misclassifications, meaning it allows some points to be wrongly classified.

$$\begin{gathered}\text{Minimize}_{w,b}\frac{1}{2}||w||^2 + C\sum_{i=1}^{m}\xi_i \\ \text{subject to } \forall_{i=1}^{m} y_i(\langle w, x_i\rangle + b) \geq 1 - \xi_i\end{gathered} \tag{8}$$

In equation 8 w is the normal vector, and b is classified as the bias. C is a positive tuning parameter, the bigger C, the more errors are accepted. Xi (ξ) shows if the observations are within the margin or misclassified. The linear Support Vector Classification (SVC) with a linear kernel, applied on the sentiment scores, is implemented with more flexibility when it comes to loss functions. It should be able to scale large numbers of samples in a better way (Pedregosa et al., 2011). The kernel setting is defined as linear. The value of the tuning parameter C defines how much misclassification is accepted from the training set. A value above 0 indicates a soft margin. The default value of C in sklearn library in Python is 1, and was used in our models. A soft margin can pick up outliers and the minimum margin is low between the classification variables. This would increase the possibility of a higher error rate, but is necessary to generate useful SVM models.

4.5.5 Decision Tree

Decision Tree is withing machine learning, often used in classification and regression problems. The model is usually used in decision making because of its visualization capabilities. The visualization looks like the branches of a tree, therefore the name (P. Gupta, 2017).

$$G = \sum_{k=1}^{K} \hat{P}_{mk}(1 - \hat{P}_{mk}) \tag{9}$$

Equation 9 is the Gini index. The index measures the total variance across the K classes. If the value of $\hat{P}_{mk}$ is one or close to zero, the Gini index takes on a small value. When the values are small, it indicates that the node mainly contains observations of a single class. Therefore, the Gini index is also referred to as a measure of purity.

The decision tree starts from a point and branches out. The length of the tree and number of branches can differ. The number of variables, features, etc. involved has an impact on the tree. One of the dangers of using to many branches is that it can lead to overfitting and the tree does not generalize the data well. On the other side, decision trees are easy to interpret (P. Gupta, 2017). The max depth function was used, both to get a feel of the data distribution, and as a tool to control the size of the tree and prevent overfitting. Overfitting can also be managed by setting a minimum sample for each “branch”. Functions can be applied to avoid bias towards the dominant class. In our case, returns were almost evenly distributed and there was little sign of a dominant class, therefore the default versions were used.

4.6 Trading Design

In our trading strategy, we have used the sentiment score of the articles and trained a predictive model using the before mentioned classifiers. The model expresses the probability of increase or decrease in the stock price for the given day. We have entered a long/short strategy, going long when our models express a rise in the price and short in the cases our models express decline in price. All trades are made, buying a companies stock at opening price and selling at closing price. Brokerage fees and liquidity problems are not accounted for in our models´ performance metrics. Stakes are calculated based on the Kelly criterion,

with staking according to our model's believed conviction of the outcomes. The Kelly criterion is in economics often referred to as the geometric mean maximizing portfolio strategy or the growth optimal strategy (Thorp, 2011).
The Kelly criterion for investment is given in equation 10:

$$K = \frac{bp - q}{b} \tag{10}$$

K is the percentage of the total assets to be invested in the security. b is the decimal odds that is always equal to 1. p is the probability of winning, and q is the probability of losing (1-p). For instance, if our models predict the chance of an increase in the stock price, p is 52%, that will result in a stake equal to 4% of the total capital. In our models, the p is given by the trained models. All models give a probability for increase or decrease. We go long when an increase is predicted and short when a decrease is predicted.

There are some drawbacks using the Kelly Criterion as the foundation of a staking plan. It assumes that the parameters, p, and q are calculated correctly, which on could argue is not possible to achieve due to the complexity of predicting the future and especially the future of stocks. Attributing a to high p could lead to a large portion of one's portfolio being at stake at once. If the probability then is miscalculated, the risk of losing a large portion of one's capital due to flaws in the staking model is definitely present. Even proponents of this staking model say, one should be careful, not putting all the eggs in one basket. Hence, methods like "Half-Kelly" or "Quarter-Kelly" are used. This is essentially just a change in the b value. According to Thorp (2011) critics says the Kelly Criterion should only be used in combination with human intuition. This could limit the potential for losing the capital. Obviously, it could just as well limit the potential gains. Empirically, the Kelly-Criterion is well tested and works well on long time-horizons. In the short run, however, the variance can be unpleasant for the investor.

5 Results

In this section, the performance comparison between the dictionaries is shown in table 11. The classification report metrics are presented for all models. All models are combinations of the sentiment extracted from the different dictionaries and classifiers. There are 15 different models for each of the three sectors. Further, the models with the greatest F1 score from each sector were included in a trading strategy. The models with positive returns are also evaluated and compared using the Sharpe ratio.

5.1 Dictionary Performance Comparison

Table 11: Dictionary Regression where Return is the dependent variable. The symbol *, denote the significance at the 5% level

Sectors	**Technology**	**Energy**	**Finance**
	VADER		
Intercept	2.089*	3.440	−0.335
Compound	−0.0006	−0.0020	−0.0020
Negative	−2.075*	−3.520	0.277
Neutral	−2.089*	−3.440	0.337
Positive	−2.088*	−3.400	0.371
R^2	0.005	0.048	0.027
Adjusted R^2	0.001	0.011	0.006
	TextBlob		
Intercept	−0.0039	0.0088	−0.0015
Polarity	−0.00032	−0.02600	0.00870
Subjectivity	0.0096	−0.0240	0.0061
R^2	0.0018	0.0390	0.0097
Adjusted R^2	0.00005	0.02000	−0.00120
	SentiWordNet		
Intercept	0.00047	−0.00220	0.00100
Senti_score	−0.000012	0.000026	0.000092*
R^2	0.000049	0.000530	0.024000
Adjusted R^2	−0.00083	−0.00910	0.01800

Table 11 illustrates the values for the three different dictionaries with return as the dependent variable. It shows every variable affects the intraday return of the stock prices. The adjusted R^2 shows how much variance in the dependent variable is explained by the independent variables, and the results illustrated are generally low. This indicates that the sentiment derive from the dictionary approach and does not explain the return well. Low R-values are not inherently bad, especially if significant variables exist. Low R-values and significant variables indicate that the independent variables are correlated with the dependent variable, but they do not explain much of the variability in the dependent variable. Table 11 show the variables significant at the 5% level. Using the VADER dictionary on the Technology sector news , we observe the variables from the sentiment extraction: negative, positive, and neutral are significant. This means they correlate with the dependent variable, return. The same goes for the variable senti_score from SentiWordNet when applied on Finance sector news. The combination of low R values and significant independent variables indicate that the variables are correlated with the dependent variable, return. However, it does not explain much of the variance in the stock price returns.

