UNIVERSITY OF EXETER

Computer Science Department

B.Sc. Computer Science
Computer Science Department

An Investigation into the Propagative Nature of Public Sentiment regarding Stock Movement

Candidate

Theodore Proctor

Student ID 76019212

Supervisor

Dr. Marcos Oliveira

University of Exeter

Academic Year
2022/2023

Abstract

This study presents a novel framework that combines Stock Correlation Networks (SCNs) and Natural Language Processing (NLP) techniques such as Sentiment Analysis to predict stock market movements. This framework suggests that public sentiment influences not just the directly associated stock but can also spread to related stocks within the same industry or index. By integrating SCNs and Sentiment Analysis, this propagative nature of public sentiment is captured and quantified, offering a more comprehensive approach to stock market prediction. The project's goal is to achieve a prediction accuracy rate exceeding 50%, representing a random walk approach. This combination of advanced techniques adds to the existing body of knowledge and provides a practical tool for potential investors. It also democratises access to technology, equipping individual investors with tools usually only used by corporations. However, while the model provides valuable insights, it does not offer explicit investment advice or risk management strategies and is not responsible for any potential loss of capital resulting from its predictions.

	Yes	No
I certify that all material in this dissertation which is not my own work has been identified.	☑	☐
I give the permission to the Department of Computer Science of the University of Exeter to include this manuscript in the institutional repository, exclusively for academic purposes.	☑	☐

Contents

List of Figures

List of Acronyms

CDSU Centralised Data Storage Unit

CSV Comma Separated Values

EMH Efficient Market Hypothesis

FTSE 100 Financial Times Stock Exchange 100

HA Human Assessment

HFT High Frequency Trading

HTML Hypertext Markup Language

IPO Initial Public Offering

JSON JavaScript Object Notation

LLM Large Language Model

NASDAQ National Association of Securities Dealers Automated Quotations

NLP Natural Language Processing

SA Sentiment Analysis

SCN Stock Correlation Network

SP500 Standard and Poor's 500

URL Uniform Resource Locator

VADER Valence Aware Dictionary for Sentiment Reasoning

1

Introduction and Motivation

1.1 Background Information

Investment opportunities have been prevalent for centuries. In the past, investments were primarily made privately; however, with the advent of the 20th century, more organisations began to list publicly, allowing a wider range of people to invest. Investment remains appealing as it can offer monetary returns over time with minimal effort. The investment industry is highly competitive, with large organisations dominating the field, leaving individual investors with fewer resources, such as time and data. Despite these challenges, individuals can still succeed in this environment, and the emergence of technology has enabled them to emulate some capabilities of large corporations.

The primary advantage these businesses possess over individual investors is their analytical capability, which allows them to employ advanced techniques to predict stock movements. This project outlines how to replicate some of the techniques used by these corporations, levelling the playing field for individual investors and enabling them to compete with corporate investors.

When an agent purchases a share of a company's stock, the agent becomes a partial owner of the company and thereafter has a stake in the company's performance. They can attempt to use fluctuations in the market to profit, by trading when conditions are favourable, however due to the instability of the market, profit is not certain.

The stock market can be described as a complex, decentralised network of agents (financial institutions, intermediaries, and market participants), all engaged in the issuance, exchange, and valuation of equity securities. It serves as a mechanism for the allocation of capital and risk in the economy, as well as a platform for price discovery[11] and liquidity provision[35].

The stock market comprises regulated exchanges like the NYSE, which offer transparent trading environments and essential market data. Stock indices, such as the Financial Times Stock Exchange 100 (FTSE 100), serve as financial indicators and benchmarks to track and analyse specific equity groups, representing market segments or industries, and providing insight into market or sector health.

Since the market can be modelled as a network, we can utilise this approach to predict equity movements based on associated stocks. By representing the market or its sub-sections as a stock correlation network, this can be achieved. Typically, this is accomplished by analysing numerical movements and using them to determine effects on neighbouring stocks. However, this project deviates from the traditional analysis of numerical data, opting instead for a combination of SCN and Natural Language Processing (NLP) techniques to leverage public sentiment as an indicator for predicting stock movements. Provided that this method proves the existence of a correlation between the propagative nature of public sentiment and stock prices, a valuable tool for all investors will be created.

The framework detailed here will serve as a tool for potential investors, offering an assessment of a stock's movement. This tool aims to aid investors in making informed decisions; however, due to inherent market volatility, we cannot guarantee the accuracy of any prediction. Consequently, the tool will not be held responsible for any trading or loss of capital resulting from trading decisions influenced by its predictions.

This project endeavours to bridge the gap between individual and corporate investors by replicating some techniques used by large corporations, thus democratising access to cutting-edge technology and predictive tools. By making these resources available to the public, the project aims to promote a more stable and ethical market direction, aligning with the objectives of exchange regulation.

1.2 Stock Picking, Market Prediction, and Inequality in Investment

Stock picking encompasses the systematic analysis and evaluation of individual equities to identify potential investment opportunities based on various financial, strategic, and economic criteria. The objective is to achieve excess returns relative to a benchmark index or the overall market by exploiting market inefficiencies or capitalising on the mispricing of securities. Despite contradicting the Efficient Market Hypothesis (EMH)[30] (which states that financial markets are informationally efficient, meaning that all available information is already reflected in the current prices, meaning any changes in price are due to new, unpredictable information), stock picking based off of historical data has often been practised with success. Thus highlighting the challenges in comprehensively understanding the market, particularly for individual investors.

Various approaches to stock picking include fundamental analysis[23], value investing[37], growth investing[13], momentum investing[24], disruptive innovation investing[15], and technical analysis[12]. Nevertheless, predicting stock performance remains a daunting task due to factors such as market complexity, information asymmetry[7], behavioural biases[6], and the inherent randomness and unpredictability of events that influence stock prices. This is succinctly described by the random walk model.

> *The random walk model[17] is a mathematical concept used to describe the path of an object or variable that moves in a series of random steps. In other words, it is a stochastic process where the future movement is determined by a random factor and is independent of past*

movements. In finance, the random walk model is often applied to stock prices or exchange rates, asserting that future price movements are unpredictable and determined by a series of random steps.

Despite this model, certain agents, particularly corporate investors, experience higher success rates than individuals, which can be attributed to market inequality. Market inequality stems from the competitive nature of public markets, where well-resourced corporate investors often have the upper hand over individual investors. Large organisations leverage advanced techniques, including big data analytics[18], Bayesian statistics[28], and high-frequency trading (HFT)[4], while individual investors typically depend on qualitative methods like fundamental analysis or value investing [1]. Although computational methods for stock prediction exist, their efficacy remains limited. By offering a tool accessible to all investors, which incorporates techniques typically exclusive to corporate investors, this project strives to bridge the gap between individual and corporate investors. By replicating corporate strategies, it democratises access to advanced technology and predictive tools, fostering a stable and ethical market in line with exchange regulation principles.

1.3 Propagative Sentiment

1.3.1 Public Opinion and Sentiment Analysis

The approach adopted in this study is a form of technical analysis. Research such as 'Widespread Worry and the Stock Market' [21] and 'Twitter mood predicts the stock market' [10] suggests that public opinion and investor emotions significantly influence stock movements. Additionally, studies such as [32] suggest there is a strong correlation between 'public sentiment' and 'market sentiment', lending validity to the employment of SA as a technique. Factors include social phenomena like herd mentality [16] or driving emotions like fear and greed. By examining media consumed by investors, we can attempt to predict their opinions on stocks using NLP techniques such as SA. This relationship has predominantly been examined in a direct manner, where the consequences of an article have an immediate impact on the associated stock. By investigating how this influence extends to related stocks within an industry or index, we can expand upon existing research and develop a more comprehensive model.

