NOVA IMS Information Management School

MDSAA

Master's Degree Program in

Data Science and Advanced Analytics

Using Consumer Online Reviews to Predict Stock Price Returns in the Ridesharing Industry

Pedro Ricardo Félix Fernandes

Master Thesis

presented as a partial requirement for obtaining a Master's Degree in Data Science and Advanced Analytics

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

Using Consumer Online Reviews to Predict Stock Price Returns in the Ridesharing Industry

by

Pedro Ricardo Félix Fernandes

Master Thesis presented as a partial requirement for obtaining the Master's degree in Data Science and Advanced Analytics, with a specialization in Data Science

Supervised by

Roberto André Pereira Henriques, PhD, NOVA Information Management School

November, 2024

STATEMENT OF INTEGRITY

I hereby declare having conducted this academic work with integrity. I confirm that I have not used plagiarism or any form of undue use of information or falsification of results along the process leading to its elaboration. I further declare that I have fully acknowledged the Rules of Conduct and Code of Honor from the NOVA Information Management School.

[Lisbon, November 2024]

DEDICATION

To my parents, since this paper only exists because of them.

They were the ones who pushed, supported and motivated me through the whole process but also who brought me to life and gave me all the tools for me to become who I am today and who I will become in the future.

Thank you.

ABSTRACT

This research explores the predictive power of online consumer reviews on weekly stock price returns within the ridesharing industry, focusing on Uber and Grab, two companies with very distinct characteristics. The user-generated reviews were obtained from the Google Play Store and leveraged to determine how their features (such as sentiment, rating, and textual content) correlate with stock performance. This data is joined with Stock data from Yahoo Finance and aggregated daily and weekly to be inputted into machine learning models, specifically Recurrent Neural Networks with Long Short-Term Memory (RNN-LSTM) and XGBoost. After identifying the optimal time lag for the predictions, the results reveal high predictive capacity from some sets of features, mainly regarding the review content itself. These findings confirm the benefits of considering user feedback in financial forecasting by showing that including features such as the most common words and the rating given by users increases performance for stock price prediction, especially in terms of the price return direction (positive or negative). By demonstrating the value added in integrating qualitative online review data into quantitative stock prediction, this work contributes to financial analytics, offering relevant insights for investors and industry stakeholders to understand how their customers influence their economic outcomes.

KEYWORDS

User Generated Content; Stock Price Returns; Prediction; Deep Learning; Ridesharing;

Sustainable Development Goals (SDG):

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

MAE	Mean Absolute Error
MDA	Mean Directional Accuracy
MSE	Mean Squared Error
RNN-LSTM	Recurrent Neural Network with Long-Short-Term Memory
UGC	User-Generated Content

1. INTRODUCTION

With the emergence of the Internet and its social platforms, it has become more accessible for people to share their opinions and feedback online. Among these platforms, the Google App Store is one of the most significant sources of user-generated content (UGC) since the consumers of the most dominant mobile operating system need it to download applications and share their opinions about a product - Smartphones with an Android system represent 70% of the mobile operating systems market share (Statcounter, 2024).

This increase in consumer feedback presents a unique opportunity to explore its prediction power on a firm's financial performance metrics, namely stock price returns. Our work investigates the potential of using consumer online reviews from the Google App Store to predict weekly stock price returns, focusing on the ridesharing industry.

The economic significance of the ridesharing industry goes beyond providing more convenient and efficient transportation solutions when compared with public transportation (while being more cost-effective than traditional taxi options). Globally, this industry is a key driver of the digital economy, reflecting trends in urban mobility, smartphone penetration and consumer preferences for shared economy models (Gössling & Michael Hall, 2019). Ridesharing companies like Uber and Grab have disrupted traditional taxi services, leveraging innovative business models to achieve rapid scalability and reach profitability and proof of that is the forecast of this market segment growing from $185.64 billion in 2024 to $294.11 billion by 2029 (Mordor Intelligence, 2024).

However, this industry relies heavily on customer satisfaction and feedback, highlighting the importance of maintaining positive customer relationships. Positive reviews can lead to increased user acquisition and retention, which consequently affects revenues and market share. On the other side, negative reviews can quickly affect companies' trust, leading to the opposite effect. Therefore, it is essential for ridesharing companies to maintain a competitive edge and drive economic growth by understanding and leveraging customer feedback (Rayle et al., 2016).

Consumer's voices are more empowered than ever, as online reviews and ratings shared on social media platforms influence potential customers and provide a rich data source for managers and investors. While previous studies have established a link between customer satisfaction and financial performance, most have focused on more traditional industries and other more conventional data sources (Fornell et al., 2006; Luo, 2007). The novelty of this research lies in its focus on leveraging unstructured user-generated content from the Google Play Store to predict stock price returns in the ridesharing industry.

Unlike traditional data sources, online reviews provide real-time qualitative insights into customer perceptions and experiences. By applying advanced natural language processing techniques and machine learning models to this data, this study quantifies and analyzes consumer sentiment at scale, offering a more immediate understanding of its impact on financial performance.

The ridesharing industry serves as an case study for analyzing the junction of technology, consumer feedback and financial markets. The industry's reliance on mobile technology and real-time data also aligns with the global trend of mobile technology dispersion, making it a good ground for applying innovative analytical methods such as machine learning (Cramer & Krueger, 2016).

By focusing on this industry, the study captures the dynamics of a global market trend where consumer feedback turns into a fundamental component of the business model and not just a consequence of a product's dispersion. The platforms are built on network effects, where the customer reviews directly influence both supply (drivers) and demand (riders), both increasing as more users participate. Analyzing this industry also provides insights with broader implications for other sectors undergoing digital transformation and emphasizes the critical role of user-generated content in today's economy.

Several theories and models highlight the relationship between consumer feedback (positive and negative), user-generated content, and stock performance. Customer satisfaction was shown to be correlated positively with financial performance metrics such as market value and stock returns. For instance, it was demonstrated that portfolios of companies with high American Customer Satisfaction Index (ACSI) scores significantly outperformed the market. Positive consumer feedback mechanisms, including repeat business and reduced marketing and acquisition costs, enhance firm valuation and stock prices (Fornell et al., 2006).

Conversely, negative consumer feedback can have the opposite effect on a firm's financial performance. Luo (2007) demonstrated that the consumer negative voice (CNV) negatively influences future stock returns, indicating the importance of managing consumer dissatisfaction to maintain stock performance.

Additionally, the increasing flow of user-generated content through platforms such as X, Amazon, and the Google Play Store has added new sources to support financial performance analysis, so much so that studies have shown that online reviews and ratings can significantly influence consumer behaviour and stock prices. Interestingly, Chevalier & Mayzlin (2006) found that negative reviews have a more pronounced impact on decreasing sales than positive reviews have on increasing them.

The research objectives of this thesis are:

1. To assess the predictive performance of consumer online reviews from the Google Play Store on the stock price returns of ridesharing companies.

2. To identify and analyze the characteristics of online reviews (such as sentiment, rating, or category) that have the highest predictive performance for stock price returns.

3. To identify the optimal time lag for reviews to influence stock price changes.

The data collected will be regarding two firms, Uber and Grab, and will be user-generated content with features such as review text, rating, and date, as well as stock price data, which includes daily stock prices and volumes. Data preprocessing will involve cleaning and reducing data, tokenizing the review text, and performing sentiment analysis and categorization using pre-trained natural language models. Data will be aggregated daily and weekly, and we'll test various machine learning models, including Recurrent Neural Networks with Long Short-Term Memory Layers and a boosting algorithm (XGBoost).

This thesis aims to contribute to the existing knowledge by providing evidence on the predictive power of consumer reviews from the Google Play Store in the ridesharing industry and identifying key review characteristics that influence stock price movements and the optimal time lag for that influence to have an effect. The findings are expected to offer valuable insights for investors, analysts, and managers, emphasizing the importance of incorporating consumer feedback into financial decision-making.

To achieve those goals in this research, we employed different steps: A Literature Review was performed to understand what previous work was done on this topic and identify research gaps, followed by defining the Methodology based on what was found in previous works, using the CRISP-DM framework. After that, the results were analyzed and discussed to infer the desired conclusions about the topic.

2. LITERATURE REVIEW

2.1. Relationship between Customer Satisfaction and Financial Performance

The linkage between shareholder returns and customer satisfaction is fundamental to a capitalistic market system, as previous research has shown that customer satisfaction positively impacts a firm's economic performance (Fornell et al., 2006).

According to the American Society for Quality, customer satisfaction is a measurement that determines how happy customers are with a company's product, influencing both micro and macroeconomic aspects of firms potentially aligning equity markets with consumer utility.

At a macro level, rising customer satisfaction boosts economic growth through increased consumer spending. In contrast, at a micro level, it leads to increased net cash flows while reducing cash flow volatility and consequently increased revenue growth for firms, rewarding shareholders who invest in companies with high customer satisfaction (Fornell et al., 2006; Tuli & Bharadwaj, 2009).

Mittal et al. (2023) confirmed that higher customer satisfaction is associated with outcomes such as positive revenue growth, profitability and shareholder value, while also enforcing the idea that customer satisfaction provides firms with a competitive advantage by reducing churn rates and instigating loyalty behaviors.

When measured by the American Customer Satisfaction Index (ACSI), customer satisfaction has a sizeable significant link to higher market value of equity, with a 1% change in ACSI score corresponding to a 4.6% change in market value. Stock portfolios based on high ACSI scores can outperform the market with a lower systematic risk, and the simple inclusion of a firm in the ACSI may increase its visibility and lead to higher returns (Fornell et al., 2006).

As Fornell tested, a customer satisfaction-focused portfolio achieved a cumulative return of 518%, significantly outperforming the S&P 500 (a stock market index tracking the stock performance of 500 of the largest companies listed on stock exchanges in the United States), which grew by only 31% over the same period of 15 years. During that period, the customer satisfaction portfolio outperformed the market (S&P 500) 14 out of 15 times. These results were allied with a significant positive correlation with positive earnings surprises, even when accounting for various risk factors, which further positively influence stock prices (Fornell et al., 2006).

Similarly, T. Gupta et al (2021) utilized high-frequency consumer spending as a proxy for customer satisfaction and showed that firms' quarterly earnings surprises could be predicted up to three quarters in advance, which shows how consumer data can decrease the gap of information between investors and firms, with higher spending patterns by satisfied customer's being correlated to positive stock performance.

2.1.1. Positive Consumer Feedback and Financial Performance

There are different mechanisms through which customer satisfaction translates into enhancing financial performance and long-term growth, such as the improvement of repeat business, products' usage levels, future revenues, positive word-of-mouth, reservation prices, market share, productivity, cross-buying and cost-competitiveness. The tendency is also to reduce consumer complaints, transaction costs, price elasticity, warranty costs, field service costs, defective goods, customer churn, and employee turnover (Fornell et al., 2006).

With all this combined, it is expected and logical to assume that an increase in customer satisfaction will promote an increase in stock prices since all those factors influence company valuations. According to Srivastava et al (1998), there are four major determinants of a company's market value:

- Acceleration of Cash Flows: Affected by the speed of buyer response to marketing efforts, customer satisfaction positively influences this component since it is less costly to persuade a satisfied customer.
- Increase in Cash Flows: Customer Satisfaction helps increase the number of repeat businesses and future revenues while helping achieve lower transaction, warranty, and marketing costs, which results in higher net cash flows.
- Increase in the Residual Value of the Business: Being a function of size, loyalty and quality of the customer base, it is related to the satisfaction level of that same customer base.
- Reduction of Risk Associated with Cash Flows: Customer satisfaction increases also significantly reduce both overall and downside systematic and idiosyncratic risk associated with future cash flows, leading to stock price growth since the reduction in the variability of cash flows reduces the cost of capital. This implies that customer satisfaction provides information to financial markets that complements traditional accounting metrics, and these findings are robust across various model specifications and alternative risk measures (Tuli & Bharadwaj, 2009).

Customer satisfaction can also act as financial guard in moments of economic uncertainty as found by Gunawardane (2023), who highlighted that better customer experiences and higher satisfaction levels contribute to more elevated financial stability, since financial institutions reported less revenue volatility and higher recovery rates after shocks the more focused they were on customer satisfaction.

For managers, customer satisfaction can lead to more stable and predictable financial outcomes, making it a crucial aspect of corporate strategy. For investors, customer satisfaction scores can be useful for assessing investment risk, particularly insulating portfolios from market downturns (Fornell et al., 2006). This supports the importance of customer satisfaction as a strategic asset for firms.

2.1.2. Negative Consumer Feedback and Financial Performance

As important and significant as positive consumer feedback might be, it is also important to analyze the effect of negative consumer feedback on firms' financial performance.

Consumer negative voice is defined as a manifestation of consumer complaint responses to dissatisfaction or the behavior of communicating the incidents of an unhappy product/service consumption experience (Luo, 2007). But how does this negative firm perception from consumers and overall signals of dissatisfaction affect a firm's financial performance?

The way consumer negative feedback can make itself heard nowadays is much more extensive thanks to the Internet, which expands further the conclusions from (Richins, 1983) in which the author highlighted the effect of the word-of-mouth of dissatisfied customers who tell eight to ten friends and other people about the unhappy experience. When angry, 20% of those might share the experience with 20 potential buyers, leading 50% of those, on average, not setting foot in that store and not being influenced by the firm's marketing campaigns.

Studying the airline industry Luo (2007) concluded that current consumer negative voice indeed has a negative influence on future idiosyncratic stock returns and that higher levels of CNV are associated with lower future idiosyncratic stock returns for firms.

This suggests that consumer negative voice has direct, relevant, and substantial implications for firms' stock value. A strong consumer negative voice hampers the existence of a valuable intangible asset, such as a large base of satisfied customers, which causes difficulties in creating customer equity (S. Gupta et al., 2004; Riley et al., 2003; Singh, 1988). This results in signalling reduced prospects of the companies' stocks to financial analysts, thus leading to less idiosyncratic excessive stock returns (Campbell et al., 2001; Fama & French, 1993, 2007).

Another intangible asset that could affect future idiosyncratic stock returns by leading to lower levels of anticipated cash flows is the lack of a firm's good brand image and solid public reputation. This is because, in the competitive marketplace, a large amount of negative consumer voice would erode the brand image and lead to diminished "reputational capital" of the firm (Richins, 1983; Singh, 1988).

Interestingly, it appears that negative feedback has a more substantial effect on firms' financial performance than positive feedback, and its predictive power is stronger:

Chevalier & Mayzlin (2006) stated that "negative review is more powerful in decreasing book sales than positive review is in increasing sales". Similarly, Su et al (2023) found that customer online satisfaction impacts stock returns in China's A-share market, with Consumer Negative Sentiment Tendency (CNST) and One-Star Tendence (OST) negatively impacting expected stock returns and showing a stronger predictive power in firms with high sentiment conditions and growth companies while negatively predicting revenue surprises, earnings surprises and cash flow stocks. This suggests that online satisfaction contains novel information about firms' fundamentals. In low sentiment periods, this predictive power is reduced. Firms with lower accounting transparency also showed a stronger negative impact of CNST on stock returns, indicating that investors rely more on consumer opinions when financial information is less transparent.

Luo (2007) also analyzed how consumer satisfaction and dissatisfaction affected a firm's stock value gap (the difference between the actual and the optimal market value – a manifestation of shortages in firms' future cash flows compared with the benchmark). Satisfaction negatively correlates with the stock value gap, supporting that higher satisfaction approximates companies to their optimal market value, while higher customer complaints increase the stock value gap. However, complaints have a relatively stronger impact than satisfaction on the stock value gap.

The works we analyzed so far show that it is viable to incorporate customer feedback into financial performance analysis, including stock price prediction, but with a great variety of results. Due to that the first research gap we identified was the lack of assessment on how well consumer feedback perform in predicting stock price returns on different industries such as, for example, the ridesharing industry.

2.2. General Impact of User-Generated Content

User-generated content is any content that comes from regular people who voluntarily contribute data, information, or media that then appears before others in a useful or entertaining way, usually on the Web—for example, restaurant ratings, wikis, and videos (Krumm et al., 2008) and it has undoubtedly been increasing with the development of the Internet.

This type of content is important because it can be seen as a proxy for an individual's consumer satisfaction level regarding a product or company. Similar to Consumer Satisfaction, prior research shows that UGC promotes effects on firms' financial performance and sales, impacting new customer acquisition and movie box office sales. It also affects variables such as buyer intention, product involvement, information sharing and brand engagement (Duan et al., 2008).

The emergence of the Internet and Social Media has changed the way investors invest through platforms that provide user-generated content. UGC in channels such as Yahoo Finance and Raging Bull enriches investors' ability to make better investing decisions by allowing them to monitor the thought process and decision-making of others. These platforms, particularly those focusing on stock market discussions, provide high-volume information in real-time that could be highly predictive of future stock movements (Antweiler & Frank, 2001).

In Marketing, it was already popular to study how consumer behavior in different formats, such as reviews, user ratings, or blogs, affects financial outcomes. Now more than ever, these types of word-of-mouth are significant tools that influence and build trust for companies because of the Internet's low-cost ability to reach a high number of people. Chevalier & Mayzlin (2006) confirmed that customer word-of-mouth affects consumer purchasing behavior by observing the relationship between the number of reviews and average ranking with sales outcomes. Duan et al (2008) concluded that businesses should focus more effort on facilitating consumer word-of-mouth exchanges instead of user ratings because the volume of postings indeed influences box office sales. Godes & Mayzlin (2002) found that blog posts volume correlates positively with future sales of music CDs.

Regarding Microblogging (from platforms such as X, Instagram, and Facebook), a few scholars have attempted to examine the relationship between predictors extracted from microblogs and future outcomes such as movie revenue, events and stock prices. Bollen et al (2011) used an extended version of the Profile of Mood States (POMS) to extract six mood dimensions from over 9 million Twitter postings and found significant correlations between extracted mood dimensions and cultural, social, economic, and political events occurring the same day. Asur & Huberman (2010) extracted sentiments from 6 million Twitter postings to predict box office revenue for movies and obtained an accuracy of 0.94 when benchmarking against the Holliwood Stock Exchange (HSX).

UGC can also be presented in the form of ratings, present in multiple platforms from online retailers such as Amazon, to application stores like the Google Play Store. Huang (2018) found that abnormal customer ratings positively predict subsequent stock returns. A portfolio that buys stocks with abnormal customer ratings (the difference between the average rating in a month and the average rating during the prior 12 months) in the top tercile and sells stocks in the bottom tercile generates an abnormal return of about 55.7 to 73.0 basis points per month. The author also found that abnormal customer ratings positively predict revenue and earnings surprises. The return predictability does not reverse in the long run and is also a significant predictor of net purchases by hedge fund managers, suggesting that sophisticated investors exploit the information contained in consumer opinions.

