Utilizing machine-learning derived news sentiment signals for modeling excess returns

VU Amsterdam

Author: Sebastian Keil

Supervisor: Martijn de Vries

August 12th, 2022

Abstract

This paper addresses the question how novel language-modelling approaches from Natural Language Processing (NLP) can be integrated to enrich the practice of asset pricing based on sentiment found in financial news articles. To this end, I devise a multi-step procedure that addresses the scraping of raw financial news, extracting sentiment of these news based on a cutting-edge NLP model, utilizing time-discounting to transform the sentiment values into a continuous time-series signal, modelling excess returns based on lagged sentiment and building a simple trading strategy to put the results to practical use. For modelling excess returns, I contrast a statistical approach to a machine-learning approach, finding that both face issues converging over longer time horizons, indicating fundamental shifts in the relationship between sentiment and excess returns overt time. The resulting sentiment-based trading strategy produces negative Sharpe ratios for three out of four companies. However, it also outperforms a baseline strategy that does not consider sentiment features. This shows that while sentiment extracted from financial news can be informative for decision-making in financial markets, more understanding is needed to be able to implement a consistent, reliable trading strategy based on sentiment in financial news.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In recent years, there has been an explosion of interest in computational methods that can understand and work with written text. Novel deep-learning models like Google's Bidirectional Transformers (BERT) [12] and OpenAI's GPT model [23] beat previous benchmark performances on common metrics related to language understanding, and demonstrated impressive abilities like writing entire essays after being given just one sentence as a prompt. For scholars of academic finance, there is an increasing interest in these types of models and how they can be used to enrich financial decision-making. After all, there is a large variety of documents that may contain information that is relevant to financial markets, for instance financial reports, shareholder calls, news articles and social media posts. These documents can inform us about how different actors in the financial markets think and feel about specific assets, asset classes or the economy as whole. But *how* can this information be utilized to enable better-informed financial decision? How can we get from raw, written text, potentially in the form of thousands of documents, to a well-informed decision on what was written? In this paper, I attempt to draw on synergies between three fields, Asset Pricing, Natural Language Processing and Machine Learning to investigate how to use freely available textual information, extract sentiment [1] from it and to model outcomes of the financial markets.

Why do I want to draw on synergies between those fields? As the literature shows, the overarching topic of *sentiment in financial markets* is of interest to scholars from all three disciplines, but there are certain strengths and weaknesses within each academic field. For instance, scholars in Asset Pricing have a strong focus on justifying modelling choices and applying economic reasoning. However, they tend to overly rely on traditional approaches within their discipline, for instance dictionary-based methods for representing language, missing novel cutting-edge developments. Scholars in Machine Learning and NLP tend to have a strong-suit in optimizing model and increasing performance, but often miss out on interpretability and economic intuition. In this paper I try to get the best of both worlds by using cutting-edge models, optimization techniques as well as economic reasoning in an end-to-end pipeline.

[1] The term 'sentiment' can have multiple, nuanced meanings depending on the context in which it is used. In finance, it is common to think of sentiment in written text as being characterized by three distinct categories: negative, neutral and positive. Alternatively, sentiment can be thought of as being divided into emotional categories like 'happy', 'sad' or 'exiting', or existing on a (multi)dimensional continuous scale, as in the circumplex model of emotion

The goal of this investigation is to build an end-to-end data pipeline that transforms raw written text into a tool for making intelligent trading decisions. This pipeline ideally answers the question *how* novel approaches from NLP and machine learning can be integrated into the pricing of assets based on sentiment. I hope this paper can serve as an inspiration for other scholars to look beyond the well of their disciplines and approach the topic of sentiment in financial markets from a holistic angle.

To achieve this over-arching goal , I devise a methodology that consists of 7 distinct steps. Here, I want to give a high-level overview of the pipeline, without mentioning too many specifics. While the specifics do matter, and are outlined in detail throughout this thesis, it is more important for the reader to develop a general understanding of the task at hand and the specific steps that are needed to achieve the overarching goal. Figure 1.1 illustrates these steps in a flow-chart, also illuminating where each step of this pipeline can be found throughout the thesis. The first step is web-scraping, where I gather textual financial news data from the internet. The second step is text cleaning. Here I take the raw, extracted text and transform it into a form that is suitable for the sentiment model used in this paper. At step 3, I perform sentiment extraction, which essentially converts the textual data into a number that represents its sentiment. [2] For this step, I utilize an 'black-box' ML-model primed for language understanding. [3] In the fourth step, I explore the extracted sentiment signal, as well as an excess returns dataset over the same time-frame, [4] to investigate the relationship between textual sentiment and market outcomes. As the reader will discover, one major problem that presents itself is that for most companies under investigation there are a large number of days for which there is no financial news available. This phenomenon is referred to as *sparsity*. On the other hand, market outcomes do happen on a daily basis, [5] as prices fluctuate consistently and so do resulting metrics like returns and volatility. To mitigate the sparsity issue, I limit my analysis to only those companies for which there is sufficient textual data available and introduce a time-discounting method that enables me to impute missing days in the sentiment signal. This time-discounting method is explored in step 5. In step 6, I finally move to building a model that predicts excess returns based on lagged values of news sentiment and past excess returns. Here, I contrast the Vector Auto-regressive (VAR) model, a commonly used statistical model, with a ML-based approach represented by the Random Forest regressor. In preparation of model building, I also explore correlations between excess returns and sentiment at different lags. In the seventh and final step, the resulting model is then tested in the construction of a simple sentiment-based trading strategy, which is contrasted against a benchmark strategy which omits sentiment features.

[2] Here, I represent sentiment as continuous value on the range -1 to +1, with -1 being the most negative value possible and +1 the most positive.

[3] The model chosen for this investigation is the *FinBERT* model. While the specifics of this model are not important for this paper, more context is given throughout the thesis and in the appendix.

[4] Excess returns are modelled by leveraging the residuals of the Fama-French-Carhart 4-Factor model. More details on this are provided later in this paper.

[5] There are usually 252 trading days per year.

Hence, the research question guiding this investigation is *how can machine-learning driven sentiment signals be used for modelling excess returns?* To approach this overall objective, the following sub-questions will be of utility: (1) what are the existing gaps in how the Natural Language Processing and Asset Pricing academic communities approach modeling sentiment? (2) how can 'black-box' sentiment models be utilized to construct a continuous sentiment time-series for individual stocks, (3) how can excess returns be modelled based on the constructed sentiment series and (4) how can an effective investment strategy be devised based on the knowledge gathered in (2) to (4). Sub-question (1) is mainly addressed in the literature review, while sub-questions (2) to (4) are answered in the methodology and results sections of this paper.

What are the expectations for the results of this thesis? I do not expect to build a model that can perfectly predict market outcomes based on textual news sentiment, as past research has shown that it is extremely difficult to predict the behavior of markets with high precision. Taking the Efficient Market Hypothesis as a baseline, which implies that sentiment bears no relation to market outcomes, it follows that market movements cannot be predicted with accuracy higher than 50% [6]. Predicting the exact magnitude of returns or excess returns is even more difficult. So, rather than asking if I can predict excess returns perfectly, a more realistic question to ask is *can I do better than mere chance?* Is it possible to draw any meaningful inferences about market outcomes based on textual sentiment? As the reader can see from the pipeline overview, the entire process is quite complex and there are a multitude of potential bottle-necks that may impede the performance. While it would be out-of-scope to address each of these, my goal is to at least highlight them to the reader and point toward possible solutions that future scholarship might employ when addressing these.

The contributions of this thesis are manifold. The main contribution is to devise a general methodology that aims to draw upon synergies in Asset Pricing, Natural Language Processing and Machine Learning. Since the scope of this project is quite large, the idea is that future research can address flaws and introduce possible improvements at each stage of the pipeline. However, the paper also introduces several improvements upon previous research. First, this paper shows how cutting-edge sentiment models can be used in a financial context. The beauty of the approach shown here is that basically any sentiment model can be used with only minor adjustments to the entire process. Second, many scholars (e.g., [18], [27]) utilize sentiment to model returns or the direction of returns. This approach provides no information on the *magnitude* of market outcomes, which can make or break intelligent decision-making. To address this, I employ a regression rather than classification approach to capture the magnitude of excess returns.[6] Third, this paper has substantial practical value, as I show how to deal with large amounts of unstructured textual data in the context of financial decision-making. Unlike most other papers, the data used for this project is very much 'from-the-wild', meaning it has not been pre-processed or prepared by another organization. Since such type of data is much more widely available than pre-processed datasets, being able to utilize such data effectively can provide additional value to the academic community.

[6] Regression and classification describes the two main approaches in supervised machine-learning, where regression is about predicting a continuous value and classification about predicting categorical data.

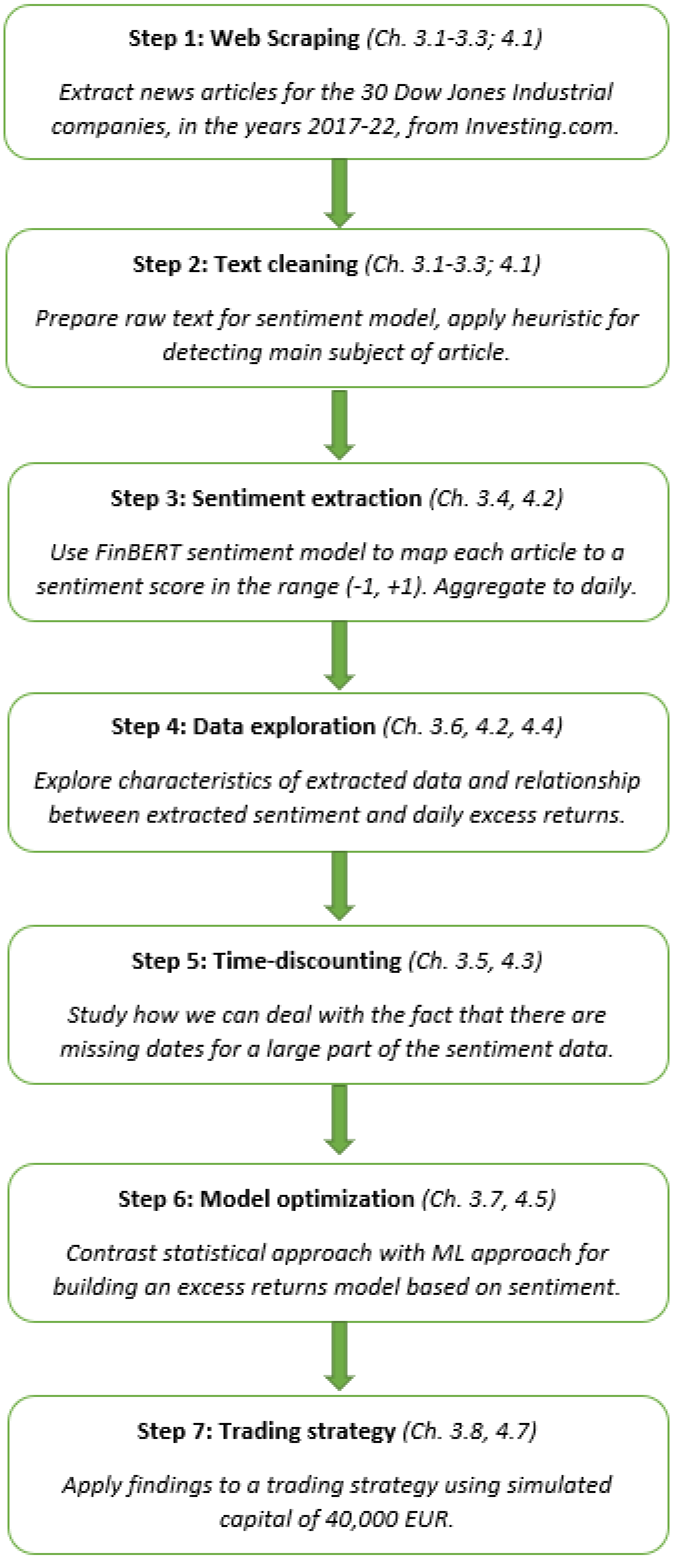
Step 1: Web Scraping (Ch. 3.1-3.3; 4.1)
Extract news articles for the 30 Dow Jones Industrial companies, in the years 2017-22, from Investing.com.
Step 2: Text cleaning (Ch. 3.1-3.3; 4.1)
Prepare raw text for sentiment model, apply heuristic for detecting main subject of article.
Step 3: Sentiment extraction (Ch. 3.4, 4.2)
Use FinBERT sentiment model to map each article to a sentiment score in the range (-1, +1). Aggregate to daily.
Step 4: Data exploration (Ch. 3.6, 4.2, 4.4)
Explore characteristics of extracted data and relationship between extracted sentiment and daily excess returns.
Step 5: Time-discounting (Ch. 3.5, 4.3)
Study how we can deal with the fact that there are missing dates for a large part of the sentiment data.
Step 6: Model optimization (Ch. 3.7, 4.5)
Contrast statistical approach with ML approach for building an excess returns model based on sentiment.
Step 7: Trading strategy (Ch. 3.8, 4.7)
Apply findings to a trading strategy using simulated capital of 40,000 EUR.

Figure 1.1: **7-Step procedure for building the data pipeline.** *This figure illustrates the seven steps that are followed in this paper to get from raw textual data, to a continuous sentiment signal, to an excess returns model based on sentiment, to the application of a trading strategy based on our models. For each steps (in parenthesis), a referenced is made where the step is dealt with in the thesis.*

Chapter 2

Literature Review

To get a grasp on the research questions formulated above, especially sub-question (1) which addresses potential 'gaps' between the NLP and Asset Pricing academic communities in how they model and understand sentiment, this section provides an overview of the relevant literature. First, the idea of sentiment in financial markets is presented and the reader is introduced to some of the debates being held around this concept. Second, distinct approaches to quantifying the idea of sentiment in finance are explained. Third, the reader is introduced to different approaches in modelling market outcomes on the basis of sentiment. Fourth, it is explained how different scholars have translated their modelling-outcomes into a trading strategy. Finally, I synthesize the findings in these four section to answer the research sub-question (1).

2.1 Sentiment in financial markets

The very idea of market sentiment in financial markets is a refutation of the Efficient Market Hypothesis (EMH), which postulates that market outcomes are based on rational decisions and information is reflected in prices immediately. Even weaker forms of the EMH, which allow for delays in the manner in which new information is absorbed in the pricing of assets, assume an underlying rational market structure. Critics of the EMH, on the other hand, argue that markets are not only driven by rational factors but that there are certain limits to the extent market actors can act rationally. The most obvious of such limits is the psychology of different market participants, whose perceptions are toned by their specific thoughts and feelings about a given asset. Crowds driven by fear or greed can push return magnitudes beyond what they would have been in a market only inhabited by rational maximizers of risk-adjusted returns. While the EMH has a long history in academic finance, only few scholars today would argue that it offers a complete explanation on the nature of financial markets [8]. For instance, Baker and Wurgler [5] point out that standard financial models that align with the EMH face considerable issues in making sense of events like the Great Crash of 1929, the Black Monday crash of 1987 or the Dot.com bubble of the 1990s.

A milestone work on sentiment in financial markets is presented by DeLong et al. [10], who show an overlapping generational model in which irrational noise traders have an effect on prices and earn higher expected returns. The existence of these noise traders generates additional risk for rational arbitrageurs, who become less inclined to trade against the crowd. This behavior causes a price divergence from fundamental values. While noise trading drives up prices in the short term, there is a mean-reversion effect in the longer term. As I show later in this paper, this price convergence from fundamental values can be quantified by utilizing the excess returns from a factor-based model, and the fact that prices are driven up by high sentiment can be used in constructing a trading strategy. Building upon this early work, Shleifer and Vishny [24] suggest a framework in which moving against investors driven by sentiment is both costly and risk-carrying. They claim that while traditional theories view arbitrage as carried out by a large number of small investors, in reality it is usually done by a small number of large, professional investors. DeVault et al. [11] also find that find that sentiment metrics relate to institutional investors' demand shocks rather than those of retail investors. These two findings show that contrary to common beliefs, the relationship between sentiment and market outcomes is not only driven by the psychology of crowds but also the behavior of large (professional) market actors. What does this imply for this paper? Ideally, I would want to approximate both the sentiment of retail investors at large, as well as that of professional investors. However, dealing with written text, the practical limitation is that documents revealing the sentiment of such investors are a lot harder to collect and analyze in a systemic fashion. Hence, although it is unfortunately not feasible to include this perspective into the analysis, the reader should keep in mind that it likely provides valuable insights.