5.2 Model Evaluation

When combining the sentiment from the dictionaries with the five different machine learning classifiers, we get an insight of what models performed the best on basis of the test set. There are some similarities between the different dictionaries combined with different classifiers. The same classifier used on the sentiment from different dictionaries in some cases show vastly different results. The results are presented in three tables, the Energy sector, Finance sector and Technology sector. The results presented are a visualisation of the four parameters: accuracy, precision, recall and F1 score. The first parameter, accuracy, is a measurement of how close or far away a given set of observations is to its true value. Precision is how close or scattered the observations are to each other. Recall is the proportion of relevant observation that were retrieved from the analysis. The last feature is the F1 score, and it is the average of the recall and precision. If two classifiers differ in value of precision and recall, where one has a high precision and one has a high recall, one can use the F1 score to decide which classifier produces the best result.

Table 12: Classification report metrics Energy. Summary of accuracy, precision, recall and F1 score for all combinations of dictionaries and machine learning classifiers

	Accuracy	Precision	Recall	F1 score
LDA Senti-WordNet	**0.63**	0.59	**0.63**	0.51
LDA TextBlob	0.59	0.49	0.59	0.50
LDA VADER	0.53	0.43	0.53	0.46
Bayes Senti Word Net	0.58	0.47	0.58	0.49
Bayes TextBlob	0.57	0.49	0.57	0.50
Bayes VADER	0.57	0.49	0.57	0.50
KNN Senti Word Net	0.59	0.58	0.59	0.59
KNN Textblob	0.58	0.57	0.58	0.57
KNN VADER	0.54	0.54	0.54	0.54
SVM Senti Word Net	0.62	0.56	0.62	0.53
SVM TextBlob	0.59	0.49	0.59	0.50
SVM VADER	0.61	0.40	0.61	0.48
Decision Tree Senti Word Net	0.61	0.49	0.61	0.50
Decision Tree TextBlob	0.62	**0.60**	0.62	**0.60**
Decision Tree VADER	0.42	0.43	0.42	0.42

As shown in table 12 regarding the Energy sector, two combinations of classifiers and dictionaries are noticeable. These models are LDA–SentiWordNet and Decision Tree–TextBlob. There are very few parameters in the classification report metrics for the Energy sector below 0.50, which indicates most of the models have a decent ability to predict the stock price movements. Based on the F1 scores, the KNN–SentiWordNet, KNN–VADER, and Decision Tree–TextBlob models were selected to be part of a trading strategy.

Table 13: Classification report metrics Finance. Summary of accuracy, precision, recall and F1 score for all combinations of dictionaries and machine learning classifiers

	Accuracy	Precision	Recall	F1 score
LDA Senti-WordNet	**0.60**	0.36	0.60	0.45
LDA TextBlob	0.60	0.36	0.60	0.45
LDA VADER	0.47	0.49	0.47	0.48
Bayes Senti Word Net	**0.60**	0.38	**0.60**	0.45
Bayes TextBlob	0.53	0.44	0.53	0.46
Bayes VADER	0.51	0.52	0.51	0.51
KNN Senti Word Net	0.53	0.56	0.53	0.54
KNN Textblob	0.49	0.51	0.49	0.50
KNN VADER	0.58	**0.60**	0.58	**0.58**
SVM Senti Word Net	**0.60**	0.36	**0.60**	0.45
SVM TextBlob	**0.60**	0.36	**0.60**	0.45
SVM Vader	0.52	0.50	0.52	0.51
Decision Tree Senti Word Net	0.57	0.41	0.57	0.45
Decision Tree TextBlob	0.46	0.48	0.46	0.47
Decision Tree VADER	0.53	0.58	0.53	0.53

For the Finance sector shown in table 13 three models stood out. These three, based on F1 score, were VADER, TextBlob and SentiWordNet all combined with KNN. These were also included in the trading strategy. Slightly lower scores in the Finance sector then in the Energy sector, which indicates that the Finance sector is marginally harder to predict than the Energy sector using our models.

Table 14: Classification report metrics Technology. Summary of accuracy, precision, recall and F1 score for all combinations of dictionaries and machine learning classifiers

	Accuracy	Precision	Recall	F1 score
LDA SentiWordNet	0.51	0.51	0.51	0.51
LDA TextBlob	0.49	0.49	0.49	0.49
LDA VADER	0.49	0.49	0.49	0.49
Bayes Senti Word Net	0.51	0.51	0.51	0.48
Bayes TextBlob	0.50	0.49	0.50	0.48
Bayes VADER	0.49	0.49	0.49	0.47
KNN Senti Word Net	0.49	0.49	0.49	0.49
KNN Textblob	0.50	0.50	0.50	0.50
KNN VADER	**0.52**	**0.53**	**0.52**	**0.52**
SVM Senti Word Net	0.51	0.51	0.51	0.51
SVM TextBlob	0.49	0.49	0.49	0.49
SVM VADER	0.49	0.49	0.49	0.49
Decision Tree Senti Word Net	0.49	0.47	0.49	0.43
Decision Tree TextBlob	0.49	0.48	0.49	0.43
Decision Tree VADER	0.49	0.49	0.49	0.46

For the Technology sector, table 14 shows most scores around and under 0.50 which should be considered as unsatisfactory. Put simply, this means most of the models applied in the Technology sector is no better than pure chance, which should give 0.50 scores. Nonetheless, the best F1 scores from Technology, were also included in the trading strategy, these were VADER combined with KNN, TextBlob combined with KNN and SentiWordNet combined with SVM.

5.3 Trading

The models inducted in our trading strategy vary in terms of conviction. The KNN model, was the model that in general had the largest spread between the probability of price rise or price fall. It resulted in a larger average stake for trades made on the bases of models with this classifier. For models using Naive Bayes, the chance of increase or decrease were often similar, with only a few percentage points difference. This in turn lead to a smaller average stake in the trades. This should be kept in mind when comparing trading results. Measures were considered to deal with these unbalances, but we deemed it fair to keep the same staking method for all models, even though the variance of the KNN trading strategy obviously is larger than the others. To account for the additional risk from staking larger parts of once bankroll, one could look at the Sharpe ratio of the trading strategy for a more risk-adjusted comparison. We have selected the models achieving the best scores at each sector to examine further in our trading strategy.