These findings provide evidence that the aggregated opinions of consumer crowds contain valuable information about cash flows and stock pricing, highlighting the role of consumers as information producers in financial markets. Compared with traditional information intermediaries such as equity analysts, consumer crowds can provide more timely information on a company's products and cash flows (Huang, 2018).

However, not all firms react in the same way to UGC, as it highly depends on the firm's characteristics and strategies. Liu (2020) demonstrated that UGC impacts firms' financial performance differently between B2B and B2C companies. This is mainly due to B2C customers being particularly more powerful on social media when compared to B2B firms' customers, since the sheer volume of UGC generated by B2C customers is much higher. Word-of-mouth information is still important for B2B companies, but just not as impactful as for B2C firms (Liu, 2020).

Finally, Chahine & Malhotra (2018) concluded that firms employing two-way interaction strategies saw a higher market reaction than those using one-way messaging. This positive market reaction was also higher for smaller firms, firms with losses, and those with a family or dominant shareholder. They also showed that the impact of social media strategies varies across industries, with significant positive reactions observed in sectors like mining, oil, gas, construction and transportation.

After having concluded that incorporating user feedback was helpful for the task of predicting financial performance of firms, the studies we analyzed in section 2.2. of this study showed us that one of the relevant forms that user feedback can have is online generated content, such as Google Play Store reviews but also that the influence of this type of content also depends on firm characteristics such as company size. This lead us to identifying a new research gap, which is the lack of understanding of what specific characteristics of those online generated reviews influence or offer the highest predictive performance for stock price returns.

2.3. PREVIOUSLY APPLIED METHODOLOGIES

Multiple methodologies were applied in previous works that can be insightful for our research, even if the main objective of their studies wasn't fully connected with our research.

Awad et al (2023) used NLP to preprocess news and media data and discern market sentiments related to stocks while also sourcing historical stock price datasets from Yahoo Finance. Darapaneni et al (2022) used the output of sentiment analysis as input for Deep Learning algorithms, namely an LSTM and a Random Forest Model, to predict stock price in the Indian market, while Koukaras et al (2022) also applied Sentiment from Microblogging as input for other seven Machine Learning models with the same objective, but validating it on Microsoft Stock. Similarly to the previous one, Xiao & Ihnaini (2023) predicted stock trends using sentiment analysis from news and tweets, by applying it to six different models, in four different companies'stocks: Amazon, Netflix, Apple and Microsoft.

Additionally to showing proof that including Sentiment Analysis outputs in stock prediction tasks shows an improvement in performance, all the previously mentioned studies pointed to the importance of text processing, namely the removal of html tags, special characters and stop words. Also, Lemmatization of text was endorsed by Darapaneni et al (2022). Another important text preprocessing step was pointed by Ho & Huang (2021) and Palomino & Aider (2022), which was the process of lowercasing text since the first work was also using

Sentiment Analysis to predict stock price movement while the second work evaluated the effectiveness of Text Pre-Processing in performing Sentiment Analysis.

Still connected with Sentiment Analysis, the previous work that more relates with our scope was performed by Mahmud et al (2022) which performed Sentiment Analysis on Google Play Store reviews from Ridesharing Companies. For that, he firstly pointed the importance of performing Tokenization and then leveraged a pre-trained DistilBERT model to perform the task.

In terms of Modelling and Evaluation, two works stand out as essential sources. Wu et al(2019) predicted long-term returns of individual stocks using Online reviews, having achieved the best results with an XGBoost model. The process of selecting the prediction target by leveraging a correlation analysis between all the features and targets was, although, a very relevant contribution for our study, similarly to the process of weekly aggregating the data, in order to deal with the 18 million consumer reviews that were available for their work.

On the second work, Mohan et al (2019) aggregated data daily and applied it to a Recurrent Neural Network with Long Short Term Memory, but contributed also to our work with the idea of using the Mean Directional Accuracy as a way to evaluate the models.

In terms of methodologies, a research gap we identified is related to the prediction lag of online reviews, specifically for the Ridesharing Industry. Although Wu et al (2019) identified 8 weeks to be the optimal prediction time lag for individual stocks for products from JD.com, we are uncertain of how that relationship behaves in our selected Industry and also if it is dependent on how data is aggregated.

3. METHODOLOGY

The methodology followed on this research was based on the CRISP-DM framework which visual representation can be seen on Figure 3.1:

Figure 3.1 – Diagram of the CRISP-DM process (Tounsi et al., 2020)

3.1. Business Understanding

The first step was to understand the business and problem areas that we are trying to solve or improve (Tounsi et al., 2020). This was achieved in the Literature Review chapter of this research, where we defined our 3 research objectives:

1. To assess the predictive performance of consumer online reviews from the Google Play Store on the stock price returns of ridesharing companies.
2. Identify the specific characteristics of online reviews that contribute most significantly to prediction performance
3. Determine the optimal time lag for the influence of consumer reviews from Google Play Store on weekly stock price returns.

To achieve these objectives it was also imperial to collect data from companies that allow us to generalize results to the rest of the Ridesharing Industry, hence we chose Uber Technologies Inc. and Grab Holdings Inc.

3.1.1. Firms Selection

The news that Uber Technologies posted its first full-year profit as a public company in 2023 and projected continue growth in the first quarter of 2024 (Rana, 2024) as well as the news that Grab posted its first profitable quarter in the fourth-quarter of 2023 (Chiang, 2024), presented a great opportunity to test the results and conclusions explored in the Literature Review in a relatively new industry within the transportation sector – the ridesharing industry.

Uber Technologies Inc. was founded in San Francisco in 2019 and it is the largest ridesharing company worldwide, being present in approximately 70 countries and 10,000 cities. Grab Holdings Inc. was founded in 2012 in Singapore and in March 2018 merged with Uber's Southeast Asia operations, taking over Uber's assets and operations as part of the acquisition done in an attempt to become the #1 online-to-offline mobile platform in Southeast Asia and a major player in food delivery (Grab, 2018).

The choice of these companies was based on key differences and similarities in their characteristics that are relevant to make a fair comparison of results while allowing us to test how different firm characteristics affect our results.

Similarities:

- Industry: Both companies act on the Ridesharing Industry, which was essential for the choice.
- IPO timing: Both companies went public recently: Uber Technologies Inc. launched its IPO on May 9th 2019, with a valuation of $75 billion, while Grab had an valuation of almost $40 billion on its IPO day in December 2nd 2021.

Differences:

- Specialization: Uber focuses mainly on transportation (taxi, food, package delivery and freight transportation, while Grab operates a "super-app" and is involved in multiple industries (food, grocery and parcel delivery, technology, transportation, e-commerce, online payments, financial services, and vehicle rental).
- Consumer base: Uber's consumer base is bigger than Grab's as it reported in the last quarter of 2023 a volume of 150 million active users per month while Grab reported for the same period almost 38 million users per month (Grab, 2024).
- Review Volume: Uber’s bigger consumer base and more diverse market presence also translates into a bigger volume of online reviews compared with Grab across the worldwide available application stores’ platforms.

Selecting only two firms could, however, introduce potential biases that could affect our results’ generalizability:

- Sample size bias and Overrepresentation of Market Leaders: In addition to a sample size of two firms being limited, with potential of not being able to capture the full diversity of the ridesharing industry, both Uber and Grab are industry leaders, which might bias the analysis to reflect the dynamics of well-established and successful firms, ignoring what would be the experience of lower-tier or emerging players in this industry that might present different relationships between consumer feedback and stock price. This means that our findings can be overoptimistic on the role of consumer reviews in influencing stock price performance, since larger firms often receive a higher volume of reviews and more frequent investor attention or could be over influenced by firm-specific characteristics (Brown et al., 1992; Elton et al., 1996).
- Technology Maturity Bias: Also related with being market leaders, both companies represent technologically mature firms in the ridesharing industry, which means that the generalization of results for other firms might not work since emerging (or less technologically advanced) firms might not exhibit the same relationship between consumer feedback and stock price.
- Industry Representation Bias: Although both firms operate in the ridesharing industry, grab also operates in multiple other industries through its "super-app" business model that includes services such as financial services and vehicle rental. This means that our results, when regarding Grab, might be reflecting the influence of the other industries in which Grab operates, which might affect the generalization process to other ridesharing firms.
- Inter-Dependence Bias: With Grab's acquisition of Uber's Southeast Asia operations in 2018, this historical relationship may create shared consumer bases or review dynamics that are not entirely independent, which might distort results.

3.2. DATA UNDERSTANDING

In this phase we determined the data to be analyzed and identified the quality of the data available (Tounsi et al., 2020).

3.2.1. Data Collection

Two types of data were collected for both companies:

- User-generated reviews of their ridesharing app.
- Historical Stock Price data

Regarding User-Generated Reviews, two datasets were acquired from Kaggle, both with a Tier of 10.0 for Usability, ensuring data quality in terms of Completeness, Credibility and Compatibility. These datasets aggregate consumer reviews from Google Play Store.

The choice of this platform (Google Play Store) for the source of data was made because we wanted to choose a platform that eliminates the most amount of barriers for users to share their opinion, to have a high volume of data but also diversification of users. Google Play Store was an easy choice since it is the main platform for downloading applications for users of smartphones with the Android operating system, which represents 70% of the global mobile operating system market share (Statcounter, 2024).

The dataset for Uber was collected through Python with the google_play_scraper API. It has 1455377 entries with 8 features that are described in Table 3.1:

Table 3.1 - Uber Dataset Features

Feature	Data Type	Description
review_id	String	An unique identifier of the review
pseudo_author_id	String	An user-unique identifier of the author of the review
author_name	String	An identifier of the author of the reivew
review_text	String	The textual content of the review
review_rating	Integer	An integer from 1 to 5 that represents the rating given by the user to the app
review_likes	Integer	An integer that represents the number of users that liked that review
author_app_version	String	An identifier of the app version the user was running when the review was posted
review_timestamp	String	A string with the date timestamp of the moment the user posted the review

The dataset for Grab was collected through Python using the RAPIDAPI API. It has 441546 entries with 7 features described in Table 3.2:

Table 3.2 - Grab Dataset Features

Feature	Data Type	Description
author_id	String	An user-unique identifier of the author of the review
author_name	String	An identifier of the author of the reivew
review_text	String	The textual content of the review
review_rating	Integer	An integer from 1 to 5 that represents the rating given by the user to the app
review_likes	Integer	An integer that represents the number of users that liked that review
author_app_version	String	An identifier of the app version the user was running when the review was posted
review_timestamp_utc	String	A string with the date timestamp of the moment the user posted the review in utc format

The data for the Stock Price Data was collected using Python with the yfinance API starting on the day of their respective IPO. We collected 624 rows of data for Grab and 1271 rows for Uber, both with the same 7 features which description can be found on Table 3.3:

Table 3.3 - Stock Data Features

Feature	Data Type	Description
Date	String	A string with of the stock market date in the format “yyyy-mm-dd”
Open	Float	The amount referent to where the stock’s price opened for trading on that given day in USD
High	Float	The amount referent to the highest price at which a stock traded during that day in USD
Low	Float	The amount referent to the lowest price at which a stock traded during that day in USD

Close	Float	The amount referent to where the stock's price closed for trading on that given day in USD
Adj Close	Float	The amount referent to where the stock's price closed for trading on that given day after adjustments for all applicable splits and dividend distributions in USD
Volume	Integer	The amount of stocks that were traded during that day in USD

3.2.2. Data Exploration

We started by analyzing Reviews data for Uber:

- review_id: Being the identifier of the review, it is unique, meaning there aren't two reviews with the same review_id
- pseudo_author_id: Being the identifier of the author of the review, there could be the case that a value could be present more than once, but it wasn't the case meaning there aren't two reviews with the same pseudo_author_id associated
- author_name: There are 822779 author_name entries equal to "A Google user", which represents 57% of the entries, while the other author names are masked with asterisks such as "al*******id". This means that we found 83911 author names that repeat at least one, but this value shouldn't represent the truth since, as an example, there are 3938 author names masked as "***", which could represent any author name with 3 characteres.
- review_text: In addition to 518 nulls, representing 0,04% of the data from Uber, there are several reviews with repeated text, being the top 5: 93721 "Good" (6% of the reviews), 29753 "Nice" (2%), 18399 "Excellent" (1.3%), 17449 "good" (1.2%) and 14075 "Very good" (1%). We also notice that some reviews such as "Good" and "good" only differ in letter capitalization. In total there are 31169 review texts that are repeated at least once.
- review_rating: This feature has a very unbalanced distribution of ratings as we can we see in Figure 3.2:

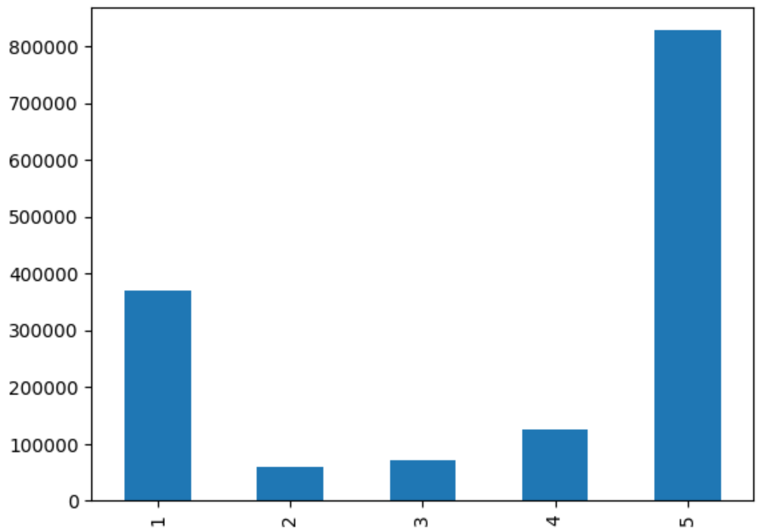


Figure 3.2 – Uber Reviews Ratings Distribution

This shows a clear tendency for users to evaluate the Uber app either very well (5) or very poorly (1).

- review_likes: This feature has a very complex distribution: The great majority of reviews has 0 likes (1324592 – 91% of the reviews), and the review with the highest amount of likes has 5571.
- author_app_version: There are 741 unique strings for this feature, with the most frequent being “4.451.10003” with 15450 entries. However, there are 371 300 nulls, representing 25,5% of the data from Uber. The distribution shows a very unbalanced usage of diferent versions as can be checked in Figure 3.3:

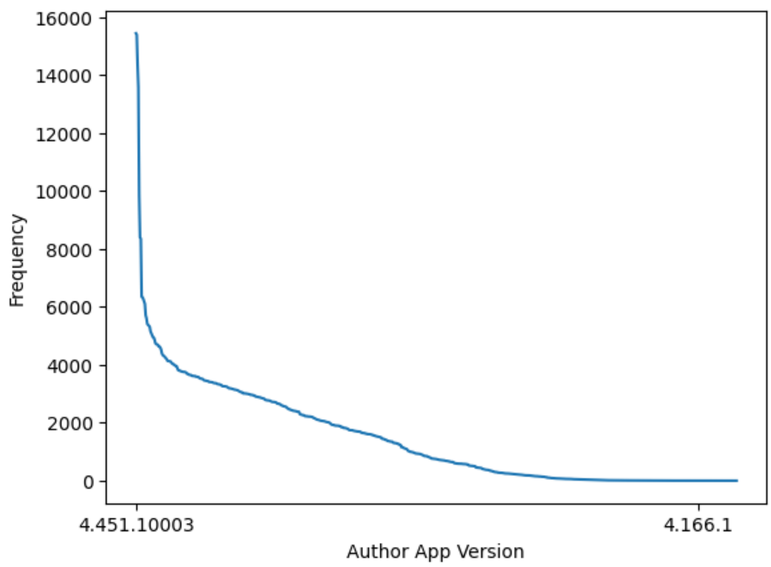


Figure 3.3 – Uber Author App Version Distribution

- review_timestamp: The 29th of January of 2017 was the day with the highest frequency, meaning 1883 reviews were posted that day, followed by the following day with 1674 reviews. The average number of reviews made per day is slightly above 355, with the median being a bit higher on 375. It is also important to note that the reviews collected started on the 3rd of November 2010 and ended on the 7th of November 2023, which basically comprises 13 years of data.

  In Figure 3.4 we can see that the reviews made on Google App Store related to Uber app have the following distribution throghout time:

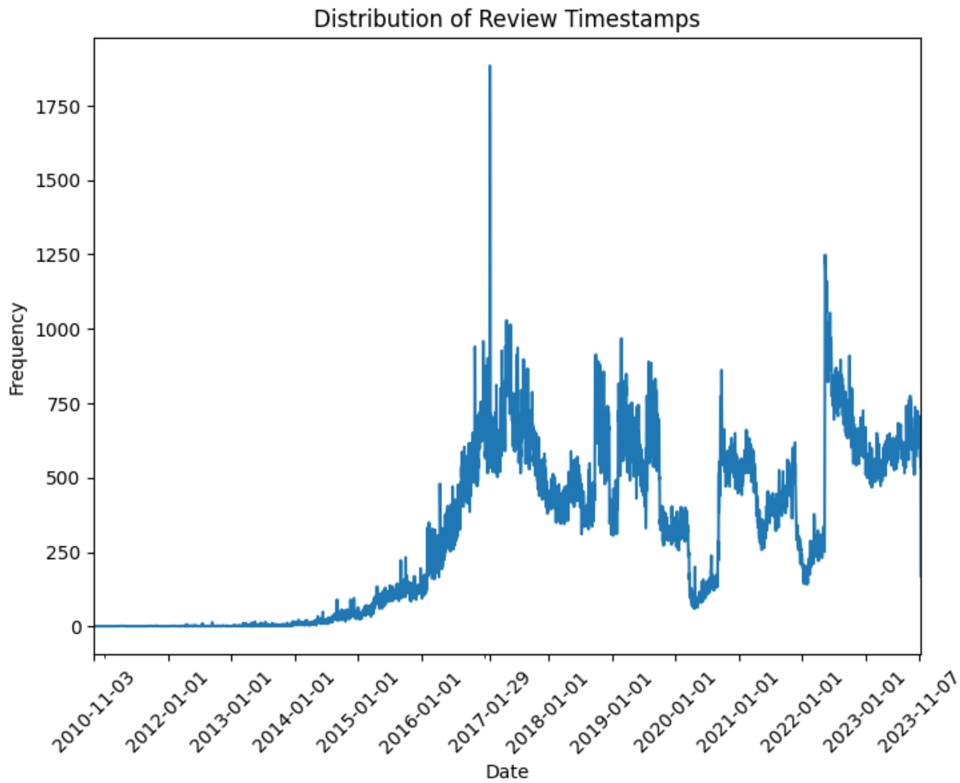


Figure 3.4 - Uber Reviews Distribution by Date

A similar analysis was performed for Grab's reviews data:

- author_id: Differently from what we saw for Uber, in the case of Grab there are 3 author ids that have 2 entries each. This feature has 72 nulls entries, representing 0,016% of the data from Grab.
- author_name: In similarity to what happened in Uber, there are 440089 cases of the author_name being "A Google user" which represents more than 99% of the data. The names are masked in the same way, using asterisks, but only 10 author names are repeated in the whole dataset.