1.3.2 Stock Correlation Network

The second key element of this service is a SCN, which is based on graph theory. Nodes represent individual stocks, and edges represent correlations between stocks. Though 'using only the SCNs to make assertions about a market is naive because they only capture the result of historic correlations' [25], combining them with other methods can yield valuable results.

1.3.3 Propagative Sentiment applied to Stock Correlation Networks

This report investigates the application of public sentiment to SCNs and examines whether public sentiment has a propagative nature concerning its effects on stock prices. By determining related stocks within an index and applying a correlation coefficient to a stock's current sentiment (which represents its current public opinion), we can study whether stock price changes due to public opinion exhibit a propagative nature. If proven true, the methods detailed in this project can be utilised as a stock prediction tool.

1.4 Aims, Objectives and Scope

1.4.1 Project Aims

The overarching goal of this project is to explore the relationship between propagative public sentiment and stock market performance, ultimately aiming to provide more accurate stock price predictions and an investment decision-making tool. To achieve this, we have identified the following specific aims:

- Prediction Accuracy: Develop a model with an accuracy exceeding 50%, demonstrating at least a weak correlation between the propagation of public sentiment for a company and its stock performance, and passing the benchmark of the random walk model.
- Provide Framework: Create a framework for users to examine certain indices or groups of companies on the stock market.
- Analyse Market Reactions to News Events: Investigate how stock prices respond to different types of news events and their corresponding sentiment, and judge if sentiment changes to a company propagate to related companies.
- Stock Prediction: Predict stock price movements using SCNs and SA.
- Improve Decision Making: Improve investment decision-making based on more accurate predictions.
- Adaptability and Continuous Learning: Ensure the developed models can adapt to changing market conditions and incorporate new data to improve prediction accuracy through the regeneration of prediction models, as well as being capable to tailor to the users specific needs.

1.4.2 Project Objectives

This project seeks to design and implement a system that combines SA and SCNs to provide valuable insights and predictions for stock market performance. To ensure the success of this endeavour, we have outlined the following key objectives:

- Short-term stock predictions: The objective of this project is to predict short-term stock movements using sentiment analysis and SCN.

- News Article Analysis: This component should automatically fetch news articles from the internet and pre-process the article. SA[19] should be performed to accurately assess articles sentiment polarity. Success can be evaluated by the regular acquisition of relevant articles, efficient pre-processing, and the correctness of sentiment scores.
- Stock Analysis and Data Management: The program should efficiently create a stock correlation network, automatically obtain relevant financial data, and save the SCN state to minimise retraining time. Success can be assessed by data accuracy, model consistency, and effective network representation.
- Integration of SA and SCN: The SCN must incorporate company sentiment and assess its impact on neighbouring companies when sentiment changes. By comparing predicted stock movements at a specific time point to actual stock movements and evaluating the model's accuracy and reliability against historical data, success can be determined.
- Visual Representation of SCN: A user-friendly interface is needed for users to interpret the project and make informed decisions. Success can be deduced through user reviews and the provision of comprehensive documentation.
- Model Optimisation: By experimenting with various heuristic functions, data sets, and hyperparameter tuning, the optimal model can be determined, additionally, anomalous data detection should be implemented to improve the quality of training data.
- Centralised Data Storage Unit (CDSU): To handle extensive data processing, a highly structured data storage unit, comprising a database and a file structure, is required. Success can be evaluated based on the speed and ease of data access, minimisation of repeated data points, and human-readability for independent research purposes.

1.4.3 Project Scope

In order to ensure the feasibility and effectiveness of this project, we have defined a clear scope, focusing on specific aspects of stock market analysis and prediction. The following points outline the limitations and boundaries within which this project will be carried out:

- Model Interpretability: The model is supposed to be used as a tool to help improve the investment decision making of users. Some degree of accuracy might be sacrificed in exchange for interpretability for users.
- Risk Management: The model will not trade on behalf of the user, nor will it offer risk management strategies such as portfolio diversification. This will be left to the user to perform.
- Investment Advice: This model will not provide explicit investment advice, only the prediction of stock movements.
- Processing and Hosting: All processing and hosting will be performed locally. This is because additional resources are necessary for off-site hosting, which are unavailable.
- Code Production: The project will primarily be written in Python, with Javascript and HTML elements.

2

Design, Methodology and Implementation

2.1 Project Design

The aim of this project is to forecast short-term stock fluctuations using sentiment analysis and SCN. We will first experiment to ascertain whether the propagative nature of public sentiment concerning stock movements is valid, and then design a testing framework for evaluation.

We will gather historical data comprising stock-related articles and corresponding numerical data for the stocks under investigation. Due to stock volatility, a significant amount of data is required for an accurate SCN model. After data acquisition, we will train the SCN and adapt it to represent a company's public sentiment score, which should correlate with stock movements as per prior studies [32][36]. The SCN will estimate sentiment change based on correlation with neighbouring stocks.

We will analyse company-specific articles to deduce sentiment and apply it to the SCN to assess the impact on neighbouring stocks. Data from widely-read sources, including unbiased and biased articles, will be collected to represent the general population's sentiment.

After obtaining articles, we will pre-process them, removing irrelevant content and identifying fundamental components. We will explore sentiment analysis methods and heuristics to determine the optimal approach for correlation coefficient identification. By refining these methods, we aim to enhance prediction accuracy.

We will develop a user-friendly live graph of the SCN, with nodes representing companies and edges indicating the correlation coefficient. Nodes will store daily sentiment changes, with real-time updates displayed for easy interpretation. Our system will cross-reference end-of-day stock prices with news articles, compensating for limited access to granular stock movement data. Lastly, we will conduct experiments to optimise the SCN, tuning hyperparameters to improve stock movement prediction.

2.2 Methodology and Implementation

For a successful product, seamless integration of all components is essential. The different parts of this program must work together, communicating via the CDSU module, which serves as the system's mediator. The diagram below demonstrates the connections among all service components.

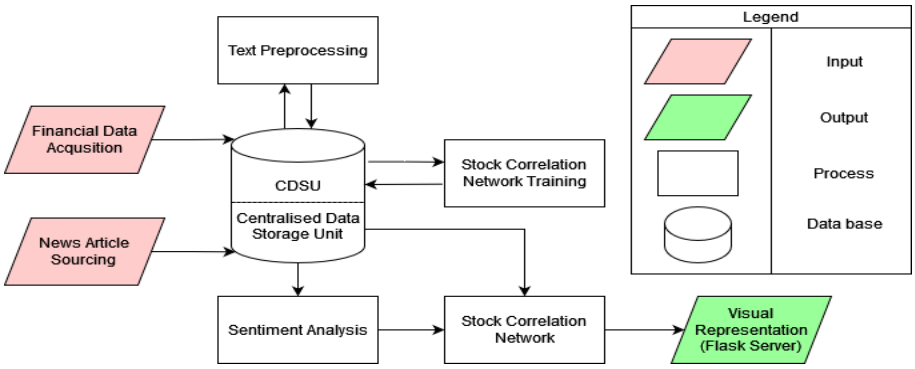

Figure 2.1: Flow Diagram of Project Elements

2.2.1 Financial Data Acquisition

Obtaining financial data is vital for training the SCN. We must identify index constituents and acquire historical data for company. For this project, we've chosen five years of data, from 01/01/2015 to 01/01/2020, to create a 'stable data set' excluding the COVID-19 pandemic's significant impact on stock prices. If adapted for current predictions, the training range would be extended. This range balances data representativeness and computational limitations. To prevent adverse effects during training, a feature for removing anomalous data points should be incorporated.

Functionality to obtain financial data for specific days is necessary for testing, as predictions are based on daily data. The data should include date, open, close, high, low, and volume of stocks traded. The data acquisition service's effectiveness can be evaluated by the proportion of accurate data returned and by implementing measures to reacquire data if the initial call fails.