While there are still ongoing debates on the nature of sentiment in financial markets, only the most loyal proponents of the EMH would claim that sentiment plays no role in financial markets at all. In recent years, rather than debating the existence of sentiment in financial markets, the focus has shifted towards debates on how the idea sentiment can be quantified and put into practical use. As Hsie et al. [16] point out, there remains a perceived gap between fields like NLP and behavioural finance within the academic literature, both of which have had rapid advancements in recent years but often times lack awareness of the progress in each others' fields. While NLP has pioneered new tools that can quantify language and meaning through computational tools, behavioral finance has advanced theoretical frameworks that enable us to better understand how psychological factors influence the outcome of markets. It is an ongoing challenge for scholars to bridge the gap between these fields.

2.2 Approaches for measuring sentiment in financial markets

The previous section established that there is wide agreement among the scholarship that sentiment does exist in financial markets and that it does influence market outcomes. But what actually *is* sentiment on a more fundamental level and how can it be measured? Baker and Wurgler (2007) [5], make the important distinction between top-down and bottom-up approaches for measuring sentiment. The top-down approach measures variations in investment based on exogenous variables (e.g., macro-economic variables from which sentiment can be deduced), while the bottom-up approach utilizes endogenous sources (e.g., textual sources directly related to the firm).

The authors themselves take a distinctly top-down approach, as they devise a sentiment measurement based on six macro-economic factors, trading volume as measured by NYSE turnover; the dividend premium; the closed-end fund discount; the number and first-day returns on IPOs; and the equity share in new issues and take their first principal component [1] as representative of the overall market sentiment. This index is widely regarded as one of the most robust indicators of market-wide sentiment. Another instance of a top-down approach to modelling sentiment is the one described by Han [15], who uses option implied volatility to infer sentiment. The author finds that the index option volatility smile is steeper and the risk-neutral skewness of monthly index returns increasingly negative when market sentiment become bearish. Conversely, the volatility smile flattens and skewness becomes positive in the case of bullish sentiment.

The work of Tetlock et al. [25] serves a good illustration of a bottom-up approach, as the authors devise a simple quantitative measure based on financial news to forecast accounting earnings and stock returns. Kearney and Liu (2014) [19] provide a classification of textual sources that are relevant to financial markets into three categories: (1) corporation-expressed, (2) media-expressed and (3) internet-expressed. Respective examples of these three include shareholder reports, news media and social media chatter. Each of these categories can potentially be used in a study on the role of sentiment in the financial markets. Atkins et al. [4], for instance, utilize StockTwits, Google-Trends, Wikipedia as the source of their textual data, while Uhl [26] uses Reuters news articles and Tetlock [25] leverages the Wall-Street Journal opinion column. In this paper I focus on media-expressed documents from different news providers. Compared to corporation-expressed content, media-expressed text addresses a wider (retail) audience and compared to social-media text, it is a lot more structured, easier to associate with specific companies and less prone to noise.

Hsie et al. [16] point out that the academic community in finance still overwhelmingly relies on dictionary-based approaches to computing sentiment signals based on textual data. A popular instance of such an approach is described by Loughran and McDonald [20], who modify the approach taken by the *Harvard Psychological Dictionary*[2] by manually classifying words if they have negative connotation in a financial setting. They sub-sequentially use their generated dictionary to analyze the language of 10-Ks during 1994 to 2008 and link it to financial outcomes like returns and volatility, trading volume and fraud. The Loughran-McDonald dictionary has been utilized by a number of scholars since, to conduct sentiment-based studies in finance. Hsie et al. [16] make an effort to move beyond the common reliance on dictionary-based approaches to textual understanding in finance by using the Bidirectional Encoder Representations (BERT) [3] model on three stocks actively discussed on the Chinese online

[1] Principal Component Analysis (PCA) is a dimension-reduction algorithm. In this case, it reduces a six-dimension input space to just one dimension, which is regarded as representing the idea of sentiment.

[2] In the HPD, psychologist labeled a large amount of English words based on their valence.

[3] BERT is a transformer-based deep-learning architecture that has been trained on a large corpus of text in an unsupervised manner. While the details of this model are not important for this paper, the curious reader is referred to [12] for more information.

platform *Weibo.com*. They find that BERT significantly improves upon existing approaches. In a similar vain, the authors also propose replacing traditional econometric approaches (e.g., linear regression) to modelling market outcomes with more advanced deep-learning methods like Long Short-term Memory (LSTM) networks.

In general, the advantage of the top-down approach is that sentiment is relatively simple to compute and that is successfully encompasses phenomena like bubbles and crashes. The benefit of using bottom-up approaches is that they can be much more fine-grained (in terms of both timing and detail), operating on a more micro-foundational level [5].

2.3 Sentiment-based modelling of market outcomes

Given some mechanism of successfully measuring the notion of sentiment in the financial market, a question that naturally arises is how this sentiment measurement can be used in a practical setting. What inferences can be made based on these sentiment measurements and what practical value do they provide? To answer these questions, a number of scholars sought to study the relationship between sentiment and market outcomes. Market outcome can refer to price outcomes like returns and volatility, but also other factors like fraud, probability of default or trading volume. This section examines the relevant literature, with a particular focus on approaches that leverage machine-learning.

Within the academic literature, most work focuses on relating sentiment to the direction of the market or specific assets (up vs. down), i.e., a classification problem, rather than predicting the variable directly in a regression setting. For instance, Hayek [18] devises bag-of-words features from annual reports and uses a neural network to predict the direction (positive/negative) of abnormal stock returns. Wei and Nguyen [27] use a combination of BERT-extracted embeddings and historical stock market data to predict the direction of future returns (up/down). Uhl [26] utilizes sentiment measured from 3.6 million Reuters articles to forecast returns of the Dow Jones Industrial Average stock index using a Vector Auto-regression (VAR) model and finds that this method outperforms prediction via traditional macro-economic factors. In a more recent effort, Akyildirim et al. [1] examine how machine learning can be utilized to predict intraday excess returns. They find that XGBoost, a tree-ensemble method, performs best in predicting the direction of excess return of individual stocks ('up' or 'down') compared to other methods. Bollen et al. [6] model twitter sentiment based on a six-dimensional representation of mood (Calm, Alert, Sure, Vital, Kind and Happy) and use their outcomes to the direction of the movement of the Dow Jones Industrial Index. Their reported accuracy of 86.7 percent has however been heavily criticized in the computational finance community, as their testing period comprises only 15 days in total, which leaves a high chance that their good results are due to mere chance. Furthermore, the authors only reported the results of their best model [4]. Ding et al. [13] propose a deep learning method for event-driven stock market prediction. They extract news text and model the the period around major news as a dense vector. This approach resembles an event-study methodology, rather than a time-series approach that requires a continuous signal. Although the event-study approach has its own merits, as it allows one to closely study the effect of the sentiment of *particular* events on market outcomes, the time-series approach allows for a more general type of analysis. In this paper, I choose to take a time-series approach, as I am more interested in the general relationship of sentiment and market outcomes.

Next to returns, another popular metric to which sentiment has been related to is volatility. Allen [2], utilizes the sentiment signal provided by the Thomson Reuters News Analytics index, which is itself devised based on ML methods, to capture the effect of sentiment on volatility via three standard econometric models (GARCH, EGARCH and GJR). Atkins et al. [4] similarly show that information extracted from news sources is better at predicting the direction of underlying asset volatility movements, rather than price or return movements. They use a simple naive Bayes classifier and achieve a 56 percent accuracy in predicting volatility movements, compared to a 49 percent accuracy in predicting price movements.

Rather than directly modelling (excess) returns or volatility, another approach to utilizing sentiment in financial decision-making is to construct a 'sentiment sensitivity' measurement in the spirit of Fama-French factors. A good starting point for constructing such an index is again Baker and Wurgler [5]. The authors divide up 10 portfolios based on their historical volatility, and find that high-growth, no-dividend, high-volatility, small and young companies are more sensitive to sentiment risk in comparison to their dividend-paying, larger, low-volatility peers. In terms of returns, it means that stocks that are highly sensitive to sentiment achieve comparatively higher returns in high sentiment periods, while they under-perform in low sentiment time periods. Du et al [14] find evidence that sentiment premium should be particularly significant on days without macroeconomic announcements, because there is a lack of information about the state of the economy at such times.

2.4 Sentiment-based trading strategies

To analyze the utility of sentiment signals and their resulting usage in market-outcome models, a number of scholars have put their results into practical use by simulating trading strategies based on sentiment. Zhang and Skiena [28] construct a sentiment-based market-neutral trading strategy, providing consistent returns with low volatility over in the period 2005-2009. The authors base their sentiment signal on the dictionary-based and graph-theoretic *Lydia* sentiment system. In their strategy, the authors rank firms by their daily measured sentiment and simply go long on positive sentiment stocks and short on negative sentiment stocks, to equal proportions. Uhl [26] proposes a simple way of utilizing the VAR model in a trading strategy. He simply makes a VAR forecast of the next month-end closing price, goes long if the current price is lower than that and short if it is higher. The merit of this approach is that it is oriented towards the long-term, meaning that it virtually does not incur trading costs. On the other hand, it has been pointed out that the longer forecasts reach into the future, the more unreliable they become [16]. Atkins et al. [4] leverage an intra-day approach by predicting movements in the direction of volatility in the next hour, and trade based on these predictions.

2.5 Conclusion

To conclude the literature review, and reflect back upon sub-question (1), which asked about the gaps between NLP and finance in how they approach modelling sentiment, here are the key takeaways. First, the literature indicates that scholars in academic finance still strongly rely on approaches that have a tradition in finance like dictionary-based approaches, rather than looking to how advances in fields like NLP and computational linguistics, who made strong strides in the understanding and processing of natural language, can be used in a financial setting. Although there have been attempts at bridging the two sides, it is not yet the mainstream approach. Second, there is a large diversity of definitions of sentiment and what the idea might mean in a financial context. There are top-down approaches which measure sentiment as residual or side-product of other activity in the financial markets and bottom-up approaches that aim to discern sentiment directly from written text or speech. While both approaches have their merit, it is not clear how they can be reconciled. Finally, there is a lot of disagreement of what usefulness sentiment can have in modelling outcomes in the financial markets. While modelling (excess) returns or volatility, direct impacts on prices, is the most common approach, sentiment has also been linked to other outcomes like fraud, volume and likelihood of default. There is still no definitive answer to which of these, if any, sentiment has a causal relationship with. This thesis aims to address these points by introducing the idea of working 'end-to-end' and devising a methodology that draws on both NLP and traditional asset pricing methods. While the outcomes of this paper are by no means conclusive, the method itself hopes to inspire other scholars to look beyond their fields and search for synergies that can enrich their analysis.

Chapter 3

Methodology

This section describes the methodology used in this research. First, the reader is introduced to some basic features of the scope of the data like the time horizon, granularity and stock selection. Second, the three different data sources used for this thesis are presented to the reader, also highlighting what could be possible alternatives. Third, the process of web-scraping is described at a high level. Fourth, the sentiment model used for the extraction of sentiment from raw text is presented. Fifth, the time-discounting method introduced for this thesis is explained in some detail. This method is necessary to deal with the fact that there are many missing dates in the sentiment data, i.e., dates for which there were no news. For these days, a method is needed to impute sentiment values in order to study the relationship between excess returns and sentiment over longer-term horizons. Fifth, I introduce a 3-step procedure that can be leveraged before fitting the excess return models. This method informs about whether the VAR can be reasonably used and about the look-back period that should be leveraged to model excess returns based on lagged sentiment. It also provides insights into the nature of the relationship between the two series and can be used synergistically in a machine-learning approach. Sixth, the two excess return models used for this thesis, VAR and Random Forest regressor, are explained to the reader, as well as the methods used to gauge the performance of these models. Finally, I explain the simple trading strategy devised for this paper and how its performance is evaluated.

3.1 Scope of the data

As regards the stock selection universe, to somewhat limit the scope and computation-time of the web-scraping operation, I choose to collect financial news for the Dow Jones Industrial 30 (DJI30) companies. The DJI30 provides a selection of the 30 largest US-companies and includes a variety of different sectors. A list of these companies can seen in the various results tables, for instance Table A.1. In contrast to most other related research, I model the sentiment of individual companies, rather than the index as a whole. The reason for this is that each company might exhibit different dynamics in how sentiment and excess returns relate. In a practical sense, it also provides investors with a more fine-grained tool compared to being only able to invest in the index as a whole.

For every company, we collect news articles from the years 2017 to 2021. These years were chosen because they represent the most recent years for which an entire year of data is available. The analysis is limited to 5 years due to computational and time restraints. Ideally, I would scrape data even further into the past, as it is reasonable to assume that long-term underlying patterns might change over time that cannot be captured in this 5-year sample.

One difficulty with data collection is that one can never be certain that there is not some kind of underlying shift that changes the relationship between sentiment and excess returns. For instance, in the sample chosen for this thesis a major event that occurred was the Covid-19 crisis which started in the beginning of 2020. It also includes the presidency of Donald J. Trump in the United States, which brought about specific market reactions on its own. The best practice for dealing with these underlying shifts without *explicitly* addressing them is to take as long a sample as possible. Methods like walk-forward validation, which is introduced later in this section, can also help to mitigate fundamental shifts over time. Compared to other research, the sample chosen in this research is still relatively long. Atkins et al. [4], for instance, only utilize Reuters news from September 2011 to September 2012.

In terms of granularity, i.e., how *fine* each time-interval should be sliced for one sentiment measurement, there are many plausible choices one could consider when analyzing the role of sentiment on stock returns, from intra-day data to long-term investment horizons, which can be several months to years. Different scholars have investigated different time frames, for instance Tetlock et al. [25] look at effects of sentiment on a daily granularity, while Uhl [26] proposes a strategy that requires monthly restructuring of the portfolio and Baker and Wurgler [5] study sentiment effects over several months and years. In this research I choose to aggregate unto a daily granularity, meaning each day for each company is associated with one sentiment value. This is done to better understand the short to mid-term effects of sentiment, which should be driven by the behavior of noise traders and the inclination of prices to be driven up by high sentiment. For practical purposes, it also also straight-forward to work with returns and OHLC data [1] on a daily granularity, as these are most commonly available. In cases where I have more than one sentiment value for a single day (i.e., the company has been referenced in more than one article on that day), the sentiment signal itself is aggregated to daily via simple averaging:

$$SENT_t = \frac{\sum_{n=0}^{N_t} SENT_n}{N_t}$$

Here, t is the given day and N_t is the number of articles the company is referenced in on that particular day. To get a further grasp on amount of days we should look into the past, correlations between different lags of our time-series are utilized to study realistic scenarios for each company. This will determine, for instance, how many days $t-1, ..., t-n$ the model looks back into the past to make its forecast for day t. Regardless of the lock-back period, the models constructed in this thesis will only be utilized to

[1] OHLC refers to Open, High, Low and Close price values. This data is needed only for purposes of implementing a trading strategy.

forecast one day into the future. This simplification has been made to keep our models comparable and also somewhat limit the scope of possible optimizations that could be made for different prediction horizons.

3.2 Data sources

3.2.1 Textual financial news data

It goes without saying that the source of the textual data chosen for this analysis is of critical importance. Its quality is likely a determining factor of the quality of the overall results. Regarding this, I have mentioned that I focus on media-expressed textual data related to the financial domain. Specifically, the textual data is scraped first-hand from the retail-investor news platform *Investing.com*. The platform was founded in 2007 under the name *Forexpros* and initially focused on foreign exchange data and discussions. It later expanded to include news and analysis on a large variety of securities, including stocks. According to a popular web-traffic analysis portal, *Investing.com* was ranked as the 189th most popular website in the world as of the time of writing. [2] Its wide popular appeal, the fact that it pools news data from multiple providers, and the relative ease with which it can be scraped make *Investing.com* a good candidate for the collection of ground-news for this thesis. Alternative news datasets that are particularly popular within the NLP community are the *Reuters21578* and *Reuters RCV1* datasets [3], both of which include a large amount of news articles from Thomson Reuters, which is a widely trusted source of financial news. However, the downside of these datasets is that the articles cover only specific years in the 1980s, which ignores recent developments when financial news data has become more readily available to retail investors. These datasets are also not particularly geared toward a financial audience, so they are not categorized by company names. Other alternatives are commercial providers like *Reuters* and *RavenPack*, which provide pre-processed sentiment signals for individual stocks. However, these datasets are not available for free academic use. Since their quality likely exceeds any non-commercially available datasets, future studies should reach out to these providers and examine their utility.

3.2.2 Excess returns data

Next to the sentiment data, the research requires a dataset of daily excess returns that matches the time-horizon and granularity of the sentiment data. For this purpose, I utilize the Fama-French-Carhart 4 factors daily dataset, which can be found on the *BetaSuite* provided by *WRDS*.[4] The dataset provides model parameters, returns and model residuals on a daily basis for all of the 30 Dow Jones Industrial companies. There are 252 trading days per year.