Table 15: Trading metrics for the Technology companies

Model	Start Capital	Gross Profit	Trades	Long Trades (won)	Short Trades (Won)	Correct predictions	Drawdown	Turnover	ROI	Annualized Returns
VADER-KNN	$100	$14.20	1142	539 (54%)	603 (52%)	53.70%	1.77	163.17	0.09%	4.56%
Textblob-KNN	$100	$−9.10	1142	577 (50.7%)	565 (48.80%)	49.90%	10.61	175.13	−0.05%	−2.92%
SentiWordNet-SVM	$100	$−0.51	1142	529 (52%)	612 (50.20%)	51%	0.51	12.43	−0.04%	−0.16%

Table 16: Trading metrics for the Finance companies

Model	Start Capital	Gross Profit	Trades	Long Trades (won)	Short Trades (Won)	Correct predictions	Drawdown	Turnover	ROI	Annualized Returns
VADER-KNN	$100	$0.09	184	91 (62%)	93 (42%)	52%	0.41	24.47	0.0040%	0.135%
Textblob-KNN	$100	$1.98	184	94 (60%)	90 (39%)	49.50%	0.70	33.78	0.0059%	2.970%
SentiWordNet-KNN	$100	$5.9	184	87 (64%)	97 (43%)	53.20%	0.06	31.08	0.1900%	8.850%

Table 17: Trading metrics for the Energy companies

Model	Start Capital	Gross Profit	Trades	Long Trades (won)	Short Trades (Won)	Correct predictions	Drawdown	Turnover	ROI	Annualized Returns
VADER-KNN	\$100	$\$-0.89$	106	47 (38.30%)	59 (64%)	52.80%	2.054	26.50	-0.034%	0.97%
Textblob-Decision Tree	\$100	\$5.33	106	27 (48%)	70 (67%)	62%	0.120	11.50	0.460%	5.81%
SentiWordNet-KNN	\$100	\$6.30	106	34 (44%)	79 (66.60%)	59.40%	0	23.80	0.265%	6.87%

Table 15, 16, and 17 show the trading metrics for the three sectors. What metrics to attribute the largest amount of emphasises on will naturally depend on an investors risk profile. For this reason, nine trading metrics are included. Annualized returns are calculated using the arithmetic average. We do acknowledge that the compound effect can cause small differences when using the geometric average. In relation to the three dictionaries, it's interesting that VADER was the only dictionary with somewhat successful results in the Technology sector. Yet it was clearly worse when applied on the Finance and Energy sector.

For the Energy sector, the short trades were considerably more successful than the long positions. The Technology sector, table 14, had only small differences in success rates for long and short trades. For the Finance sector, in table 13, the win ratio of long trades is greater with success rates reaching 64% for the SentiWordNet–KNN model. As seen in table 12, the opposite is the case. Short trades are much more successful with success rates up to 67% for the TextBlob–Decision Tree model. This should be seen in context with the differences in general behaviour of the stocks within the three sectors. During the period examined, the basket of technology stocks, have all shown considerable growth. Some of the finance stocks and most of the energy stocks, saw no growth over the period that makes up our training sets.

The highest ROI (Return on Investment) was found in the Energy sector, using the TextBlob–KNN model. An expected return on 0.46% on each trade was the highest across all trading-strategies. The sustainability of this performance can be questioned, as the Energy sector was only tested on 106 trades. The reason being that there were not as many articles published about these companies, as compared with the technology companies. The turnover–how many times over one puts the initial capital to work, was also low on this strategy. This indicates that the conviction the model had for each trade were modest, with mostly just a few percentages in favour of one of the outcomes (long or short).

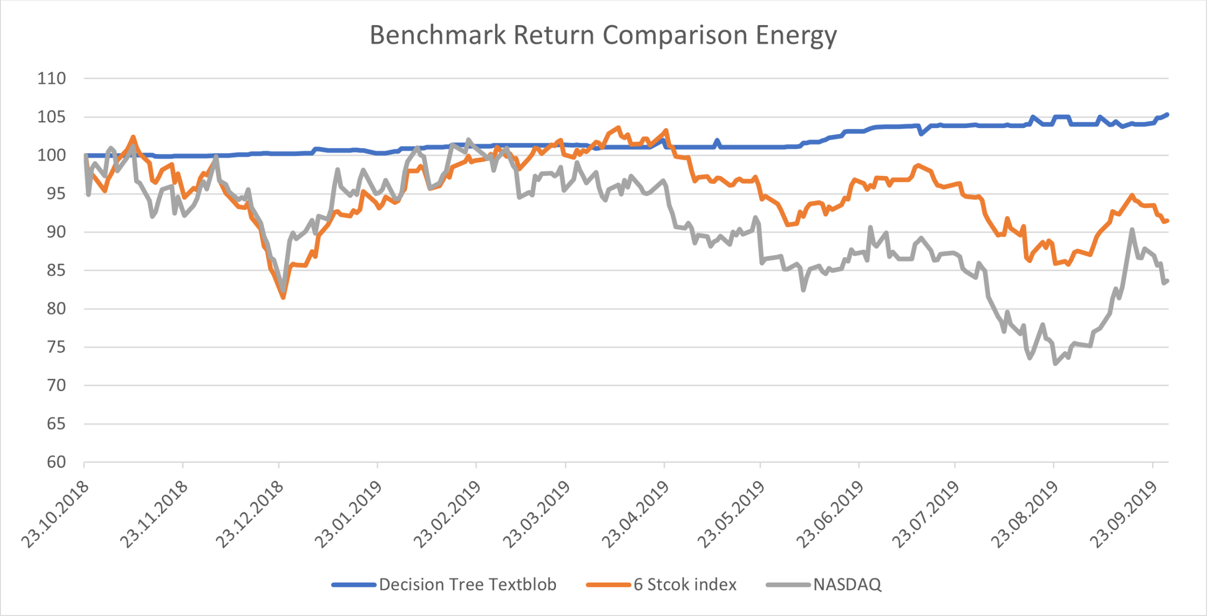

Figure 2: Comparing energy trade results, 6 stock sector index (British Petroleum, Chevron, ConocoPhilips, Enbridge, Exxon Mobile, and Sclumberger) & Index benchmark (Nasdaq)