2.2.2 News Article Sourcing

Obtaining dated articles from relevant sources on companies within an index is essential for our project. The source's bias or political agenda is less important, as content is widely consumed and influences public sentiment.

These articles enable sentiment analysis, which is applied to the SCN to assess the effect of sentiment on stock value and correlated neighbours. As long as articles represent public sentiment, this service component is successful. For best performance, the service must quickly and regularly acquire data from diverse sources.

2.2.3 Data Storage

The CDSU is the central component of the entire service, handling concurrent requests efficiently and ensuring database accessibility during resource-intensive tasks such as SCN training. A config file is used for communication between program elements, enhancing performance and reliability. It has three main components:

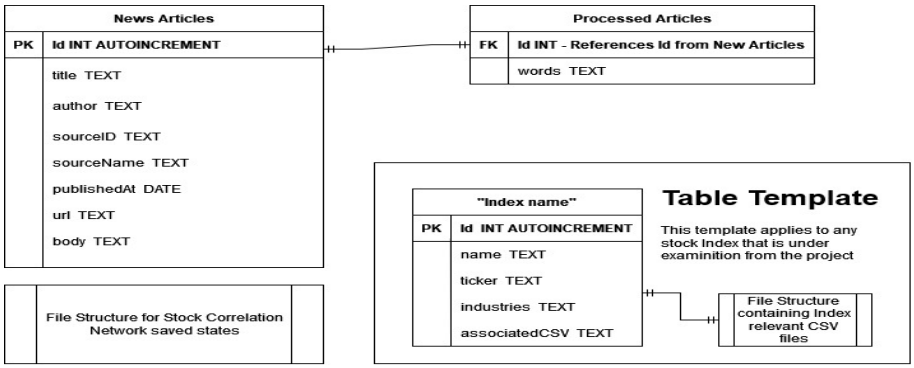

Figure 2.2: Database Organisation

Database

This project aims to create an efficient database system to store and organise large volumes of interconnected data, including links to external data structures like Comma Separated Values (CSV) files. Success depends on database integrity and completeness, handling missing or incomplete data, and efficient data storage and retrieval processes.

CSV storage

A large, organised structure of numerical data is crucial for SCN training. CSV files offer an efficient solution, being easier to access and interpret than JavaScript Object Notation (JSON) files or multiple database requests. Project success will be reflected in a structured file system with relevant CSV files, efficient data access and storage for model training, and an organised file structure.

Store Graph States

Quick access to graphs is vital for training multiple models in time-sensitive finance settings. Pre-training and saving multiple models under different conditions allow swift access and switching between models as needed. Success will be evident in efficient model saving and loading processes and the ability to seamlessly and rapidly switch between models.

2.2.4 Text Pre-processing for Sentiment Analysis

Pre-processing the acquired data is essential for accurate sentiment analysis. Conducting this separately improves the system's effectiveness by allowing concurrent processing and

increased efficiency. The pre-processing effectiveness can be evaluated by comparing processed text to the original text. The text must undergo several processes in a specific order:

Please reformat this content into a coherent sentance or two:

1. Remove punctuation and capital letters: Simplifying text and eliminating superficial differences not contributing to sentiment.
2. Tokenise: Breaking the text into individual words or phrases for easier analysis.
3. Remove stopwords, numbers, and noise: Excluding elements that do not significantly contribute to sentiment for more accurate analysis.
4. Lemmatize: Reducing words to their base or dictionary form[38], ensuring different forms of the same word are treated as one during analysis.

Implementing these processes as consecutive functions streamlines pre-processing workflow and prepares text for effective sentiment analysis.

2.2.5 Sentiment Analysis

The sentiment analysis component evaluates article sentiment to predict its impact on stock prices, emulating public opinion. It extracts sentiment data and integrates it with the stock correlation network, benefiting investors. Three main approaches are:

- Knowledge-based: Uses lexicons with sentiment scores, e.g., Valence Aware Dictionary for Sentiment Reasoning (VADER) [26] and SentiWordNet [5].
- Statistical: Applies machine learning algorithms to learn patterns from labelled data, handling complexity but requiring large training data. Examples include: Bollen et al's [9].
- Hybrid: Combines both methods for improved performance.

Due to time and complexity, we will use a knowledge-based approach. VADER is designed for informal text and idiomatic expressions, meaning it is adaptable to finance-related articles, although it may require customisation for domain-specific terms. SentiWordNet suits formal English but may need customisation for finance-specific terms.Both methods produce compound sentiment score. VADER is preferable due to its adaptability and ability to handle informal language. Success evaluation involves comparing predictions to a labelled data set using accuracy, precision, and recall metrics. Customising the lexicon ensures accurate sentiment analysis, contributing to the stock correlation network's predictions.

2.2.6 Stock Correlation Network

The SCN is vital for processing sentiment changes and applying the propagative nature of public sentiment. It should create a network representing stock correlations within an index, with nodes as companies and edges denoting correlation. Users can specify industry-specific sub-graphs and exclude weak correlations. Success is determined by comparing networks from training and testing data sets, and presenting predictions in a user-friendly format.

Optimisation involves exploring various correlation calculation methods, heuristic functions, data sets, and hyperparameter tuning. Implementing features to save and load trained SCN states is necessary due to computational costs.

Evaluating the model's success involves testing the effects of an article on the SCN, comparing suggested changes to actual stock price movements, and considering a threshold value for disregarding weak predictions. Assessing the ratio of true to false predictions and accuracy, precision, and recall metrics help judge overall effectiveness.

2.3 Implementation

Python was chosen for this project due to its extensive libraries, readability, versatility, and cross-platform compatibility. Flask, a lightweight web framework, offers seamless integration with Python, extensibility, and easy development. We will focus on the secondary market [27], where previously issued shares are traded among investors. This market allows access to historical data for training the SCN. Yahoo Finance [45], a financial information platform, provides such data and has a built-in Python library, ensuring no API call restrictions. We will limit our experiments to certain indices: FTSE 100, National Association of Securities Dealers Automated Quotations (NASDAQ), and in some cases Standard and Poor's 500 (SP500). Acquired financial data will be organised in a directory structure with the index as the folder name and CSV files containing company data named after the company's ticker symbol. File names will be stored in the database for efficient retrieval and management.

2.3.1 Stock Correlation Network

The SCN is established through an in-depth analysis of historical data from a specified index's companies, focusing on stock movement variations to discern correlations. In this network, companies or stocks are nodes, and the links between them, or edges, depict correlations. The degree of this relationship is quantified through the 'correlation coefficient', which ranges from -1 to 1. This study examines various heuristic functions to assess these stock correlations.

The first algorithm considered compares the number of days where two companies' stock movements align or oppose each other. This comparison is captured in a polarity list where '1' represents identical movement, '0' indicates no movement, and '-1' signifies divergent movement. Applying the following equation to the polarity list:

$$corrCoef = 2\left(\frac{n_1}{n_1 + n_{-1}} - 0.5\right) \tag{2.1}$$

where n_1 is the count of 1's, n_{-1} the count of -1's, and n_0 the count of 0's, generates a normalised correlation coefficient between -1 and 1. If $n_1 = n_{-1} = 0$, the correlation coefficient defaults to 0.

An iterative method is also proposed, augmenting the correlation coefficient based on

consecutive polarity values. This approach employs the formula:

$$corrCoef \mathrel{+}= \frac{p \cdot n \cdot (1 - |x|)^2}{z} \tag{2.2}$$

where p is the current polarity, n is the count of successive polarity values, and z is the list length. The component $p \cdot (1 - |x|)^2$ is designed to decelerate the convergence of x to either 1 or -1. This method aims to highlight insights into the correlation strength and direction not readily evident with conventional correlation coefficients.