As mentioned before, I want to model the notion of excess returns, in the sense of returns that cannot be explained by a rational (factor) model. Given such a model,

[2] see https://web.archive.org/web/20211222070439/https://www.alexa.com/siteinfo/investing.com

[3] see https://archive.ics.uci.edu/ml/datasets/

[4] https://wrds-www-wharton-upenn-edu.vu-nl.idm.oclc.org/pages/get-data/beta-suite-wrds/beta-suite-by-wrds/

I hypothesize that the returns not explained by this rational model can be explained by the irrational aspect of market behavior, namely sentiment. From a theoretical perspective, if I assume that sentiment does indeed have an effect on stock returns, the model can help us to filter out the effect of other (macro-economic) factors on returns. While there are a large variety of such factor models, each with their own merits and disadvantages, I choose the Fama-French-Carhart Four Factor Model because it is the most comprehensive model for which sufficient data was available. There is also precedent in the research, as Du et al. [14] also examine the relationship between sentiment and the Fama-French-Carhart model.

A simpler alternative could be the market model, however for this thesis it is important that the model subsumes as many rational market factors as possible, in order to properly isolate what may be deemed a sentiment effect. The Fama-French-Carhart model can be described as follows, where EXMKT is the market sensitivity factor, HML is the high-minus-low book-to-market ratio factor, SMB is the small-minus-big size factor and UMD is the up-minus-down momentum factor:

$$RET_{i,t} = \alpha + \beta_{mkt} * EXMKT_{i,t} + \beta_{HML} * HML_{i,t} + \beta SMB * SMB_{i,t} + \beta_{UMD} * UMD_{i,t} + \epsilon_{i,t}$$

The residual of this model $\epsilon_{i,t}$ represents the notion of excess or abnormal return. In other words, it is the return that was achieved beyond (or below) what the model would have predicted. If I assume that the model explains all rational market factors appropriately, I am left with the idea of sentiment in the residual $\epsilon_{i,t}$. Hence, $\epsilon_{i,t}$ will be the target variable for my market-outcome model.

3.2.3 OHLC data

For the implementation of the trading strategy, I also need to leverage OHLC (Open, High, Low, Close) price data for the companies of interest. To this end, I utilize *Yahoo! Finance* [5], where such data can easily be obtained.

3.3 Web Scraping

The web scraping algorithm is implemented using the *Python* packages *BeautfifulSoup* and *requests*. In essence, the scraper goes to the news page of each company using a modified URL. For instance, *https://www.investing.com/equities/apple-computer-inc-news* can be used to access the news page of *Apple*. By the adding a page number to this URL, I can scrape further into the past. Financial news articles can be found via specific HTML tags, and are distinguished from commercials and advertisements on the website. A more detailed explanation on how the Web Scraper works can be found in Appendix. The code can be found on my GitHub repository. [6]

[5] see https://finance.yahoo.com/

[6] see https://github.com/sebkeil

3.4 Sentiment model selection

As regards to the choice of sentiment model, this is largely to be regarded as a "black-box", meaning that not a lot of effort is spent on explaining or improving the model itself. This is done to avoid going down the rabbit-hole of model optimization and selection, which may require several research papers on its own. Also, the idea behind this paper is to devise a methodology that can be used with *any* cutting-edge sentiment model. Here, I am taking the working assumption that there are models that can do accurate sentiment analysis, which enables us to focus on how such information can be used in the context of modelling market outcomes and constructing a trading strategy.

The model chosen for this thesis is the *FinBERT* model, which is an open source model that has strong language understanding capabilities and has been fine-tuned on a finance-specific dataset [3]. FinBERT is a good model this task, as it addresses the issue of domain specificity in the BERT model, meaning that sentiment models do not generally perform well outside of the domain they were trained on. To address this gap, the author fine-tuned the BERT model on the *FinancialPhrasebank*, a dataset collected by Malo et al. [21] which consists of financial sentences. Araci [3] reports state-of-the-art results for the classification of this dataset using his *FinBERT* model. [7].

Since the *FinBERT* model has been trained on a polarity detection task (i.e., positive, neutral or negative), rather than a regression, we need to convert the model output to a continuous signal before further processing. One simple method is to equate a 'negative' label to -1, 'neutral' label to 0 and 'positive' label to 1, multiply each of these integers by the model certainty, which is automatically provided by the model output, and aggregate all signals over a daily interval by taking the average. This is how we could express this formulation, where t is the current day, N_t is the number of news sources for that day, λ_n is the model certainty for its prediction of article n and Θ_n is the mapping to -1, 0 or 1 as described above:

$$SENT_t = \frac{\sum_{n=1}^{N_t} \lambda_n * \Theta_n}{N_t}$$

3.5 Time-discounting

As the reader will discover later in this paper, one characteristic of the the news sentiment data collected for this paper is that it is extremely sparse, meaning that for most companies there are more days in which they are not mentioned in the news compared to days in which they are. Sparsity issues are caused by two characteristics of our data, namely that (1) some companies are a lot less in the news than others and may not have any news for one or more days and (2) sometimes although a company is mentioned in a given news article, the article is not mainly about that company.

[7] The public library *Huggingface* offers over 215 out-of-the-box sentiment models that can be easily accessed via their Python API. From these, the *FinBERT* model appears most promising in classifying financial news data, as it has been fine-tuned on exactly that task. See https://huggingface.co

For the latter case, common examples are market overviews, which are published by *Reuters* and *Investing.com* itself and might only mention the company performance in one sentence, or articles that are about competitor companies and make reference to the company of interest (e.g., an article that deals with a new Microsoft product might make references to what Apple is doing). To deal with the fact that some articles might only briefly mention the company of interest, I employ a simple heuristic: if the company name or ticker is contained in the headline of the news article, the entire document will be utilized for extracting the value of the sentiment signal. If, on the other hand, this is not the case, only the sentences which specifically mention the company or ticker are extracted for the signal construction. In this manner, I avoid assigning the sentiment to the company of interest of entire articles that mention the company only a few times.

The first type of sparsity, missing entire days, poses a more serious issue, as I generally require a smooth time-series signal without missing values for the modeling of excess returns. To achieve this, I require the use of some sort of imputation. There are several approaches that could be taken here, for instance mean imputation, which simply replaces all missing values with the mean, or forward fills, which propagates the last existing value into the future. While these approaches may work in practice, they do not make much sense in the context of sentiment analysis. The approach taken here is to introduce a discount factor γ_i for each company i, which signifies the diminishing strength of the sentiment signal. This approach is inspired by Atkins et al. [4], who introduce a decay function as a way to weight the importance of news events towards the more recent time. In this manner, if there are some significant news on day t for company i, but no significant news on day $t + 1$, the sentiment value can still be calculated in the following manner:

$$SENT_{i,t+1} = \gamma_i SENT_{i,t}$$

It should be noted that the sentiment signal converges towards zero as the number of days without relevant news increases, which makes sense logically. Also, γ_i depends on each individual company, so it is a hyper-parameter that can be optimized. The logic here is that the pace at which a company becomes irrelevant in the news might depend on the company itself and characteristics like its sector, size, etc. Alternatively, γ_i could be replaced by one factor γ that is deployed for all companies.

3.6 3-step procedure prior to model fitting

Before building the excess returns model, the following three steps should be taken into consideration to inform my modelling choices:

1. Test for stationarity of target time-series

2. Test for significant auto-correlations in the target series (excess returns)

3. Test for significant correlation between different series

While these steps are necessary to build a VAR model effectively, they can also be used synergistically for building a machine-learning model. For instance, the tests for significant auto-correlations and cross-correlations inform about how far the model should look into the past to make inferences about the present. This is also something that may vary per company, and studying these outcomes allows me to reason about the effect sentiment has on excess returns. Here, I describe these three steps in more detail.

3.6.1 Testing for stationarity in the target series

An important requirement for the VAR is that both time-series have to be stationary, meaning they have a constant mean and variance, no trends and no seasonality. To test if this is the case, we conduct the Augmented Dickey Fuller (ADF) test. This tests for the following hypotheses:

$$H_0 : \textit{Time-series is not stationary}$$

$$H_a : \textit{Time-series is stationary}$$

3.6.2 Testing for significant auto-correlation in the target series

Auto-correlations inform us on whether or not lagged excess returns ϵ_{t-s} contain predictive power on the excess returns at present ϵ_t. It is common practice to inspect the auto-correlation function graphically before fitting a VAR model, and the significant auto-correlations inform about which lags to include for the AR component of the VAR model. For the machine-learning approach, they also inform about which lagged excess returns the model should utilize as feature. The mathematical details of the underlying process, the Yule-Walker Equations, is omitted here. The curious reader is referred to [9] for more detail. The implementation for this paper is done using the scientific computing library *SciPy*. Instead of plotting the entire auto-correlation function for each company, I only highlight the significant lags.

3.6.3 Testing for significant cross-correlation between the two time-series

Cross-correlation analysis is useful to determine if there is any relation between our two series, the excess returns and sentiment signal, and if so at which lag. Given two time-series Y_1 and Y_2, we want to measure the correlation between our (non-lagged) target series $Y_{1,t}$, which in our case are the excess returns, and different s-lagged version of the predictor series $Y_{2,t-s}$, in our case the sentiment signal. So, for any given lag s, I want to measure the Pearson correlation coefficient:

$$\rho_{Y_{1,t},Y_{2,t-s}} = \frac{Cov(Y_{1,t}, Y_{2,t-s})}{\sigma_{Y_{1,t}}\sigma_{Y_{2,t-s}}}$$

3.7 Excess returns model selection

This section describes how I select and optimize the excess returns model. First, I highlight the choice between modelling excess returns predictions as a regression or classification, and justify why I chose a regression setting for this paper. Second, I describe the VAR model. Third, I explain the Random Forest regressor. Fourth, I highlight the evaluation metrics used in this paper and their utility.

3.7.1 Regression vs. Classification

Excess returns can be modelled either as a regression problem, where the excess return value is predicted directly, or as a classification problem, where the direction (up or down) or sign (positive or negative) of excess returns is predicted. As the literature review shows, the classification approach is generally the more popular one. In this paper, I still choose to pose the problem as a regression problem. This is done because predicting only the direction of excess returns causes the loss of a very important piece of information, which is the magnitude of the change. For instance, an excess return of 0.01% and 10% would have the same classification target (up), but these two values likely have vastly differing underlying reasons. I want the model to at least capture some of these dynamics, which I believe is better done by posing a regression problem.

3.7.2 Vector Autoregressive Model

The Vector Autoregressive (VAR) model is a linear time-series model that, unlike typical AR-models, can include exogenous variables. In the VAR, each variable is described as a linear combination of past values of itself and past values of the other variable in the system. Since here we have two time series (excess returns and sentiment), the VAR model can be described as a system of two equations. Mathematically, the VAR model can be described as follows:

$$Y_t^i = \alpha^i + \sum_{s=1}^{l} \beta_s^i Y_{t-s}^i + \epsilon_t^i$$

where Y_i^t is the time-series signal (sentiment and abnormal returns) for series i at time t, A_i is a time-invariant constant for each series, l is the length of the looking-back window, β_s^i is the coefficient matrix for the s-lag vector Y_{t-s}^i and ϵ_t^i is the error term. As series i can here be, for instance, the excess return series or the sentiment series. One benefit of the VAR over most machine-learning approaches is its interpretability. Like in a linear regression, the β coefficients can be read as a indicator of the strength of the effect that different lags of the series have on the output variable. Also, results by Husani [17] show that VAR serves as a good trend line for trading and it has been successfully applied by other scholars in a sentiment-related context [26].

3.7.3 Random Forest Regressor

As an alternative to the traditional VAR approach, machine learning methods can also be utilizes to establish the link between the sentiment signal and excess returns. For instance, Support-Vector-Machines (SVM) and tree-based ensembles like Random Forest and XGBoost have an ability to deal with time-series data ”out-of-the-box”, meaning their architecture does not need to be modified, and can learn which lags are important through their respective learning processes. This can also be achieved through more complex methods like Long Short-Term Memory neural networks, which are outside the scope of this thesis. In contrast to the VAR, these method require less manual decision-making, although understanding the data and its relationship certainly help to inform modelling decisions.

Another major advantage of machine learning methods, unlike the VAR, is that they can capture complex, non-linear relationships. In this thesis, I consider a Random Forest regressor as representative of the machine-learning based approach to modelling, as opposed to statistical models like the VAR. I choose Random Forest due to its well-known ability to achieve state-of-the-art results without much fine-tuning efforts. The Random Forest Algorithm is a tree-ensemble method that was first described by Leo Breiman in 2001 [7]. In the supervised ML set-up, we are given a dataset $\mathcal{D} = ((x_1, y_1), ...(x_n, y_n))$. We can describe the feature space as $\mathcal{F}$. One fundamental principle underlying Random Forest is bootstrap aggregation, i.e., bagging, where each sub-tree is grown on the basis on a random subset (with replacement) of the training set. Bagging limits the variance of each individual tree and hence combats over-fitting. Algorithm 1 below describes the Random Forest Algorithm in pseudo-code. As Breiman notes, a good method to reduce the variance of the model is to replace regular decision trees with weak learners, which can be decision stumps (decision trees of length 1) or decision trees of a fixed depth. Hence, the depth of the decision tree d is another hyper parameter that can be optimized. Next to this, I also optimize for the number of trees in this thesis.

Algorithm 1 This algorithm describes the Random Forest regressor algorithm. The algorithm takes a bootstrapped sample (with replacement) from the dataset and samples a random subset of features. From these samples the algorithm constructs sub-trees. The procedure is repeated a number of times and the average prediction of the sub-trees represents the regression output.

```
for i=1, ...., B do
    Choose boostrap sample $\mathcal{D}_i$ from $\mathcal{D}$
    Choose random subset of $m$ features, $\mathcal{F}_i$
    Construct tree $\mathcal{T}_i$ using $\mathcal{D}_i$ splitting only on features in $\mathcal{F}_i$
end for
Given new instance $x$, take the average over $T_1, ..., T_B$
```

3.7.4 Model Evaluation

Error metrics

In order to have some degree of comparability between our models, I need to establish common error metrics. In this paper, I want to consider two errors that inform about *absolute* deviance and *relative* deviance. The measure for absolute deviance is the

Root-Mean-Squared Error (RMSE), which measures and sums up the squares distance between any two points and takes the root of the overall sum. The benefit of RMSE is that its unit is the same as our output space, which is percentage points. The RMSE also more strongly penalizes points that are further away, as compared to points that are closer, due to its squared component. Mathematically, we can describe the RMSE as follow:

$$RMSE = \sqrt{\frac{\sum_{t=1}^{T} (\hat{y_t} - y_t)^2}{T}}$$

where t represent each time-step and T the total number of time-steps we want to predict. $\hat{y_t}$ and y_t represent the predictions and actual values, respectively.

To quantify relative deviance, which is sensitive to changes in the scale of the underlying data, I utilize the Mean-Average-Percentage Error (MAPE). The unit of the MAPE is percentage (this reader is to note the difference between percentage points and percentage). This formulation describes the MAPE:

$$MAPE = \frac{100\%}{T} \sum_{t=1}^{T} |\frac{y_t - \hat{y_t}}{y_t}|$$

Walk-forward validation

Given the fact that I am dealing with time-series data, I need to employ a special mechanism called *walk-forward validation* when training and validating the model. Walk-forward validation is a method to split our data into training and validation sets along the time axis without allowing test-set leakages [8] from the future or present into the past. Pardo [22] lists as main benefits of using walk-forward analysis in designing trading strategies their ability to do well in real-time trading environments, provide measurement of the robustness of trading strategies and the ability to adapt to changing market conditions. There are slight difference in how walk-forward validation can be implemented for the VAR and machine-learning models, which are highlighted here.

Walk-forward validation for machine learning
I first split off one rolling window of size 100 (meaning 100 days). Within this window, I make a division of 70 percent training data and 30 percent validation data. I train the model on the 70 training instances and compute an error metric on the validation samples. Then, in order not to loose the validation data, I also give it to the model for training. Finally, I move the rolling window to encompass the next 100 days and repeat the procedure.

Walk-forward validation for VAR
To apply walk-forward validation to the VAR model, I need to make slight adjustments

[8] Test-set leakages means that information from the test set accidentally shows up in the training sample, for example by averaging over time.

in the implementation. As VAR is a statistical time-series model that should consider all available data-points until the current point in time, the window-shifting approach is replaced by a window-enlargement approach. This means that each time I make the window larger, I initialize a new VAR model that receives the entirety of the new window as input. Instead of taking a validation percentage (e.g., 30 percent), I take a fixed validation size of 5 (i.e., the last 5 instances of the enlarged window).

