Google Stock Trend Analysis

Faiz Shaikh

July 16th, 2024

Abstract

This theoretical project aims to lay out the foundation for a future coding project that will analyze Google (Alphabet Inc.) stock trends using advanced time series forecasting techniques and generative AI to provide insights into these trends. The primary goal is to create a comprehensive visual dashboard that displays trends, forecasts, and accuracy metrics. Understanding stock price trends and forecasting future prices is crucial for investors, analysts, and financial institutions. Accurate forecasts can lead to better investment decisions, risk management, and financial planning.

The stock market is known for its volatility, driven by various factors such as economic indicators, market sentiment, and company-specific news. Currently, traditional forecasting methods often fall short in capturing the complex patterns and seasonality that are often inherent in stock prices. This project addresses these challenges by leveraging modern data management practices (including data collection, pre-processing, storage, and visualization techniques) as well the rise of AI and data processing technologies.

The analysis involves several key steps: data collection, data cleaning, exploratory data analysis, model development, and evaluation. The data is sourced from a reputable historical Google stock price dataset from Kaggle[1] and stored in a relational database to ensure structured data handling and retrieval. Data cleaning processes are applied to handle missing values, remove outliers, and perform feature engineering. Exploratory data analysis helps in identifying trends, seasonality, and correlations between variables.

For the forecasting models, various techniques are explored, including ARIMA (AutoRegressive Integrated Moving Average), Prophet by Meta, and LSTM (Long Short-Term Memory). Each model has its strengths and weaknesses, and the performance is evaluated using metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE). The ARIMA model is particularly noted for its ability to capture short-term trends and seasonality patterns effectively. These models are discussed extensively in the research paper ”Forecasting Time Series With Complex Seasonal Patterns Using Exponential Smoothing” by De Livera et al. (2011), which provided me the necessary foundational knowledge of these topics.

[1] `https://www.kaggle.com/datasets/varpit94/google-stock-data/data`

In addition to traditional forecasting methods, generative AI will be used to provide narrative insights. OpenAI's GPT-4o is a newly launched model that is used to generate explanations and context for the observed trends and forecasts, which can help add a qualitative layer to the more quantitative analysis. This dual approach ensures that the insights are both data-driven and easily understandable, making them more actionable for stakeholders.

The plan is to use the results of the analysis and forecasting to create a visual dashboard (or potentially application), which includes graphs and charts that display the stock trends, forecasts, and accuracy metrics. The dashboard is designed to be user-friendly, providing a clear and intuitive interface for users to interact with the data.

This project not only demonstrates the application of advanced time series forecasting techniques but also highlights the potential of combining these techniques with generative AI for enhanced interpretability and insight generation. The integration of AI-generated insights helps bridge the gap between raw data and actionable knowledge, providing a more comprehensive understanding of the factors driving stock price movements This is something I am deeply interested in as linking both accurate results and comprehensibility is the key factors towards making a successful product. While currently this paper is theoretical, I hope that one day by implementing this project fully I can gain a better understanding of these two aspects.

Future work could involve exploring more sophisticated models and incorporating additional features to further improve forecast accuracy. Expanding the dataset to include more historical data and other relevant variables could also enhance the robustness of the models. Overall, this project provides a valuable tool for investors and analysts, offering a blend of quantitative and qualitative insights into Google stock trends. By addressing the limitations of existing forecasting methods and incorporating new technologies, this project sets a foundation for more advanced and interpretable financial analysis tools. The findings and methodologies presented here can be extended to other stocks and financial instruments, contributing to the broader field of financial analytics and decision-making.

1 Introduction

The project aims to analyze Google (Alphabet Inc.) stock trends on a yearly basis using advanced time series forecasting techniques. This analysis will involve creating a comprehensive visual dashboard that displays trends, forecasts, and accuracy metrics. Furthermore, I will incorporate generative AI to provide insights into these trends.