- review_text: In addition to 111 nulls, representing 0,025% of the data from Grab, 10585 review texts are repeated at least once, being the top 5: “Good” 21968 (5% of reviews) , “good” 15646 (3.5%) , “Ok” 4659 (1.1%), “Nice”4338 (1%) and “ok”4256 (1%).
- review_rating: The ratings distribution for Grab (Figure 3.5) is very similar to the one we observed for Uber:

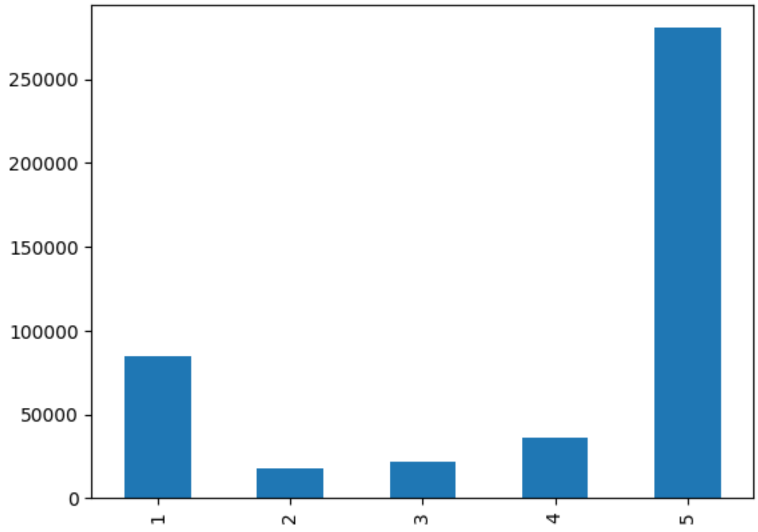


Figure 3.5 - Grab Reviews Ratings Distribution

- review_likes: Also in Grab’s case the great majority of the reviews has 0 likes, 392950 reviews or 0,89% of the data to be more specific, with the most liked review having 2316 likes.
- author_app_version: 479 is the amount of unique strings for this feature, with the most frequent showing up 6406 times with the value “5.55.1”. We also noticed 68 260 null rows, representing 15,5% of the data from Grab. The distribution is again very similar to the one observed for Uber and can be seen in Figure 3.6:

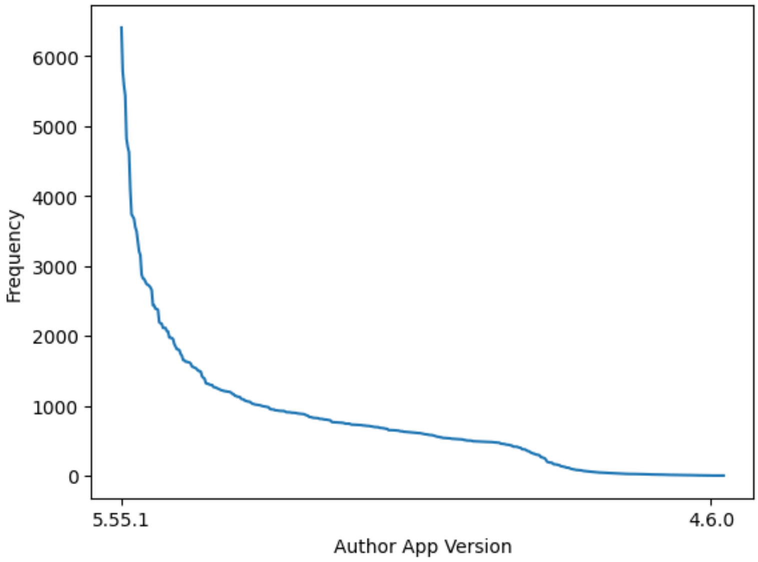


Figure 3.6 - Grab Author App Version Distribution

- review_datetime_utc: For Grab, the day with the most amount of reviews was the 18th of May 2019 with 1413 reviews, followed by the 17th, 19th, 20th and 21st of May from the same year. On average there are almost 119 new posted reviews per day, with the median above 96. The collected reviews from Grab started 3 years later than Uber's, on the 31st of May 2013, and include 10 years of reviews, until the 9th of November 2023. The distribution of reviews through time is the following presented in Figure 3.7:

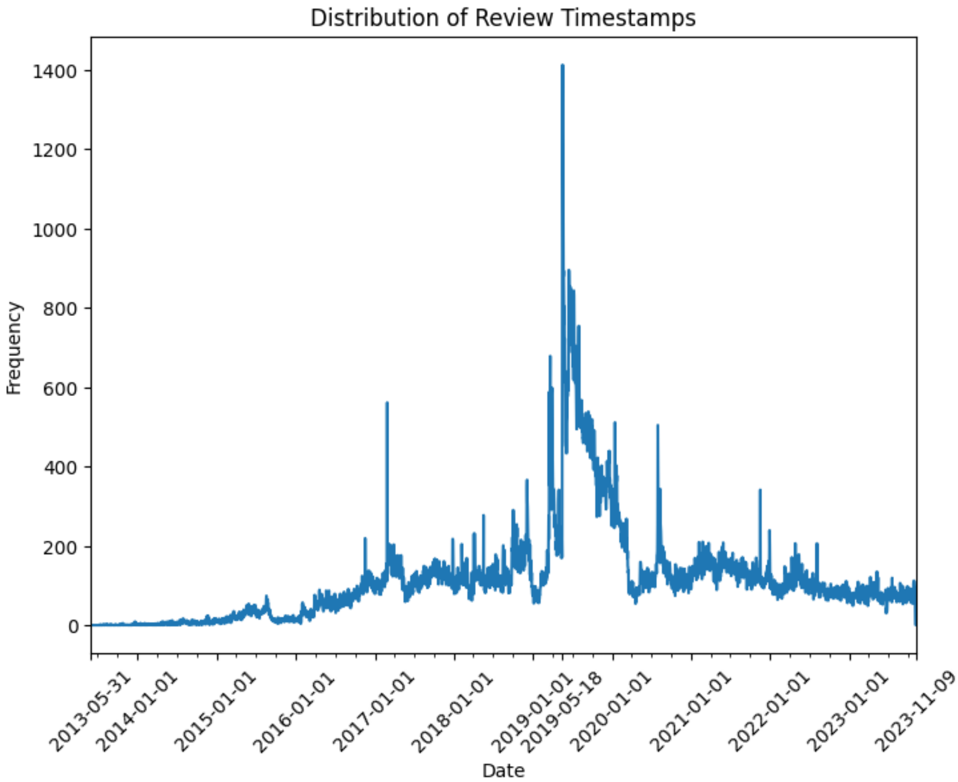


Figure 3.7 - Grab Reviews Distribution by Date

After analyzing the data referent to the user generated reviews, we proceeded with exploring the Historical Stock Data.

Data regarding stock prices did not have any empty entries in any of the cases, but it is important to notice that Date is not continuous, since the stock market is not open during the weekends, which means that dates referent to the non-business days are skipped.

Also, the amount of data is not equal for both companies, due to their IPO dates being different: Uber Stock Data starts on the 10th of May 2019 while Grab's counterpart starts on the 2nd of December 2021, but we collected Stock Price Data for both companies until the same day, the 4th of September 2024.

In terms of Descriptive Statistics, the following Table 3.4 and Table 3.5 describe how data for Uber and Grab behave, respectively:

Table 3.4 - Uber Stock Data Descriptive Statistics

	Open	**High**	**Low**	**Close**	**Adj Close**	**Volume**
Mean	42.42	43.21	41.59	42.41	42.41	2.45e+07
Standard Deviation	14.22	14.32	14.19	14.19	14.19	1.80e+07
Minimum	15.96	17.80	13.71	14.82	14.82	1.52e+06
Maximum	81.94	82.14	80.79	81.39	81.39	3.64e+08

Table 3.5 - Grab Stock Data Descriptive Statistics

	Open	**High**	**Low**	**Close**	**Adj Close**	**Volume**
Mean	3.51	3.60	3.41	3.50	3.50	2.08e+07
Standard Deviation	1.09	1.11	0.98	1.04	1.04	1.48e+07
Minimum	2.33	2.33	2.19	2.31	2.31	2.45e+06
Maximum	13.055	13.29	8.55	9.00	9.00	1.79e+08

It is noticeable that there is a significant difference between the values referent to Uber and Grab, with Uber's being relatively higher, and we can see that any of the features' minimum value for Uber is higher than any of the features' maximum value for Grab. There is also no difference, for any of these companies, between the statistics of the Close and Adj Close features.

When checking the distributions for all the features excluding Volume we got the following graphs present in Figure 3.8 (Uber data) and Figure 3.9 (Grab data):

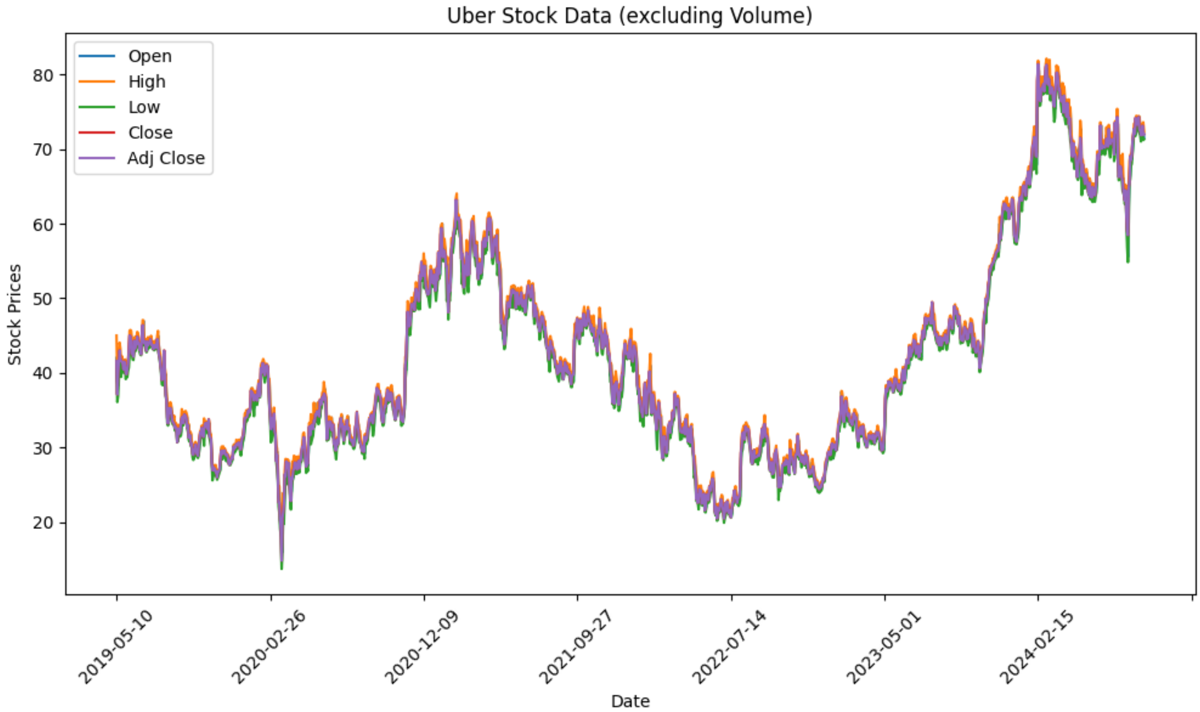


Figure 3.8 - Uber Stock Data Distribution excluding Volume

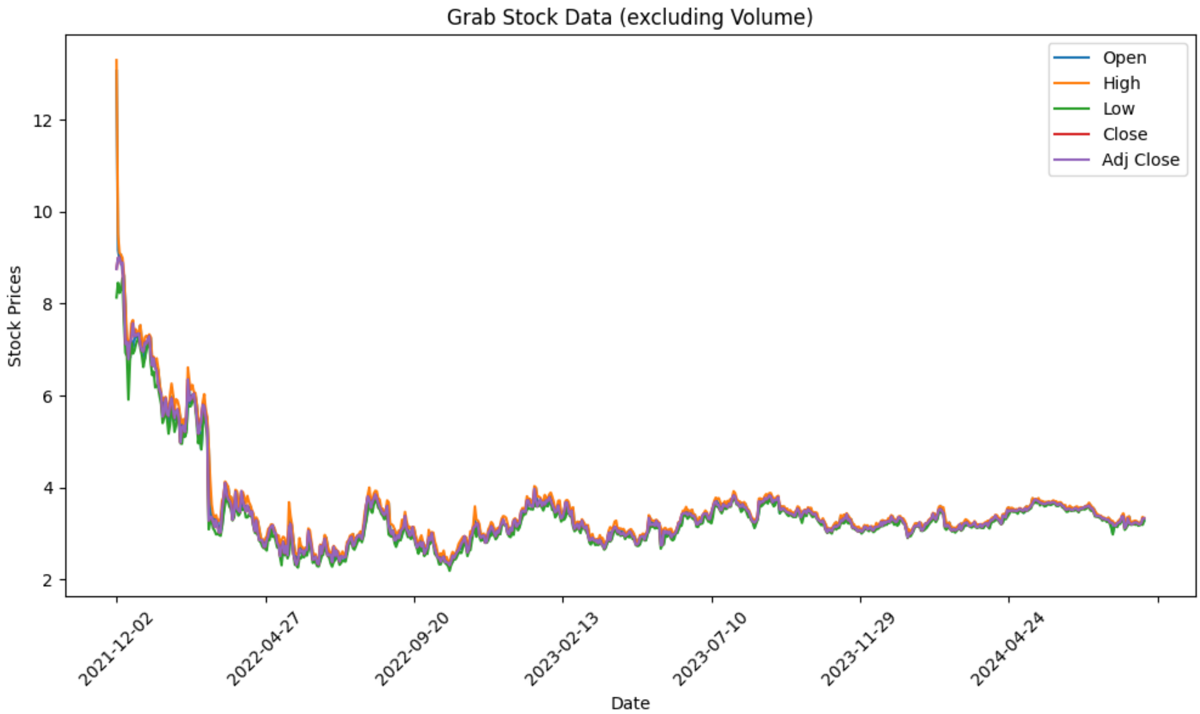


Figure 3.9 - Grab Stock Data Distribution excluding Volume

These graphs confirm the small difference in values we saw in Table 3.2.1. and Table 3.2.2. between features, as they follow the same distribution. Another observation is made possible by these graphs, as we notice that, while Uber's Stock has been following a positive trend since it became public, Grab's stock followed the opposite trend, having stabilized since the beginning of 2023.

Regarding volume, we observe a very similar distribution between both firms (Figure 3.10 and Figure 3.11), in exception for the highest traded Volume of Uber, which stands out as a very big outlier on the 15th of December of 2023:

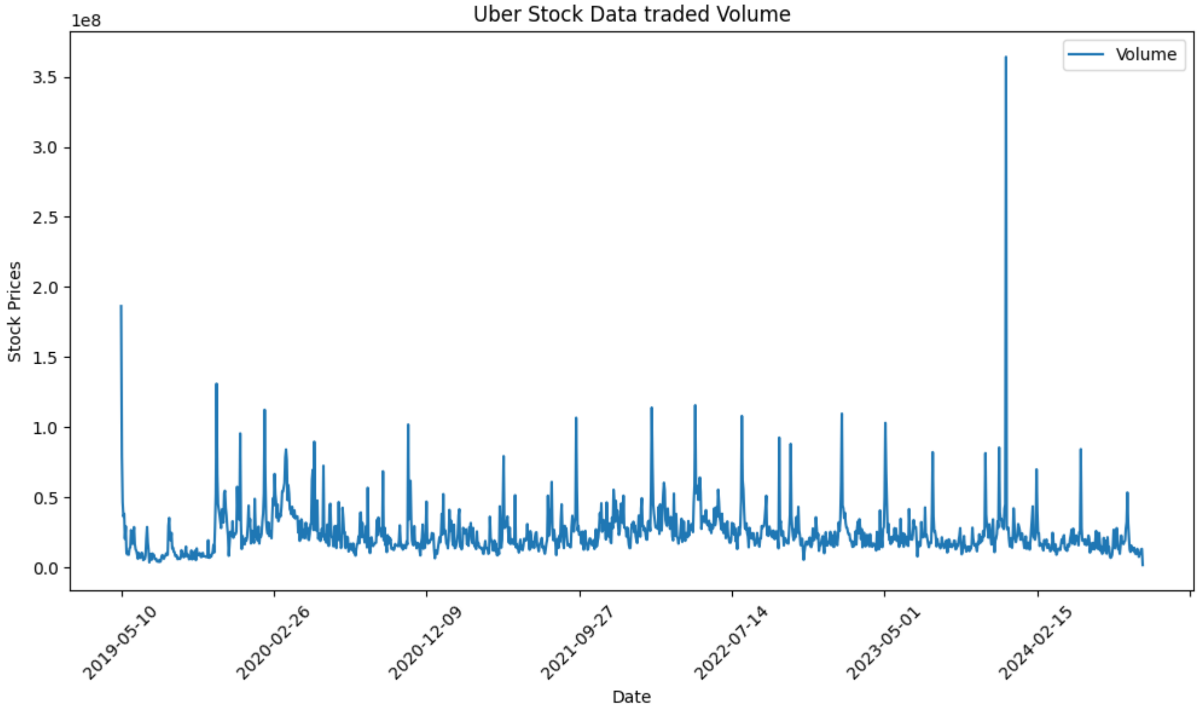


Figure 3.10 - Uber Stock Data Trading Volume Distribution

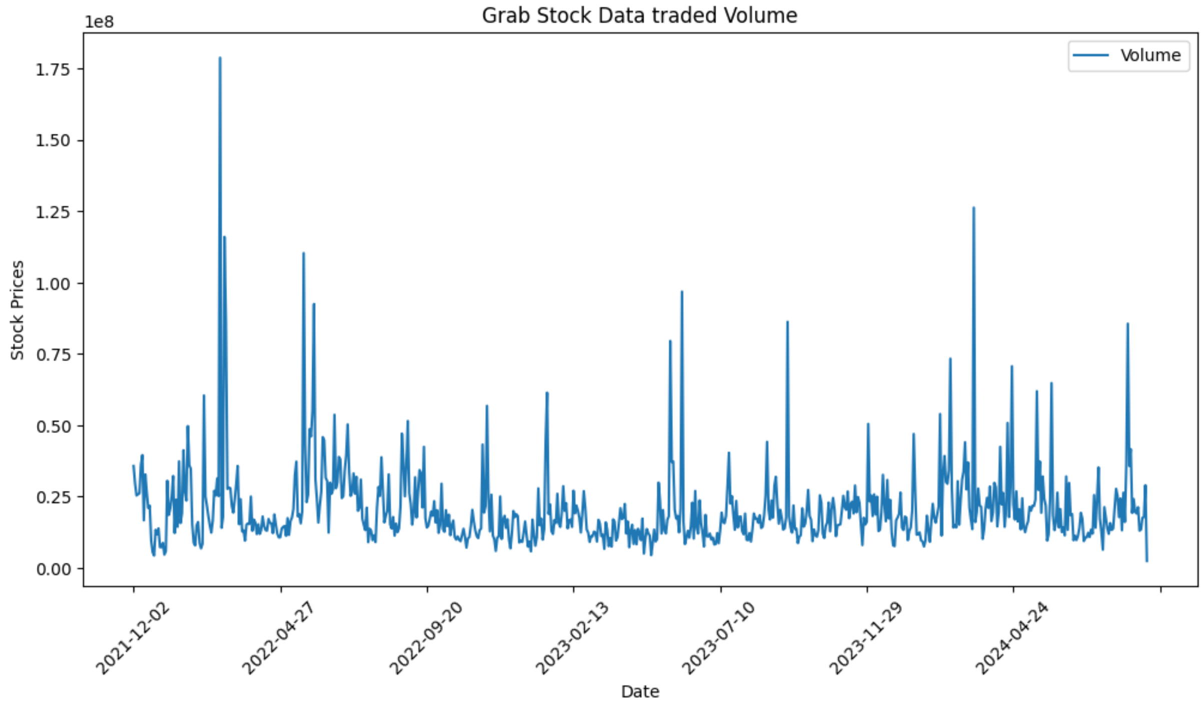


Figure 3.11 - Grab Stock Data Trading Volume Distribution

3.3. DATA PREPARATION

This phase relates to the steps needed to construct the final dataset, from the raw data. It includes cleaning data, labeling it, and massaging it so that it is compatible with the algorithms that will be used. (Tounsi et al., 2020)

The data regarding user-generated content was preprocessed in a pipeline comprising of different steps: Data Cleaning, Feature Engineering (Tokenization, Sentiment Analysis, Topic Modelling), Data Aggregation and Target Selection.

3.3.1. Data Cleaning

Different techniques were employed to clean the data:

1. Renaming Columns: To keep consistency column "pseudo_author_id" from Grab's reviews dataset was renamed to "author_id" to match the naming from Uber's dataset, but the column "review_datetime_utc" from Uber's dataset was renamed to match the column "review_timestamp" from Grab's dataset.
2. Removal of rows with empty reviews: As seen before, the number of reviews with empty text was minimal, totalling 629 in both datasets, so we decided to remove them, as it represented an minimal percentage of our total data.

3. Removal of html tags, special characters and stop words: Very typical in textual data such as ours, this type of character clutters the data without much additional relevant content, so its removal will facilitate the processing of data while also helping improve the accuracy of predictive models (Awad et al., 2023; Darapaneni et al., 2022; Koukaras et al., 2022; Xiao & Ihnaini, 2023).
4. Lowercasing: This step is important as it helps standardizing the text and ensuring two words with the same content, such as "Good" and" good", like we seen during exploration, are not treated differently. This technique has been proven to help improving the performance of predictive models (Ho & Huang, 2021; Palomino & Aider, 2022).
5. Lemmatization: Similarly to lowercasing, this technique helps normalizing the text and making it easier for it to be processed, helping improving accuracy of predictive models (Darapaneni et al., 2022).
6. Filling in missing values: After solving the issue with the empty reviews, other feature that presented missing values was “author_app_version”, but on the contrary to “review_text”, the total of nulls regarding this feature represents a very significant amount of our total data, so removing those rows was not the solution. Instead, to solve this, empty rows were filled with the mode of “author_app_version” of all the reviews posted on the same day by assuming that most users have their app updated to the latest version, and consequently are using the same app version. Keeping this information might be valuable to capture some trends regarding poor optimized app versions.
7. Feature removal: Features that are used as unique identifiers do not include any content related information that could be helpful when making predictions. As we saw before “author_name” had, in great majority, “A Google user” as input, and the rest were masked/filled with asterisks, so it was removed from both datasets, as well as author_id which also solved the issue of the final 72 null values we had in Grab’s dataset. Similarly to the previously mentioned features, review_id was also removed from Uber’s reviews dataset.

3.3.2. Tokenization

We tokenized the review texts as it is one of the most important processes when dealing with Google Play Store data from Ridesharing companies (Mahmud et al., 2022) since it helped transforming the unstructured text into a structured format more suitable for analysis.

3.3.3. Sentiment Analysis

Mohan et al (2019) predicted Stock Price of S&P500 companies using News Sentiment Analysis and found that there is a strong relationship between textual information and stock price direction. To do that, different models were tested, but the one that consistently got the best results was the one using text sentiment as input, even better than the one using the text itself as input.

To get the sentiment analysis from ridesharing companies' mobile applications reviews Mahmud et al (2022) tried different models, but the one that achieved the best performance was the pre-trained model DistilBERT. Similarly to this author, the model was applied directly to the review text, so a copy of the original content of this feature was kept before the previous steps of Data Cleaning and Tokenization. The model outputs a number from 1 to 5 that represents the sentiment from negative to positive in an ascending way. We extracted 3 sentiments: Positive (reviews with output of 4 and 5), Negative (1 and 2 as output) and Neutral (3 as output). The distribution of these sentiments is very similar for both firms (Figure 3.12 for Uber's reviews, Figure 3.13 for Grab's reviews), with the majority of the reviews being positive, followed by negative and finally neutral:

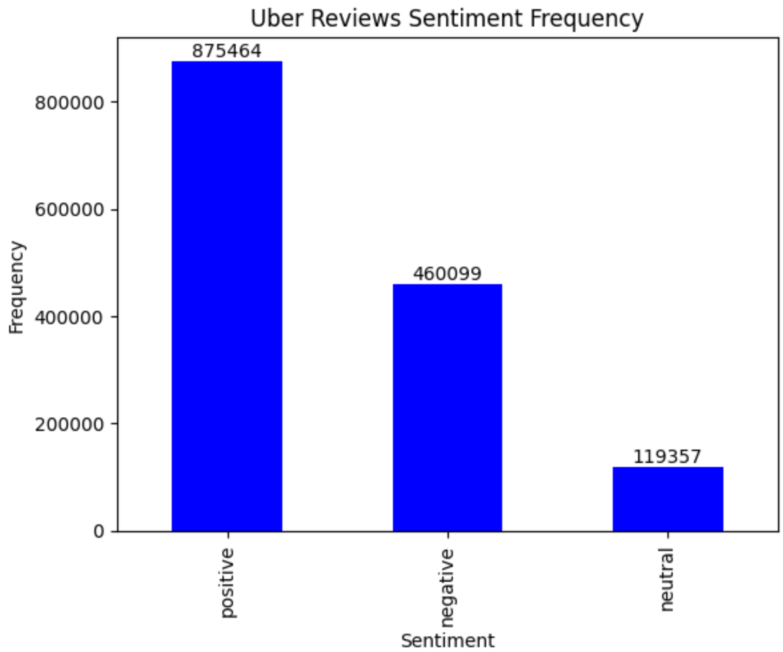


Figure 3.12 - Uber Reviews Sentiment Frequency

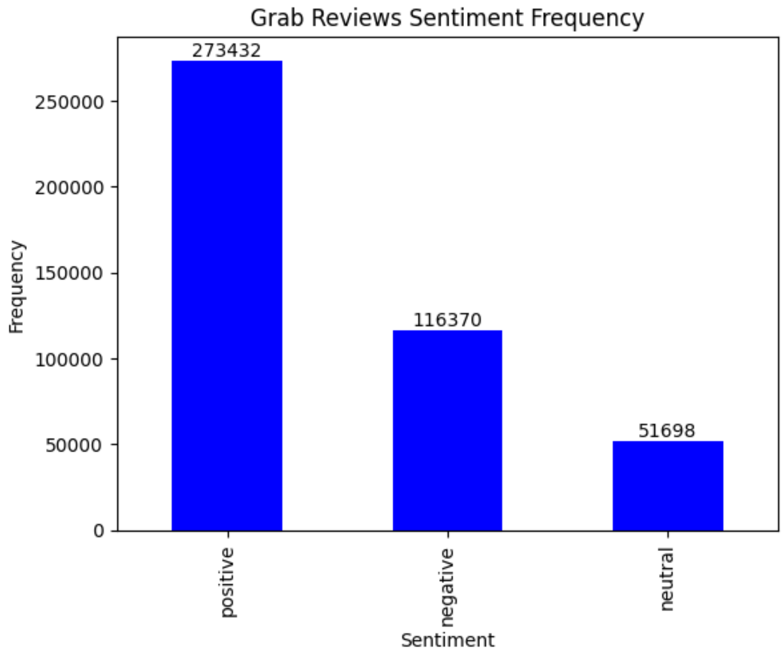


Figure 3.13 - Grab Reviews Sentiment Frequency

3.3.4. Topic Modelling

In this step, the customer reviews are classified into meaningful categories/topics so that the models, besides having access to the sentiment transmitted by the text, can also leverage patterns in terms of what the reviews are related to.

To do this BERTopic was used, which is a topic modelling technique that leverages hugging face transformers and cTF-IDF to create dense clusters allowing for easily interpretable topics (Grootendorst, 2022). This can be used to apply a technique called Zero-shot Topic Modelling, which allows to find topics in large amounts of documents that were predefined (BERTopic, n.d.).

Although with this technique it is possible for topics to be detected based on the review textual content, it is also possible to define candidate labels. With that in mind, 14 categories will be proposed to promote data standardization, described in Table 3.6:

Table 3.6 - Reviews proposed categories

Category	Description
Pricing and Fees	Discussions about fare prices, surge pricing, and overall cost of rides.
Service Quality	Comments on driver behavior, professionalism, and overall service experience.

App Functionality	Feedback related to app performance, bugs, crashes, and user interface issues.
Ride Experience	Reviews about the comfort, safety, and overall quality of the ride itself.
Customer Support	Experiences with customer service, response times, and resolution of issues.
Driver Availability	Availability of drivers, wait times, and ease of getting a ride.
Payment Issues	Problems related to payment methods, billing errors, and refunds.
Promotions and Discounts	Comments on the availability and usability of promotional offers and discounts.
Navigation and Routing	Feedback on the accuracy of navigation, route choices, and efficiency.
Safety and Security	Concerns or praise regarding the safety measures in place for both riders and drivers.
Account Management	Issues or feedback related to account setup, login problems, and profile management.
Rating and Review System	User opinions on the rating and review system for both drivers and passengers.
Vehicle Quality	Comments on the condition, cleanliness, and type of vehicles used for rides.
Accessibility	Feedback related to the accessibility of the app and rides for people with disabilities.
Miscellaneous	Any review that does not qualify into the other topics

The results were significantly unbalanced, due to the relatively low number of reviews classified under “Customer Support”, “Navigation and Routing”, “Promotions and Discounts”, “Account Management” and “Miscellaneous”, so we decided to aggregate all of the reviews that were assigned with these topics to a new topic “Other”. With these adjustments, the resulting distributions can be seen in Figure 3.14 for Uber and Figure 3.15 for Grab:

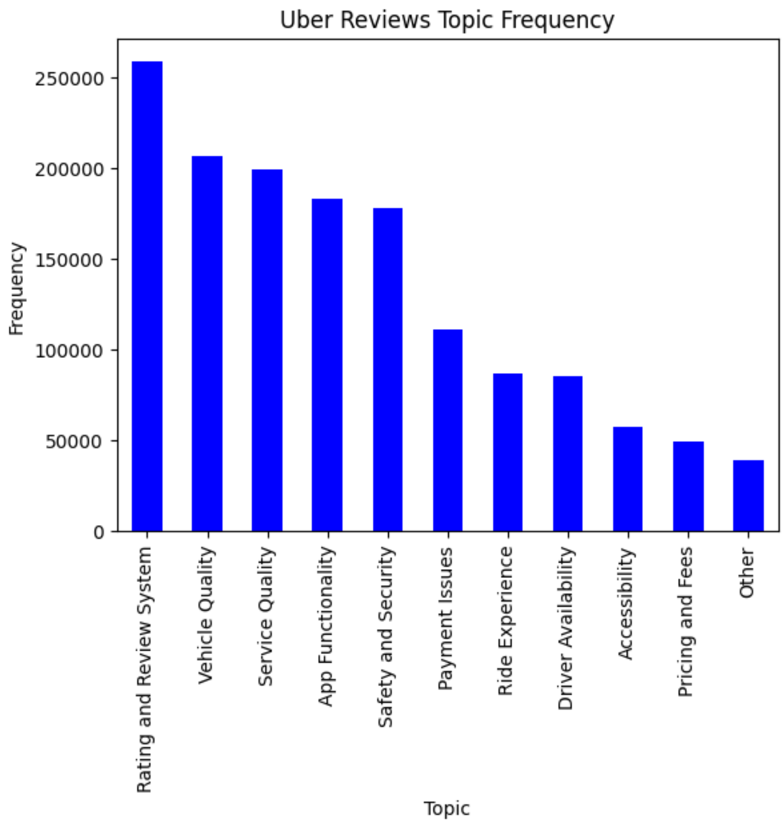


Figure 3.14 - Uber Reviews Topic Frequency

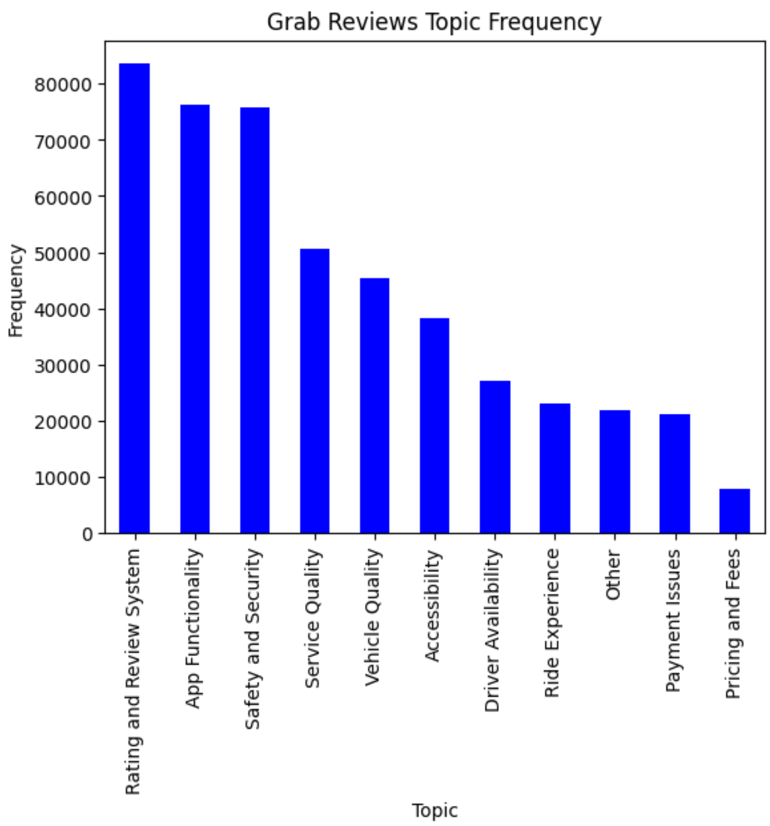


Figure 3.15 - Grab Reviews Topic Frequency

It is interesting to notice that the distribution of topics changes from one firm to another, but in both "Rating and Review System" were classified as the major reviews topic, followed by "Vehicle Quality" and "Service Quality" for Uber, and "App Functionality" and "Safety and Security" for Grab.

3.3.5. Data Aggregation and Feature Set Creation

Since data at this point is still scattered around, it is important to join the different datasets and aggregate data to facilitate the analysis, especially while creating new features with the intention of capturing trends, such as the ones we will describe in this section.

Data was aggregated in two different ways:

- Weekly aggregation approach: Data will be aggregated weekly and per firm, based on what Wu et al (2019) did to deal with 18 million consumer reviews from JD.com. Analysing data weekly is also helpful for identifying trends based on the entirety of trading periods, since the stock market is only open during week days. In this case, every review will be aggregated with any review made during the same week, by taking day of that week's Monday as reference. This aggregation approach lead to a final dataset of 315 rows of data.

- Daily aggregation approach: Due to the reduced amount of rows originated by aggregating data weekly, we decided to also test the approach of Mohan et al (2019) when predicting stock price using news sentiment analysis, which was to aggregate data by day. This approach should retain more information for the models to train on, as our final dataset with data aggregated daily was around 5 times bigger than the one in the previous approach, having 1601 rows of data.

Starting with the Weekly Aggregation Approach, the final dataset comprises of multiple features, with the following features described in Table 3.7 being used as the aggregation feautes:

Table 3.7 - Weekly Aggregation Features

Feature	Data Type	Description
week_date	Object (Timestamp)	A string with the date of the monday of the week
firm	Integer	A binary variable that distinguishes Uber (0) from Grab (1) reviews

We started by creating features that would be used as a reference for the aggregation process. The first feature we created was “week_date”, which takes the day of the Monday of the review’s week. Additionally, when joining Uber’s and Grab’s datasets, we tagged the rows with a new feature “firm”, that purely identifies the dataset that row came from by taking the value “0” if it came from Uber’s dataset or “1” if it came from Grab’s.