The third algorithm uses the same polarity list and is iterative, boosting the correlation coefficient based on consecutive polarity runs. The formula is as follows:

$$corrCoef \mathrel{+}= \frac{p \cdot n \cdot (1 - |x|)^2}{z} \tag{2.3}$$

Here, p is the current polarity, n is the count of consecutive polarity values, and z is the list length. The term $p \cdot (1 - |x|)^2$ is designed to decelerate the convergence of x to either 1 or -1. Two versions of this method are explored, determined by the starting value of n. If n starts at 0, the algorithm emphasises stability and long-term relationships, while if n starts at 1, it focuses on variability and sensitivity to short-term patterns.

The next method employs the Pearson correlation coefficient [3], quantifying the linear relationship between two variables:

$$corrCoef = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{2.4}$$

Here, x_i and y_i are ith observations of variables X and Y while $\bar{x}$ and $\bar{y}$ are their sample means respectively. n denotes the total sample size. For lists x and y, we can either utilise daily stock price differences, or a polarity list ('1' for increase, '0' for no change, and '-1' for decrease).

The Spearman correlation coefficient, a non-parametric metric, quantifies the monotonic relationship between two variables. Unlike the actual values, it leverages ranked data values, which enhances its suitability for assessing non-linear relationships and reduces sensitivity to outliers. The equation is as follows:

$$corrCoef = 1 - \frac{6 \sum d_i^2}{n(n^2 - 1)} \tag{2.5}$$

Here, d_i represents the rank difference between corresponding values in two data sets, while n denotes the total observations. The options for x and y are identical to those in the Pearson coefficient. All heuristic functions must consider boundary conditions to circumvent computational errors, such as division by zero.

The suitability of each heuristic for our graphing needs can be evaluated using a training and testing data set across multiple indices. By comparing correlation coefficients, we can gain insights into the performance of each heuristic, noting that different heuristics may excel with different data sets or indices. Key comparison metrics include the ratio of stable correlation

coefficients across both data sets, the number of unusable results due to low-quality data, and the average variation between the correlation coefficients of the data sets, which can indicate the relative accuracy of each heuristic.

Our goal is to select a heuristic that optimises these criteria, with a primary focus on minimising the ratio of incorrect assessments. While reducing variation is beneficial, it's not crucial as the actual graph implementation will be trained on all available data, thus mitigating the impact of variations in correlation coefficients. To optimise the program, trained graphs should be saved due to the intensive training process. The Python library 'joblib' [40] will be used for storage, with 'LZ4' [29] for compression. Stored in a dedicated folder, the graphs will be named based on the index and the heuristic used, facilitating efficient retrieval and management of these trained graph objects.

2.3.2 Sentiment Analysis

While the ideal application of SA would be on analyst reports or financial statements, accessibility constraints necessitate an alternative approach. News articles, though slightly delayed, echo the content of analyst reports, making them suitable for our purposes. However, this limits us to predicting daily rather than instantaneous movements, as other investors have earlier access to new data. Despite this, our SA component can interpret articles faster than human readers, granting us an edge, given a frequently refreshed data stream.

For article retrieval, we employed newsAPI [33], an aggregator of articles from various news providers. The Uniform Resource Locator (URL)s of these articles were fetched and retrieved using the Python 'requests' library [43]. The raw Hypertext Markup Language (HTML) was processed with 'Beautiful Soup 4'[39] to extract crucial information which was then stored in a locally hosted SQLite3[41] database for subsequent SA and integration into the SCN. Upon pre-processing the text, SA was conducted using libraries like Vader Sentiment [44] and NLTK [34], testing methodologies of analysing individual words versus whole sentences. We also explored the TextBlob library [42] and investigated the effects of unprocessed data on SA methods.

To evaluate these SA models, we compared their sentiment polarity calculations with those of a Large Language Model (LLM), like ChatGPT [14], and human interpretations, using pre-collected articles from two distinct periods, one within the training range, and one outside of the training range.

After deriving the SA value from an article, it's incorporated into the SCN to study its propagative effects. We examined three distinct propagation techniques for connected nodes: maximum propagative value, average value, and first value, each with varying propagative depths. The choice of propagative depth is vital; higher depths could lead to repeated node visits, particularly when longer network paths hold greater compound correlation values than shorter ones. Additionally, we tested the approach of discounting prediction values falling below a specified threshold, as they lacked sufficient conviction to substantiate a reliable prediction. We utilized Flask [20], a lightweight web framework, for user-friendly live graph visualisation and result interpretation.

3 Results

3.1 Sentiment Analysis Validation

In order to evaluate the efficacy of our sentiment analysis methodologies, we employ a myriad of testing procedures aimed at deducing their accuracy. These methods include validating prediction accuracy against pre-established data sets, in addition to a comparative analysis of the predictions made by our model and the genuine movements exhibited by the stock market. We test our hypotheses on the NASDAQ, FTSE 100 and SP500, although in some cases we cannot perform tests on the latter due to its size complexity.

3.1.1 Deducing the Most Effective SA Method

To improve prediction accuracy, we evaluate the best SA methods by comparing them on a consistent text corpus to measure accuracy and sensitivity to sentiment changes. This illuminates each model's proficiency in detecting extreme or subtle sentiments, aiming for accurate sentiment polarity.

Our main focus is sentiment polarity accuracy. We create test samples using a LLM, contrast it with a Human Assessment (HA), and examine the deviation in results of various SA methods against these benchmarks. Each method was tested, and average sentiments across data sets were computed, as shown in graph 3.1.

The y-axis displays the SA score, with each bar group representing different test data. Results are inconsistent and skewed positive, indicating potential inaccuracies in our predictions.

Curiously, even mildly positive data results in high sentiment scores. Methods 4 and 8 show a rise but misinterpret slightly negative articles as positive. VADER Method 1 is fair but yields SA scores of a low magnitude. Adding a scalar could help, but lacks rigour. VADER method 2 and NLTK VADER, similar to VADER method 1, produce correct polarities but with a positive skew. Textblob methods show a consistent rise but are heavily skewed positive, indicating potential need for score normalisation. However, we should focus on improving the SA method

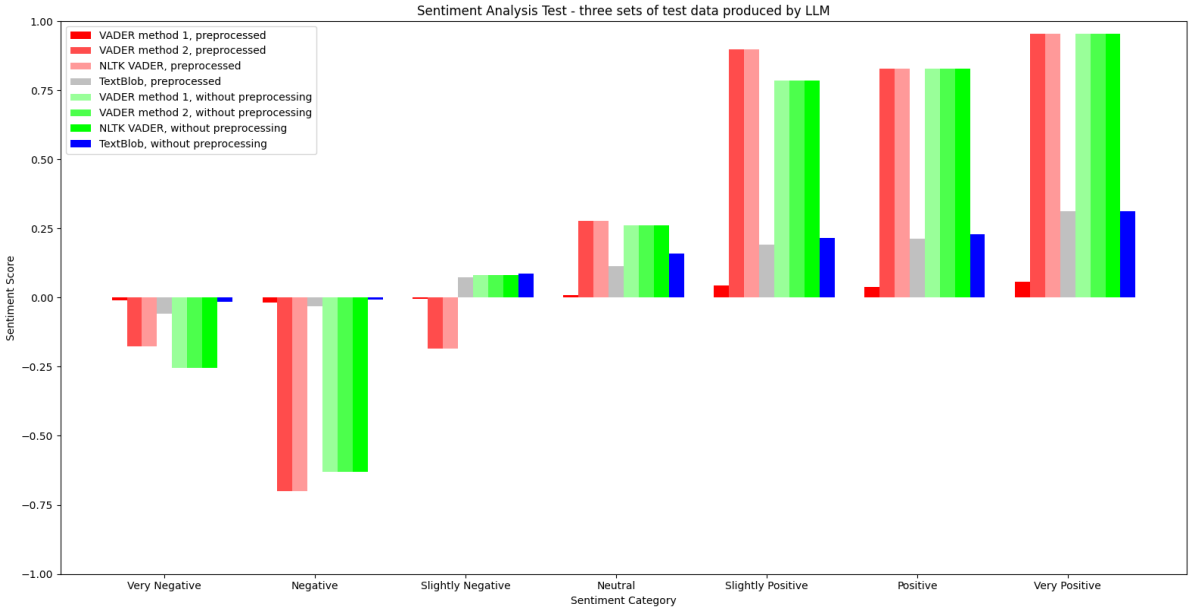

Figure 3.1: Examining SA methodologies across texts with predefined diverse sentiments

itself rather than post hoc adjustments.