3.8 Trading Strategy

Algorithmic trading nowadays accounts for around 85% of market trading volume [4]. It is of great practical value to be able to apply the results found in this paper in a simulated trading environment. This shows an indication of the practical utility of the sentiment pipeline and excess return model. Hence, the final step in the analysis is to use our machine-learning sentiment signals and the resulting model for each company and apply it to a trading strategy. In order to assess the performance of the strategy, the following Key Performance Indicators (KPIs) are used:

Name	Description	Formulation
Sharpe Ratio	Average return earned in excess of the risk-free rate per unit of volatility	$\frac{R_p - R_f}{\sigma_p}$
Max Drawdown	Maximum observed loss from a peak to a trough of a portfolio, before a new peak is attained	$\frac{ThroughValue - PeakValue}{PeakValue}$

Table 3.1: **Trading Strategy KPIs.** *This table shows the performance indicators used to measure the quality of our trading strategy and their mathematical formulation.*

The strategy employed in this paper is a simple long-short strategy that utilizes the one day-ahead forecast, incorporating both sentiment model and excess returns forecasting model. For each stock, I use the sentiment model to generate a sentiment value for each day for which there is textual data available. To impute missing values, I use the time-discounting factor γ_i, which is optimized for each stock through experimentation. Then, I construct the excess returns model, which receives the lagged sentiment values and lagged excess returns as its input and predicts the excess returns at time t.

I also utilize the returns as predicted by the Fama-French-Carhart model. Denoting the predicted returns of the excess return model and Fama-French-Carhart model for company i at time t as $m_{i,t}$ and $f_{i,t}$, respectively, we can construct the predicted close price in the following manner:

$$Predicted\ Close_{i,t} = Close_{i,t-1} * (1 + f_{i,t} + m_{i,t})$$

This means that the predicted close price for the today is equal to the close price yesterday, multiplied by any returns the Fama-French-Carhart model would predict **plus** the excess returns predicted by my model. After making these predictions for each day t, each morning the trading strategy will compare $\textit{Predicted Close}_t$ against the the current open price $\textit{Open}_t$. If $\textit{Predicted Close}_t$ is larger than $\textit{Open}_t$ plus some error margin ω, I go long in the stock. I fix ω at 1%. If $\textit{Predicted Close}_t$ is smaller than $\textit{Open}_t$ minus ω, I take a short position against the stock. If none of these criteria is met, I just hold the current position. In the case that I am currently holding the opposite position (e.g., I am short but the model indicates me to go long), the current position is liquidated and the opposite position is taken. I trade stocks individually with an available capital of 10,000 EUR each.

To ensure 'real-world' conditions, I only train the model on the first four year of data, 2017-2020, and test the strategy on the year 2021 .The model is adjusted by using walk-forward methodology when casting the predictions for 2021. Hyper-parameter configurations and time-discounting are chosen according to the outcomes of the experimental results.

To see if the trading strategy benefits from adding the sentiment features, I construct a benchmark strategy. In this strategy, I utilize models that only consider lagged excess returns as features, omitting the sentiment.

Chapter 4

Results

This section shows the results of this paper. First, I illustrate the outcomes of the web-scraping operation, highlighting important characteristics like the number of articles scraped, the different news providers, the amount of articles scraper per company and the proportion each company is the main subject in the articles where it is mentioned, as described by the headline heuristic in the methodology. Second, I explore both the extracted sentiment signals and the excess returns dataset. Based on these explorations and their implications, I limit the stock selection to four companies, which are Apple (AAPL), Boeing (BA), Goldman Sachs (GS) and JP Morgan (JPM). Third, I show the effects of time discounting using the γ-factor introduced in the methodology. This is done both visually and via experimentation. Fourth, I study the three steps in preparation of building the model, which are testing for stationarity in the target series, testing for significant auto-correlations and testing for significant cross-correlations between sentiment and excess returns. I contrast the VAR model with the Random Forest regressor based on the results of walk-forward validation. Finally, I show the outcome of applying the resulting model and time-discounting configurations in a trading strategy.

4.1 Web scraping

Here, I discuss the textual data that has been scraped from the *Investing.com* platform and some of its characteristics. In total, there are 29,787 articles scraped, according to the method outline in the methodology section, including their date, headline and the ticker of the company that is mentioned in it. Some articles, for instance market overviews, can have multiple companies associated with them. Of these 29,787 articles, 3,589 are given as not-a-number (NaN), meaning that either the parser was not able to extract the text properly, which can sometimes happen as different providers structure their HTML tags [1]differently, or it was blocked by the platform itself (e.g., some providers like *CNBC* link their content on *Investing.com* but it requires to go to an external page to view the full content). 43 articles had issues with the date. While the latter needs to be discarded, we can still utilize the headlines of those articles that

[1] There are different HTML tags like headlines (h1), divisions (div) and paragraphs (p), which can be named. Web scrapers work by looking for associations between names and tags, but sometimes there can be inconsistencies in how the website is designed, which makes it more difficult to scrape. The web-scraping process is described in more detail in the appendix.

could not be fully scraped as a proxy. The full dataset, as well as datasets associated with intermediate steps, can be found on my GitHub repository. [2]

4.1.1 News providers distribution

As Figure 4.1 below shows, the majority of news on the *Investing.com* website is provided by *Reuters* and *Investing.com* itself. The authors have been identified by parsing the first sentence of the article, which usually mentions who it is written and edited by. The residual category 'unknown' bundles a number of smaller news outlets as well as articles by individual authors. While some diversity in the news providers is desirable to eliminate provider-specific biases, it is generally positive that *Reuters* is so pre-dominant on the platform, since it is known to be a quality news provider that adheres to high editorial standards. As the reader discovered in the literature review of this study, *Reuters* articles also have a long history of being used for sentiment analysis by scholars in both Natural Language Processing and Asset Pricing. I have manually checked some of the articles written by individual authors and get the impression that the quality of the writing fluctuates strongly compared to an established news outlet like *Reuters*. I could either choose to eliminate these texts from lesser-known outlets or keep them in the sample. Here, since I want to prioritize variance and want limit the impact of sparsity (which would only increase by deleting instances), I choose to keep all data providers in the sample.

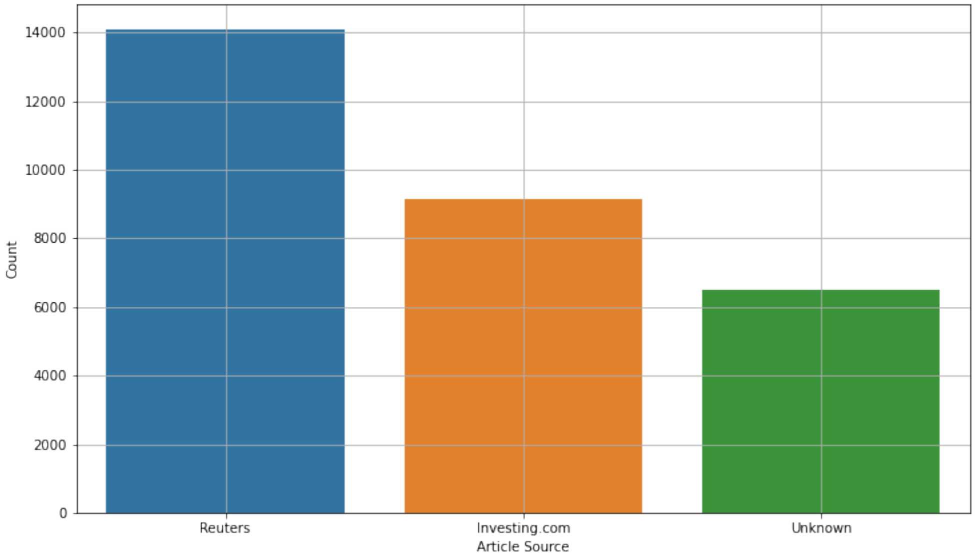

Figure 4.1: **Article Source Distribution.** *This figure shows the amount of articles per news providers on Investing.com. It can be seen that Reuters provides around 14,000 articles of the sample, while Investing.com itself provides around 9,000. The category Unknown subsumes smaller outlets and independent writers, which together provide around 6,200 articles.*

[2] https://github.com/sebkeil

4.1.2 Amount of articles per company

Figure 4.2 shows the amount of articles that is available for each company. It can be seen that Apple (AAPL) has by far the most articles available on the platform, around 7000 instances, which reflects the popularity of the company and the fact that it is of high interest to news writers. The next most-popular companies are Boeing (BA), Goldman Sachs (GS), JP Morgan Chase (JPM) and Microsoft (MSFT), which have between 1000 and 4000 instances. The majority of companies are represented by less than 500 instances, highlighting that they tend to be largely ignored by the news unless there are specific events relating to it. Seeing that there are such large differences in the amount of news for companies already means that it quite likely that a single item of news has a different impact depending on the company. For Apple (AAPL), which is in the news all the time, one piece of news likely matters less than a piece news does for a company like Coca Cola (KO), which rarely receives any mentions.

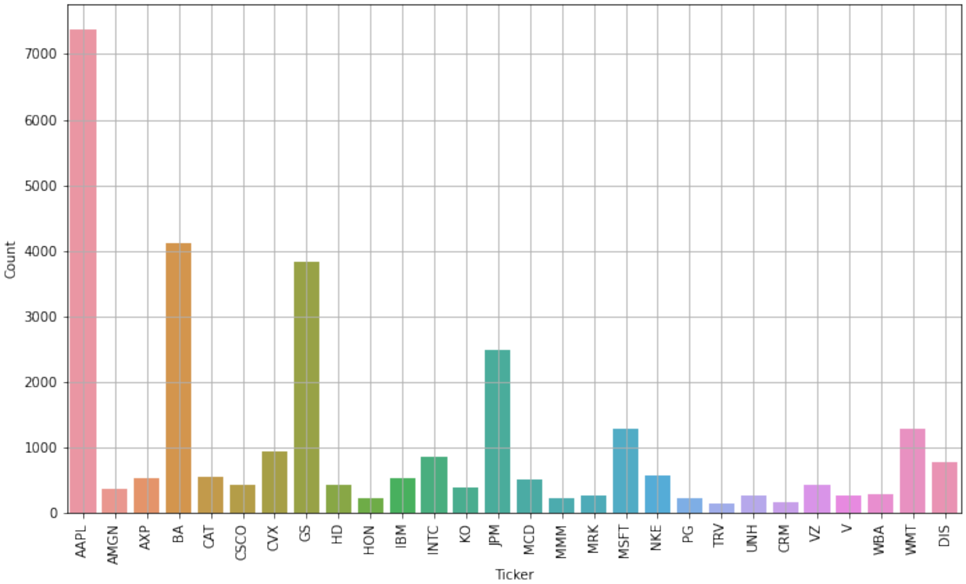

Figure 4.2: **Amount of articles per company.** *The figure shows the number of articles scraped between 2017-2021 for each of the Dow Jones Industrial 30 companies. Apple (AAPL) is by far the most prominent company, followed by Boeing (BA), Goldman Sachs (GS), JP Morgan (JPM) and Microsoft (MSFT). Most companies are represented by less than 500 instances.*

4.1.3 Main subject heuristic

Another variable of interest is the extend to which each ticker is the main subject in the articles which it is associated with. This implies whether I extract the entire article or just the headline for the sentiment measurement. As discussed in the methodology section, whether or not a company is defined as the main subject of the article is found by checking if the company name or ticker appears in the headline of the article. The idea behind using this filter operation is that I don't want the sentiment of a market overview article to represent the sentiment of a company if that company was merely mentioned in the market overview. Figure 4.3 below illustrates the proportion each company. It can be seen that only a few companies are the main subject in more than half of the articles they are mentioned in. Some companies (e.g., McDonalds (MCD)) are not even the main subject once, meaning they had no articles dedicated only to them in the sample. The fact that some companies are rarely have their dedicated articles makes it imperative to extract the sentences in which these companies are mentioned as a proxy.

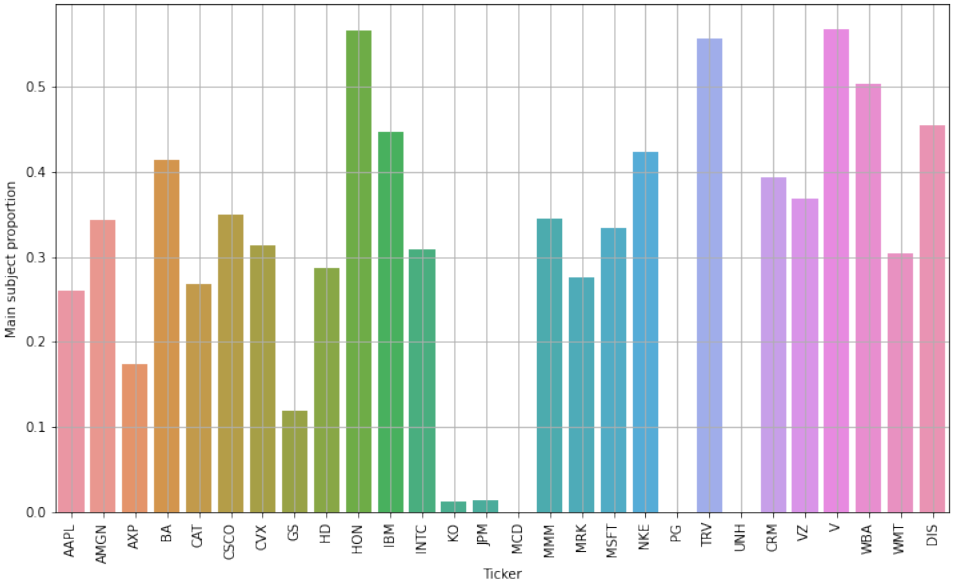

Figure 4.3: **Proportion of company being the main subject.** *The figure shows the proportion each company is the main subject in its referenced articles. Investing.com references companies in market overview if they are mentioned there, but I do not want the sentiment of an entire market overview to represent the sentiment of the mentioned company. It can be seen that there is a lot of variation between companies, some being the main subject in around 50 percent of the sample while others being closer to 0.*

4.2 Data exploration

4.2.1 Sentiment data

The next step in my analysis is to compute the continuous time-series signal for all 30 Dow-Jones Industrial companies. The sentiment values are computed from raw text using the *FinBERT* model as outlined in the methodology. Table 4.1 shows summary statistics for these sentiment signals, already aggregated to a daily level, and displays the sparsity ratio [3] for each company. In the Appendix A.1 the reader can also find a visual depiction of the extracted sentiment over time. It can be observed that the

[3] The sparsity ratio refers to the proportion of days in which the company is **not** mentioned in the news.

sparsity ratio is well above 80 percent for most companies, meaning that these do not receive any mention in the mainstream news for the majority of days. This is certainly a major issue, as it is hard to imagine to conduct a meaningful analysis if around four-fifth of the time-series has to be imputed using time-discounting. The sparsity will likely be the bottleneck for a decent model and trading strategy performance, as sparse series generally provide less information than dense series. The companies that have a sparsity-ratio below 50 percent are Apple (AAPL), Boeing (BA), Goldman Sachs (GS) and JP Morgan (JPM).