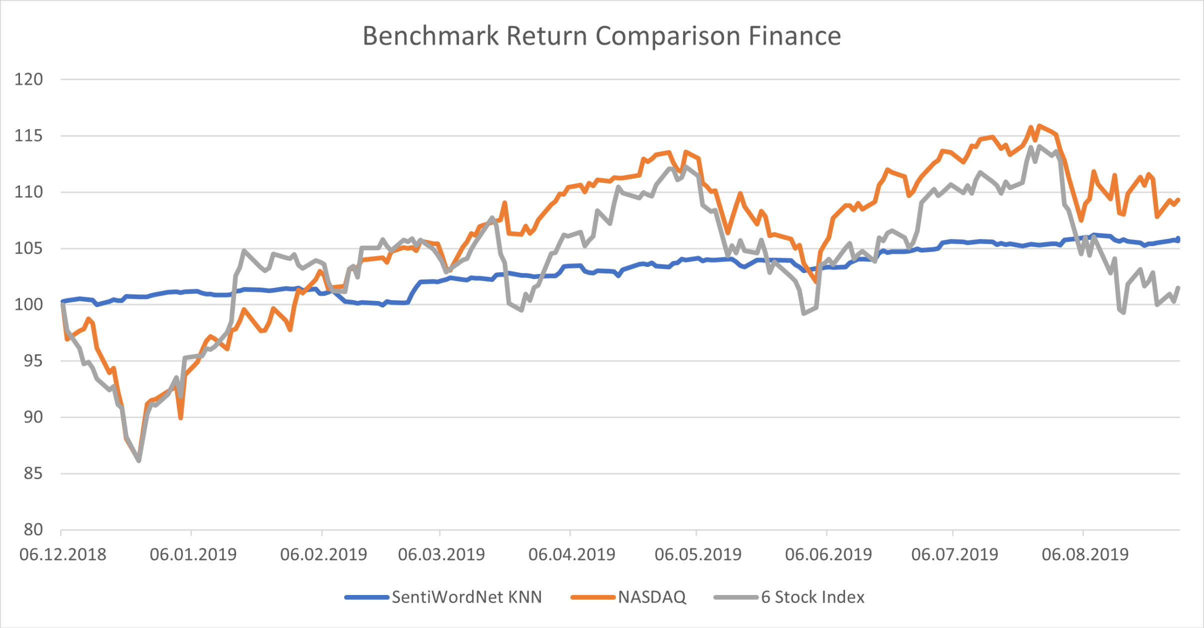

Figure 3: Comparing finance trade results, 6 stock sector index (Bank of America, Blackrock, Citigroup, Goldman Sachs, JP Morgan Chase & Co, and Wells Fargo) & Index benchmark (Nasdaq)

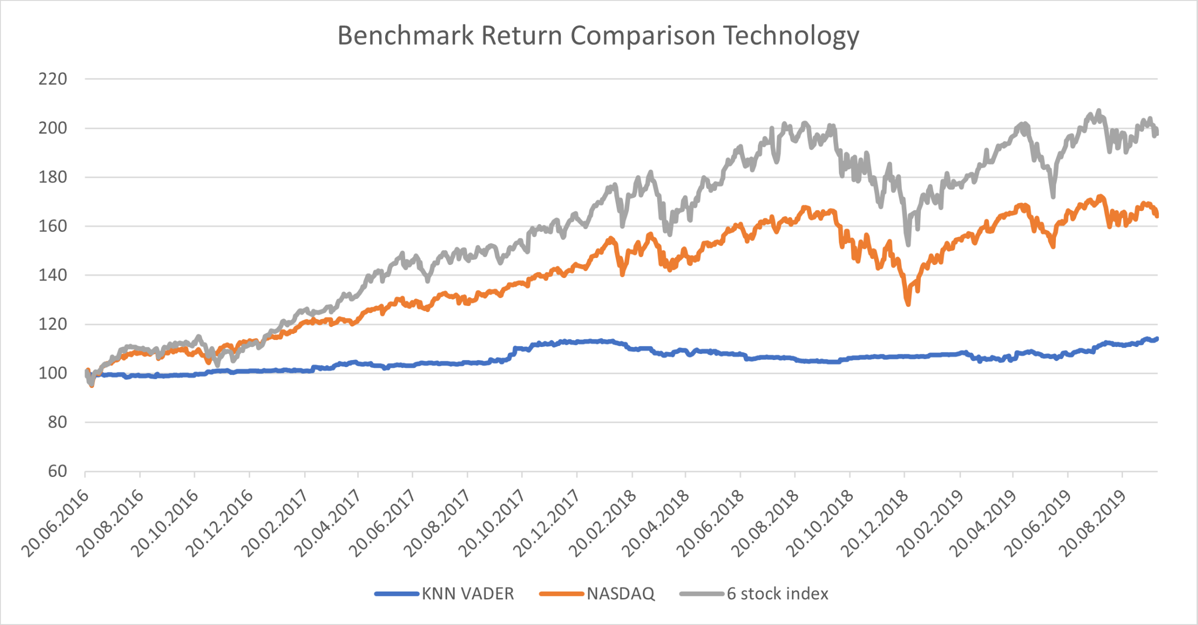

Figure 4: Comparing technology trade results, 6 stock sector index (Alphabet, Amazon, Apple, Meta, Microsoft, and Tesla) & Index benchmark (Nasdaq

Figure 2, 3 and 4 shows the return for the best models based on ROI from the trading strategy, compared to the Nasdaq index and the six stocks constituting the selected sector (also made into an index with equal weight) over the test set period. The test period for the Technology sector stretches over a longer time-period than the two other sectors. The reason being, that the number of articles written on the Technology companies skewed towards the earlier parts of the time-period. Comparing the figures should consequently be conducted with this in mind. Figure 2 outperform both the Nasdaq index and the sector index over the time-period. The Nasdaq and sector index were down in the period simultaneously. The Decision Tree–TextBlob model was slightly up, suggesting it is uncorrelated with the indexes. In Figure 3 the SentiWordNet–KNN model outperform the sector index, but underperform the Nasdaq index over the period. It does so with considerably lower volatility of the returns. The Technology sector, in Figure 4, underperforms both indexes over the time-period. This is also true when evaluating the shorter time-period equalling the period of the Energy and Finance sector.

The Sharpe ratio is a good measurement to evaluate the risk adjusted return for our trading models.

$$Sharpe - ratio = \frac{R_x - R_F}{\sigma_x} \tag{11}$$

In equation 11, $\mathrm{R_x}$ is the return of the investment, $\mathrm{R_F}$ is the risk-free rate and the square root of the returns variance, σ_x is the standard deviation of the returns. $\mathrm{R_F}$ was selected as the average from the 10 Year US Treasury Rate over the test periods.

Table 18: Selected models Sharpe ratio

	Sharpe Ratio
Technology VADER KNN	0.444
Finance SentiWordNet KNN	2.929
Finance TextBlob KNN	0.268
Energy TextBlob Decision Tree	2.030
Energy SentiWordNet KNN	1.643

The Sharpe ratio is a measure of the risk adjusted returns. As shown in table 18, the Sharpe ratio are in a rather large range of values. When the Sharpe ratio value is below zero it does not represent any meaningful significance and is useless to investigate. Therefore, only the models that generated positive Sharpe ratio were included in the table 18. For Finance only SentiWordNet–KNN were included. In the Energy sector TextBlob–Decision

Tree and SentiWordNet–KNN were inspected. All the scores are over 1 and two of them over 2, with one almost reaching the score of 3. These scores, if sustainable over time, are good and indicates an excess return relative to the volatility for the selected models. The Technology sector KNN–VADER and Finance sector TextBlob–KNN has scores below 1 and close to 0. These values are normally below the market performance and would most likely not yield risk adjusted return in the long run. From a trading point of view, it could be more appealing to include models with high Sharpe ratios.