In Table 3.8 are described the features resulting of aggregating Stock Price data in such a way that allowed us to keep absolute as well as average information about that week:

Table 3.8 - Weekly Aggregated Stock Price Features

Feature	Data Type	Description
week_open	Float	That week's open price (Monday)
week_high	Float	That week's highest traded price
week_low	Float	That week's lowest traded price
week_close	Float	That week's closing price (Friday)
week_adj_close	Float	That week's adjusted closing price (Friday)
week_volume	Float	That week's total traded volume
week_avg_open	Float	That week's average opening price
week_avg_high	Float	That week's average highest traded price
week_avg_low	Float	That week's average lowest traded price
week_avg_close	Float	That week's average closing price
week_avg_adj_close	Float	That week's average adjusted closing price
week_avg_volume	Float	That week's average traded volume

Then we moved to the Reviews datasets and the next set of features we created were the “Review Features” which, as the name suggests, are features that directly relate to the user-generated reviews.

Textual aggregation posed a challenge that we solved by creating 3 new features: “most_common_1”, ”most_common_2” and “most_common_3”, which respectively contain the 3 most common words (by frequency) used in the textual reviews made during that week, after removing stop words. This is expected to help with the identification of trends as it is expected that major events/problems in a specific week will be the focus of feedback in that week's reviews.

Additionally, we aggregated the number of likes per review by taking the average of that week’s likes in a feature named “avg_review_likes”, but also kept a track of how many reviews were made that week in the feature “review_nr”. The average of review’s likes was included as a proxy of how many people relate to the textual content of the review’s made during that week, while the number of reviews was thought in order to capture trends of high and low reviews made.

Finally, we also created a feature “author_app_version_mode” which takes the mode of the app versions used when posting the reviews for that week. This feature should help the models to capture trends about what app versions are targeted by reviews positively and negatively.

These Review Features are described in the Table 3.9:

Table 3.9 - Review Features

Feature	Data Type	Description
most_common_1	Object	The most used word in that week's reviews
most_common_2	Object	The second most used word in that week's reviews
most_common_3	Object	The third most used word in that week's reviews
avg_review_likes	Float	The average number of likes received by that week's reviews
review_nr	Integer	The total number or reviews made that week
author_app_version_mode	Object	The most common app version used by users posting reviews during that week

In sequence, to capture the rating the users gave to the app, we created the set of features “Star Features”, described in Table 3.10. These track the percentage of the feature “review_rating” that was equal to each of the possible values a user can attribute to an app (1,2,3,4 and 5) and are expected to capture trends in customer satisfaction and dissatisfaction throughout time.

Table 3.10 - Star Features

Feature	Data Type	Description
star_1_ratio	Float	Percentage of 1-star reviews in that week
star_2_ratio	Float	Percentage of 2-star reviews in that week
star_3_ratio	Float	Percentage of 3-star reviews in that week
star_4_ratio	Float	Percentage of 4-star reviews in that week
star_5_ratio	Float	Percentage of 5-star reviews in that week

“Sentiment features”, similarly to the “Star Features”, are expected to capture trends in customer satisfaction and dissatisfaction throughout time by representing the percentage of each sentiment present in that week's reviews. These were aggregated using the features that originated from our sentiment analysis process and are described in Table 3.11:

Table 3.11 - Sentiment Features

Feature	Data Type	Description
positive_tendency_ratio	Float	Percentage of reviews with positive sentiment in that week
neutral_tendency_ratio	Float	Percentage of reviews with neutral sentiment in that week
negative_tendency_ratio	Float	Percentage of reviews with negative sentiment in that week

Finally, the “Category Features”, described in Table 3.12, are expected to capture trends in the topics addressed by customers in the reviews throughout time and were calculated using the features that resulted from our Topic Modelling process:

Table 3.12 - Category Features

Feature	Data Type	Description
app_functionality_ratio	Float	Percentage of reviews under “App Functionality”
pricing_and_fees_ratio	Float	Percentage of reviews under “Pricing and Fees”
service_quality_ratio	Float	Percentage of reviews under “Service Quality”
ride_experience_ratio	Float	Percentage of reviews under “Ride Experience”
driver_availability_ratio	Float	Percentage of reviews under “Driver Availability”
payment_issues_ratio	Float	Percentage of reviews under “Payment Issues”
safety_and_security_ratio	Float	Percentage of reviews under “Safety and Security”
rating_and_review_system_ratio	Float	Percentage of reviews under “Rating and Review System”
vehicle_quality_ratio	Float	Percentage of reviews under “Vehicle Quality”
acessibility_ratio	Float	Percentage of reviews under “Acessibility”
other_ratio	Float	Percentage of reviews under “Other”

Regarding the Daily Aggregation Approach, the Aggregation Features caracterized in Table 3.13 are very similar, being the only difference the "week_date" feature that was replaced with "Date" since now we are aggregating data by day.

Table 3.13 - Daily Aggregation Features

Feature	Data Type	Description
Date	Object (Timestamp)	A string with the date of the review
firm	Integer	A binary variable that distinguishes Uber (0) from Grab (1) reviews

Additionally, the features sets that resulted from the Reviews Datasets are the same (adapted to be aggregated daily), also following the same nomenclature. However, the "Stock Price Features" are a bit different since it would be useless to calculate the average for the stock price features, as they will be the same for every review recorded during that day. To solve that, we kept the original daily features, but also calculated the absolute values relative to the week, in an effort to capture some trends in low scale as seen in Table 3.14:

Table 3.14 - Daily Aggregated Stock Price Features

Feature	Data Type	Description
Open	Float	That day's open trade price
High	Float	That day's highest traded price
Low	Float	That day's lowest traded price
Close	Float	That day's closing price (Friday)
Adj Close	Float	That day's adjusted closing price (Friday)
Volume	Float	That day's total traded volume
week_open	Float	That week's open price (Monday)
week_high	Float	That week's highest traded price
week_low	Float	That week's lowest traded price
week_close	Float	That week's closing price (Friday)
week_adj_close	Float	That week's adjusted closing price (Friday)
week_volume	Float	That week's total traded volume

After this aggregation process we considered our data to be ready to be fully joined, so we started by merging Uber's Reviews Dataset with Uber's Stock Price Dataset on the "week_date" for the weekly aggregated data and on "Date" for the second data aggregation approach and repeated the process for Grab's data. For both firms, data was merged using a Left Join on the review data, which lead to many rows with null entries on the columns regarding stock price data. We proceded with removing all the rows in which that was the case, which gave us two final datasets with 335 rows for the Weekly Aggregated data and 1625 rows for the Daily Aggregated data.

3.3.6. Targets Selection

After aggregating Reviews data and Stock Price, the next step consisted of selecting the target feature we are trying to predict. To achieve that goal, we followed a similar approach to Wu et al (2019) when using online reviews to predict long-term returns of individual stock, which consisted of calculating the correlation between each one of his features and his target for different points in time. Since it was estimated that online review features may take 1 to 12 weeks to take any effect, we tested the correlation between each feature and the target for the week correspondent to our reviews as well as the twelve following weeks.

The target feature itself is Week Stock Price Returns, which was calculated following the formula shown in Figure 3.16 for both datasets (Weekly Aggregated and Daily Aggregated):

$$Week\ Stock\ Price\ Return = \frac{(Week\ Closing\ Price) - (Week\ Opening\ Price)}{Week\ Opening\ Price} \times 100$$

Figure 3-16 - Weekly Stock Price Return Formula

To keep consistency, we removed any rows from the datasets that had a null value for any of the features representing the week stock price return before computing the correlations. In terms of descriptive statistics, Table 3.15 shows our target features for Uber data while Table 3.16 shows the same regarding Grab data.

Table 3.15- Uber's Week Stock Price Returns Descriptive Statistics

Feature	Mean	Standard Deviation	Minimum	Maximum
weekly_stock_price_return	0.003984	0.073832	- 0.207	0.366
weekly_stock_price_return_1	0.004324	0.073943	- 0.207	0.366
weekly_stock_price_return_2	0.004002	0.073767	- 0.207	0.366
weekly_stock_price_return_3	0.003947	0.073769	- 0.207	0.366
weekly_stock_price_return_4	0.003813	0.073860	- 0.207	0.366
weekly_stock_price_return_5	0.003627	0.073710	- 0.207	0.366
weekly_stock_price_return_6	0.003727	0.073696	-0.207	0.366
weekly_stock_price_return_7	0.003489	0.073741	-0.207	0.366

weekly_stock_price_return_8	0.003166	0.073677	- 0.207	0.366
weekly_stock_price_return_9	0.003357	0.073533	- 0.207	0.366
weekly_stock_price_return_10	0.003923	0.074067	- 0.207	0.366
weekly_stock_price_return_11	0.004363	0.074200	- 0.207	0.366
weekly_stock_price_return_12	0.004517	0.074288	- 0.207	0.366

Table 3.16 - Grab's Week Stock Price Returns Descriptive Statistics

Feature	Mean	Standard Deviation	Minimum	Maximum
weekly_stock_price_return	-0.006493	0.107719	-0.426	0.3399
weekly_stock_price_return_1	-0.001456	0.103832	-0.426	0.3399
weekly_stock_price_return_2	0.001237	0.101627	-0.426	0.3399
weekly_stock_price_return_3	-0.000045	0.101100	-0.426	0.3399
weekly_stock_price_return_4	-0.000853	0.101233	-0.426	0.3399
weekly_stock_price_return_5	-0.000652	0.101149	-0.426	0.3399
weekly_stock_price_return_6	0.000203	0.101096	-0.426	0.3399
weekly_stock_price_return_7	0.001269	0.100606	-0.426	0.3399
weekly_stock_price_return_8	0.001602	0.100503	-0.426	0.3399
weekly_stock_price_return_9	0.000930	0.100565	-0.426	0.3399
weekly_stock_price_return_10	0.000591	0.100707	-0.426	0.3399
weekly_stock_price_return_11	-0.000210	0.100235	-0.426	0.3399
weekly_stock_price_return_12	0.000465	0.100089	-0.426	0.3399

3.4. MODELING

This phase includes the choice, parameterization and the test of different algorithms (Tounsi et al., 2020), which was based on previous works that we reviewed:

Mohan et al (2019) predicted the stock price using news sentiment analysis by employing versions of Recurrent Neural Networks with Long Short-term Memory with different combinations of inputs, having achieved good results when compared with other models such as Arima or Facebook Prophet. On the other hand, Wu et al (2019) achieved the best results in predicting individual stocks from online reviews when using Boosting algorithms, with the best being XGBoost.

In order to dynamically perform feature selection and choose the best parameters, both models were fitted with every set of features for 50 trials in a random search approach, each trial with a different set of parameters using Optuna, an automatic hyperparameter optimization software framework, particularly designed for machine learning.

Feature selection was made in sets, not by individual features, being the sets considered the following presented in Table 3.17:

Table 3.17 - Combination of Features for Modeling

Feature set	**Description**
stock_prices	All the "Stock Features" + "Firm"
review_features	All the "Stock Features" + "Firm" + All the "Review Features"
star_features	All the "Stock Features" + "Firm" + All the "Star Features"
sentiment_features	All the "Stock Features" + "Firm" + All the "Sentiment Features"
category_features	All the "Stock Features" + "Firm" + All the "Category Features"
all_features	All the features combined

Additionally, to help both models capturing sequential dependencies in data, we created a function that would create sequencies of data, meaning it would append to each row of data the data correspondent to the *t* previous data periods. This meant that, if the parameter t of this function was equal to 3, the row equivalent to November 15th 2023 would also include data from November 14th, 13th and 12th . This also worked for weekly aggregated data, as the week of 13th of November 2023 would also be inputted with data from the week of the 6th of November, 30th of October and 23rd of October 2023. Both models were trained with different values for the "time_step" parameter: 1, 3, 5 and 7.

Structurally-wise, the RNN-LSTM created was rather simple, valuing adaptability and control over complexity in an effort to maximize the ability of the model in generalizing on new data, as finding the best structure possible for the RNN was not the focus of this research. Following the works of Qiu et al (2020) forecasting stock prices with LSTM networks based on attention and Zaheer et al (2023) that performed multi parameter forecasting for Stock Time Series Data using LSTM and Deep Learning Models the following structure were created:

1. LSTM Layer with tunable amount of units, ranging between 50 and 200, allowing the model to dynamically adjusting its capacity to retain sequential dependencies while avoiding overfitting. This choice aligns with Zaheer et al (2023) and Qiu et al (2020) who showed that scalable capacity is essential for financial forecasting models to dynamically adapt to data complexities.

2. Second LSTM Layer: To further help capture temporal patterns in the data, a similar LSTM layer was added as this multi-layer setup has been shown to enhance model robustness and prediction accuracy in financial time series, by ensuring that the model captures both short-term trends as well as long-term data structures (Qiu et al., 2020).
3. L2 Regularization: Regularization parameters were applied to the LSTM layers to help reduce overfitting, a major issue in this type of problems, with a value between 1e-5 and 1e-2. This selected range was based on the work from Qiu et al (2020), which showed that L2 enhances model robustness without compromising adaptability.
4. Tunable Dropout Layer: As overfitting was a relevant issue, a Dropout Layer was included in order to prevent it further. As the name suggests, this type of layer drops part of the units during training to ensure the model does not rely too much on a specific part of the network, promoting generalization. In our case, the percentage of units dropped is between 0.2 and 0.5. This type of layer has shown to be effective in capture volatile trends without being misled by short-term fluctuations, which promotes robustness while handling noisy datasets (Zaheer et al., 2023).
5. Output Layer: The final layer of the model is a fully connected Dense Layer with a single Neuron able to output a continuous value, suitable for this regression problem. We used Mean Squared Error (MSE) as the loss function with Mean Absolute Error (MAE) as an additional metric, as the heavy penalty given to larger errors by MSE is very helpful in regression tasks such as financial prediction tasks, helping the model handle with possible relevant fluctuations in stock price returns.
6. Compiler: The choice for the compiler was between the Adam and RMSProp optimizers. The first option is robust for noisy gradients, which is typical in stock market data, while RMSProp usually performs well with sequential data. This choice allows the models to adapt to different convergence behaviors.

Regarding the XGBoost, there are also multiple parameters that needed to be optimized:

- Number of Trees (n_estimators): Controls the number of boosting trees in the model, ranging from 50 to 500 trees, to balance out complexity with performance since, although a large number of estimators allows the model to capture more patterns, it also can lead to increased overfitting.

- Maximum Tree Depth (max_depth): Tuned between 3 and 10, optimizing the maximum depth of each tree also allows us to control the complexity of the model and prevent overfitting. Deeper trees capture more detailed patterns, which can be helpful for short-term price predictions, however might lead to overfitting.
- Learning Rate: This parameter was optimized between 0.01 and 0.2 following a log-uniform distribution and allows the model to balance between speed and accuracy of convergence. Usually lower learning rates result in more accurate models, but require more boosting rounds, which decreases speed.
- Subsample and Column Subsampling: With a similar purpose of the Dropout Layer in the LSTM model, these parameters control the proportion of data and features used to build each tree, respectively, which helps reducing overfitting by introducing variability and enhancing the model's generalization ability. The values for these features range between 0.6 and 1.
- L2 (lambda) and L1 (alpha) Regularization: To help with further reducing overfitting, these terms were optimized between 1e-3 and 1e1. L1 helps reduce feature dependence while L2 minimizes the impact of individual weights, ensuring robustness.
- Objective Function: For the optimization process we used Mean Squared Error as the objective function, similarly to our deep leaning approach since it is suitable for continuous values prediction such as our case.

While the created models were structurally relatively simple, mainly to ensure adaptability, future works can extend the architectures to futher enhance performance with approaches such as:

- RNN-LSTM's Attention Mechanisms: Attention layers could be integrated into the RNN-LSTM structure to help the model focus on the most relevant parts of the input. According to Qiu et al. (2020), attention mechanisms significantly improve forecasting accuracy by weighting the importance of different time steps.
- Transformer-Based Models: Being models that rely entirely on attention mechanisms such as the ones mentioned in the previous point, might help further capture long-term dependencies.
- Hybrid Models: Creating an ensemble framework that could leverage the strengths of both RNN-LSTMs and XGBoost could prove to be very useful.

XGBoost could, for instance, be used for feature ranking, while RNN-LSTM consequently handles the sequential patterns.

3.5. Evaluation

As mentioned before, Mean Squared Error (MSE) and Mean Absolute Error (MAE) were the metrics selected to evaluate the models during the optimization process, but afterwards, for model comparison, we decided to analyze two other metrics: R-squared and Mean Directional Accuracy (MDA).

R-squared was important to include in this analysis because we want to understand how well does the model fit the data and generalizes on it.

Additionally, we are mostly interested in understanding if the models are able to predict accurately if the stock price return will be positive or negative (the direction), similarly to what was done by Mohan et al (2019) while predicting stock price using news sentiment analysis, since that information is often the most important piece for manager or analysts, not the actual value of the return. The Mean Directional Accuracy (MDA) achieves that by comparing the forecast direction of the week stock price returns (positive or negative) to the actual realized direction and will assess if the stock price predictions have the accurate direction.

4. RESULTS AND DISCUSSION

In this research we leveraged Machine Learning models proven to be effective in forecasting stock price, namely Recurrent Neural Networks with Long Short-Term Memory (Mohan et al., 2019) and the Boosting Algorithm XGBoost (Wu et al., 2019), in order to:

- Determine the optimal time lag for the influence of consumer reviews on stock returns
- Assess the predictive performance of consumer reviews on stock price returns
- Identify the specific characteristics of online reviews that contribute most significantly to the prediction performance

4.1.1. Feature to target correlation

As explained in the Methodology, data was aggregated in two ways: daily and weekly. Aggregating data daily means that we grouped reviews by date (since reviews made in the same day will have the same date) and by firm (since we want the models to predict prices for each firm independently instead of having them all mixed together). Aggregating data weekly worked in a similar way, but reviews were grouped by date of the first day of their corresponding week but also by firm. This means that when aggregating weekly, we have access to less data points to train the models, but each data point aggregates more information.