Overall, our sentiment analysis methods require refinement to enhance prediction accuracy. The positive bias is a concern as it could skew results and mislead conclusions. This could be due to an overly simplistic model or inadequate test set. Further investigation is necessary to ensure reliable sentiment analysis.

3.1.2 Validating SA Methods Against Pre-assessed Data

To validate our sentiment analysis methods, we compare their results with HAs and LLM assessments [14]. We apply sentiment analysis on our article dataset, with some assessed by humans and LLM for sentiment polarity. This helps gauge our SA's accuracy. The articles were labelled 15% by humans, 60% by LLM, and 100% by SA. We utilise VADER method 1 with pre-processing for its consistency in producing polarity values, as per our prior analysis. The results are displayed in 3.2.

The y-axis represents assessment types proportions. 'Same assessment' means both methods agree on sentiment, 'different assessment' indicates disagreement, and 'neutral assessment' denotes low SA scores (0.05 or less) and neutral HA or LLM scores. Key comparisons are between 'same assessments' and 'different assessments'. SA compared to HA achieves 67.7% accuracy, while SA to LLM results in 75.0% accuracy. The difference can be attributed to the 84.6% similarity between LLM and HA assessments. Data limitations account for imperfect alignment since not all articles were evaluated using all methods. Other methods, like TextBlob, performed less well, scoring 64.6% accuracy against HA.

Figure 3.2: Verifying SA assessments

In conclusion, our SA methods lack high accuracy compared to human interpretation. This may be due to poor quality testing data. Further human labelling on a larger data set is required for more accurate comparisons. To truly assess SA accuracy, we should compare predictions to actual stock market movements.

3.1.3 Comparing SA Methods to Actual Market Movement

It's also essential to evaluate the accuracy of the SA methods against genuine market movements. To accomplish this, we compared the predictions made by the SA on our test data set with the actual market movement for that day. We categorised the outcomes into true positives, false positives, true negatives, false negatives, and neutral assessments, as seen on 3.3 the latter representing no market movements. The accuracy is determined based on true and neutral classifications, as these results would not lead to a loss.

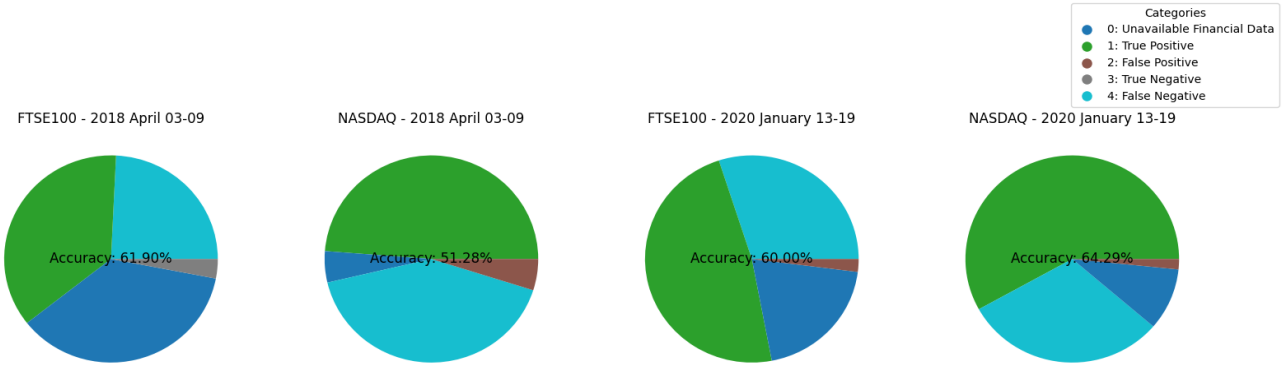

Figure 3.3: Accuracy of SA using TextBlob, without Pre-processing

Our SA methods average around 60% accuracy, indicating considerable imprecision. An exception is the consistently lower accuracy in the NASDAQ 2018 test set, possibly due to data anomalies. This inaccuracy risks undermining our SCN's final results. This might be due to the acceptance of minimal sentiment score changes, which could lead to incorrect predictions.

To address this, we've implemented a prediction threshold. As seen in 3.1, SA score magnitude varies significantly across methods. For instance, VADER Method 1 yields low magnitudes, while VADER NLTK provides high SA scores.

Upon testing SA Methods like VADER method 1, VADER NLTK, and TextBlob (with preprocessing), we found that introducing a threshold improves prediction consistency. However, the standard remains suboptimal. The subsequent graph, using the TextBlob method, implements the threshold, showing various thresholds across the test set, with epoch results consolidated into a single pie, as shown on 3.4.

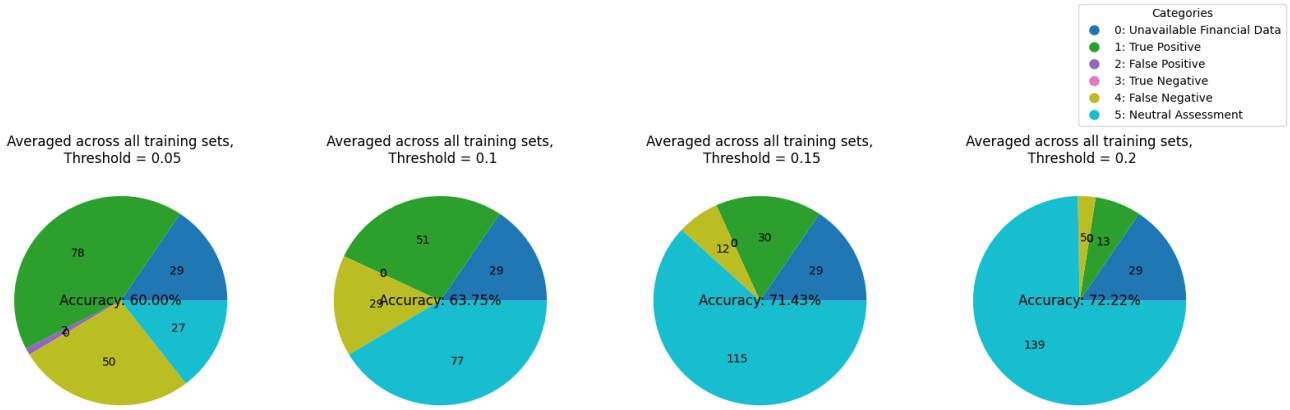

Figure 3.4: SA Predictions with thresholds, Using TextBlob

Additional instances of thresholds on various heuristics are depicted in the appendices 5.1, 5.2, although they performed worse than the TextBlob assessment. The data demonstrates a general trend, wherein the accuracy of the sentiment analysis method improves as the threshold increases. This pattern, however, is not consistent across all SA methods. A notable consequence of raising the threshold is the increase in null predictions, which, while not ideal, is decidedly better than incorrect predictions. It is also noteworthy that negative sentiment analysis values are almost nonexistent, and the few that appear are incorrect as the threshold increases. This phenomenon can likely be attributed to the positive skew observed in 3.1 and warrants further exploration in subsequent research.