6 Discussion

This chapter is arranged thematically answering the research questions raised. Firstly, market efficiency implication are discussed, following this, dictionaries implication, classifier implication and trading implication are considered.

6.1 Market Efficiency Implication

In this section we will try to elaborate sub-research question 1: *"Does the EMH make it unviable to achieve abnormal returns for dictionary-based Sentiment Analysis?"*

Our results show that the majority of our models were unable to use the sentiment in news articles posted on Nasdaq News to achieve investment returns superior to those of the market. This supports the strongest form of the EMH presented by Fama (1965). This form of EMH claims that models such as ours should not be able to make use of the information, as all information both public and non-public are already priced into the market.

If one were to apply aspects from the EMH and try to explain why our models were mostly unsuccessful, two key features are central to consider: the quality of the information, and the absorption of the information. As Nasdaq News publicises articles from many news providers, the impact and influence of various news providers is also crucial for the reception of the news. After all, Day and Lee (2016) found significant dissimilarities in the level of influence between the different news providers. The argument that the news are already priced in, is perhaps more relevant as only news articles posted outside the markets opening hours were included. Several studies, one of which, Dodevska et al. (2019) states that the time the market needs to absorb the news is so short that our models do not pick up the market's reaction to the news at all.

The evidence against the strongest form of EMH is not fully conclusive. An argument is to be made against it, as some models were able to perform better than the comparable indexes in times of stock price decline. This occurred in the Energy sector – a sector that over the entire timeframe of our included data saw prices decline slightly. This could indicate that our models are better at predicting declines rather than increases in price. Bird and Yeung (2012) find good support for their hypothesis that investors faced with uncertainty, react differently to positive and negative news. Investors tend to react more irrationally when faced with negative news, leading to inefficiencies in the prices following a negative news event. This contradicts the strong form for market efficiency and lends support to the weak form. The weak form suggests that our models can obtain an information advantage over the general market, thus making it possible to use the news sentiment to earn abnormal returns in down trending markets.

Further investigation into periods of low stock price returns for the different sectors were considered, as this is where our models performed best compared to the market. Nonetheless, cherry picking data from these periods were not a fallacy we wanted to commit. In addition to this, limitations in our article dataset would have made this unviable. The reason being a shortage in company specific articles in periods of down trending stock prices for the Technology and Finance sector.

Other studies like Bing et al. (2014) and Coval et al. (2005) supports the theory that there is a window for exploiting such inefficiencies in the market pricing. This window of opportunity applies to both a down trending period for stock prices, and the window after a news drop until the news is fully priced in. Others like Philippas (2014) claim it is so difficult to know whether the window is open at the time one would like to make the investments, that it is not even worth trying to exploit it. These windows of opportunities can often seem obvious in retrospect, but are easily missed when living them.

Overall, our findings do not coincide with most of the included literature on this field presented in section 2.4. Most dictionary-based methods presented in table 2, as well as non-dictionary-based methods in table 3, argue firmly against the strong form for market efficiency in the markets they examined. The study by C. Wang and Luo (2021) is an example of similar research whose results were more in line with ours. Our method does not conclusively support the strong form for EMH, neither should the arguments against it be considered robust enough to argue the opposite. Our works' contribution in terms of furthering the EMH, is to suggest that dictionary-based approaches of Sentiment Analysis used to improve stock price predictions based on Nasdaq News, does not conclusively contradict even the strongest form for EMH.

6.2 Dictionaries Implication

In this section we will try to elaborate sub-research question 2: *"How does the choice of sentiment dictionary affect prediction of stock price movement?"*

VADER, TextBlob and SentiWordNet have all previously shown encouraging results in finance applications. As the dictionaries are an important component in understanding the output of our research's results, addressing why they provide different outcomes is useful. As shown in table 10, the Sentiment Analysis provided by the dictionaries are unable the explain much of the variance in the returns. This is in line with Gobel and Zinfrazi (2019) who found that dictionaries were not good at predicting price movements, but more suitable for predicting trading volume and volatility. The low scores of R^2 and Adjusted R^2 could also indicate that the positive returns from the trading strategies may have been achieved due to randomness. However, it could be the case that the classifiers used to train the models identify important patterns in the sentiment scoring values and returns. Consecutively leading to higher prediction accuracy and sustainable trading strategies. Nonetheless, as trading application are a key feature of our thesis, it is only natural that the dictionaries' trading results will be the basis of discussion in this section.

The VADER dictionary had the most deviating results when applied in our trading strategies. As illustrated in table 15, it was the only dictionary able to make a profit on the Technology sector. Given the fact that the dictionary was trained on social media posts, it could be that words associated with popular companies on social media are better represented within the dictionary. The large technology companies seem to get a disproportionately large part of the traction in investment discussions on Twitter (Bing et al., 2014). The jargon used to describe events concerning these companies also seem to deviate from other sectors. These observations backed by our models' result could indicate that the structural training design of VADER has given it an advantage over the other two dictionaries when addressing sentiment on the technology companies.

TextBlob offer relatively good trading metrics in the Finance and Energy sector. The results are perhaps surprisingly good, considering that TextBlob is arguable the simplest dictionary out of the three, with under 3 000 scored words. News articles concerning the sectors, to a larger extend stems from more conservative news providers like Reuters and Bloomberg. High level analysis of the vocabulary differences of the news providers suggests that these use less colourful, more professional language. It seems TextBlob is able to capture the sentiment in these articles well. The previous research making use of TextBlob is ambiguous in regards of where its application is best suited. Biswas et al. (2020) and I. Gupta et al. (2022) argues that TextBlob is a great choice for more colourful language like that of social media. Nemes and Kiss (2021) found it effective within business news, pointing out its structure was a good base for news articles and headlines. Our findings lend more support to the latter's way of applying TextBlob.

SentiWordNet showed the best overall Sharp ratios and annualised returns in our trading strategies. This was not surprising given the convincing results from earlier studies and the fact it is the most advanced of the three dictionaries. It was, however, surprising that we were not able to find a profitable trading strategy using SentiWordNet analysing the technology articles. After all, Bing et al. (2014) and Deng et al. (2011) was able to achieve this, analysing mostly technology companies. It could simply be that the companies constituting the Technology sector are priced more efficiently now than when the other research was conducted.