The mean sentiment for most companies is slightly negative but close to zero, indicating that the overall sentiment in the given time-frame is slightly negative to neutral. This may reflect a certain bias towards neutral language in the news, especially from providers like *Reuters*. The standard deviation is around 0.5 for most values, while the sentiment ranges between -0.97 to 0.95. The reader is reminded that while these summary statistics do reflect the nature of the textual data, they are also the result of modelling choices that have been made. For instance, the range (-1, +1) is manually defined and could be replaced by other values. But even mapping to another scale would not change the underlying distribution of positivity/negativity in the news, they would only be represented by different values. Most variation between companies is shown in the 25 percent and 75 percent confidence intervals, while the median, maximum and minimum values barely deviate.

	count	mean	std	min	25%	50%	75%	max	sparsity
AAPL	1311	-0.070	0.458	-0.973	-0.351	0.000	0.181	0.954	0.282
AMGN	249	-0.038	0.524	-0.974	0.000	0.000	0.000	0.944	0.864
AXP	281	-0.022	0.542	-0.974	-0.223	0.000	0.000	0.956	0.846
BA	971	-0.123	0.528	-0.974	-0.484	0.000	0.035	0.955	0.468
CAT	295	-0.026	0.564	-0.974	-0.353	0.000	0.004	0.951	0.838
CRM	77	0.032	0.579	-0.975	0.000	0.000	0.506	0.956	0.870
CSCO	238	-0.023	0.621	-0.974	-0.483	0.000	0.422	0.954	0.751
CVX	454	-0.054	0.589	-0.974	-0.484	0.000	0.000	0.954	0.361
DIS	324	-0.046	0.612	-0.974	-0.498	0.000	0.113	0.955	0.879
GS	1167	-0.036	0.421	-0.973	-0.243	0.000	0.077	0.956	0.923
HD	221	0.015	0.620	-0.973	-0.454	0.000	0.473	0.955	0.850
HON	141	-0.001	0.675	-0.969	-0.780	0.000	0.739	0.951	0.768
IBM	274	-0.043	0.540	-0.974	-0.007	0.000	0.000	0.957	0.877
INTC	423	0.023	0.600	-0.973	-0.119	0.000	0.474	0.954	0.490
JPM	932	-0.041	0.474	-0.975	-0.316	0.000	0.070	0.955	0.838
KO	224	0.071	0.559	-0.974	0.000	0.000	0.494	0.954	0.933
MCD	296	-0.088	0.547	-0.973	-0.515	0.000	0.000	0.954	0.905
MMM	122	-0.063	0.477	-0.971	0.000	0.000	0.000	0.946	0.677
MRK	173	-0.020	0.487	-0.974	0.000	0.000	0.000	0.953	0.864
MSFT	590	-0.003	0.448	-0.972	0.000	0.000	0.000	0.960	0.925
NKE	249	-0.131	0.681	-0.975	-0.921	0.000	0.366	0.955	0.949
PG	137	0.079	0.589	-0.971	0.000	0.000	0.671	0.952	0.899
TRV	94	-0.058	0.751	-0.970	-0.965	0.000	0.887	0.917	0.958
UNH	184	-0.021	0.722	-0.973	-0.898	0.000	0.764	0.949	0.879
V	116	-0.156	0.726	-0.972	-0.967	0.000	0.636	0.938	0.936
VZ	221	0.007	0.545	-0.973	0.000	0.000	0.000	0.954	0.922
WBA	142	-0.122	0.623	-0.976	-0.773	0.000	0.000	0.956	0.687
WMT	571	-0.018	0.581	-0.973	-0.472	0.000	0.417	0.953	0.823

Table 4.1: **Sentiment Signal Summary Statistics.** *This table shows the count (in days), mean, standard deviation, minimum, quantile ranges, maximum and sparsity of the constructed sentiment signal for all DJI30 companies in the years 2017-2021.*

In Figure 4.4, a histogram of the sentiment values for all the DJI30 companies (before the application of time-discounting) is shown. It can be seen that the majority of articles are classified as 'neutral', which translates to a sentiment value of zero. The data exhibits strong (tri)modality, with peaks around zero and the most negative (-1) and positive (+1) values.

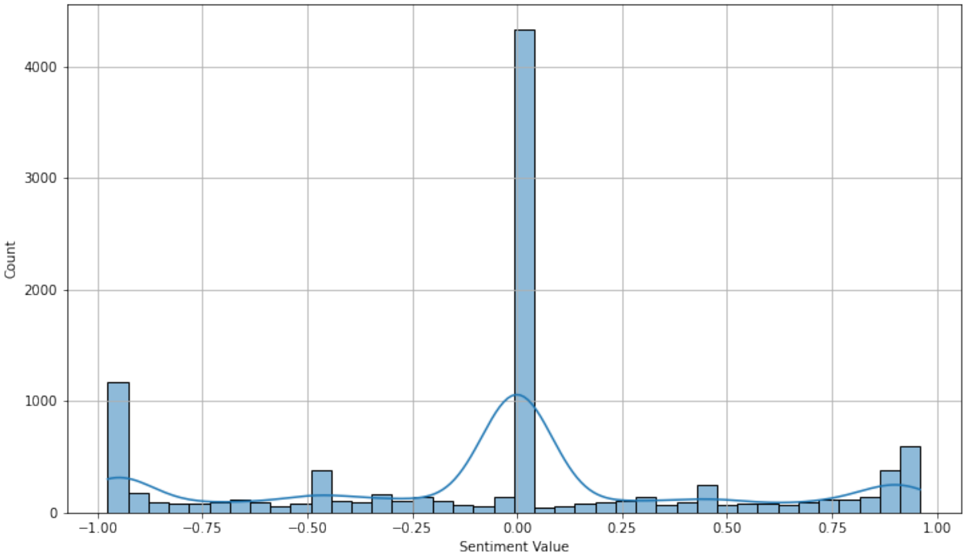

Figure 4.4: **Histogram of Sentiment Values.** *The figure illustrates a histogram of the constructed sentiment values. The data exhibits a dominant peak around zero, and two lesser peaks around the close to -1 and 1.*

4.2.2 Fama-French-Carhart excess returns

Table 4.2 shows the summary statistics for the Fama-French-Carhart daily returns model over the years 2017-2021. The average return is 0.07%, with a standard deviation of 1.8%. This translates to an average excess return, the residual of what the Fama-French-Carhart would have predicted the return to be, of only -0.0002%. While the Fama-French-Carhart is a good predictor of returns on average, the minimum excess return of -15% and maximum excess return of 24% show that the model can miss the actual reality by a substantial margin in some cases. Regarding the four factors, the average β_{Mkt}, the sensitivity to market movements, is close to 1, showing that the Dow Jones Industrial 30 generally gives a good representation of the entire market (the entire market sample would have a β_{Mkt} of exactly 1). The β_{SMB} averages at around -0.18, meaning there is a negative size premium in the chosen sample. The average β_{HML} 0.12 shows that there is a small value premium, the standard deviation is highest here among all the factors with a value of 0.5. The average momentum factor β_{UMD} of -0.04 indicates that stocks that did well in the past tend to perform slightly worse in the future.

	Returns (in %)	α	β_{Mkt}	β_{SMB}	β_{HML}	β_{UMD}	Excess Returns (in %)
mean	0.07	0.00006	0.97	-0.18	0.12	-0.04	-0.00024
std	1.82	0.00066	0.26	0.23	0.50	0.31	1.23230
min	-23.85	-0.00330	0.29	-1.11	-1.31	-1.28	-15.35100
25%	-0.65	-0.00030	0.79	-0.32	-0.22	-0.21	-0.55755
50%	0.08	0.00010	0.98	-0.20	0.04	-0.02	-0.00590
75%	0.83	0.00050	1.15	-0.06	0.40	0.16	0.54255
max	26.04	0.00250	1.76	0.81	1.60	1.28	24.20990

Table 4.2: **Fama-French-Carhatt summary statistics.** *This table shows summary statistics to the Fama-French-Carhart collected for all DJI30 companies in the years 2017-2021. The columns represent the returns (in %), alpha parameters, the beta parameters for the four factors (market, small-minus-big, high-minus-low, and momentum) and the excess returns (in %), which are the residuals $\epsilon_{i,t}$ from the Fama-French-Carhart model.*

Figure 4.5 shows a histogram of both returns and excess returns, overlayed by a KDE plot which approximates the underlying continuous function. According to most standard econometric approaches, returns are assumed to exhibit a Gaussian distribution. Figure 4.5 however shows that returns and excess return distributions have a high degree of kurtosis. Compared to returns, excess returns also exhibit a higher degree of right skewness. To check if the returns and excess returns indeed are not normally distributed, I run a Shapiro-Wilkinson test on the data, we test for the following hypotheses:

$$H_0 : \textit{Data is normally distributed}$$

$$H_a : \textit{Data is not normally distributed}$$

For the return data, we receive a test statistic around 0.845 and a p-value of 0.0. For the excess returns, we receive a test statistic of 0.86 and a p-value of 0.0. This means that for both returns and excess returns we reject the null hypothesis that the data is normally distributed with a high degree of certainty ($\alpha = 0.01$). This implies that standard approaches, such as a Linear Regression, are likely not going to suffice for modelling excess returns. Indeed, it suggests that a machine-learning approach might be more suitable, as it can more effectively capture non-normal distributions.

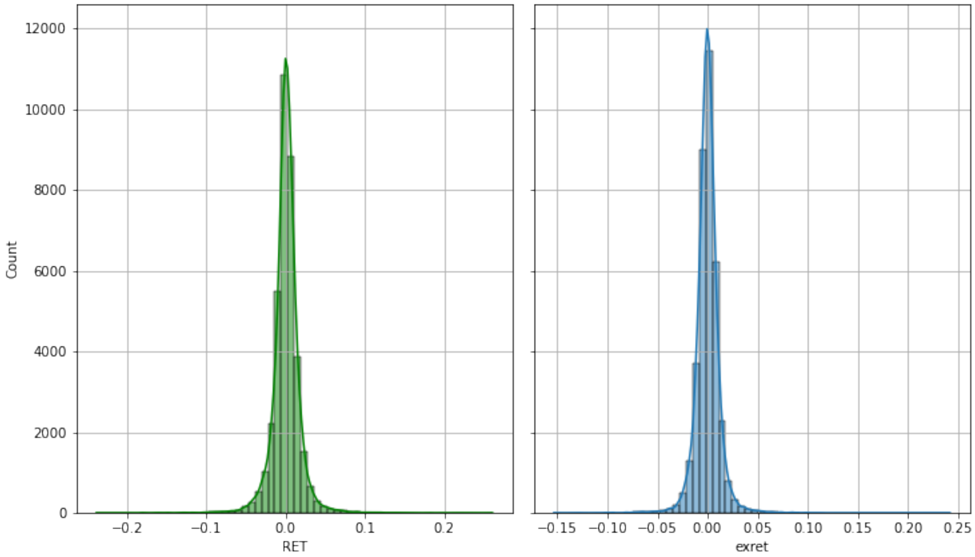

Figure 4.5: **Histogram of Returns and Excess Returns.** *The figure illustrates the distributions of both returns and excess returns. Compared to a normal distribution, we observe a high degree of kurtosis in both series. The excess returns exhibit a higher degree of right-skewness than the returns.*

4.2.3 Adjusted stock selection

As the data shows, the sparsity of for most companies is so high that it will likely be unfeasible to find any meaningful results from them. Hence, we choose to disregard all but the four least sparse companies, which are Apple (AAPL), Boeing (BA), Goldman Sachs (GS) and JP Morgan (JPM). The rest of this analysis focuses only on these four companies.

4.3 Time-discounting

4.3.1 Visual exploration

As described earlier, I introduce a time-discounting factor γ_i to deal with the fact that there are a lot of missing dates in our sentiment time-series signal. This imputation method allows me to transform the original sparse time-series into a continuous series, which is needed to predict the excess return, for which there is no issue with missing dates. This is done because I want to build a model that can be used throughout the entire year. An alternative would be to focus only on regions where there is a lot of sentiment data and study how it impacts excess returns. To give the reader an impression how the effect of such an imputation looks like, Figure 4.7 below shows both the original and imputed sentiment signal for the four selected companies. Here, for illustration purposes, I only show the results using a value of 0.5 for γ and the year 2017. The same graph with other configurations for γ can be found in Appendix A.2 to

A.5. Visually, it can be observed that the imputation method succeeds in generating a smooth, continuous sentiment signal by effectively filling in the 'gaps' that arise from bursts of sentiment at different points in time.

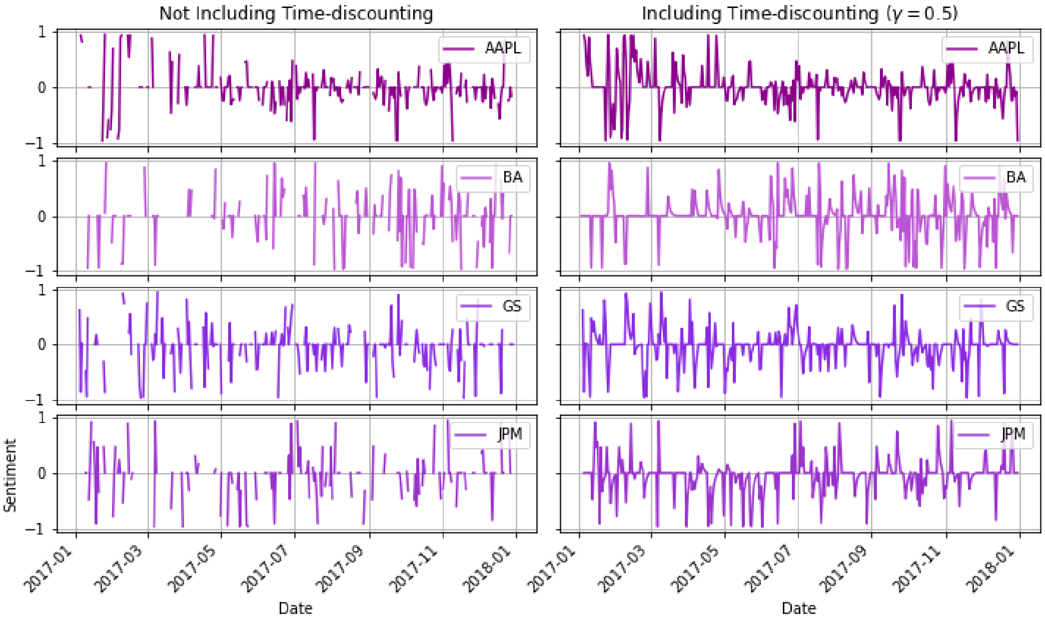

Figure 4.6: **Before-after comparison for time-discounting (selected sample).** *The graph shows the sentiment series for the four chosen companies in the year 2017 before and after applying time-discounting, with a value of* $\gamma = 0.5$.

I want to give the reader an impression on how different values of γ_i influence the behavior of the resulting time-series. To this end, I zoom in on the period starting on March 15th, 2017 for the company Boeing (BA). This section is a sample of a date for which we had one sentiment signal (on March 16th) and then rely on imputation. It can be seen that the behavior of the time-series fluctuates strongly depending on which value I use for γ_i. With a low value of 0.3, the series has a steep decline and flattens after around 3 days. A value of 0.7 causes an almost linear decline. For a high value of 0.99, the series remains nearly unchanged (flat) over the shown sample. In the next section, I show how to optimize these values for each company, by keeping a fixed underlying excess returns model and only changing the values for γ_i.

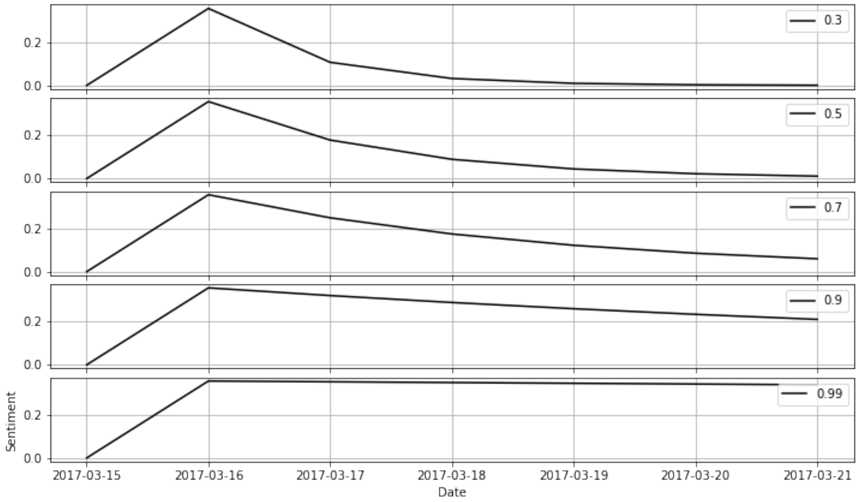

Figure 4.7: **Sentiment signal behavior for different values of γ_i (selected sample).** *The figure shows the behavior of the sentiment signal for five different values of γ_i. The sample chosen for illustration purposes here is from the company Boeing (BA) in the period March 15th, 2017 to March 21st, 2017.*

4.3.2 Exploring correlation under different gamma configurations

To develop a deeper understanding of how the different configuration for γ_i changes the relationship between excess returns and sentiment, Table 4.3 shows the Pearson correlation coefficients between the excess returns series and the sentiment series for different values of γ_i. The first row of this table shows the correlation *without* the usage of any γ_i, meaning we simply concatenate the the time-series for any non-missing dates and compute the correlation at the intersection of both series. The general pattern that can be observed is that the correlation is significant for three companies Apple (AAPL), Boeing (BA) and JP Morgan (JPM), but not for Goldman Sachs (GS). The correlation-coefficient tends to decrease slightly with an increasing value of γ_i, but the magnitude of this change is not strong enough to induce a change in the correlation significance (in parenthesis). This implies that the particular choice of γ_i may not be a strong bottleneck for the overall outcomes of the analysis, as it does not shift the underlying relationship between excess returns and sentiment in a fundamental way. However, it may still be possible that the underlying relationship changes *locally*, meaning during specific time-periods. This is certainly something that deserves future study.