3.2 SCN Validation

We also need to validate that, in conjunction with the provided SA scores, we can improve the quality of predictions by utilising a SCN. We will examine numerous methods for deriving the correlation coefficient between nodes, and judge what combination of hyperparameters results in the most effective overall model.

3.2.1 Establishing Best Heuristic for determining Correlation Coefficients

Optimising the heuristic for SCN training is crucial for boosting accuracy, and we evaluated several based on their correlation coefficients. We partitioned financial data to ascertain the

consistency of correlation coefficients across heuristics, validating the best one for network integration. The reliability of each heuristic was measured by dividing data and determining the fraction of stable correlation coefficients. We examined predicted correlation coefficient variability across sets to assess model accuracy. We used two data division methods: the 'nth element split' (n=4), and a pivot-based split at the final 25% of data points, ensuring equal data quantities for both tests (3.5).

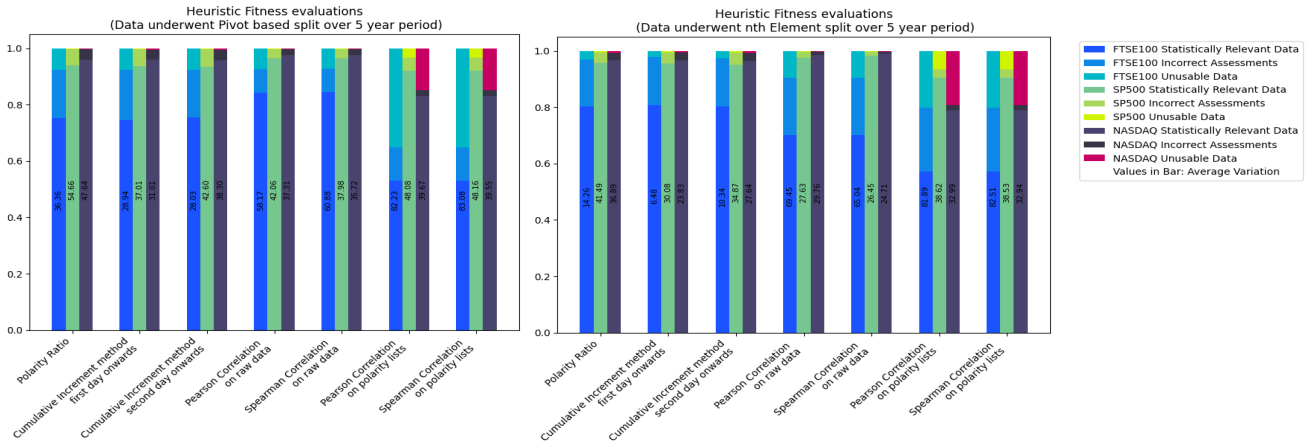

Figure 3.5: Testing Validity of SCN correlation coefficients

Our graph's y-axis represents data proportion, with unusable data, due to financial inconsistencies, seen across all models. Each bar cluster symbolises a heuristic method, with sub-bars denoting the data set used (NASDAQ, FTSE 100, SP500). Statistically relevant segments and erroneous assessments within each sub-bar reflect correlation coefficient polarity and variance across training and testing sets.

The primary objective is to minimise variation and the ratio of incorrect assessments to statistically relevant data, striving for graph consistency. The FTSE 100 often contains more unusable data, indicating a need for better data gathering methods, while the NASDAQ and SP500 present minimal unusable data. In the nth element split graph, methods 1-3 excel for the FTSE 100. Still, excluding FTSE 100 data, Spearman correlation performs best. Pivot split tests reveal Pearson and Spearman correlation surpassing methods 1-3. The cumulative increment method (from the first day) exhibits the least variation, with Spearman correlation also showing low variation on NASDAQ and SP500. The reliable performance of the cumulative increment and Spearman methods suggest their aptness for SCN training. The final model will utilise all available data for maximum accuracy, reducing variation's importance.

In summary, the cumulative increment and Spearman methods emerge as the most favourable heuristics, pending further validation. By ensuring similarity between training and testing data sets, we expect enhanced results after applying SA to the SCN.

3.2.2 Determining Optimal Final Model

After analysing hyperparameters and their affects on the the SCN, we test numerous combinations of the best ones, along with some other hyperparameters we can see in **??**.

Sentiment Analysis Model	Propagation Depth (range 0-3)	Propagation Method (1,2,3)	Correlation Coefficient Threshold (range 0-0.7)	Correlation Coefficient Heuristic	Sentiment Threshold (range 0.05-0.15)	Average Accuracy	Number of Predictions
VADER1	2	1	0.1	2	0.15	0.6892174432497012	930
VADER1	2	1	0.1	5	0.15	0.6892174432497012	930
TextBlob	2	1	0.1	2	0.1	0.6828255675029867	930
VADER1	2	1	0.1	2	0.125	0.6663082437275986	930
VADER1	2	1	0.1	5	0.125	0.6663082437275986	930
VADER1	2	1	0.1	2	0.1	0.6634109916367981	930
VADER1	2	1	0.1	5	0.1	0.6634109916367981	930
TextBlob	2	1	0.1	5	0.15	0.6572281959378734	930
VADER1	2	1	0.1	2	0.075	0.6495221027479092	930
VADER1	2	1	0.1	5	0.075	0.6492234169653525	930

Figure 3.6: Final Results Table

Owing to the simulations' computational demands, this study exclusively used NASDAQ data, requiring several days for a single index processing, suggesting potential model adjustments for speed. Model success was measured via the 'accuracy' column and prediction count. With all top models producing identical prediction numbers, this metric isn't a success determinant. It's noteworthy that prediction numbers varied across models due to differing propagation depth values. As observed in 3.6, propagation method 1 (maximum propagative value) consistently emerges as the optimal propagation technique. This pattern extends beyond the tabulated data, indicating a clear superiority over the other two methods (average value and first value). Furthermore, a correlation coefficient threshold of 0.1 characterises the most accurate models, implying its optimality or near-optimality. This hyperparameter increased in increments of 0.1, showing that the best models only excluded very low correlation thresholds, preserving most edges. Our correlation coefficient heuristics seem equally credible, given their identical outputs for the same hyperparameters set, with the latter as the independent variable. The SA threshold (increment of 0.025) decreases under VADER 1 SA, suggesting higher thresholds increase accuracy. However, when TextBlob yields higher accuracy, sentiment thresholds do not decrease, although they do not begin from the lowest value, indicating the importance of a relatively strict threshold. Though TextBlob generates some highly accurate models, VADER 1 demonstrates more consistent accuracy.

Comparing the accuracy of the best VADER 1 and TextBlob models with their respective Pie charts 3.4 and 5.1, derived from both FTSE 100 and NASDAQ training sets, revealed interesting insights. VADER 1's accuracy improved by 4.2% in our final model compared to the optimal pie configuration, whereas TextBlob's accuracy decreased by 3.3%. These conflicting and relatively insignificant changes suggest that SCN doesn't necessarily enhance accuracy but broadens prediction scope, generating 930 predictions compared to the pies' 231. This fourfold increase in predictions supports the integration of SCN and SA.

Unfortunately, the maximum propagation depth could not be achieved due to memory constraints. Tests with the best models at the maximum propagation depth resulted in either similar or reduced accuracy, suggesting a propagation depth of two strikes an effective balance.

4 chapter

4.1 Discussion and Ethics

Our article data collection methods are efficient for recent results, specifically in collating term-specific articles (with the term being a company). However, gathering historical data poses a challenge, complicating the creation of a robust training set. Enhancing this set could enable better SA verification, potentially reducing the skew observed in our SA results and refining the model. Improvements are required to bolster prediction accuracy and eliminate bias, either by augmenting the testing set or reconsidering our SA approach.