Understanding stock price trends and forecasting future prices is crucial for investors, analysts, and financial institutions. Accurate forecasts can lead to better investment decisions, risk management, and financial planning. This project combines traditional time series analysis with modern machine learning techniques to improve forecast accuracy and provide deeper insights.

While largely theoretical, this project will explain the foundational principles that underlie undertaking such a project. Moreover, code snippets that are shared will be expanded upon and explained clearly on their uses for accomplishing this project. The rest of this project will follow a structural approach with identifying the problem followed by my proposed interpretation of how to make my solution to it.

2 Project Definition

2.1 Problem Statement

The main objective of this project is to effectively predict the fluctuating prices of Google stocks. These prices are highly unstable and shaped by various elements such as market sentiment, economic indicators, and company-specific developments. Precise forecasting offers significant benefits to investors, analysts, and financial institutions. By comprehending the factors influencing stock prices and anticipating future trends, stakeholders can make informed choices that enhance their financial results.

2.2 Research Question

How can advanced time series forecasting techniques, combined with generative AI insights and the data management practices learned in this course, create an accurate and interpretable Google stock price prediction model? This research question addresses the need to not only predict stock prices accurately but also to understand the underlying reasons behind these predictions (ie. AI used for explanations——something not commonly present in current stock prediction tools. By combining quantitative forecasting with qualitative insights from AI, I aim to bridge the gap between raw data and actionable knowledge.

2.3 Strategic Aspects

This project strategically leverages advanced forecasting techniques to improve accuracy, utilizes generative AI for enhanced explanatory capabilities, and develops a user-friendly visual dashboard for effective data presentation. This comprehensive approach aligns with modern data management and analytics practices covered in our coursework, such as data collection, data pre-processing, data storage (ie. relational databases), and data visualization techniques. Each step of the project, from data acquisition to model deployment, is designed to maximize the usability and interpretability of the result to ensure that the insights gained are both robust and actionable.

Moreover, as mentioned, the time series framework that goes into this project is inspired by ”Forecasting Time Series With Complex Seasonal Patterns Using Exponential Smoothing” by De Livera et al. (2011). This paper focuses on analyzing techniques to work with time series that involve complex seasonal changes (ie. month-to-month) such as Stock Prices. Understanding the concept of exponential smoothing was essential towards creating a model that is not only accurate but also stable/reliable for the long-term picture.

2.4 Relation to Existing Literature

As mentioned, the project builds on the techniques discussed in the research paper ”Forecasting Time Series With Complex Seasonal Patterns Using Exponential Smoothing” by De Livera et al. (2011). The paper introduces an innovative state space modeling framework for forecasting complex seasonal patterns using sophisticated techniques such as Box-Cox transformations and ARMA error correction. My project will aim to leverage these approaches to handle the complex seasonal patterns observed in stock price data, expanding on them by integrating generative AI insights.

This paper also focuses on analyzing techniques to work with time series that involve complex seasonal changes (ie. month-to-month) such as Stock Prices. Understanding the concept of exponential smoothing was essential towards creating a model that is not only accurate but also stable/reliable for the long-term picture. By understanding the practices mentioned in this paper I can gain a more comprehensive understanding of the factors influencing stock prices prediction from a technical perspective, which is essential for developing robust predictive models.

3 Novelty and Importance

3.1 Significance

The importance of this project lies in its potential to enhance the accuracy of stock price forecasts, which is critical for decision-making in the financial sector. Prediction of stocks is largely important in driving economic decisions in all industries. Many companies rely on accurate stock price prediction models in order to invest into the market/purchase other smaller firms. Moreover, the average consumer uses stock trading platforms, such as E-Trade, on a daily basis, meaning that accurate future trends will help them make an informed decision from the earliest of stages, thereby saving potential losses. The integration of generative AI will provide novel insights into the underlying factors driving stock price movements, making the analysis even more detailed and actionable since it will provide suggestions in plain English for the analyst to think over and review potential next steps. Accurate forecasting models can help investors optimize their portfolios, reduce risks, and capitalize on market opportunities, thereby improving overall financial performance.