γ_i	AAPL	BA	GS	JPM
None	(0.1407, 0.0)	(0.1995, 0.0)	(0.0489, 0.1289)	(0.1104, 0.0019)
0.3	(0.1284, 0.0)	(0.1784, 0.0)	(0.046, 0.1032)	(0.0872, 0.002)
0.5	(0.1283, 0.0)	(0.18, 0.0)	(0.045, 0.1111)	(0.0836, 0.003)
0.7	(0.1281, 0.0)	(0.1791, 0.0)	(0.0425, 0.132)	(0.0783, 0.0055)
0.9	(0.1274, 0.0)	(0.1741, 0.0)	(0.0385, 0.1728)	(0.0707, 0.0122)
0.99	(0.1268, 0.0)	(0.1695, 0.0)	(0.0361, 0.2007)	(0.0664, 0.0185)

Table 4.3: **ρ-correlation under different configurations of Gamma.** *This table shows the correlation between excess returns and sentiment, both non-lagged, under different configurations of γ_i. In parenthesis, it shows both ρ-correlation and the p-value on the test of the significance of the correlation. In general, correlation appears to slightly decrease with the introduction of time-discounting.*

4.3.3 Gamma optimization

Having developed some intuition on how time-discounting influences the sentiment time-series itself and its relation to the excess returns series, I show here how firm-specific discount factors γ_i can be optimized. To isolate the effect of γ_i, I utilize a Random Forest regressor with fixed hyper-parameter configurations. [4] I also fixed the maximum number of lags (i.e., the look-back period) at 15 to keep conditions fixed and simplified. The table below shows the results for different values of γ_i. These results are generated using the walk-forward methodology described in the methodology. It can be seen that the results vary per company and only change slightly when modifying the γ-value.

For Apple (AAPL), the best validation RMSE is achieved using $\gamma = 0.5$. For Boeing, it is $\gamma = 0.3$, while it is $\gamma = 0.7$ for Goldman Sachs and $\gamma = 0.3$ for JP Morgan. However, the changes in RMSE are only very slight, so it appears that the γ-value does not have a strong influence on the predictive power of our model in absolute terms. The MAPE error shows stronger fluctuations and appears to be more informative. As mentioned before, the RMSE informs on *absolute* deviations, while the MAPE is a *relative* error metric, meaning it varies based on the scale of the underlying predictions. Since I am dealing with very small numbers, and the absolute deviations in terms of RMSE are almost negligible, I want to base my decision-criteria for choosing γ_i on the MAPE here. When using the MAPE as decision criteria, it can be observed that 0.3, 0.7, 0.5 and 0.7 are the best γ_i-values for Apple (AAPL), Boeing (BA), Goldman Sachs (GS) and JP Morgen (JPM), respectively. Hence, we choose these four γ-values for the remainder of the analysis.

[4] Here, I have taken a Random Forest regressor with 1000 trees and a maximum depth of 5

	γ_i	RMSE	MAPE
AAPL	0.3	1.297	3.628
	0.5	1.292	4.787
	0.7	1.302	4.617
	0.9	1.296	4.285
	0.99	1.284	3.761
BA	0.3	2.080	1.927
	0.5	2.090	1.826
	0.7	2.104	1.742
	0.9	2.118	1.797
	0.99	2.126	1.789
GS	0.3	1.220	2.155
	0.5	1.216	2.121
	0.7	1.214	2.122
	0.9	1.224	2.212
	0.99	1.234	2.210
JPM	0.3	0.744	2.036
	0.5	0.747	2.103
	0.7	0.745	1.975
	0.9	0.757	2.221
	0.99	0.758	2.207

Table 4.4: **Gamma optimization validation metrics for fixed RF regressor** *The table summarizes the gamma optimization results for the four chosen companies Apple (AAPL), Boeing (BA), Goldman Sachs (GS) and JP Morgan (JPM), keeping a fixed Random Forest regressor and maximum lags. It shows the company ticker, the γ value and validation RMSE and MAPE errors. The results are generated using walk-forward validation*

4.4 3-step procedure prior to model fitting

As mentioned in the methodology, the three steps taken before the models are (1) testing for stationarity in the target series, (2) testing for significant auto-correlations in the target series and (3) testing for cross-correlation between the target series and lagged version of the predictor series. The outcome of these tests can be used in a synergistic manner for both VAR fitting and the optimization of the Random Forest regressor, by informing *how far back* into the past the model should look to infer excess returns from sentiment. Table 4.5 summarizes the results of these tests, for the four chosen companies. Here, the excess returns at time t is the target variable and lagged versions of the sentiment signal, here limited to the scope $t-1$ to $t-30$, as well as the lagged excess returns, are the input variables. Since the final aim is to construct a short-term trading strategy, I choose to limit the maximum lag we inspect to 30 here. Regarding time-discounting, I choose the γ_i values optimized for in the previous step.

It can be seen that the test statistic for the ADF-fuller for the sentiment series is significant at $\alpha = 5\%$ for all companies, although the magnitude of the test statistic lies in a range of -9.12 (JP Morgan) to -35.59 (Goldman Sachs). Given that the excess returns pass the stationarity test, it is reasonable to construct a VAR model. In case of trends or clusters, the usage of the VAR model would not be advisable, but the machine learning approach would still be feasible. The significant auto-correlation (AR) lags (also at $\alpha = 5\%$) vary strongly between the different companies. For Apple (AAPL), there is a positive auto-correlation at lag 16, meaning that excess returns of 16 days ago may be a predictor for excess returns today. The other three companies show a mixture of positive and negative auto-correlations for excess returns, indicating that there might be reversion effects at hand here. Apple (AAPL) and JP Morgan (JPM) also show no significant cross-correlation between excess returns and lagged sentiment. This shows that it is likely difficult to predict excess returns based on past sentiment. Boeing (BA) and Goldman Sachs (GS), on the other hand, do show significant cross-correlation at various lags, both positive and negative. It can be seen that sentiment values that lie further into the past generally have a negative coefficient, while more recent lags have a positive cross-correlation. This again supports the hypothesis that there might be a mean-reversion effect in how sentiment influences excess returns. Sentiment appears to be driving excess return up in the short-term, and down in the medium-term. For the curious reader, a full version of this table (including all 30 Dow Jones Industrial companies) with a fixed γ value can be found in Appendix A.1.

	ADF-fuller	Sign. AR Lags	Sign. CC Lags
ticker			
AAPL	-34.58 (0.00)	16(+)	24(-)
BA	-18.47 (0.00)	1(+),2(+),3(-),4(-),5(-),6(-),21(-),22(-)	1(+),2(+),14(+),19(-)
GS	-35.59 (0.00)	12(-),27(+)	
JPM	-9.12 (0.00)	2(+),4(-),5(+),11(+),15(-)	

Table 4.5: **Three steps prior to model optimization.** *This table shows the ADF-fuller test statistic (and p-value), the significant auto-correlation lags and the significant lagged cross-correlation between the excess returns and sentiment (measured by the ρ-coefficient) for the four chosen companies Apple (AAPL), Boeing (BA), Goldman Sachs (GS) and JP Morgan (JPM). Significance is determined by the threshold $\alpha = 5\%$.*

4.5 Model Optimization

4.5.1 VAR fitting

Table 4.6 below shows the average validation RMSE and MAPE for the VAR model, according to the walk-forward methodology adapted to the VAR, as outlined earlier in this thesis. The sentiment data is imputed using the γ_i values found earlier and the look-back horizon (i.e., the number of lagged excess returns and sentiment considered as features) is chosen based on the results from the 3-step procedure prior model fitting. The modifications for each company are indicated in the table.

The table shows an average validation RMSE of 1.25 percentage points for Apple (AAPL), 1.7 percentage points for Boeing (BA), 1.635 for Goldman Sachs (GS) and

0.765 for JP Morgan (JPM). The MAPE values are 4.2%, 3.2%, 13.3% and 2.9% for the four companies, respectively. This shows that JP Morgan (JPM) has the overall best predictive performance and Goldman Sachs (GS) the worst in terms of relative deviation. Figure 4.10 shows that the VAR model has difficulties converging, as the validation RMSE moves up and down in a zick-zack pattern. This is an indication that the underlying relationship between excess returns and sentiment changes fundamentally over time, which spikes up the error rate. The model then learns the new pattern, causing the error to fall again. In the MAPE error curve, extreme spikes can be observed. These reflect changes in magnitude in the underlying target series, compared to the model predictions. In this particular case, it reflects the fact that excess returns can suddenly be on a different *scale* if underlying conditions change. Overall, the graphs highlight the difficulty of the model to converge over time. Convergence appears to happen only locally.

	RMSE (in %-points)	MAPE (in %)
AAPL ($\gamma = 0.3$, lags=24)	1.250	4.211
BA ($\gamma = 0.7$, lags=22)	1.722	3.235
GS ($\gamma = 0.5$, lags=27)	1.635	13.343
JPM ($\gamma = 0.7$, lags=15)	0.765	2.947

Table 4.6: **VAR walk-forward validation results** *This table illustrates the average validation RMSE and MAPE, using the VAR model for the four chosen companies Apple (AAPL), Boeing (BA), Goldman Sachs (GS) and JP Morgan (JPM).*

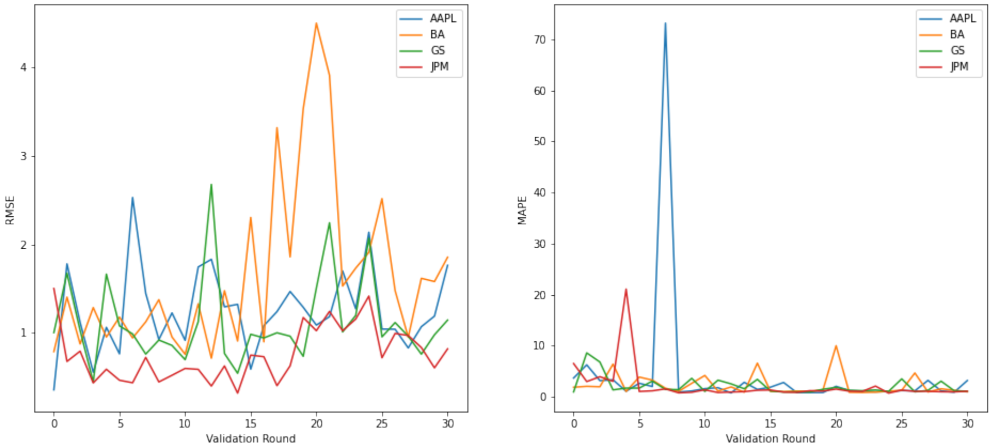

Figure 4.8: **VAR Validation RMSE Curve** *The figure shows the error behavior for the VAR model over time, for the chosen four companies. We see that while the error does go down at times, it keeps rising again and even spiking, indicating that something fundamental has changed in the time-series that the model does not pick up on.*

To give the reader some visual input on what is happening 'under the hood', Figure 4.11 shows the VAR predictions against actual excess return values for a randomly sampled validation window. These graphs show that the model at times gets the direction of the movements of excess returns right, but fails to predict the correct magnitude (e.g., see August 14th for Boeing (BA)). In other times (e.g., August 21st for Apple (AAPL)) the model also misses the direction. From this graph, as well as the previous graph on RMSE and MAPE behavior over multiple samples, it is still difficult to deduce whether or not the model is learning any fundamental pattern or whether it just happens to get it right sometimes due to mere chance.

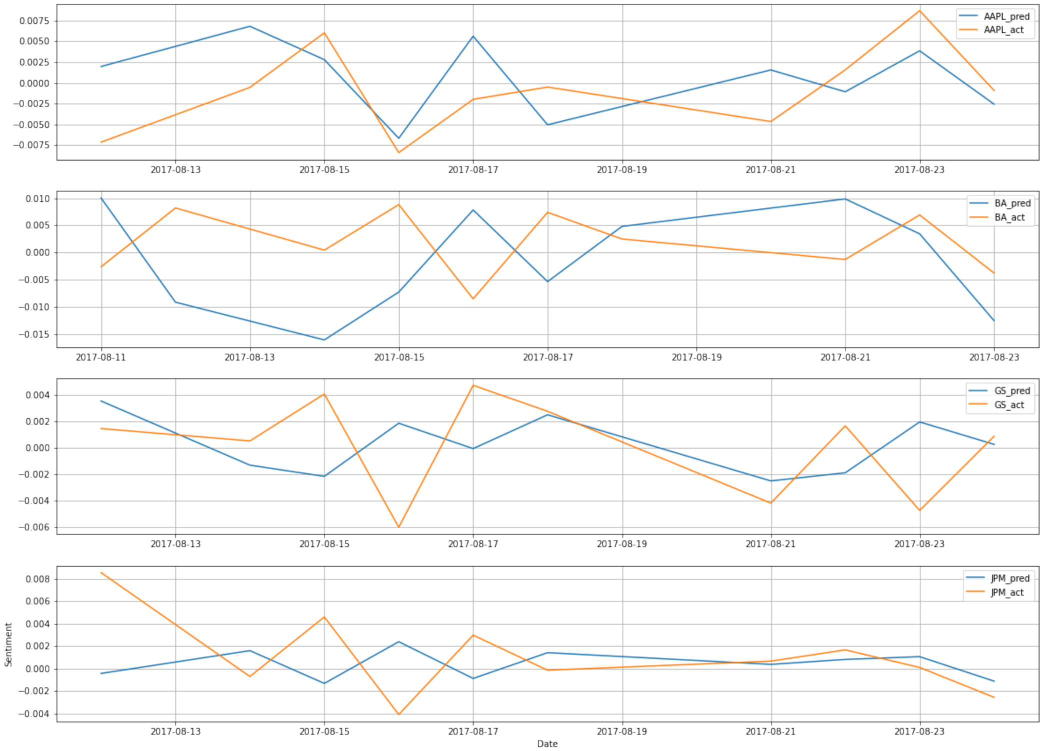

Figure 4.9: **VAR Validation vs. Actuals plot for a randomly chosen validation window.** *The graph indicates the VAR excess return predictions against actual values for the period August 12th, 2017 to August 24th, 2017. Each row shows one of the four chosen companies, the predicted values are indicated in blue and the actual values in orange.*

4.5.2 Random Forest regressor optimization

Here, I want to show the results of fitting a Random Forest regressor instead of a VAR model. Just like for the VAR model, the γ_i values, as well as the number of lags the model looks to the past, are chosen based on previous experimentation. The results have been generated using the walk-forward methodology described earlier in this paper. Here, I only show the outcomes that had to best RMSE and MAPE. In case that one modification had both the best RMSE and MAPE, only that modification is shown. The full table can be found in Appendix A.2. The overarching take-away is that while the VAR appears to perform better in terms of RMSE, meaning absolute terms, the Random Forest regressor does better in terms of MAPE, or relative terms. In other words, the Random Forest regressor better captures extreme variations in excess returns, which drive up the MAPE error in the VAR model.

	Hyper-parameters	RMSE (in %-points)	MAPE (in %)
AAPL	Trees:1500, Max.Depth:5	1.315	3.121
	Trees:500, Max.Depth:10	1.319	3.011
BA	Trees:500, Max.Depth:5	2.143	1.702
	Trees:1500, Max.Depth:5	2.147	1.661
GS	Trees:1500, Max.Depth:5	1.187	2.183
JPM	Trees:500, Max.Depth:5	0.748	2.001

Table 4.7: **Random Forest Regressor Optimization** *The table summarizes the optimization for the four chosen companies AAPL, BA, GS and JPM, indicating both their validation RMSE and MAPE errors. Here, I only show a selection of the best results, the full table can be found in the Appendix.*

Similarly to what I did with the VAR, I show the RMSE and MAPE curves over the validation windows for the Random Forest regressor. Note that the validation windows are larger for the Random Forest approach, due to the modifications of the walk-forward method outlined earlier. It can be observed that similar to the VAR, the Random Forest regressor has issues converging. The error appears to be dropping in a linear fashion, before jumping up again. This tendency even increases over time. Similar patterns can be observed for both RMSE and MAPE error. The interpretation of these results is that even a non-linear model like Random Forest appears to be confronted by underlying shifts in the relationship between sentiment and excess returns, and faces difficulties in modelling these properly.