Our SA models predict with over 50% accuracy but are not particularly efficient. Implementing a SA threshold improved prediction quality, albeit reducing their number – a valuable trade-off to preserve capital. Embedded in the SCN, this feature slightly enhances prediction quality, with our best model achieving nearly 70% accuracy on the testing set. However, considering the testing set's non-exhaustiveness, its substantial expansion is advisable before drawing firm conclusions on our SA predictions' accuracy.

Our financial data acquisition method, particularly for the FTSE 100, needs refining due to yfinance's inconsistency. Consider corroborating financial data sources or introducing contingency options for incomplete or missing data. Also, expanding the SCN training set could be beneficial.

While the SCN's impact on prediction accuracy remains unconfirmed, it significantly broadens prediction scope. This feature allows greater portfolio diversification [22] and more frequent stock updates, even in the absence of recent articles or related data. Improving the SA component could yield promising results, but requires more rigorous testing for substantiation.

While we've identified promising hyperparameter values, they warrant fine-tuning. Further experimentation around our current 'optimal' solutions, using smaller increments until the difference between the best solutions becomes negligible, is necessary.

In summary, our best models surpass 50% accuracy, suggesting their utility in stock prediction. However, simulating a portfolio over an extended period would be prudent to truly gauge

our model's efficacy and potential as a stock prediction tool.

4.2 Ethical considerations

Our system's predictions may not always be accurate, potentially leading to financial losses. We cannot be held responsible for any such losses, as we don't offer automated trading, but merely provide market analysis. Therefore, the final trading decision rests with the user. It's advisable to add a disclaimer outlining our non-liability for financial outcomes based on our system's use.

4.2.1 Improvements

The sub-optimal performance of our SA methods warrants exploration of alternative techniques, such as hybrid or machine learning SA methods [10]. Enhanced SA accuracy could translate into improved prediction precision. Further, a comprehensive testing set for our SA model would enable rigorous testing and potentially reduce variation observed in results. Exploring other SA forms, like feature analysis [31], may provide more precise sentiment scores, encompassing multiple companies from one article.

Developing the representation of SCN correlation coefficients might also be beneficial. A polynomial model could offer more nuanced connections between stock entities. If resources permit, each correlation coefficient could even be modelled as a machine learning algorithm, aligning with Agrawal et al's method [2]. This would necessitate accumulating more textual data to incorporate sentiment scores into machine learning models. Random Forest [8], for example, could be trained on sentiment scores and historical stock prices to predict sentiment's impact on future prices.

Momentum investing techniques [24] could offer an alternative to correlation coefficients, though their complexity might increase computation time. Coupled with our propagation approach, this could demand considerable computing power.

It might yield more valuable insight to experimenting with both resetting SA scores daily and allowing them to carry over across days.

Finally, preliminary user testing suggests a preference for a unified user interface or webpage for easier access. Users found the program interpretable but needed assistance during initialisation. Therefore, a combined user interface and automatic stock market integration would improve the user experience.

References

[1] Jeffrey S Abarbanell and Brian J Bushee. "Fundamental analysis, future earnings, and stock prices". In: *Journal of accounting research* 35.1 (1997), pp. 1–24.

[2] Manish Agrawal et al. "Stock prediction based on technical indicators using deep learning model". In: *Computers, Materials & Continua* 70.1 (2022), pp. 287–304.

[3] Haldun Akoglu. "User's guide to correlation coefficients". In: *Turkish journal of emergency medicine* 18.3 (2018), pp. 91–93.

[4] Irene Aldridge. *High-frequency trading: a practical guide to algorithmic strategies and trading systems*. Vol. 604. John Wiley & Sons, 2013.

[5] Stefano Baccianella, Andrea Esuli, Fabrizio Sebastiani, et al. "Sentiwordnet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining." In: *Lrec*. Vol. 10. 2010. 2010, pp. 2200–2204.

[6] H Kent Baker and Victor Ricciardi. "How biases affect investor behaviour". In: *The European Financial Review* (2014), pp. 7–10.

[7] Ricardo N. Bebczuk. *Asymmetric Information in Financial Markets*. Cambridge Books 978052 1797320. Cambridge University Press, Dec. 2003. URL: https://ideas.repec.org/b/cup/cbooks/9780521797320.html.

[8] Gérard Biau and Erwan Scornet. "A random forest guided tour". In: *Test* 25 (2016), pp. 197–227.

[9] Johan Bollen and Huina Mao. "Twitter Mood as a Stock Market Predictor". In: *Computer* 44.10 (2011), pp. 91–94. DOI: 10.1109/MC.2011.323.

[10] Johan Bollen, Huina Mao, and Xiaojun Zeng. "Twitter mood predicts the stock market". In: *Journal of Computational Science* 2.1 (2011), pp. 1–8. ISSN: 1877-7503. DOI: https://doi.org/10.1016/j.jocs.2010.12.007. URL: https://www.sciencedirect.com/science/article/pii/S187775031100007X.

[11] Jonathan Brogaard, Terrence Hendershott, and Ryan Riordan. "High-Frequency Trading and Price Discovery". In: *The Review of Financial Studies* 27.8 (June 2014), pp. 2267–2306. ISSN: 0893-9454. DOI: 10.1093/rfs/hhu032. eprint: https://academic.oup.com/rfs/article-pdf/27/8/2267/24450035/hhu032.pdf. URL: https://doi.org/10.1093/rfs/hhu032.

[12] David P. Brown and Robert H. Jennings. "On Technical Analysis". In: *The Review of Financial Studies* 2.4 (May 2015), pp. 527–551. ISSN: 0893-9454. DOI: 10.1093/rfs/2.4.527. eprint: https://academic.oup.com/rfs/article-pdf/2/4/527/24416052/020527.pdf. URL: https://doi.org/10.1093/rfs/2.4.527.

[13] Louis K.C. Chan and Josef Lakonishok. "Value and Growth Investing: Review and Update". In: *Financial Analysts Journal* 60.1 (2004), pp. 71–86. DOI: 10.2469/faj.v60.n1.2593. eprint: https://doi.org/10.2469/faj.v60.n1.2593. URL: https://doi.org/10.2469/faj.v60.n1.2593.

[14] *ChatGPT*. URL: https://openai.com/blog/chatgpt.

[15] Clay Christensen, Michael E Raynor, and Rory McDonald. *Disruptive innovation*. Harvard Business Review Brighton, MA, USA, 2013.

[16] Ha V. Dang and Mi Lin. "Herd mentality in the stock market: On the role of idiosyncratic participants with heterogeneous information". In: *International Review of Financial Analysis* 48 (2016), pp. 247–260. ISSN: 1057-5219. DOI: https://doi.org/10.1016/j.irfa.2016.10.005. URL: https://www.sciencedirect.com/science/article/pii/S1057521916301570.

[17] Ali F Darrat and Maosen Zhong. "On testing the random-walk hypothesis: a model-comparison approach". In: *Financial Review* 35.3 (2000), pp. 105–124.

[18] Bin Fang and Peng Zhang. "Big data in finance". In: *Big data concepts, theories, and applications* (2016), pp. 391–412.

[19] Ronen Feldman. "Techniques and applications for sentiment analysis". In: *Communications of the ACM* 56.4 (2013), pp. 82–89.

[20] *Flask documentation*. URL: https://flask.palletsprojects.com/en/2.3.x/.

[21] Eric Gilbert and Karrie Karahalios. "Widespread Worry and the Stock Market". In: *Proceedings of the International AAAI Conference on Web and Social Media* 4.1 (May 2010), pp. 58–65. DOI: 10.1609/icwsm.v4i1.14023. URL: https://ojs.aaai.org/index.php/ICWSM/article/view/14023.