3.2 Excitement and Innovation

The innovative aspect of this project is the combination of advanced time series forecasting techniques with the new rising technology of generative AI. This approach not only aims to improve forecast accuracy but also to offer deeper explanations of trends, which can be a valuable tool for both novice and experienced investors. By using AI to generate narrative explanations for the data, I can provide context that goes beyond mere numbers, helping prospective users understand the broader economic and market forces at play in the long-term picture.

As for excitement, anything and everything related to AI brings me joy to learn about and grow my skills. I am excited to learn about designing a full-stack project end-to-end and also honing in my data mangement and processing skills. Moreover, I believe that this project has some great potential once implemented and it will serve a useful purpose in the lives of many. With all these attributes I am excited to write this research paper and lay the foundation out of this project.

3.3 Existing Issues

Current data management practices in stock price forecasting often struggle with handling complex patterns such as sudden market changes and/or seasonal changes. Moreover, these models do not provide interpretable text explanations for trends and/or predictions. The techniques proposed in the referenced paper by De Livera et al. address these issues by offering a robust framework for forecasting and decomposition of complex time series. My project aims to build upon these techniques and incorporate additional insights from generative AI (as well as machine learning + neural networks to stay up-to-date with exact trends) to further enhance comprehension. By addressing the limitations of existing models and incorporating new technologies, I can develop a more holistic approach to stock price forecasting.

4 Progress and Contribution

4.1 Data Collection and Storage

I used a reputable historical Google stock price dataset from Kaggle[2]. The plan data is for the data to be stored in a relational database (e.g., MySQL) for structured data handling and retrieval. This ensures that the data is easily accessible and can be efficiently queried for analysis. For a brief overview of the attributes of this dataset, variables include: daily stock prices from 8/19/2004, trading volumes, and other relevant financial metrics (such as daily high/low prices). These aspects put together provide a comprehensive view of Google's market performance over time.

4.2 Data Cleaning and Transformation

In order for data to be ready for processing/analysis, data cleaning techniques must be employed to handle missing values, outliers, and ensure data quality. This can be done using Python libraries such as Pandas and NumPy. Moreover, feature engineering should be leveraged to create a better and more accurate model. Based on what I read from the study by De Livera et al, I believe a necessary step will also be to decompose the time series (data that is recorded over consistent intervals of time) to identify seasonal and trend components. All these steps are crucial as it will prepare the data for analysis by removing ”noise” and highlighting the underlying patterns that the models will use for forecasting.

While this is a theoretical project, I thought of sharing a simple data cleaning script that could be used to remove outlier and use basic feature engineering principles.

Listing 1: Data Cleaning Script

```
import pandas as pd
import numpy as np

# Load data
data = pd.read_csv('GOOGL.csv')

# Handle missing values
data.fillna(method='ffill', inplace=True)

# Remove outliers
q_low = data['Close'].quantile(0.01)
q_high = data['Close'].quantile(0.99)
data = data[(data['Close'] > q_low) & (data['Close'] < q_high)]

# Feature engineering
```

[2] https://www.kaggle.com/datasets/varpit94/google-stock-data/data

```
data['Date'] = pd.to_datetime(data['Date'])
data['Year'] = data['Date'].dt.year
data['Month'] = data['Date'].dt.month
data['Day'] = data['Date'].dt.day

# Save cleaned data
data.to_csv('cleaned_googl.csv', index=False)
```

4.3 Exploratory Data Analysis (EDA)

Before getting into code to tune my model, it's important to truly understand the data. This can be done using practices from Exploratory Data Analysis (EDA). I plan to visualize stock price trends, seasonality, and other patterns using Matplotlib and Seaborn. These steps will help to identify correlations between variables and extract potential insights which can help develop my model. EDA is a critical step towards understanding the data, uncovering hidden patterns, and generating hypotheses for further analysis. By visualizing the data, I can clearly identify trends, seasonal effects, and anomalies that might influence the stock prices. This will come handy while designing my model.