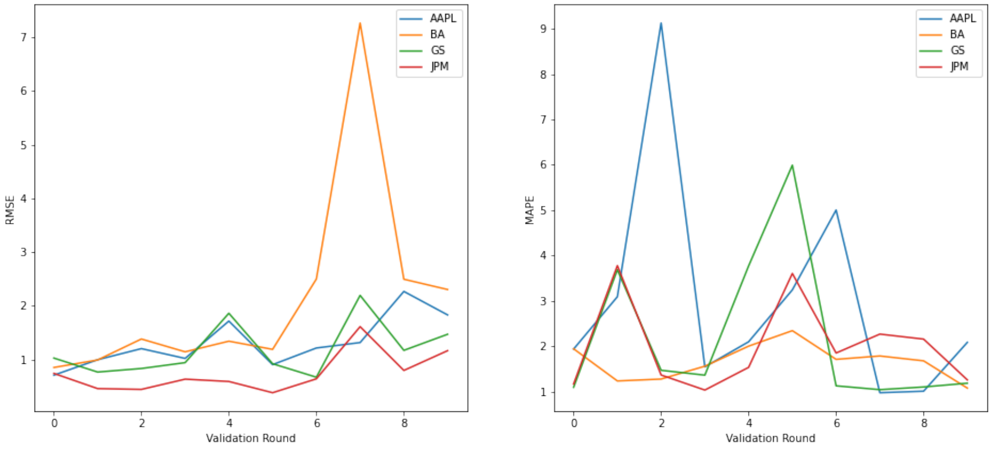

Figure 4.10: **Random Forest Validation RMSE and MAPE Curves** *The figure shows the error behavior, in terms of RMSE and MAPE, for the Random Forest model over time. Displayed are the results for the chosen four companies. It can be seen that while the error does go down at times, it keeps rising again and even spiking, indicating that something fundamental has changed in the relationship between the two time-series that the model does not pick up on.*

To get a deeper understand on how the model predicts excess returns, I again visualize a randomly chosen validation window. Compared to the VAR model, the Random Forest regressor appears to do a better job at capturing the underlying movements of the data. However, the model is still quite off for some predictions and fails to capture many of the spikes in excess returns. Overall, the predictions move more closely around zero, which is the mean of the excess returns series. This behavior is an indication that the excess returns do not really follow any pattern that can be predicted based on sentiment, and so the model learns its best guess: a prediction close to the average.

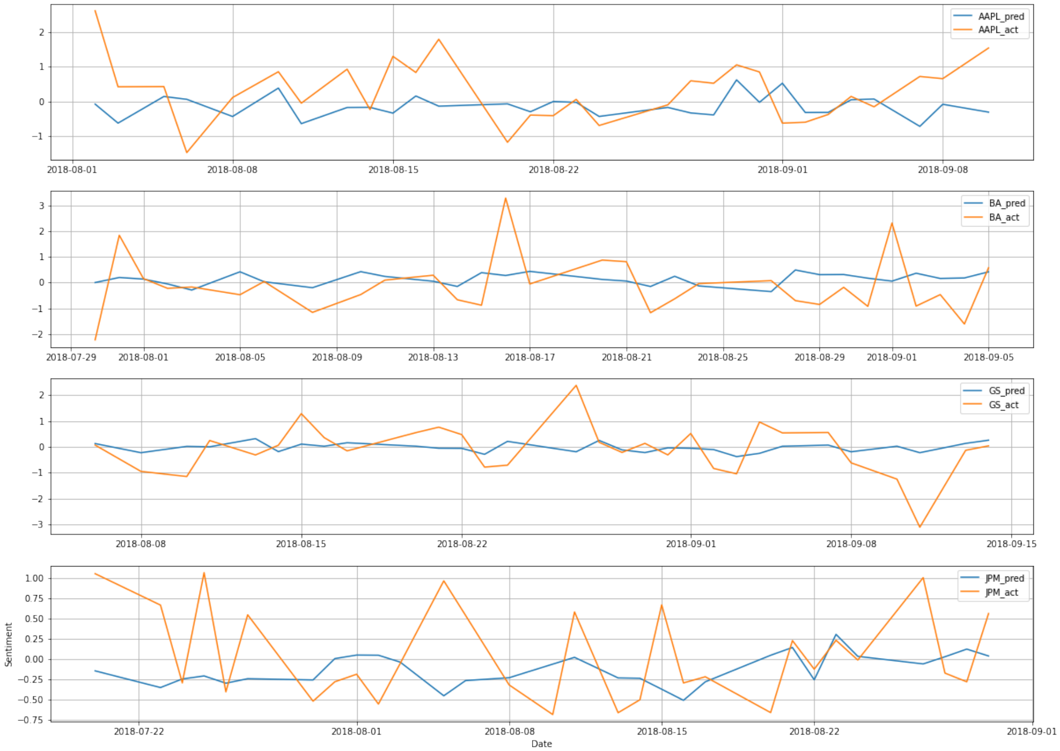

Figure 4.11: **Random Forest Validation vs. Actuals plot for a randomly chosen validation window.** *The graph shows the Random Forest excess return predictions against actual values for the period August 1st, 2018 to September 9th, 2018. Each row shows one of the four chosen companies, the predicted values are indicated in blue and the actual values in orange.*

4.5.3 Model choice

While VAR and Random Forest regressor perform similar in terms of absolute error, quantified by the RMSE, the latter does considerably better in terms of relative error (MAPE). For this reason, and also because the implementation and re-training process is more straight-forward, I choose the Random Forest approach in the execution of the trading strategy. As hyper-parameter configuration, I choose the models that achieved the highest validation MAPE.

4.6 Trading Strategy

The figure below illustrates the results of the trading strategy using the Random Forest regressor. The regressor has been trained for the years 2017-2020 and the strategy is applied to the out-of-sample year 2021, as described in the methodology section above. The model parameter, look-back period and time-discounting values (γ_i) are chosen based on previous experimentation. The model has been re-trained every 5 days, to allow it to capture emerging patterns. As risk-free rate, I assume a rate of 0%, which is in line with the historical global average over the past couple of years. It can be observed that the results of the trading strategy are quite underwhelming, as the strategy consistently looses money for all four chosen stocks. Table 4.8 summarizes the Key-Performance Indicators. It shows both Sharpe ratio and Maximum drawdown for the sentiment-based trading strategy and the benchmark strategy which omits sentiment features (*in parenthesis*). While I only achieve a positive Sharpe ratio for Boeing (BA), the sentiment-based strategy consistently outperforms the benchmark, in both Sharpe ratio and maximum draw-down, indicating that the sentiment features do provide valuable information. The maximum draw-down lies near 200% for all four companies, indicating that strong losses are still possible when relying on this trading strategy. However, the reader is reminded that this strategy is only executed on the year 2021, which has been an extremely turbulent year for financial market due to Covid-19. To better understand the performance, other years should be taken into consideration for future research.

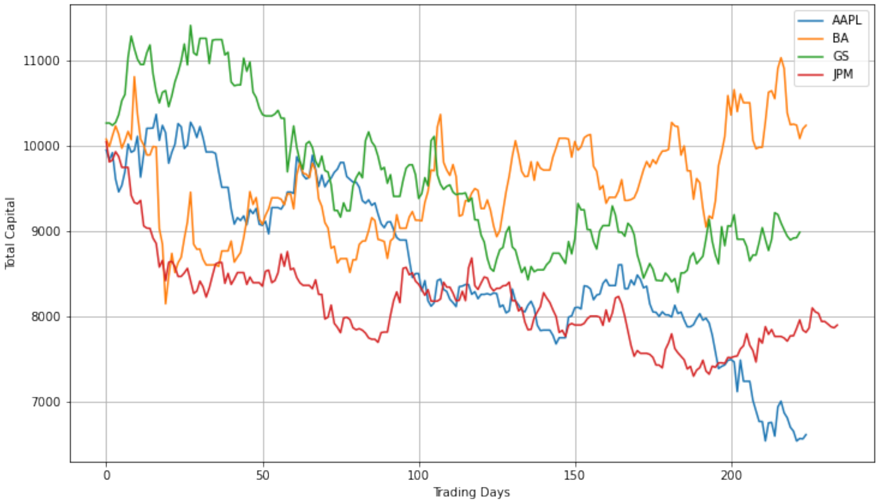

Figure 4.12: **Trading Strategy Results** *The figure shows the amount of capital held at each point in time over the out-of-sample year 2021, for the four chosen companies.*

	Sharpe Ratio	Max. Drawdown (in %)
AAPL	-1.730 *(-1.887)*	-191.076 *(-191.080)*
BA	0.220 *(-0.170)*	-230.790 *(-281.132)*
GS	-0.428 *(-1.161)*	-209.445 *(-209.706)*
JPM	-1.113 *(-1.213)*	-176.068 *(-176.069)*

Table 4.8: **Trading Strategy KPIs.** *This table shows the Key-Performance indicators, Sharpe ratio and Maximum Draw-down, for the four chosen stocks. On the left side, the result for the sentiment-based trading strategy is shown. On the right (in parenthesis) I show the results for the benchmark strategy which omits sentiment features. Except for Boeing (BA), all strategies are loosing, implying a negative Sharpe ratio. The Maximum Drawdown is also rather high for all stocks and strategies.*

Chapter 5

Conclusion and Future Work

5.1 Conclusion

In this paper, I show how to utilize web-scraping and a cutting-edge NLP model to extract sentiment from financial news articles. I then relate this sentiment signal to daily excess returns, finding that there are significant correlations, both lagged and non-lagged, and that these can differ per company. I introduce and optimize a time-discounting factor γ_i and show that while the introduction of such a factor reduces correlations between sentiment and excess returns slightly, its exact configurations matters less than one would assume, as model performance changes only slightly based on different configurations of this γ_i. I show that the statistical approach, represented by the VAR, outperforms the machine-learning approach slightly in terms of absolute error (RMSE), but does worse in terms of relative error (MAPE). Both models have difficulty converging over time, highlighting what might be shifts in the underlying relationship between excess returns and sentiment. In terms of trading strategy, it is not possible to create a reliable, profitable strategy based on lagged sentiment and excess returns, as indicated by negative Sharpe ratios for three out of four stocks. However, the strategy still outperform a benchmark strategy that omits sentiment features, showing that there is at least some utility in the sentiment of financial news for trading in the financial markets.

5.2 Future Work

The results of this paper indicate how difficult it is to predict excess returns, even using what are known to be well-performing models in other contexts. It could just be that the complexity of the financial markets is so enormous that it is impossible to predict excess returns with a high degree of accuracy. However, there are also several potential bottle-necks in this paper that could potentially be improved in future research. These are described in some detail here.

The first bottle-neck is the dataset itself. As was shown earlier, there are a lot of missing dates, meaning days where there are no news published about a company, resulting in a sparse time-series. This was the case on the retail-investment platform *Investing.com*, but there are other sources of textual data one could consider for modelling sentiment. For instance, future scholars can look to social media chatter in the absence of major news publications in an attempt to gauge the idea of crowd sentiment. An alternative approach could be to closely monitor the textual data produced by people close to the company, or investors. The fact that most companies were rarely in the news most of the time also warranted a drastic reduction of the company sample used for this thesis. Given less missing dates, more different companies could also be investigated and the effect of sentiment on excess returns can be compared in the cross-section.

The second bottle-neck is the sentiment model. Here, I have chosen the *FinBERT* model, which claims to outperform other cutting-edge models on financial news classification. However, the idea behind this thesis is that any sentiment model can replace this particular choice, so future research should investigate closer how the predictions of different models changes the resulting sentiment signal, and how that changes the relationship to excess returns.

Third, the time-discounting method introduced in this paper is a major bottle-neck. Although the correlation analysis showed that it does not materially affect the relation between excess returns and sentiment, in an ideal scenario a study can be conducted without time-discounting.

The fourth bottle-neck is the excess returns model. Although I spend some effort on finding and optimizing an appropriate model, much more could be done in this regard. For instance, one could leverage novel deep-learning approaches that achieve peak performances on other tasks for relating sentiment to excess returns.

Another issue is related to the Fama-French-Carhart model. It is by no means guaranteed that there are no correlations between the factors and sentiment. For instance, the size premium might already subsume certain sensitivities to sentiment, making it an inefficient filter to isolate the sentiment aspect of returns. Future studies should closely examine whether such correlations exists and what this implies for modelling the notion of excess returns.

The final bottleneck is the trading strategy. Here, a very simple (non-contrarian) strategy is chosen, but there are a lot more ways this strategy could be modified. For instance, one might construct a strategy based on the mean-reversion effect of sentiment or study how to leverage sentiment in a contrarian manner. Another approach could be to construct portfolios based on company sensitivity to sentiment and go long/short in these portfolios instead of individual stocks.

Appendix A

Appendix

A.1 Web Scraping Algorithm Explanation

The web-scraping algorithm is implement using the Python packages *BeautifulSoup*, *Pandas*, and *requests*. First, I define a dictionary that maps each stock ticker to the name it is defined as on the *Investing.com* platform. For instance, Apple (AAPL) is defined as 'apple-computer-inc' on the platform, and the Apple news page can be accessed via 'https://www.investing.com/equities/apple-computer-inc-news/'. This basic pattern hold for all companies. I use the *requests* library to connect to the platform. I loop through all of these company names, and start scraping articles by checking for the html tags. For instance, each article title has a named 'title' tag, which belongs to the 'a' html class. The article itself is named 'WYSIWYG articlePage' and belongs to the 'div' class. After extracting the headline, text and date for each article, I store it in a *Pandas* dataframe. For each article, I first check the date and if the date goes below 2017, I stop the scraping operation for the current company and move to the next. The reader is referred to my GitHub page [1] and can also contact me via e-mail [2] for any questions relating to the code.