Overall, the three dictionaries all have their pros and cons. If our work can serve as an indication of which dictionary to choose when mining for sentiment, there is no absolute recommendation. It is important to be aware of the characteristics of the news one is trying to examine. As a general choice, the SentiWordNet dictionary seems the safest. It can show to attractive Sharpe ratios in both the Energy and Finance sector. If speed is of importance for investors, it should be noted that SentiWordNet uses significantly more compute due to its complexity. Our research strongly indicates that there are great payoffs to being considerate in picking a dictionary suitable for one's target corpus. Every dictionary has their fields of expertise. The choice of dictionary should be evaluated on basis of the content. For our application on articles from Nasdaq News, there were significant differences, and SentiWordNet clearly transcended the others. Everything else held equal, the choice of dictionary can as shown by our results be the difference from a profitable and non-profitable trading strategy.

6.3 Classifiers Implication

In this section we will try to elaborate sub-research question 3: *"How does the choice of classifier affect dictionary-based stock movement prediction?"*

As showed in chapter 5, the results from the different models utilizing various classifiers were similar in terms of the classification report metrics. However, KNN and Decision Tree differ from the other three classifier in an interesting way. Decision Tree and KNN express more conviction in regards of the likelihood of rise or fall than the other three classifiers. KNN and Decision Tree generally performed best across sectors in term of F1 score.

What classifier to choose to predict stock movement is debated in the literature. A. Khedr et al. (2017) used different types of classifiers on three companies to predict stock market behaviour. They considered KNN as the best approach to the problem, achieving the highest accuracy in the comparable literature. Bing et al. (2014) applied SVM, with good results. Gidofalvi and Elkan (2001) did the same using the Naive Bayes classifier. Our results had the overall best scores when the KNN–Decision Tree model were applied. This lends support to A. E. Khedr, Yaseen, et al. (2017), supporting their belief that the complexity of the sentiment variables are better accounted for using more complex classifiers like KNN and Decision Tree. As the F1 score is usually highest for the KNN or Decision Tree classifier, there is good basis to support these classifiers over the SVM, Naive Bayes and LDA. Interestingly, SVM, Naive Bayes and LDA to a large extent agrees with one another in the direction they are predicting, resulting in similar results. KNN and Decision Tree, tends to interpret the sentiment variables differently in terms of confidence of the stock direction prediction.

Why the KNN and partially the Decision Tree classifiers yielded the best results should be due to their ability to recognise patterns. In terms of the of the news articles, this could be that articles with a certain composition of sentiment are likely to influence the stock price in one direction. Such patterns are harder to recognize for the linear classifiers: LDA, SVM and Naive Bayes. Making the models work better by tuning the parameters of the classifiers would obviously improve the models' accuracy. For the KNN classifier – our best performing classifier overall, further improvement could possibly be achieved by changing the weighting of the nearest neighbours. In our models, we have a uniform approach, meaning all neighbours are weighted equal. However, it could be the case that weighting the distance from the observation makes sense. Given the nature of the way dictionary scores sentiment, this approach could produce more accurate and better findings. As with KNN, SVM could also be adjusted by a validation set. We decided the default option for SVM was fitting for the dataset and found it unnecessary to use a validation set before applying the classifier on the test set. Ensemble methods, the action of combining several weak models with each other to produce better results is an option. This has been a popular way to go and is usually the winning solution in various data science competitions (Bojer & Meldgaard, 2021). Obviously, the choice of classifier influence our predictions, as all our models are tested on the same sentiment scores. Generally, Decision Tree and KNN were the best classifiers over the sectors, this could be because of their non-linear features.

6.4 Trading Implication

Results from our trading strategy show that there may not be a way to get abnormal returns when using only our suggested models and strategies in the Finance, Technology or Energy sector. Our returns are over the researched periods lower for both Finance and Technology sector compared to the Nasdaq Index. However, we do manage to find models that return a positive ROI. This could indicate that if we were to include more than the six companies in each sector into our sentiment mining and prediction, we could see more profitable trades included in our trading implementation. The previous literature on the field, however, tend to show diminishing returns for models as the number of included companies increase.

All the trading results show that to achieve an annualized return close to what one could naturally compare to, like the Nasdaq or S&P 500 Index, is hard to achieve by only acting on the trading signals and staking strategy given by our models. The most fair comparison for the trading strategies and a comparable index would be to compare the annualized returns. In our models, the trading models yield relatively modest annualized returns across all sectors. For Technology the best model, VADER–KNN, had an annualized return of 4.56%. In the Finance sector, the SentiWordNet–KNN model produced the best annualized return of 8.850%. For the last sector, Energy, the SentiWordNet–KNN model generated the best annualized return value of 6.87%. In a profitable trading strategy, annualized returns can easily be compensated for by increasing or decreasing the leveraged capital. Therefore, the Sharpe ratio reflects the strength of our models better. It can be argued that these findings could be best fitted for signals to try to implement in more sophisticated trading techniques, as these methods alone would not be sufficient to beat the indexes.

Low drawdown across all models, makes it plausible to discuss even larger staking sizes per trade. Obviously, this would give the same ROI as before, but the Turnover and annualized returns would increase. After all, that is one of the main arguments for choosing trading in favour of a passive investment style. Large turnover can cause even small positive returns per trade to be significant over time.

Our findings could potentially serve as a part of an investors larger trading strategy, where the sentiment methods serve as a trading signal. It could be combined with other potential profitable edges in the market. The research suggests that when one combines the dictionary-based approach with other classification techniques, the accuracy increases. The same could be true if one were to combine the trading strategies. Say you traded only when both our model and the other model agreed. This was not tested by us, so the suggestion only serves as a string of thought following similar conclusion in other papers. It is also worth noting that there are always uncertainties associated with making predictions of the future with historical data. Additionally, transaction costs and liquidity were not accounted for. Even a great predictive model can cause losses for investors if the bankroll and trades are not managed carefully. For this reason, these implications are discussed, so that the models created are not mismanaged when the findings of our research are put to action in a trading strategy.

7 Conclusion

Previous research on the relationship between dictionary-based Sentiment Analysis and stock price prediction has little or no included material for trading applications. The previous work provide no clear consensus on whether it's possible to improve trading decisions by utilizing news sentiment extracted by sentiment dictionaries.

This thesis has proposed models that suggest it's possible to use dictionary-based Sentiment Analysis on Nasdaq News articles to create a successful trading strategy. Our best performing models show strong results in regards of accuracy of predicting the direction of the stock movements. For the Energy and Finance sector, we found models achieving high Sharpe ratios, proving our models can be of interest for traders looking for high risk-adjusted returns. The best Sharpe ratio was found when applying the SentiWordNet–KNN model in the Finance sector. In terms of classification report metrics, the results shows the greatest F1 score (60%) from the TextBlob–Decision Tree model in the Energy sector. The findings are, however, not conclusive across all sectors. The Technology sector does not yield the same promising results. There is a possibility that the smaller sample size of articles examined in the Energy and Finance sector has an impact on the result. The Technology sector has the largest sample size with the most news articles. At the same time it is also most in accordance with the EMH. Consequently our thesis' best performing models are arguably not indicative of sustainable long term abnormal trading returns.