[22] William N Goetzmann and Alok Kumar. "Equity portfolio diversification". In: *Review of Finance* 12.3 (2008), pp. 433–463.

[23] Anthony C Greig. "Fundamental analysis and subsequent stock returns". In: *Journal of Accounting and Economics* 15.2 (1992), pp. 413–442. ISSN: 0165-4101. DOI: https://doi.org/10.1016/0165-4101(92)90026-X. URL: https://www.sciencedirect.com/science/article/pii/016541019290026X.

[24] Bruce D. Grundy and J. Spencer Martin Martin. "Understanding the Nature of the Risks and the Source of the Rewards to Momentum Investing". In: *The Review of Financial Studies* 14.1 (June 2015), pp. 29–78. ISSN: 0893-9454. DOI: 10.1093/rfs/14.1.29. eprint: https://academic.oup.com/rfs/article-pdf/14/1/29/24432029/29.pdf. URL: https://doi.org/10.1093/rfs/14.1.29.

[25] Nicholas Huang, Leo Keselman, and Vincent Sitzmann. "Beyond correlation networks". In: ().

[26] Clayton Hutto and Eric Gilbert. "Vader: A parsimonious rule-based model for sentiment analysis of social media text". In: *Proceedings of the international AAAI conference on web and social media*. Vol. 8. 1. 2014, pp. 216–225.

[27] Will Kenton. *What is the secondary market? how it works and pricing*. Dec. 2022. URL: https://www.investopedia.com/terms/s/secondarymarket.asp.

[28] Dennis Victor Lindley. *Bayesian statistics: A review*. SIAM, 1972.

[29] *LZ4 compression library bindings for python¶*. URL: https://python-lz4.readthedocs.io/en/stable/.

[30] Burton G. Malkiel. "Efficient Market Hypothesis". In: *Finance*. Ed. by John Eatwell, Murray Milgate, and Peter Newman. London: Palgrave Macmillan UK, 1989, pp. 127–134. ISBN: 978-1-349-20213-3. DOI: 10.1007/978-1-349-20213-3_13. URL: https://doi.org/10.1007/978-1-349-20213-3_13.

[31] Justin Martineau and Tim Finin. "Delta tfidf: An improved feature space for sentiment analysis". In: *Proceedings of the International AAAI Conference on Web and Social Media*. Vol. 3. 1. 2009, pp. 258–261.

[32] Anshul Mittal and Arpit Goel. "Stock prediction using twitter sentiment analysis". In: *Standford University, CS229 (2011 http://cs229. stanford. edu/proj2011/GoelMittal-StockMarketPredictionUsingTwitterSentimentAnalysis. pdf)* 15 (2012), p. 2352.

[33] *News API – search news and blog articles on the web*. URL: https://newsapi.org/.

[34] *NLTK VADER Documentation*. URL: https://www.nltk.org/_modules/nltk/sentiment/vader.html.

[35] Maureen O'Hara. "Presidential Address: Liquidity and Price Discovery". In: *The Journal of Finance* 58.4 (2003), pp. 1335–1354. DOI: https://doi.org/10.1111/1540-6261.00569. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/1540-6261.00569. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/1540-6261.00569.

[36] Venkata Sasank Pagolu et al. "Sentiment analysis of Twitter data for predicting stock market movements". In: *2016 international conference on signal processing, communication, power and embedded system (SCOPES)*. IEEE. 2016, pp. 1345–1350.

[37] Joseph D. Piotroski. "Value Investing: The Use of Historical Financial Statement Information to Separate Winners from Losers". In: *Journal of Accounting Research* 38 (2000), pp. 1–41. ISSN: 00218456, 1475679X. URL: http://www.jstor.org/stable/2672906 (visited on 04/26/2023).

[38] Joël Plisson, Nada Lavrac, Dunja Mladenic, et al. "A rule based approach to word lemmatization". In: *Proceedings of IS*. Vol. 3. 2004, pp. 83–86.

[39] Leonard Richardson. *Beautiful Soup documentation¶*. URL: https://beautiful-soup-4.readthedocs.io/en/latest/.

[40] *Running python functions as pipeline jobs¶*. URL: https://joblib.readthedocs.io/en/latest/.

[41] *SQLITE3 - DB-API 2.0 interface for SQLite databases*. URL: https://docs.python.org/3/library/sqlite3.html.

[42] *Textblob sentiment analysis library*. URL: https://pypi.org/project/textblob/.

[43] *The Python Standard Library*. URL: https://docs.python.org/3/library/.

[44] *VADER Sentiment library documentation*. URL: https://pypi.org/project/vaderSentiment/.

[45] *Yfinance*. URL: https://pypi.org/project/yfinance/.

Acknowledgments

I would like to thank:

Dr. Marcos Oliveira - for all his help during the course of the thesis.

Frederico Richardson,

Pamela Fernandez Figueiredo,

David Proctor - for their assistance in providing human assessments for the sentiment analysis training set, and providing user feedback on pertinent additions to the program.

5

Appendix

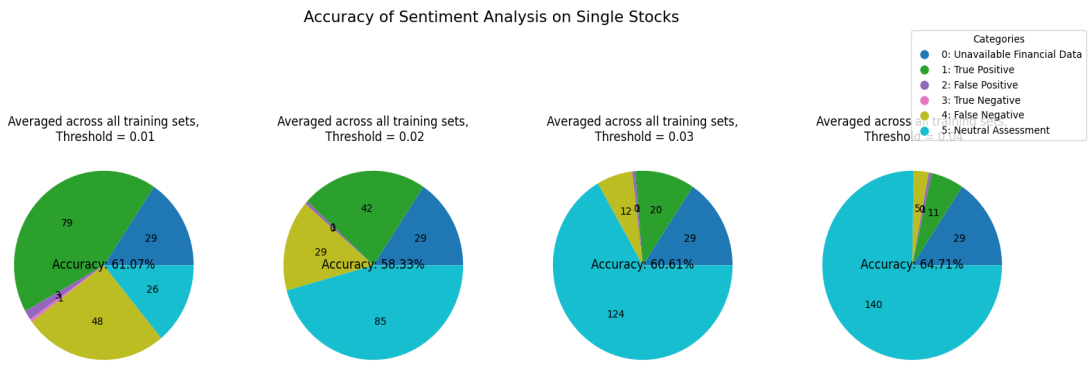

Figure 5.1: SA Predictions with thresholds, VADER Method 1, with Pre-processing

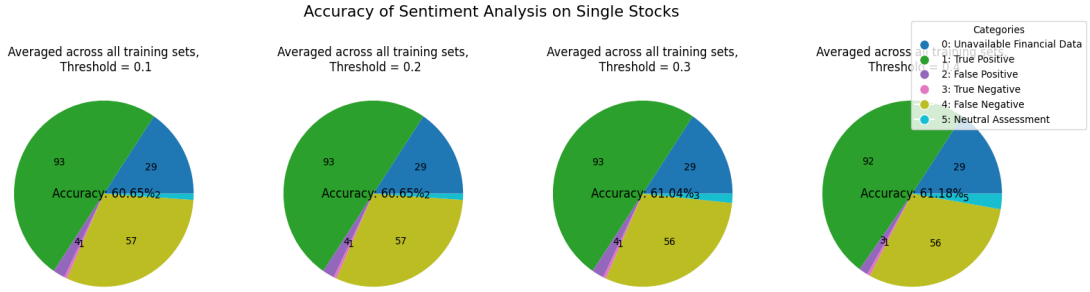

Figure 5.2: SA Predictions with thresholds, VADER NTLK, with Pre-processing