To determine the time difference that maximizes the prediction performance, we calculated the correlation between every feature in our dataset and the weekly stock price return for the same week as the reviews and the following 12 weeks, totaling 13 weeks tested and results can be seen in Table 4.1 and Table 4.2 for Weekly-aggregated features and Daily-aggregated features, respectivelly:

Table 4.1 - Weekly-aggregated features correlation

Feature	Correlation Sum
Weekly Stock Price Return (current week t0)	3.568719
Weekly Stock Price Return 1 (week t1)	3.267808
Weekly Stock Price Return 2 (week t2)	3.572008
Weekly Stock Price Return 3 (week t3)	3.419863
Weekly Stock Price Return 4 (week t4)	3.348661
Weekly Stock Price Return 5 (week t5)	3.126128
Weekly Stock Price Return 6 (week t6)	3.488301
Weekly Stock Price Return 7 (week t7)	3.372382

Weekly Stock Price Return 8 (week t8)	3.039250
Weekly Stock Price Return 9 (week t9)	3.091304
Weekly Stock Price Return 10 (week t10)	2.702698
Weekly Stock Price Return 11 (week t11)	2.817149
Weekly Stock Price Return 12 (week t12)	2.798546

Table 4.2- Daily-aggregated features correlation

Feature	Correlation Sum
Weekly Stock Price Return (current week t0)	4.820676
Weekly Stock Price Return 1 (week t1)	5.369014
Weekly Stock Price Return 2 (week t2)	5.675543
Weekly Stock Price Return 3 (week t3)	5.903063
Weekly Stock Price Return 4 (week t4)	5.930153
Weekly Stock Price Return 5 (week t5)	5.932833
Weekly Stock Price Return 6 (week t6)	6.003929
Weekly Stock Price Return 7 (week t7)	5.960192
Weekly Stock Price Return 8 (week t8)	5.834459
Weekly Stock Price Return 9 (week t9)	5.793848
Weekly Stock Price Return 10 (week t10)	5.636226
Weekly Stock Price Return 11 (week t11)	5.241279
Weekly Stock Price Return 12 (week t12)	4.477274

Analyzing tables 4.1.1. and 4.1.2, we conclude that aggregating the features daily provided an higher correlation, on average, to the target features. The weekly stock price return for the 2nd week was the one that correlated the most with the features when these were aggregated weekly. Aggregating features daily increased the optimal time lag, which sits on the 6th week. This complies relatively better with the result shown by Wu et al (2019) which calculated the optimal time lag to be on the 8th week for predicting long-term returns of individual stocks using Online Reviews.

With our targets chosen as “week_stock_price_returns_6” for the dataset with data aggregated by day and “week_stock_price_return_2” for the one with data aggregated by week we plotted the distributions of these targets for both firms, seen in Figure 4.1 (for daily aggregated data) and Figure 4.2 (for weekly aggregated data):

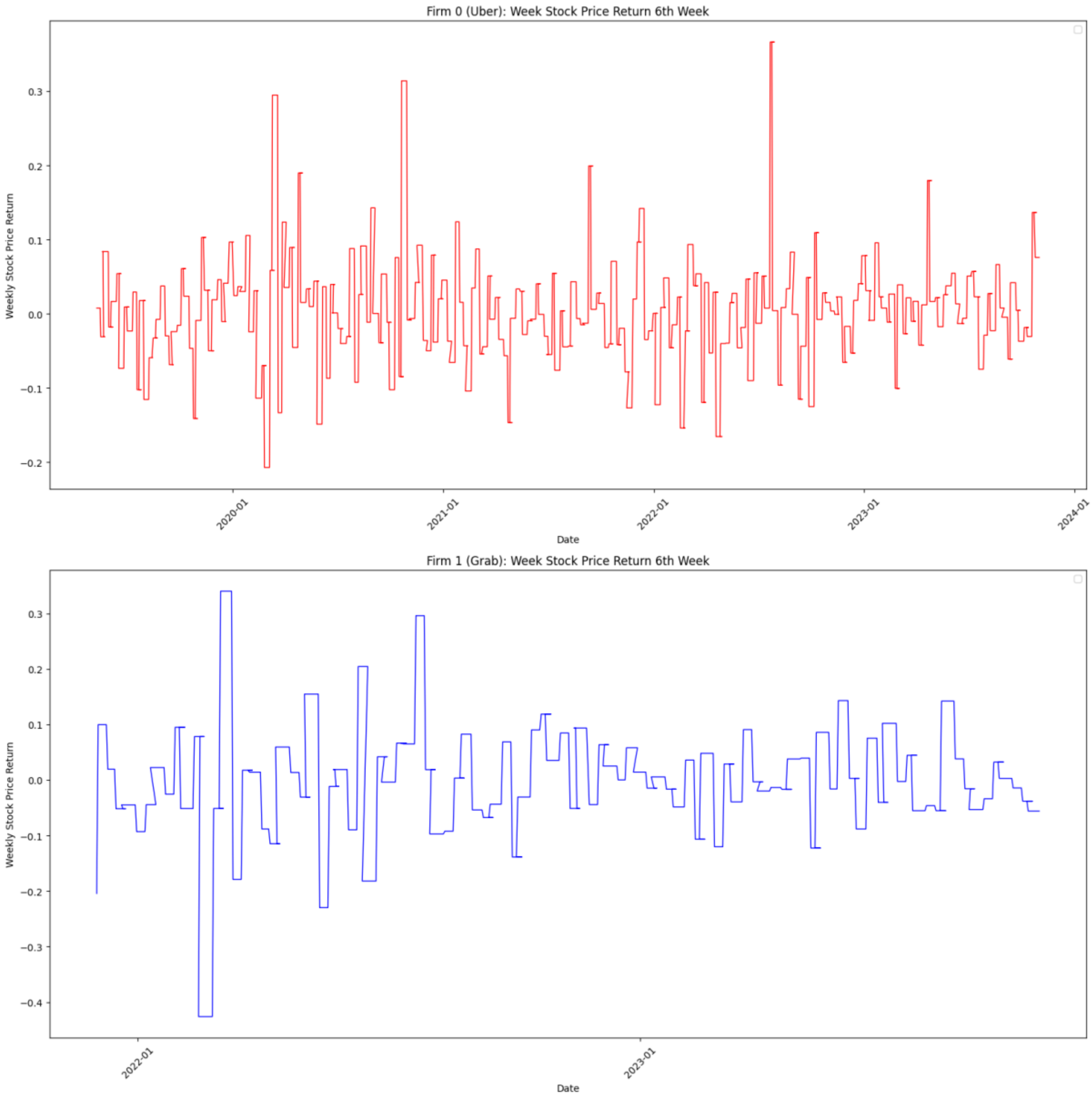
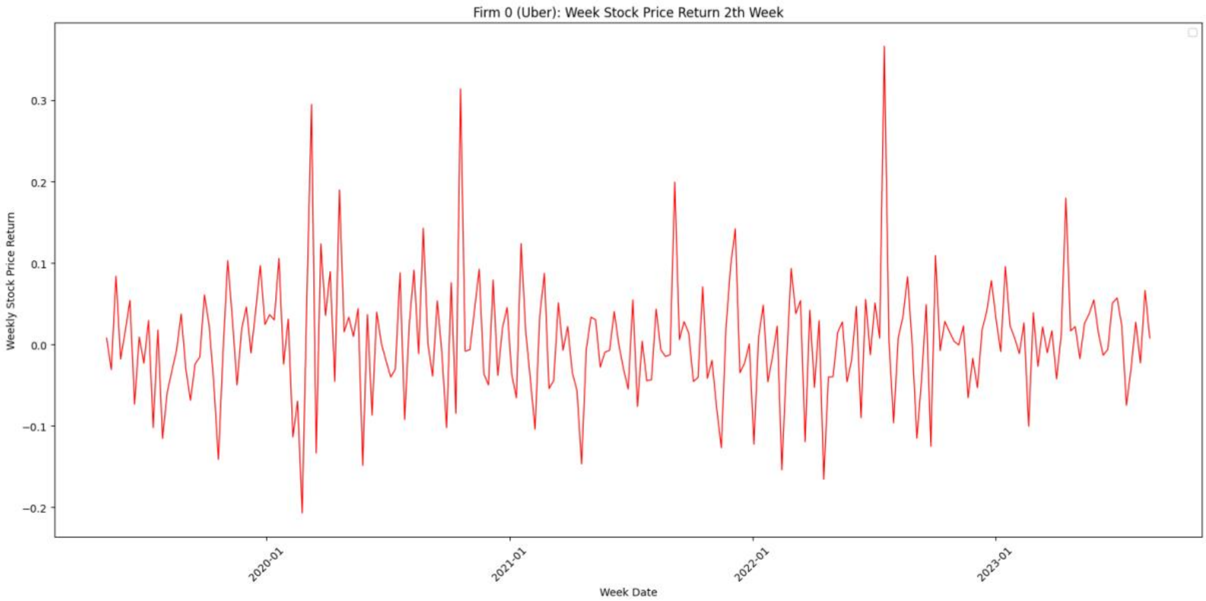


Figure 4.1 - Distribution of target week_stock_price_returns_6 for daily aggregated dataset for both firms



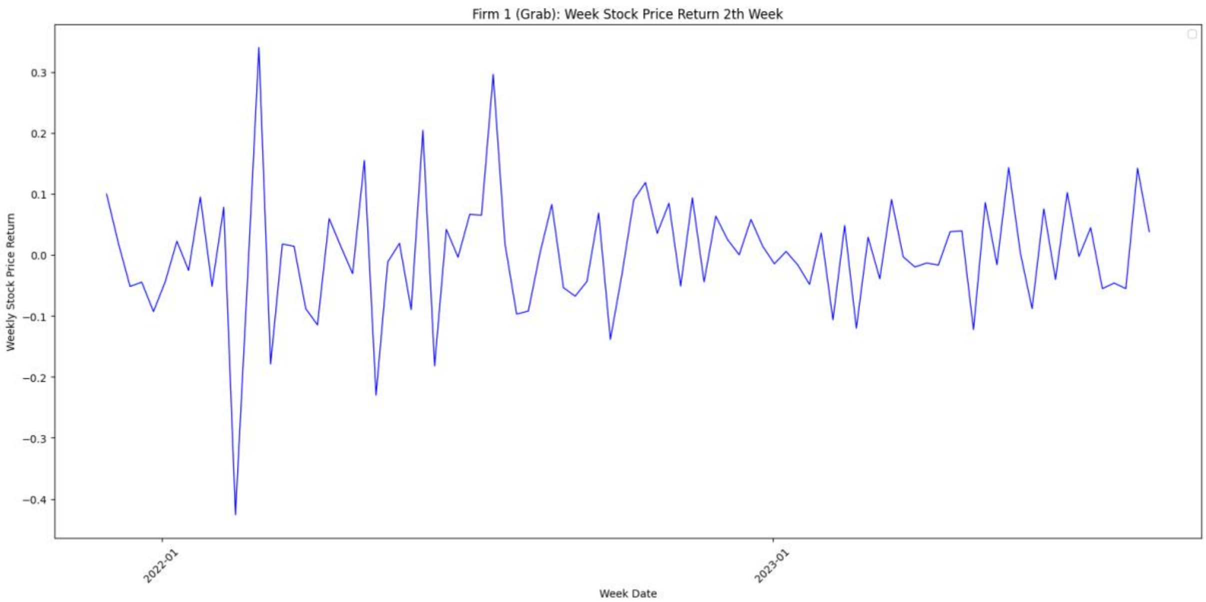


Figure 4.2 - Distribution of target week_stock_price_returns_2 for weekly aggregated dataset for both firms

We can notice that for the first set of graphs (Figure 4.1) there are periods were stock price returns stabilizes and this is due to, as mentioned before, the non-existence of data referent to the weekend, since the stock market is not open.

Additionally, besides the increased number of data points in the first set, it is possible to see that the distribution looks very similar besides the different granularity level, as it was expected based on the descriptive statistics analysis performed after we creation of these target features.

4.2. MODELLING RESULTS

The following Table 4.3 (Weekly aggregated data on XGBoost algorithm), Table 4.4 (Daily aggregated data on XGBoost algorithm), Table 4.5 (Weekly aggregated data on RNN-LSTM algorithm) and Table 4.6 (Daily aggregated data on RNN-LSTM algorithm) show the best results that we acheived for each set of features and time step (and best absolute results per metric in bold):

Table 4.3 - Weekly Aggregated XGBoost Results

Model	Time Step	MSE	MAE	R2	MDA
Xgboost_stock_prices_week_1	1	0.0057	0.0544	-0.0021	0.0000
Xgboost_stock_prices_week_3	3	0.0071	0.0572	-0.0455	0.3710
Xgboost_stock_prices_week_5	5	0.0082	0.0606	-0.1029	**0.6393**
Xgboost_stock_prices_week_7	7	**0.0044**	**0.0518**	-0.1755	0.0000
Xgboost_word_features_week_1	1	0.0059	0.0544	-0.0021	0.0000
Xgboost_word_features_week_3	3	0.0071	0.0572	-0.0460	0.4677
Xgboost_word_features_week_5	5	0.0083	0.0606	-0.1157	**0.6393**
Xgboost_word_features_week_7	7	**0.0044**	**0.0518**	-0.1755	0.0000
Xgboost_star_features_week_1	1	0.0057	0.0565	**0.0022**	0.4516
Xgboost_star_features_week_3	3	0.0071	0.0572	-0.0456	0.0000
Xgboost_star_features_week_5	5	0.0082	0.0605	-0.1090	0.5246
Xgboost_star_features_week_7	7	**0.0044**	**0.0518**	-0.1755	0.0000
Xgboost_sentiment_features_week_1	1	0.0057	0.0544	-0.0021	0.0000
Xgboost_sentiment_features_week_3	3	0.0071	0.0572	-0.0456	0.0000
Xgboost_sentiment_features_week_5	5	0.0081	0.0602	-0.0913	0.5738
Xgboost_sentiment_features_week_7	7	**0.0044**	**0.0518**	-0.1755	0.0000
Xgboost_category_features_week_1	1	0.0057	0.0544	-0.0021	0.0000
Xgboost_category_features_week_3	3	0.0077	0.0645	-0.1348	0.5645
Xgboost_category_features_week_5	5	0.0084	0.0617	-0.1261	0.4426
Xgboost_category_features_week_7	7	**0.0044**	**0.0518**	-0.1755	0.0000
Xgboost_all_features_week_1	1	0.0058	0.0536	-0.0121	0.5323
Xgboost_all_features_week_3	3	0.0070	0.0568	-0.0310	0.4355
Xgboost_all_features_week_5	5	0.0083	0.0610	-0.1127	0.0000
Xgboost_all_features_week_7	7	**0.0044**	**0.0518**	-0.1755	0.0000

Table 4.4 - Daily Aggregated XGBoost Results

Model	Time Step	MSE	MAE	R2	MDA
Xgboost_stock_prices_day_1	1	0.0057	0.0534	0.0859	0.6395
Xgboost_stock_prices_day_3	3	0.0051	0.0503	0.2576	0.6520
Xgboost_stock_prices_day_5	5	0.0034	0.0397	0.5319	**0.7210**
Xgboost_stock_prices_day_7	7	0.0060	0.0501	0.2910	0.6918
Xgboost_word_features_day_1	1	0.0057	0.0541	0.0850	0.6301
Xgboost_word_features_day_3	3	0.0058	0.0542	0.1425	0.5956
Xgboost_word_features_day_5	5	**0.0033**	**0.0382**	**0.5405**	0.7179
Xgboost_word_features_day_7	7	0.0062	0.0540	0.2647	0.6635
Xgboost_star_features_day_1	1	0.0058	0.0562	0.0598	0.5799
Xgboost_star_features_day_3	3	0.0055	0.0520	0.1920	0.5737
Xgboost_star_features_day_5	5	0.0046	0.0478	0.3569	0.6489
Xgboost_star_features_day_7	7	0.0068	0.0545	0.1932	0.6352
Xgboost_sentiment_features_day_1	1	0.0056	0.0540	0.1020	0.5768
Xgboost_sentiment_features_day_3	3	0.0050	0.0499	0.2602	0.6709
Xgboost_sentiment_features_day_5	5	0.0043	0.0457	0.4007	0.7022
Xgboost_sentiment_features_day_7	7	0.0069	0.0537	0.1828	0.6761
Xgboost_category_features_day_1	1	0.0057	0.0549	0.0747	0.6207
Xgboost_category_features_day_3	3	0.0056	0.0523	0.1795	0.6019
Xgboost_category_features_day_5	5	0.0055	0.0506	0.2361	0.6113
Xgboost_category_features_day_7	7	0.0073	0.0565	0.1367	0.6321
Xgboost_all_features_day_1	1	0.0060	0.0555	0.0360	0.5956
Xgboost_all_features_day_3	3	0.0060	0.0545	0.1246	0.5925
Xgboost_all_features_day_5	5	0.0036	0.0449	0.4930	0.6959
Xgboost_all_features_day_7	7	0.0066	0.0579	0.2207	0.5881

Table 4.5 - Weekly Aggregated RNN-LSTM Results

Model	Time Step	MSE	MAE	R2	MDA
RNN_stock_prices_week_1	1	0.0057	0.0544	**-0.0001**	0.5484
RNN_stock_prices_week_3	3	0.0072	0.0575	-0.0536	**0.6129**
RNN_stock_prices_week_5	5	0.0085	0.0622	-0.1482	0.2295
RNN_stock_prices_week_7	7	0.0062	0.0631	-0.6426	0.5082
RNN_word_features_week_1	1	0.0057	0.0545	-0.0002	0.5000

RNN_word_features_week_3	3	0.0072	0.0574	-0.0507	0.5645
RNN_word_features_week_5	5	0.0075	0.0582	-0.0162	0.5246
RNN_word_features_week_7	7	0.0040	0.0490	-0.0772	0.5902
RNN_star_features_week_1	1	0.0057	0.0544	-0.0054	0.4194
RNN_star_features_week_3	3	0.0068	0.0560	-0.0007	**0.6129**
RNN_star_features_week_5	5	0.0086	0.0628	-0.1642	0.3934
RNN_star_features_week_7	7	0.0076	0.0712	-1.0194	0.5410
RNN_sentiment_features_week_1	1	0.0057	0.0544	-0.0015	0.4032
RNN_sentiment_features_week_3	3	0.0073	0.0581	-0.0766	0.3871
RNN_sentiment_features_week_5	5	0.0079	0.0596	-0.0651	0.5410
RNN_sentiment_features_week_7	7	0.0040	0.0505	-0.0867	0.5082
RNN_category_features_week_1	1	0.0057	0.0544	-0.0000	0.5323
RNN_category_features_week_3	3	0.0076	0.0594	-0.1150	0.3548
RNN_category_features_week_5	5	0.0085	0.0621	-0.1474	0.5082
RNN_category_features_week_7	7	0.0045	0.0524	-0.1994	0.5902
RNN_all_features_week_1	1	0.0057	0.0544	-0.0028	0.4194
RNN_all_features_week_3	3	0.0072	0.0575	-0.0533	0.5645
RNN_all_features_week_5	5	0.0075	0.0601	-0.0052	0.4918
RNN_all_features_week_7	7	**0.0038**	**0.0484**	-0.0078	0.5902