Interestingly, our models seemed to be largely uncorrelated with both the market index and the comparable companies included in the sectors. This was especially true for the Energy and Finance sector. This indicates that there are some periods of opportunity for inefficiencies to be exploited by our models. Traders looking to protect their portfolios from downtrends in the general market could find it worth while to investigate this further.

For a trader, the goal is to achieve excess return over the market index for a longer period. A few of our models show signs that could be promising for this purpose. If one were to implement the models as a tool in a trading strategy, more backtesting should be considered as the number of observations are limited. Nonetheless, we have through our trading strategy provided a proof of concept of long/short trading strategy that shows promising risk adjusted returns. They should be interpreted as a strong indicator that corporate announcements are likely to be a useful addition as a basis for decision making in an automated trading application.

Our findings show that it is important for traders wanting to utilize a dictionary-based approach, to carefully choose the dictionary and classification techniques. The nature of the articles or text one analyses is important when choosing dictionary. In our case, the SentiWordNet dictionary performed best overall. This is not to say that it would work for all kinds of corpuses. For articles concerning the technology companies, the VADER dictionary generated best results. Regarding classifiers, the non-linear classifiers, KNN and Decision Tree, generated the best results on the given corpus. Some of our models managed to achieve abnormal returns. Without more backtesting of these models with larger sample sizes, the significance of these models are unclear. This makes the findings insufficient to argue even against the strongest form of EMH.

7.1 Future Work

Opportunities for further experimentation within the field are many. It would be interesting to study the relationship between sentiment and stock prices on a more frequent level. It is possible to do this in time periods below even one second. In the world of sentiment dictionaries, there are also much still to be tried. One example is to use dictionaries specifically designed to analyse the sentiment of text of a specific sector or even company. Another is to create a dictionary where the words or phrases are scored based on how the stock prices reacted to them. Moreover, as corporate news does not exist in a vacuum, one could expect multimodal models might improve predictions. Multimodal models combine the data set with other information. This could be quarterly reports, press releases or even stock price history. All these examples provide additional context to the news and could be useful. As the stock markets are accessible for traders and investors all over the world, the inclusion of multilingual dictionaries could help discover otherwise missed sentiment. Articles, social media, or other text written in other languages could very well affect the market sentiment. Admittedly, the field of Sentiment Analysis is increasingly moving in direction machine learning approach. Yet, the opportunities suggested are all intriguing possibilities to explore to further the dictionary-based form for for sentiment mining with focus on stock price prediction.

References

Bing, L., Chan, K. C., & Ou, C. (2014). Public sentiment analysis in twitter data for prediction of a company's stock price movements, 232–239.

Bird, R., & Yeung, D. (2012). How do investors react under uncertainty? *Pacific-Basin Finance Journal*, *20*(2), 310–327.

Biswas, S., Sarkar, I., Das, P., Bose, R., & Roy, S. (2020). Examining the effects of pandemics on stock market trends through sentiment analysis. *Journal of Xidian University*, *14*(6), 1163–1176.

Bojer, C. S., & Meldgaard, J. P. (2021). Kaggle forecasting competitions: An overlooked learning opportunity. *International Journal of Forecasting*, *37*(2), 587–603.

Burton, N. (2018). *Burton g. malkiel's: A random walk down wall street*. https://doi.org/10.4324/9781912281169

Chantona, K., Purba, R., & Halim, A. (2020). News sentiment analysis in forex trading using r-cnn on deep recurrent q-network. *2020 Fifth International Conference on Informatics and Computing (ICIC)*, 1–7.

Colianni, S., Rosales, S., & Signorotti, M. (2015). Algorithmic trading of cryptocurrency based on twitter sentiment analysis. *CS229 Project*, *1*(5).

Coval, J. D., Hirshleifer, D. A., & Shumway, T. (2005). Can individual investors beat the market?

Danielsson, P.-E. (1980). Euclidean distance mapping. *Computer Graphics and image processing*, *14*(3), 227–248.

Day, M.-Y., & Lee, C.-C. (2016). Deep learning for financial sentiment analysis on finance news providers, 1127–1134.

Deng, S., Mitsubuchi, T., Shioda, K., Shimada, T., & Sakurai, A. (2011). Combining technical analysis with sentiment analysis for stock price prediction, 800–807.

Dodevska, L., Petreski, V., Mishev, K., Gjorgjevikj, A., Vodenska, I., Chitkushev, L., & Trajanov, D. (2019). Predicting companies stock price direction by using sentiment analysis of news articles. *Proceedings of the 15th Annual International Conference on Computer Science and Education in Computer Science*, 37–42.

Elbagir, S., & Yang, J. (2019). Twitter sentiment analysis using natural language toolkit and vader sentiment. *122*, 16.

Esuli, A., & Sebastiani, F. (2006). Sentiwordnet: A publicly available lexical resource for opinion mining.

Esuli, A., & Sebastiani, F. (2007). Sentiwordnet: A high-coverage lexical resource for opinion mining. *Evaluation*.

Fama, E. F. (1965). The behavior of stock-market prices. *The Journal of Business*, *38*, 34.

Gandhi, R. (2018). Support vector machine — introduction to machine learning algorithms.

Geekofgeeks. (2021). Removing stop words with nltk in python.

Gidofalvi, G., & Elkan, C. (2001). Using news articles to predict stock price movements. *Department of Computer Science and Engineering, University of California, San Diego*, 17.

Gobel, N. N., & Zinfrazi, A. J. (2019). The sentiment of news articles and trading activity of cryptocurrencies.

Guo, L., Shi, F., & Tu, J. (2017). Textual analysis and machine leaning: Crack unstructured data in finance and accounting. *The Journal of Finance and Data Science*, *2*. https://doi.org/10.1016/j.jfds.2017.02.001

Gupta, I., Madan, T. K., Singh, S., & Singh, A. K. (2022). Hisa-smfm: Historical and sentiment analysis based stock market forecasting model. *arXiv preprint arXiv:2203.08143*.

Gupta, P. (2017). Decision trees in machine learning.

Haff, M. D. (2010). Sentiment analysis, hard but worth it.