[1] https://github.com/sebkeil

[2] basti.keil@hotmail.de

A.2 Sentiment Signal DJI30 Graph

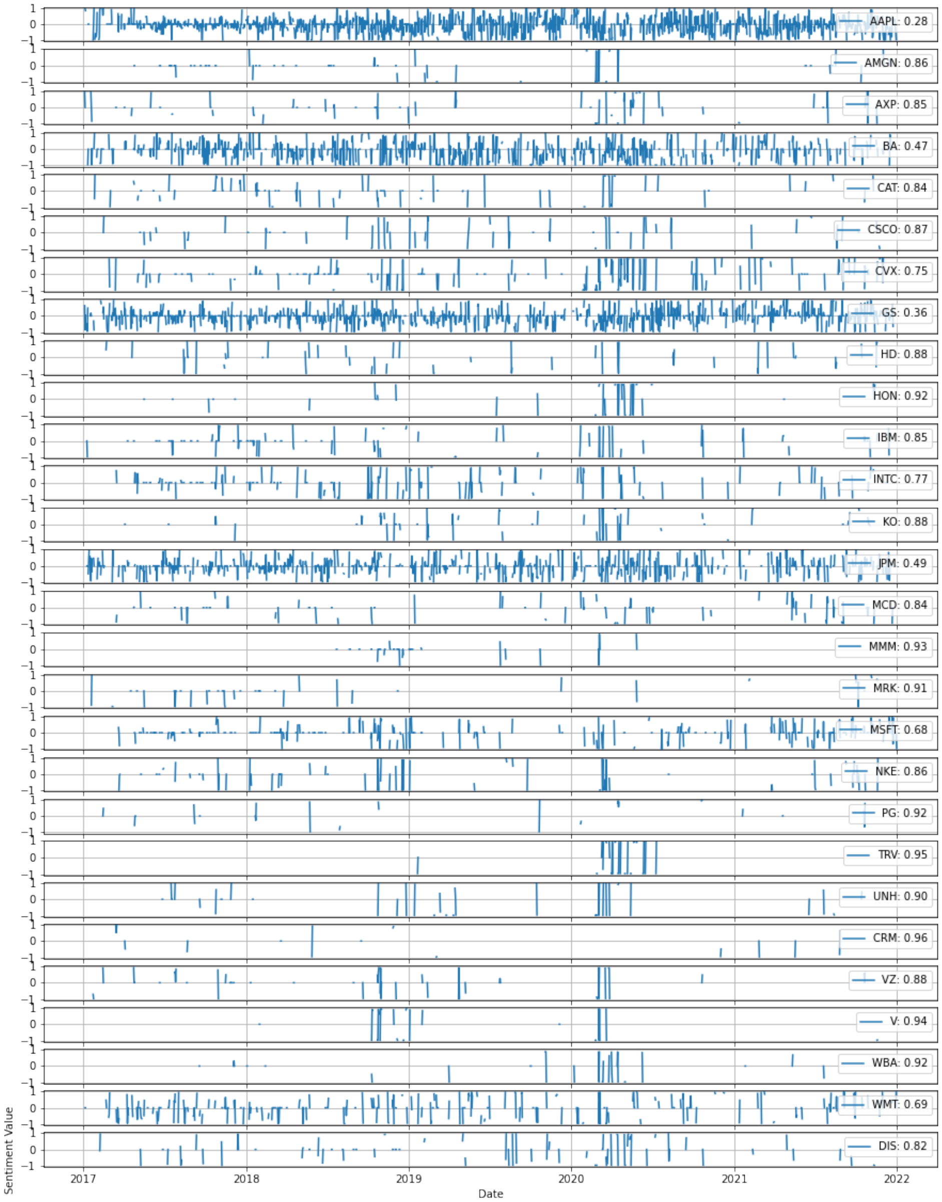

Figure A.1: DJI30 Sentiment Values

A.3 Imputation

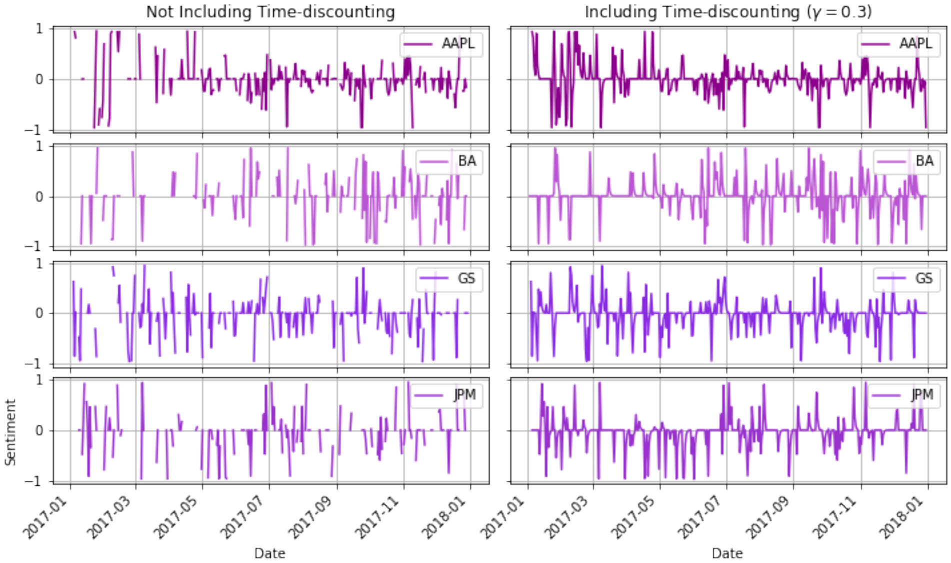

Figure A.2: Imputation with $\gamma = 0.3$

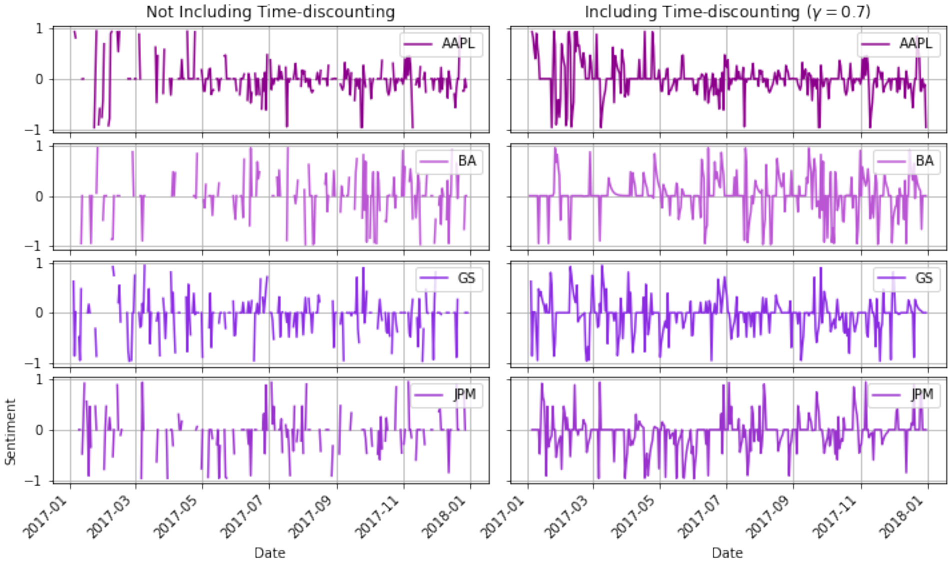

Figure A.3: Imputation with $\gamma = 0.7$

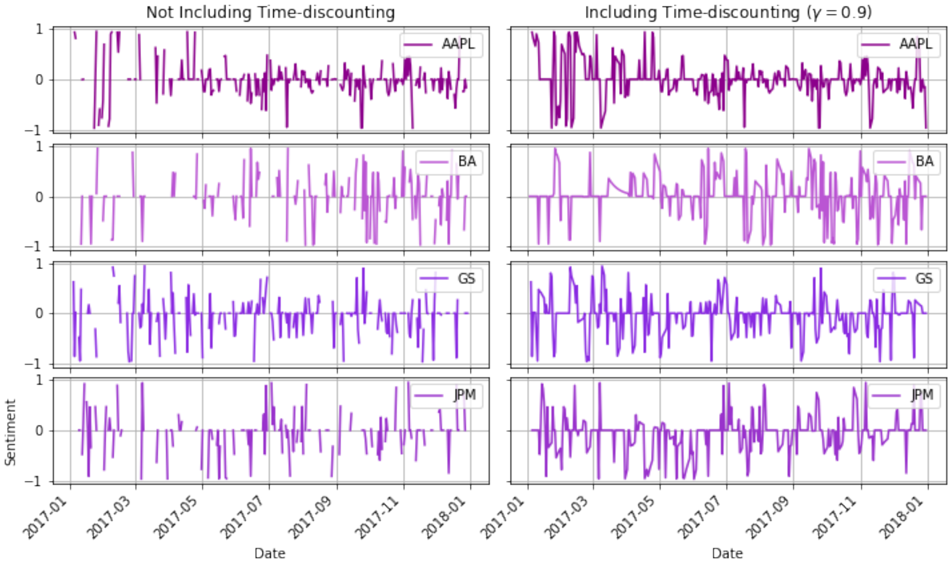

Figure A.4: Imputation with $\gamma = 0.9$

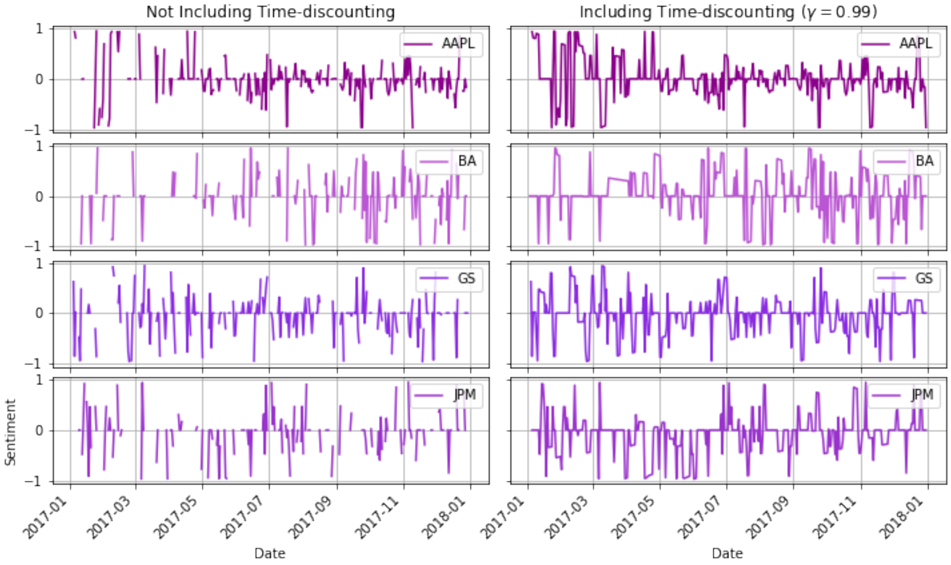

Figure A.5: Imputation with $\gamma = 0.99$

A.4 Auto-correlation and lagged correlation (full table)

	ADF-fuller	Sign. AR lags	Sign. CC lags
ticker			
AAPL	-34.58 (0.00)	16(+)	6(-)
AMGN	-34.63 (0.00)	8(-),14(+)	
AXP	-19.03 (0.00)	2(-),5(-),17(-),18(+),24(-),25(+)	
BA	-18.47 (0.00)	1(+),2(+),3(-),4(-),5(-),6(-),21(-),22(-)	2(+)
CAT	-23.72 (0.00)	2(+),12(+),18(-),20(-)	13(-)
CSCO	-34.52 (0.00)		6(+)
CVX	-20.02 (0.00)	1(+),2(+),9(-),30(-)	1(-),6(-),8(+),13(+)
GS	-35.59 (0.00)	12(-),27(+)	
HD	-17.27 (0.00)	5(-),14(-)	8(-),9(-)
HON	-36.38 (0.00)	6(-),7(+)	2(-),3(-),7(+),9(-)
IBM	-26.05 (0.00)	29(+)	1(+),9(+)
INTC	-36.48 (0.00)		
KO	-27.01 (0.00)	12(-)	4(-),12(-)
JPM	-9.12 (0.00)	2(+),4(-),5(+),11(+),15(-)	
MCD	-6.69 (0.00)	5(-),6(-),13(+),18(+),20(+)	11(-)
MMM	-27.19 (0.00)	2(-),30(-)	2(-),13(+)
MRK	-18.68 (0.00)	12(-)	2(-)
MSFT	-37.09 (0.00)	21(+)	
NKE	-13.89 (0.00)	8(-)	12(-)
PG	-14.05 (0.00)	1(+),5(-),6(-),7(+)	10(-),11(-)
TRV	-19.37 (0.00)	1(-),4(-),9(-),18(-),27(-),30(+)	6(-),12(+)
UNH	-35.71 (0.00)	5(+),18(-)	9(-)
CRM	-34.28 (0.00)	11(+),12(-),25(-),29(+)	3(+)
VZ	-32.34 (0.00)	1(+),22(-)	2(+),5(-),9(+),12(-)
V	-26.08 (0.00)	15(+)	2(-),4(+),8(+),13(+)
WBA	-33.17 (0.00)	1(+),5(-),16(-),21(+)	1(-),3(-),9(+),12(-),13(-)
WMT	-25.69 (0.00)	7(-)	12(-),13(-)
DIS	-14.84 (0.00)	2(-),5(-),14(+),21(+)	1(+),2(+)

Table A.1: **Econometric analysis in preparation for VAR fitting on all DJI30 companies in the years 2017-2021**. *The table shows the company tickers, sector, ADF test statistic and significance (in parenthesis), significant AR lags at* $\alpha = 0.5$ *and significant cross-correlation lags at* $\alpha = 0.5$*. For this analysis, a discount rate of* $\gamma = 0.5$ *has been chosen for imputation*

A.5 Random Forest optimization (full table)

	Hyper-parameters	RMSE	MAPE
AAPL	Trees:500, Max.Depth:5	1.317	3.287
	Trees:1000, Max.Depth:5	1.315	3.396
	Trees:1500, Max.Depth:5	1.315	3.121
	Trees:500, Max.Depth:10	1.319	3.011
	Trees:1000, Max.Depth:10	1.317	3.288
	Trees:1500, Max.Depth:10	1.317	3.081
	Trees:500, Max.Depth:15	1.318	3.039
	Trees:1000, Max.Depth:15	1.317	3.250
	Trees:1500, Max.Depth:15	1.317	3.019
BA	Trees:500, Max.Depth:5	2.143	1.702
	Trees:1000, Max.Depth:5	2.144	1.677
	Trees:1500, Max.Depth:5	2.147	1.661
	Trees:500, Max.Depth:10	2.144	1.705
	Trees:1000, Max.Depth:10	2.146	1.707
	Trees:1500, Max.Depth:10	2.147	1.688
	Trees:500, Max.Depth:15	2.145	1.711
	Trees:1000, Max.Depth:15	2.146	1.703
	Trees:1500, Max.Depth:15	2.147	1.688
GS	Trees:500, Max.Depth:5	1.195	2.275
	Trees:1000, Max.Depth:5	1.188	2.200
	Trees:1500, Max.Depth:5	1.187	2.183
	Trees:500, Max.Depth:10	1.193	2.296
	Trees:1000, Max.Depth:10	1.188	2.220
	Trees:1500, Max.Depth:10	1.187	2.206
	Trees:500, Max.Depth:15	1.194	2.312
	Trees:1000, Max.Depth:15	1.189	2.233
	Trees:1500, Max.Depth:15	1.188	2.221
JPM	Trees:500, Max.Depth:5	0.748	2.001
	Trees:1000, Max.Depth:5	0.748	2.060
	Trees:1500, Max.Depth:5	0.748	2.076
	Trees:500, Max.Depth:10	0.752	2.079
	Trees:1000, Max.Depth:10	0.752	2.136
	Trees:1500, Max.Depth:10	0.751	2.168
	Trees:500, Max.Depth:15	0.752	2.093
	Trees:1000, Max.Depth:15	0.751	2.145
	Trees:1500, Max.Depth:15	0.751	2.177

Table A.2: **Random Forest Regressor Optimization** *The table summarizes the Random Forest optimization for the four chosen companies AAPL, BA, GS and JPM, indicating both their validation RMSE and MAPE errors.*

Bibliography

[1] E. Akyildirim, D. K. Nguyen, A. Sensoy, and M. Šikić. Forecasting high-frequency excess stock returns via data analytics and machine learning. *European Financial Management*, 2021.

[2] D. Allen and A. Singh. Machine news and volatility: The dow jones industrial average and the trna sentiment series, Jan. 2014.

[3] D. Araci. Finbert: Financial sentiment analysis with pre-trained language models. *CoRR*, abs/1908.10063, 2019.

[4] A. Atkins, M. Niranjan, and E. Gerding. Financial news predicts stock market volatility better than close price. *The Journal of Finance and Data Science*, 4(2):120–137, 2018.

[5] M. Baker and J. Wurgler. Investor sentiment in the stock market. Working Paper 13189, National Bureau of Economic Research, June 2007.

[6] J. Bollen, H. Mao, and X. Zeng. Twitter mood predicts the stock market. *Journal of Computational Science*, 2(1):1–8, 2011.

[7] L. Breiman. Random forests. *Machine Learning*, 45:5–32, 2001.

[8] J. Brock. The myth of the rational market: A history of risk, reward, and delusion on wall street. *International Review of Economics Education*, 10(1):130–132, 2011.

[9] W. Chen, B. D. Anderson, M. Deistler, and A. Filler. Solutions of yule-walker equations for singular ar processes. *Journal of Time Series Analysis*, 32(5):531–538, 2011.

[10] J. B. De Long, A. Shleifer, L. H. Summers, and R. J. Waldmann. Noise trader risk in financial markets. *Journal of Political Economy*, 98(4):703–738, 1990.

[11] L. DeVault, R. Sias, and L. Starks. Sentiment metrics and investor demand. *The Journal of Finance*, 74(2):985–1024, 2019.

[12] J. Devlin, M. Chang, K. Lee, and K. Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018.

[13] X. Ding, Y. Zhang, T. Liu, and J. Duan. Deep learning for event-driven stock prediction. In *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, page 2327–2333. AAAI Press, 2015.

[14] D. Du and O. Hu. The sentiment premium and macroeconomic announcements. *Review of Quantitative Finance and Accounting*, 50(1):207–237, 2017.

[15] B. Han. Investor sentiment and option prices. *Review of Financial Studies*, 21(1):387–414, 2007.

[16] J. Z. G. Hiew, X. Huang, H. Mou, D. Li, Q. Wu, and Y. Xu. Bert-based financial sentiment index and lstm-based stock return predictability, 2019.

[17] P. Hushani. Using autoregressive modelling and machine learning for stock market prediction and trading, 2018.

[18] P. Hájek. Combining bag-of-words and sentiment features of annual reports to predict abnormal stock returns. *Neural Computing and Applications*, 29(7):343–358, 2017.

[19] C. Kearney and S. Liu. Textual sentiment in finance: A survey of methods and models. *International Review of Financial Analysis*, 33:171–185, 2014.

[20] T. Loughran and B. McDonald. When is a liability not a liability? textual analysis, dictionaries, and 10-ks. *The Journal of Finance*, 66(1):35–65, 2011.

[21] P. Malo, A. Sinha, P. Takala, P. J. Korhonen, and J. Wallenius. Good debt or bad debt: Detecting semantic orientations in economic texts. *CoRR*, abs/1307.5336, 2013.

[22] R. Pardo. *The evaluation and optimization of trading strategies.* John Wiley amp; Sons, Inc., 2 edition, 2009.

[23] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever. Improving language understanding by generative pre-training. 2018.

[24] A. Shleifer and R. Vishny. The limits of arbitrage. *Journal of Finance*, 52:35–55, 1997.

[25] P. C. Tetlock, M. Saar-Tsechansky, and S. Macskassy. More than words: Quantifying language to measure firms' fundamentals. *The Journal of Finance*, 63(3):1437–1467, 2008.

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

[27] F. Wei and U. Nguyen. Stock trend prediction using financial market news and bert. *Proceedings of the 12th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management*, 2020.

[28] W. Zhang and S. Skiena. Trading strategies to exploit blog and news sentiment. 01 2010.