Table 4.6 - Daily Aggregated RNN-LSTM Results

Model	Time Step	MSE	MAE	R2	MDA
RNN_stock_prices_day_1	1	0.0062	0.0550	0.0037	0.5078
RNN_stock_prices_day_3	3	0.0068	0.0565	0.0032	0.5110
RNN_stock_prices_day_5	5	0.0073	0.0591	-0.0210	0.5549
RNN_stock_prices_day_7	7	0.0083	0.0627	0.0103	0.5315
RNN_word_features_day_1	1	0.0053	0.0528	0.1394	0.6050
RNN_word_features_day_3	3	0.0056	0.0531	0.1777	0.6082
RNN_word_features_day_5	5	0.0029	0.0396	0.5988	0.7492
RNN_word_features_day_7	7	0.0053	0.0477	0.3693	0.7138
RNN_star_features_day_1	1	0.0063	0.0546	-0.0171	0.5016
RNN_star_features_day_3	3	0.0068	0.0560	0.0086	0.5392
RNN_star_features_day_5	5	0.0071	0.0580	0.0108	0.5329
RNN_star_features_day_7	7	0.0082	0.0624	0.0312	0.5629
RNN_sentiment_features_day_1	1	0.0062	0.0548	-0.0003	0.5078

RNN_sentiment_features_day_3	3	0.0069	0.0580	-0.0123	0.5204
RNN_sentiment_features_day_5	5	0.0070	0.0581	0.0201	0.5611
RNN_sentiment_features_day_7	7	0.0081	0.0620	0.0336	0.5723
RNN_category_features_day_1	1	0.0062	0.0563	-0.0031	0.5392
RNN_category_features_day_3	3	0.0070	0.0587	-0.0322	0.5549
RNN_category_features_day_5	5	0.0069	0.0572	0.0340	0.5423
RNN_category_features_day_7	7	0.0082	0.0625	0.0303	0.5472
RNN_all_features_day_1	1	0.0063	0.0592	-0.0223	0.5580
RNN_all_features_day_3	3	0.0058	0.0548	0.1548	0.6176
RNN_all_features_day_5	5	**0.0026**	**0.0362**	**0.6397**	**0.7524**
RNN _all_features_day_7	7	0.0051	0.0440	0.4005	0.7327

When analyzing the results above a significant difference in performance is aparent between each model when data is aggregated weekly and their daily aggregated counterparts.

Starting by analyzing the XGBoost Models for weekly aggregated data, we see that in terms of Mean Directional Accuracry (MDA), the best results come from using solely stock prices or including word features with data from the previous 5 weeks, with a value of 0.6393, which means that on average the model is able to predict the direction of the stock price returns (positive or negative) almost 64% of the times, which sounds promissing until we analyze the r-squared metric (which tells us how well the data fit in the model). Only one combination of parameters produced an positive r-squared, and even that was very low (0.0022), meaning that the majority of these models fit the data worse than just applying the average of the weekly stock price returns for all the data points. This ultimately labels these models as useless, without the need to analyze other metrics to confirm it.

The same conclusion was drawn in the RNN-LSTM models with weekly aggregated data, since there is no combination of parameters that promote the models to fit the data better than simply applying the average. Analyzing the MDA, we can still find two models that on average predict the weekly stock price direction above 61% of the times: One using only the stock features and the second including star features. It is interesting to notice that, unlike XGBoost, these models acheive these results when we consider the 3 previous weeks of data as well, which might suggest that this deep learning approach could reach a similar level of performance with a more modest amount of data available.

When analyzing the models with data aggregated daily, for example the XGBoost models, it is possible to see that all the models have a positive r-squared metric, which shows an significant improvement compared with the previously analyzed models. The model which includes word features and considers the previous 5 days of data acheived the best results for this algorithm when we consider the R-squared, Mean Squared Error and Mean Absolute Error metrics, with values of 0.5405, 0.0033 and 0.0382 respectively. It only falls short to the model that only considers stock prices and the previous 5 days of data in the MDA metric, but even that is particularly similar in terms of performance, with the first having a prediction direction accuracy of 71.79% and the latter 72.10%.

Lastly, we take a look into the RNN-LSTM models with daily aggregated data and we observe that there is a combination of parameters that outperforms every other in all the considered metrics. This model includes all the features available and data from the previous 5 days, which is a relevant difference from the models using data aggregated weekly, which could achieve their best performance with less data (when compared with the XGBoost models using a similar type of aggregation) . With a MSE of 0.0026 and an MAE of 0.0362, the model is able to explain on average 63.97% of the variability in the weekly stock price returns and predicts the correct stock price direction above 75% of the times. Another set of parameters comes close in performance, which includes only word features, with an r-squared of 59.88% and 74.92% of MDA. This result suggests that, although including all the features might promote an high performance in this model, the bulk of the prediction is carried by the word features which allow on average an increase of 10% to 20% of prediction direction accuracy when compared with models that only consider stock price, for the same amount of previous data considered.

4.3. Comparative Analysis of the RNN-LSTM and XGBoost Best Models

Having the previous results in mind, in order to ultimately choose the best model, we will compare 4 models: Xgboost_stock_prices_day_5, Xgboost_word_features_day_5, RNN_word_features_day_5 and RNN_all_features_day_5. These were chosen since they are the best 2 models per algorithm and all offer a different level of complexity in terms of features, since Xgboost_stock_prices_day_5 resorts to using just stock prices while Xgboost_word_features_day_5 and RNN_word_features_day_5 also include word features. RNN_all_features_day_5, being the best model overall, is the initial best candidate to be our best model, but its higher complexity in terms of data/features needed might be considered a negative point for this model, considering the reduced performance increase when compared with RNN_word_features_day_5 model.

We can start by analyzing the parameters that resulted from the hypertuning process on Table 4.7, Table 4.8, Table 4.9 and Table 4.10:

Table 4.7 - Best parameters for Xgboost_stock_prices_day_5

Parameter	Value
n_estimators	476
max_depth	10
learning_rate	0.0168
subsample	0.9812

colsample_bytree	0.7692
lambda	0.0017
alpha	0.0041

Table 4.8 - Best parameters for Xgboost_word_features_day_5

Parameter	Value
n_estimators	344
max_depth	8
learning_rate	0.1471
subsample	0.8393
colsample_bytree	0.8452
lambda	0.0128
alpha	0.0015

Table 4.9 - Best parameters for RNN_word_features_day_5

Parameter	Value
lstm_units_1	150
l2_1	8.22e-05
lstm_units_2	200
l2_2	0.0037
dropout_rate	0.4
optimizer	Adam
batch_size	64

Table 4.10 - Best parameters for RNN_all_features_day_5

Parameter	Value
lstm_units_1	100
l2_1	3.24e-05
lstm_units_2	150
l2_2	0.0025
dropout_rate	0.4
optimizer	Adam

batch_size	48

From these tables, it is possible to notice a trend: The more features we add, the less computationally expensive the models become in terms of parameters. From model Xgboost_stock_prices_day_5 to model Xgboost_word_features_day_5 we see an decrease in the number of estimators needed of 132, which represents a decrease of almost 28%, but also a 20% decrease in terms of max_depth, from 10 to 8. In the RNN models, increasing the feature complexity reduces the number of LSTM units in the first and second layer by 50, to 100 and 150 respectively, but also the batch_size is reduced in 25%, from 64 to 48.

This shows that the increase in number of considered features has a positive impact in our models, while decreasing the computational cost of running them, so we will proceed this comparison by considering only the Xgboost_word_features_day_5 and RNN_all_features_day_5 models. We see that in terms of evaluation metrics, the RNN_all_features_day_5 model surpasses the XGBoost variant in every way, but we should also look how these values look like graphically on Figure 4.3 for model RNN_all_features_day_5 and Figure 4.5 for model Xgboost_word_features_day_5:

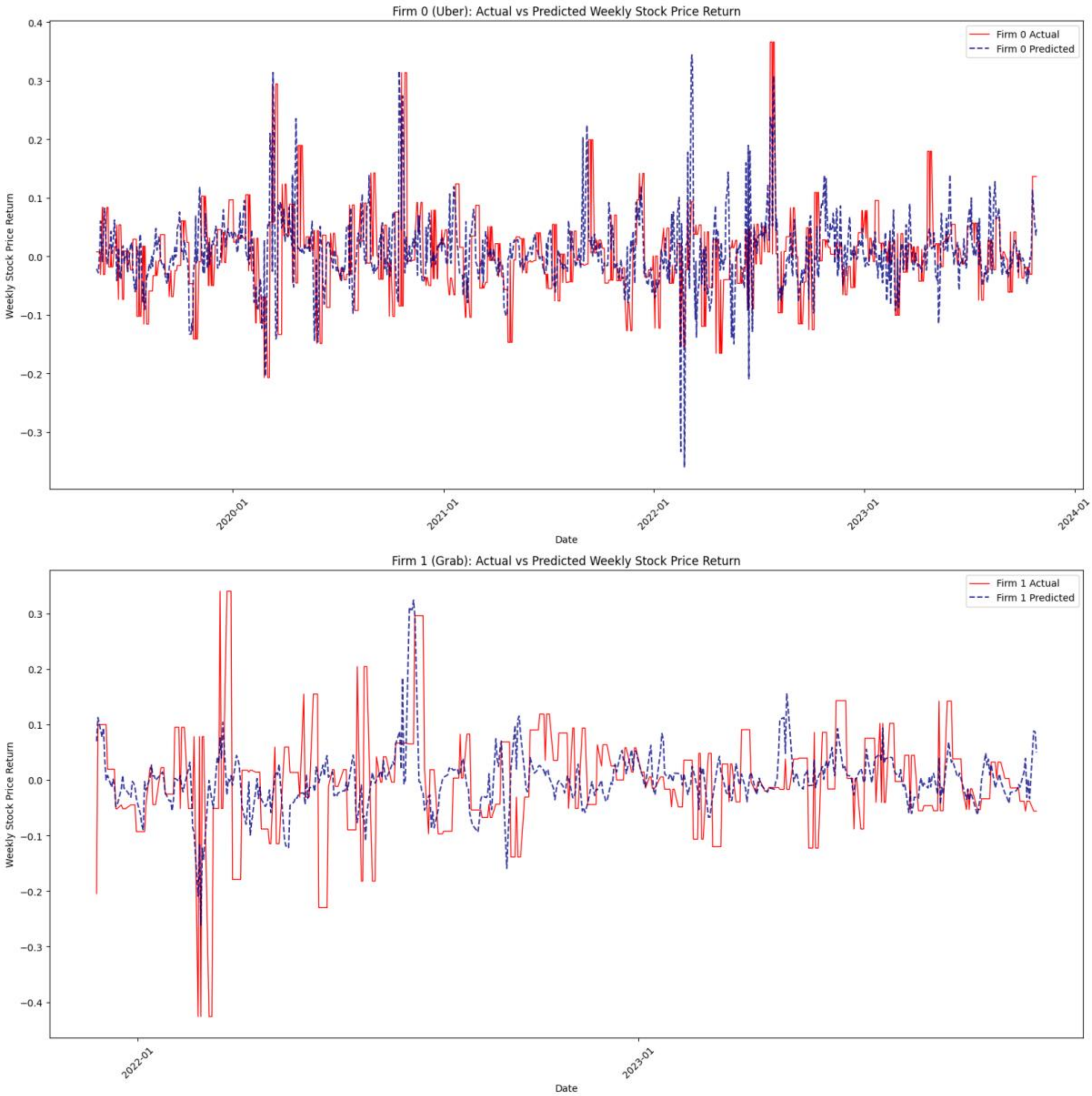


Figure 4.3 - RNN_all_features_day_5 model predicted vs actual weekly stock prices for firms 0 and 1

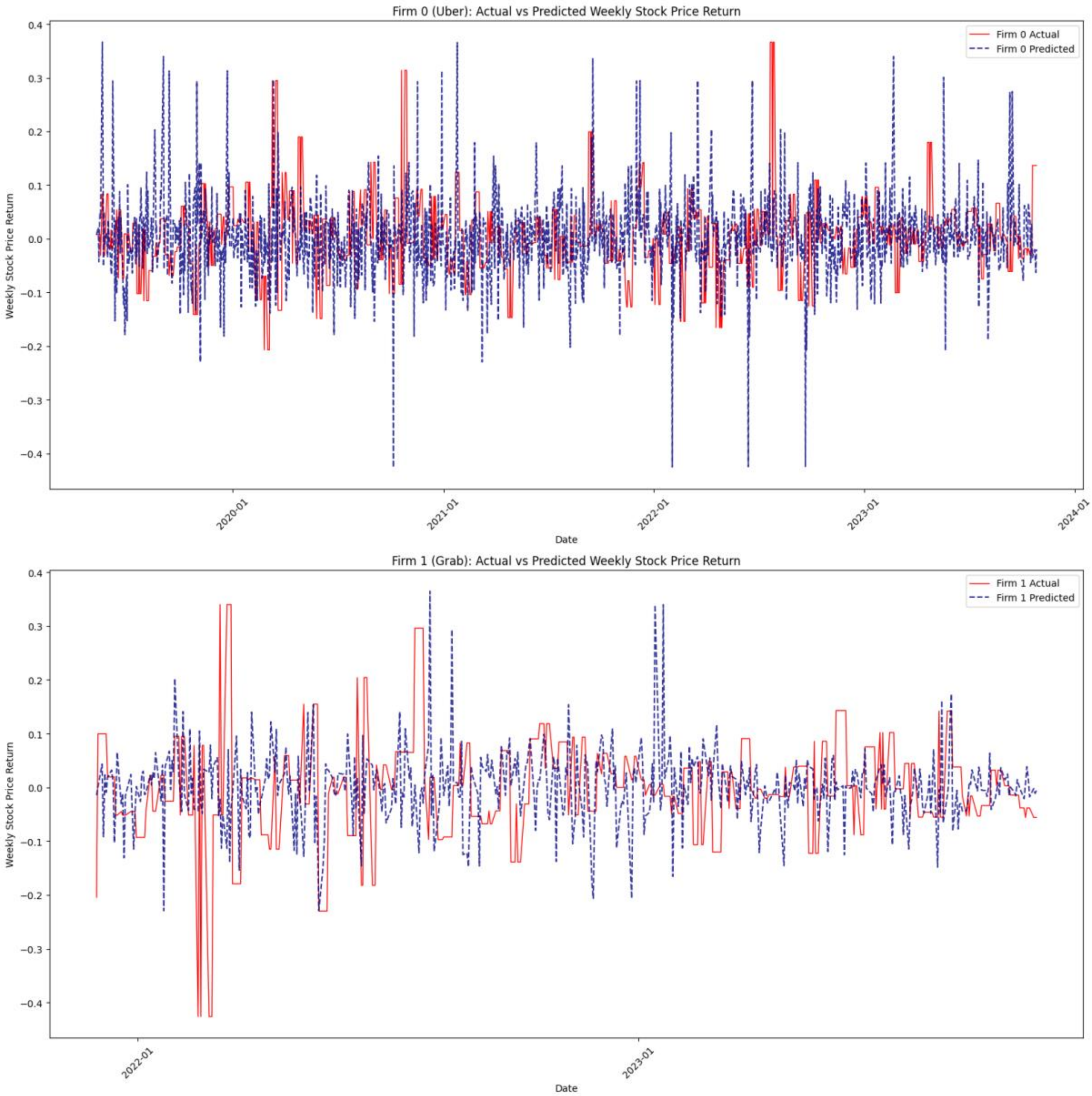


Figure 4.4 - Xgboost_word_features_day_5 model predicted vs actual weekly stock prices for firms 0 and 1

When analyzing these graphs it is possible to compare this models in two ways: How they perform comparatively to each other, but also how they perform for different types of firms.

Let's start by analyzing the Deep Learning model. For the firm 0, Uber, we see that the RNN-LSTM model fits very well to the data, with some major price overshootings (negatively and positively) over the year 2022. It was also unable to predict a stock price return spike in the first third of 2023, having then overshot a negative stock price return straight away. Regarding firm 1, we see that the model is taking a very conservative approach, often under predicting (in absolute) the weekly stock price return. This result could point us to the significant impact that firms' characteristics might have on our model's performance, in this case probably due to the difference in data available, both volume and time-span-wise.

On the other hand, our boosting algorithm shows to be a bit less “controlled” with the predictions. For firm 0 it often overpredicts the stock price return in both directions, while being unable to predict the majority of the abnormal positive stock price returns like the one in the fourth quarter of 2020 or the one right after the middle of 2022. In the case of the smaller firm 1, the results are largely similar, often overshooting returns when stock prices are more stable, but being unable to predict actual spikes (positive or negative) like the ones in the first quarter of 2022 or some smaller ones throughout 2023. Although it still achieves an high value regarding MDA, the fact that the model overpredicts stock prices so often and in such a great scale, might bring excessive optimism or pessimism for a manager trying to leverage user generated content to plan the future according to these models predictions, which we believe its an huge disadvantage when compared with the RNN-LSTM model we are still considering.

4.4. Performance by Firm

With our final model selected (RNN_all_features_day_5), it is important to understand if this model can be applied to any company, ie. how does the model perform when applied to firms with different characteristics. For that, we will look into the same evaluation metrics, but segregated by company, seen in Table 4.11 for Train data and Table 4.12 for Test data.

Table 4.11 - Train evaluation metrics by firm

Metric	Firm 0	Firm 1
Mean Squared Error	0.0007	0.0009
Mean Absolute Error	0.0180	0.0204
R-Squared	0.8728	0.9179
Mean Directional Accuracy	0.8809	0.9220

Table 4.12 - Test evaluation metrics by firm

Metric	Firm 0	Firm 1
Mean Squared Error	0.0016	0.0024
Mean Absolute Error	0.0293	0.0328
R-Squared	0.7463	0.5093
Mean Directional Accuracy	0.7991	0.7419

First thing we notice is that both models show some level of overfitting, significantly higher for the smaller firm 1: While the Mean Squared Error increases by 0,0009 for firm 0, firm 1 produces an increase of 0.0015 but Mean Absolute Error also sees an increase of 0,0113 for the biggest firm (0) and 0,0124 for firm 1, which can be considered a small difference. For R-squared we have a very different result, since the fit performance decreases around 13% for firm 0, but more than 40% for firm 1, which is a very significant difference and might be an impactful factor when applying this model to a firm with a lower market share. Finally, during training the model was able to predict correctly the direction of the stock price return for around 8% more data than when testing for firm 0, while for firm one, during testing it predicted correctly around less 18% of data points.