Hamid, K., Suleman, M., Shah, S., & Akash, R. (2010). Testing the weak form of efficient market hypothesis: Empirical evidence from asia-pacific markets. *International Research Journal of Finance and Economics*, *58*, 121–133. https://doi.org/10.2139/ssrn.2912908

Harrison, O. (2018). Machine learning basics with the k-nearest neighbors algorithm.

Hassan Yousef, A., Medhat, W., & Mohamed, H. (2014). Sentiment analysis algorithms and applications: A survey. *Ain Shams Engineering Journal*, *5*. https://doi.org/10.1016/j.asej.2014.04.011

Hirshleifer, D., Hsu, P.-H., & Li, D. (2012). Innovative efficiency and stock returns. *Journal of Financial Economics*, *107*. https://doi.org/10.2139/ssrn.1799675

Husnain, M., Missen, M. M. S., Akhtar, N., Coustaty, M., Mumtaz, S., & Prasath, S. (2019). *A systematic study on the role of sentiwordnet in opinion mining*. https://doi.org/10.1007/s11704-019-9094-0

Hutto, C., & Gilbert, E. (2015). Vader: A parsimonious rule-based model for sentiment analysis of social media text. *Proceedings of the 8th International Conference on Weblogs and Social Media, ICWSM 2014*.

Jagdale, R. S., Shirsat, V. S., & Deshmukh, S. N. (2018). Review on sentiment lexicons, 1105–1110. https://doi.org/10.1109/CESYS.2018.8723913

Jishag, A., Athira, A., Shailaja, M., & Thara, S. (2020). Predicting the stock market behavior using historic data analysis and news sentiment analysis in r, 717–728.

Jivani, A. G. (2011). A comparative study of stemming algorithms ms .

Karlsson, M. (2011). The immediacy of online news, the visibility of journalistic processes and a restructuring of journalistic authority. *Journalism*, *12*, 279–295. https://doi.org/10.1177/1464884910388223

Kazemian, S., Zhao, S., & Penn, G. (2016). Evaluating sentiment analysis in the context of securities trading, 2094–2103.

Khedr, A., S.E.Salama, & Yaseen Hegazy, N. (2017). Predicting stock market behaviort using data mining technique and news sentiment analysis. *International Journal of Intelligent Systems and Applications*, *9*, 22–30.

Khedr, A. E., Yaseen, N. et al. (2017). Predicting stock market behavior using data mining technique and news sentiment analysis. *International Journal of Intelligent Systems and Applications*, *9*(7), 22.

Kim, Y., Jeong, S. R., & Ghani, I. (2014). Text opinion mining to analyze news for stock market prediction. *Int. J. Advance. Soft Comput. Appl*, *6*(1), 2074–8523.

Kim, Y. B., Kim, J., Kim, W., Im, J., Kim, T., Kang, S., & Kim, C.-H. (2016). Predicting fluctuations in cryptocurrency transactions based on user comments and replies. *PLOS ONE*, *11*, e0161197. https://doi.org/10.1371/journal.pone.0161197

Kohut, A., Doherty, C., Dimock, M., & Keeter, S. (2010). Americans spending more time following the news. *Pew Research Center*.

Lamon, C., Nielsen, E., & Redondo, E. (2017). Cryptocurrency price prediction using news and social media sentiment. *SMU Data Sci. Rev*, *1*(3), 1–22.

Li, S. Z., & Jain, A. (2009). Fisher´s criterion.

Li, X., Xie, H., Chen, L., Wang, J., & Deng, X. (2014). News impact on stock price return via sentiment analysis. *Knowledge-Based Systems*, *69*, 14–23.

Ligthart, A., Catal, C., & Tekinerdogan, B. (2021). Systematic reviews in sentiment analysis: A tertiary study. *Artificial Intelligence Review*, *54*(7), 4997–5053.

Loria, S. (2018). Textblob documentation.

Loria, S. (2020). Textblob: Simplified text processing.

Makrehchi, M., Shah, S., & Liao, W. (2013). Stock prediction using event-based sentiment analysis. *1*, 337–342.

Mehta, A. (2019). Linear discriminant analysis.

Modigliani, F., & Modigliani, L. (1997). Risk-adjusted performance. *Journal of Portfolio Management - J PORTFOLIO MANAGE*, *23*, 45–54. https://doi.org/10.3905/jpm.23.2.45

Nemes, L., & Kiss, A. (2021). Prediction of stock values changes using sentiment analysis of stock news headlines. *Journal of Information and Telecommunication*, *5*(3), 375–394.

Niederhoffer, V. (1971). The analysis of world events and stock prices. *The Journal of Business*, *44*(2), 193–219.

Pagolu, V. S., Reddy, K. N., Panda, G., & Majhi, B. (2016). Sentiment analysis of twitter data for predicting stock market movements, 1345–1350.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Philippas, N. (2014). Did behavioral mutual funds exploit market inefficiencies during or after the financial crisis? *Multinational Finance Journal*, *18*(1/2), 85–138.

Sazzed, S. (2020). *Development of sentiment lexicon in bengali utilizing corpus and cross-lingual resources*.

Schwert, G. (2003). *Chapter 15 anomalies and market efficiency* (Vol. 1). https://doi.org/10.1016/S1574-0102(03)01024-0

Shleifer, A., Summers, L., Long, J., & Waldmann, R. (1990). Noise trader risk in financial markets. *Journal of Political Economy*, *98*, 703–38. https://doi.org/10.1086/261703

Shynkevich, Y., McGinnity, T., Coleman, S., & Belatreche, A. (2015). Stock price prediction based on stock-specific and sub-industry-specific news articles, 1–8. https://doi.org/10.1109/IJCNN.2015.7280517

Stenqvist, E., & Lönnö, J. (2017). Predicting bitcoin price fluctuation with twitter sentiment analysis.

Thorp, E. O. (2011). The kelly criterion in blackjack sports betting, and the stock market. *The kelly capital growth investment criterion: Theory and practice* (pp. 789–832). World Scientific.

Țițan, A. (2015). The efficient market hypothesis: Review of specialized literature and empirical research. *Procedia Economics and Finance*, *32*, 442–449. https://doi.org/10.1016/S2212-5671(15)01416-1

Uhl, M. W. (2014). Reuters sentiment and stock returns. *Journal of Behavioral Finance*, *15*(4), 287–298.

Wang, C., & Luo, B. (2021). Predicting $gme stock price movement using sentiment from reddit r/wallstreetbets. *Proceedings of the Third Workshop on Financial Technology and Natural Language Processing*, 22–30.

Wang, W., Ho, K.-Y., Liu, W.-M. R., Wang, K. T., et al. (2013). The relation between news and stock price jump: An analysis based on neural network.

Zhang, H. (2004). The optimality of naive bayes. *Aa*, *1*(2), 3.

NTNU

Kunnskap for en bedre verden