This observation is probably caused by the significant difference in data volume between companies, since firm 0 (Uber) has been in the stock market for longer and has a relatively higher market share, so naturally that translates into more reviews to feed into the model, allowing it to be able to generalize better than for firm 0.

In practical terms, this translates in the model showing an Mean Squared Error reduction of 33% and Mean Absolute Error reduction of 11% when predicting for firm 0. Also, the model has a fit performance almost 24% higher and almost 5% directional prediction performance increase for firm 0 when compared with firm 1. Although we believe the model can still be applied to smaller firms like firm 1, these results show that the data available for firm 0 allows the model to output way more satisfactory results in comparison.

5. CONCLUSIONS AND FUTURE WORKS

In this work we explored the predictive power of consumer online reviews from Google Play Store of ridesharing companies, focusing on Uber and Grab, in forecasting weekly stock price returns. We aimed to answer three core research objectives:

1. Assess the predictive performance of consumer reviews on stock price returns: We extensively tested different models based on different algorithms (XGBoost and Recurrent Neural Networks), with different parameters and data aggregation approaches. Our final model, a Recurrent Neural Network with Long-Short Term Memory, which includes every feature we have at our dispose and data from the previous 5 days aggregated daily, was able to achieve overall very low Mean Squared Error (0.0026) and Mean Absolute Error (0.0362) while maintaining a good data fit performance (R-squared = 0.6397) and predicting correctly if the weekly stock price return was positive or negative above 75% of the times. When we consider a model with similar structure and parameters but that only considers stock price-related features (MSE=0.0073, MAE=0.0591, R2=-0.0210 and MDA=0.5549) we conclude that consumer reviews indeed carry information that allows for a better performance in predicting stock price returns. Still related to this research objective, it is important to notice that one of the most important factors for the model performance was how data was aggregated, since when data was aggregated by day our model significantly outperformed the models in which we aggregated data by week. This might be related to the fact that, by aggregating data daily, we are able to keep more data points to feed to the model, maintaining possible data trends more visible, that might be overshadowed by other more volumous but less relevant trends within the reviews when aggregating data weekly.

2. Identify the specific characteristics of online reviews that contribute most significantly to prediction performance: Although our final model leverages all the features we had available (Stock Price Features, Word Features, Start Features, Sentiment Features and Category Features) to achieve the best performance possible, we saw that the performance increase from just including the word features additionally to the stock features was very limited. This suggests that the word features (Daily Average Review Likes, Total Number of daily reviews, Most common, Second most and Third Most common words in those reviews) contribute the most for the model predictabililty of stock price returns. This said, we also noticed that, by including all the other features, we could reduce models' complexity and consequently reduce computational costs, even if performance didn't increase very significantly.

3. Determine the optimal time lag for the influence of consumer reviews on stock returns: We also concluded that data correlates the best with stock price returns

visible 6 weeks after the reviews were posted. This suggests that the ridesharing industry might feel a quicker effect from user-generated online content than other Industries such as Retail, as the optimal time lag for the latter was concluded to be 8 weeks by Wu et al (2019). The reduced optimal lag can be seen as an benefit for decision-making processes, since managers can leverage the effects of user generated feedback quicker, but also poses the negative effect that, in case the model outputs pessimistic results, the responsible entities are left with less time available to counteract those results. Finally, it is also important to notice that, although we are predicting stock price returns for the 6^{th} week after the reviews were posted, our models still benefit from including data from previous days, like our best model which keeps into account data from the 5 previous days, outperforming the same model that includes the 3 or 7 previous days of data.

In conclusion, this study highlights that consumer online reviews can be used as a valuable data source to predict stock price return trends in the ridesharing Industry. These types of companies rely heavily on consumer satisfaction and feedback, so these findings offer valuable insights for investors, analysts and managers, emphasizing the importance of incorporating consumer feedback into financial decision-making.

Furthermore, we believe our findings suggest that UGC, particularly online reviews, can complement traditional financial indicators by providing real-time insights into consumer sentiment and behaviors, which demonstrates significant potential for financial analytics beyond the ridesharing industry. Any industry in which consumer feedback is crucial to financial outcomes such as retail, technology, hospitality or even automotive, is a potential industry to which our approach can be extended upon.

As far as how future works can develop on this research, there could be an expansion of the study in the following areas:

- Industry Expansion: Despite selecting two firms with different characteristics for the purpose of assessing the predictive power of UGC, we understand that even companies with similar characteristics might lead the model to output significantly different results due to other factors that we are not accounting for such as user's demographics. Exploring that could bring important knowledge on how to fine tune the models leveraging UGC for different firm-specific cases.
- Feature Expansion: Although we consider the feature set we selected abrangent enough to take conclusions for our research objectives, exploring other data sources and acquiring new features through feature engineering could pose an important step into unlocking new results that might improve our solution and help explain further what other specific details of online consumer-generated content contribute to the predictive power of stock price returns.

- Model Structure Optimization: In this work we were more focused on assessing the predictive capabilities of Customer Online Reviews of Week Stock Price Returns, not so much on building the perfect model for the task, due to the inherent complexity of Deep Learning models which could offer virtually infinite combinations of structures. We suggest that the results could be improved by deep diving further into our model, tweaking its structure with diferent layers (such as attention layers for the RNN-LSTM) and parameter values.

As a direct consequence of Feature Expansion and Model Structure Optimization, we believe future works should also dive deeper into analyzing the trade-offs between model complexity and interpretability for this real life use case. On one hand, complex deep learning models such as RNN-LSTM excel in capturing temporal dependencies and non-linear relationships in data, being highly adaptable, but they can really difficult the interpretabililty aspect of the model, which might difficult the explanation of how data relates to the output for stakeholders or investors that are on the lower technical spectrum, which might affect transparency in decision-making processes. Boosting algorithms, such as the XGBoost, although still offering a high level of complexity but not as good performance in this case, have their interpretability made a bit easier due to tool such as feature importance rankings or partial dependency plots which can make easier the communication process to the final decision-makers.

Besides model choices, data preprocessing should also be a subject of these future studies as we saw that simple decisions, such as how to aggregate data, highly affect not only the outcomes but also the complexity of the models, which should be always kept in mind when we are building prediction models that might be put into production and available for a large variety of users.

BIBLIOGRAPHICAL REFERENCES

Antweiler, W., & Frank, M. Z. (2001). Is All That Talk Just Noise? The Information Content of Internet Stock Message Boards. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.282320

Asur, S., & Huberman, B. A. (2010). Predicting the Future with Social Media. *2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, 492–499. https://doi.org/10.1109/WI-IAT.2010.63

Awad, A. L., Elkaffas, S. M., & Fakhr, M. W. (2023). Stock Market Prediction Using Deep Reinforcement Learning. *Applied System Innovation*, *6*(6), 106. https://doi.org/10.3390/asi6060106

BERTopic. (n.d.). *Zero-Shot Topic Modelling* . Retrieved June 2, 2024, from https://maartengr.github.io/BERTopic/getting_started/zeroshot/zeroshot.html

Bollen, J., Mao, H., & Zeng, X. (2011). Twitter mood predicts the stock market. *Journal of Computational Science*, *2*(1), 1–8. https://doi.org/10.1016/j.jocs.2010.12.007

Brown, S. J., Goetzmann, W., Ibbotson, R. G., & Ross, S. A. (1992). Survivorship Bias in Performance Studies. *Review of Financial Studies*, *5*(4), 553–580. https://doi.org/10.1093/rfs/5.4.553

Campbell, J. Y., Lettau, M., Malkiel, B. G., & Xu, Y. (2001). Have Individual Stocks Become More Volatile? An Empirical Exploration of Idiosyncratic Risk. *The Journal of Finance*, *56*(1), 1–43. https://doi.org/10.1111/0022-1082.00318

Chahine, S., & Malhotra, N. K. (2018). Impact of social media strategies on stock price: the case of Twitter. *European Journal of Marketing*, *52*(7/8), 1526–1549. https://doi.org/10.1108/EJM-10-2017-0718

Chevalier, J. A., & Mayzlin, D. (2006). The Effect of Word of Mouth on Sales: Online Book Reviews. *Journal of Marketing Research*, *43*(3), 345–354. https://doi.org/10.1509/jmkr.43.3.345

Chiang, S. (2024, February 23). Ride-hailing giant Grab posts first profitable quarter, announces $500 million share buyback. *CNBC*. https://www.cnbc.com/2024/02/23/grab-posts-first-profitable-quarter-announces-share-buyback.html

Cramer, J., & Krueger, A. B. (2016). Disruptive Change in the Taxi Business: The Case of Uber. *American Economic Review*, *106*(5), 177–182. https://doi.org/10.1257/aer.p20161002

Darapaneni, N., Paduri, A. R., Sharma, H., Manjrekar, M., Hindlekar, N., Bhagat, P., Aiyer, U., & Agarwal, Y. (2022). *Stock Price Prediction using Sentiment Analysis and Deep Learning for Indian Markets*. http://arxiv.org/abs/2204.05783

Duan, W., Gu, B., & Whinston, A. B. (2008). Do online reviews matter? — An empirical investigation of panel data. *Decision Support Systems*, *45*(4), 1007–1016. https://doi.org/10.1016/j.dss.2008.04.001

Elton, E. J., Gruber, M. J., & Blake, C. R. (1996). Survivor Bias and Mutual Fund Performance. *Review of Financial Studies*, *9*(4), 1097–1120. https://doi.org/10.1093/rfs/9.4.1097

Fama, E. F., & French, K. R. (1993). Common risk factors in the returns on stocks and bonds. *Journal of Financial Economics*, *33*(1), 3–56. https://doi.org/10.1016/0304-405X(93)90023-5

Fama, E. F., & French, K. R. (2007). The Anatomy of Value and Growth Stock Returns. *Financial Analysts Journal*, *63*(6), 44–54. https://doi.org/10.2469/faj.v63.n6.4926

Fornell, C., Mithas, S., Morgeson, F. V., & Krishnan, M. S. (2006). Customer Satisfaction and Stock Prices: High Returns, Low Risk. *Journal of Marketing*, *70*(1), 3–14. https://doi.org/10.1509/jmkg.2006.70.1.3

Godes, D., & Mayzlin, D. (2004). Using Online Conversations to Study Word-of-Mouth Communication. *Marketing Science*, *23*(4), 545–560. https://doi.org/10.1287/mksc.1040.0071

Gössling, S., & Michael Hall, C. (2019). Sharing versus collaborative economy: how to align ICT developments and the SDGs in tourism? *Journal of Sustainable Tourism*, *27*(1), 74–96. https://doi.org/10.1080/09669582.2018.1560455

Grab. (2018, March 26). *Grab Merges with Uber in Southeast Asia*. https://www.grab.com/sg/press/business/grab-merges-with-uber-in-southeast-asia/

Grab. (2024). *Grab Reports Fourth Quarter and Full Year 2023 Results and Announces Inaugural Share Repurchase Program of Up to $500 Million*. https://www.grab.com/sg/press/others/grab-reports-fourth-quarter-and-full-year-2023-results-and-announces-inaugural-share-repurchase-program-of-up-to-500-million/

Grootendorst, M. (2022). *BERTopic: Neural topic modeling with a class-based TF-IDF procedure*. http://arxiv.org/abs/2203.05794

Gunawardane, G. (2023). Enhancing customer satisfaction and experience in financial services: a survey of recent research in financial services journals. *Journal of Financial Services Marketing*, *28*(2), 255–269. https://doi.org/10.1057/s41264-022-00148-x

Gupta, S., Lehmann, D. R., & Stuart, J. A. (2004). Valuing Customers. *Journal of Marketing Research*, *41*(1), 7–18. https://doi.org/10.1509/jmkr.41.1.7.25084

Gupta, T., Leung, E., & Roscovan, V. (2021). Online Appendix: Consumer Spending and the Cross-Section of Stock Returns. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.3969435

Ho, T.-T., & Huang, Y. (2021). Stock Price Movement Prediction Using Sentiment Analysis and CandleStick Chart Representation. *Sensors*, *21*(23), 7957. https://doi.org/10.3390/s21237957

Huang, J. (2018). The customer knows best: The investment value of consumer opinions. *Journal of Financial Economics*, *128*(1), 164–182. https://doi.org/10.1016/j.jfineco.2018.02.001

Koukaras, P., Nousi, C., & Tjortjis, C. (2022). Stock Market Prediction Using Microblogging Sentiment Analysis and Machine Learning. *Telecom*, *3*(2), 358–378. https://doi.org/10.3390/telecom3020019

Krumm, J., Davies, N., & Narayanaswami, C. (2008). User-Generated Content. *IEEE Pervasive Computing*, *7*(4), 10–11. https://doi.org/10.1109/MPRV.2008.85

Liu, X. (2020). Analyzing the impact of user-generated content on B2B Firms' stock performance: Big data analysis with machine learning methods. *Industrial Marketing Management*, *86*, 30–39. https://doi.org/10.1016/j.indmarman.2019.02.021

Luo, X. (2007). Consumer Negative Voice and Firm-Idiosyncratic Stock Returns. *Journal of Marketing*, *71*(3), 75–88. https://doi.org/10.1509/jmkg.71.3.075

Mahmud, M. S., Jaman Bonny, A., Saha, U., Jahan, M., Tuna, Z. F., & Al Marouf, A. (2022). Sentiment Analysis from User-Generated Reviews of Ride-Sharing Mobile Applications. *Proceedings - 6th International Conference on Computing Methodologies and Communication, ICCMC 2022*, 738–744. https://doi.org/10.1109/ICCMC53470.2022.9753947

Mittal, V., Han, K., Frennea, C., Blut, M., Shaik, M., Bosukonda, N., & Sridhar, S. (2023). Customer satisfaction, loyalty behaviors, and firm financial performance: what 40 years of research tells us. *Marketing Letters*, *34*(2), 171–187. https://doi.org/10.1007/s11002-023-09671-w

Mohan, S., Mullapudi, S., Sammeta, S., Vijayvergia, P., & Anastasiu, D. C. (2019). Stock price prediction using news sentiment analysis. *Proceedings - 5th IEEE International Conference on Big Data Service and Applications, BigDataService 2019, Workshop on Big Data in Water Resources, Environment, and Hydraulic Engineering and Workshop on*

Medical, Healthcare, Using Big Data Technologies, 205–208. https://doi.org/10.1109/BigDataService.2019.00035

Mordor Intelligence. (2024). *Ride Hailing Industry Size & Share Analysis - Growth Trends & Forecasts (2024 - 2029)*. https://www.mordorintelligence.com/industry-reports/ride-hailing-market

Palomino, M. A., & Aider, F. (2022). Evaluating the Effectiveness of Text Pre-Processing in Sentiment Analysis. *Applied Sciences*, *12*(17), 8765. https://doi.org/10.3390/app12178765

Qiu, J., Wang, B., & Zhou, C. (2020). Forecasting stock prices with long-short term memory neural network based on attention mechanism. *PLOS ONE*, *15*(1), e0227222. https://doi.org/10.1371/journal.pone.0227222

Rana, P. (2024). Uber Posts First Annual Profit Since Its IPO. *The Wall Street Journal*. https://www.wsj.com/business/earnings/uber-q4-earnings-report-2023-4e0d59f6

Rayle, L., Dai, D., Chan, N., Cervero, R., & Shaheen, S. (2016). Just a better taxi? A survey-based comparison of taxis, transit, and ridesourcing services in San Francisco. *Transport Policy*, *45*, 168–178. https://doi.org/10.1016/j.tranpol.2015.10.004

Richins, M. L. (1983). Negative Word-of-Mouth by Dissatisfied Consumers: A Pilot Study. *Journal of Marketing*, *47*(1), 68. https://doi.org/10.2307/3203428

Riley, R. A., Pearson, T. A., & Trompeter, G. (2003). The value relevance of non-financial performance variables and accounting information: the case of the airline industry. *Journal of Accounting and Public Policy*, *22*(3), 231–254. https://doi.org/10.1016/S0278-4254(03)00021-8

Singh, J. (1988). Consumer Complaint Intentions and Behavior: Definitional and Taxonomical Issues. *Journal of Marketing*, *52*(1), 93. https://doi.org/10.2307/1251688

Srivastava, R. K., Shervani, T. A., & Fahey, L. (1998). Market-Based Assets and Shareholder Value: A Framework for Analysis. *Journal of Marketing*, *62*(1), 2. https://doi.org/10.2307/1251799

Statcounter. (2024). *Mobile Operating System Market Share Worldwide from April 2023 to May 2024*. https://gs.statcounter.com/os-market-share/mobile/worldwide

Su, Z., Wu, D., Zhou, Z., Wu, J., & Yin, L. (2023). *The Impact of Customer Online Satisfaction on Stock Returns: Evidence from the E-commerce Reviews in China*. http://arxiv.org/abs/2306.12119

Tounsi, Y., Anoun, H., & Hassouni, L. (2020). CSMAS. *Proceedings of the 3rd International Conference on Networking, Information Systems & Security*, 1–7. https://doi.org/10.1145/3386723.3387851

Tuli, K. R., & Bharadwaj, S. G. (2009). Customer Satisfaction and Stock Returns Risk. *Journal of Marketing*, *73*(6), 184–197. https://doi.org/10.1509/jmkg.73.6.184

Wu, J., Xu, K., & Zhao, J. (2019). *Online reviews can predict long-term returns of individual stocks*. http://arxiv.org/abs/1905.03189

Xiao, Q., & Ihnaini, B. (2023). Stock trend prediction using sentiment analysis. *PeerJ Computer Science*, *9*, e1293. https://doi.org/10.7717/peerj-cs.1293

Zaheer, S., Anjum, N., Hussain, S., Algarni, A. D., Iqbal, J., Bourouis, S., & Ullah, S. S. (2023). A Multi Parameter Forecasting for Stock Time Series Data Using LSTM and Deep Learning Model. *Mathematics*, *11*(3), 590. https://doi.org/10.3390/math11030590

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação

Universidade Nova de Lisboa