While I am not too familiar with Matplotlib and Seaborn below is a sample script I created based on the mentioned tips from an article from BuiltIn Tutorials. This source taught me basic commands using these two libraries and how to create simple graphs based on them. It was a good introduction to learn these two skills and in the future I plan to learn a lot more about them. Moreover, through looking through De Liveria et al.'s study I was able to learn how to leverage seasonal decomposition packages to break down the stock price time series by year. In the future, I plan to expand on the capabilites of these packages by utilizing more sophisticated approaches to handle time series such as Box-Cox transformations and ARMA error correction.

Listing 2: EDA Script

```
import matplotlib.pyplot as plt
import seaborn as sns
import pandas as pd
import numpy as np

data = pd.read_csv('GOOGL.csv')

# Plot closing price over time
plt.figure(figsize=(14, 7))
plt.plot(data['Date'], data['Close'])
plt.title('Google␣Stock␣Closing␣Price␣Over␣Time')
plt.xlabel('Date')
plt.ylabel('Closing␣Price')
plt.show()
```

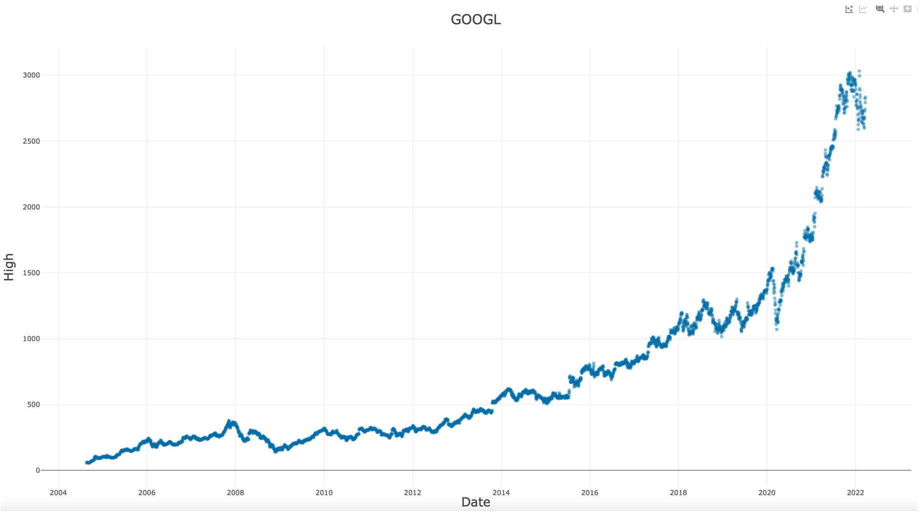

Figure 1: High Prices vs. Date

```
# Seasonal decomposition
from statsmodels.tsa.seasonal import seasonal_decompose

result = seasonal_decompose(data['Close'], model='multiplicative', period=365)
result.plot()
plt.show()
```

Figures 1 through 4 are some sample graphs that I generated from an online CSV compiler. These graphs illustrate trends in key attributes of the data:

4.4 Forecasting Models and Machine Learning Techniques

Moving on to the core of the project, I explore several time series forecasting and machine learning models, including ARIMA (AutoRegressive Integrated Moving Average), Prophet by Meta, and LSTM (Long Short-Term Memory). These models were all discussed equally in De Liveria et al.'s research paper. Each model has its strengths and weaknesses, and by comparing their performance, I can select the most appropriate one for our specific use case.

4.4.1 ARIMA (AutoRegressive Integrated Moving Average)

- **Strengths:**
  - Well-suited for short-term forecasting.
  - Effective in handling seasonality and linear trends.
  - Widely used.

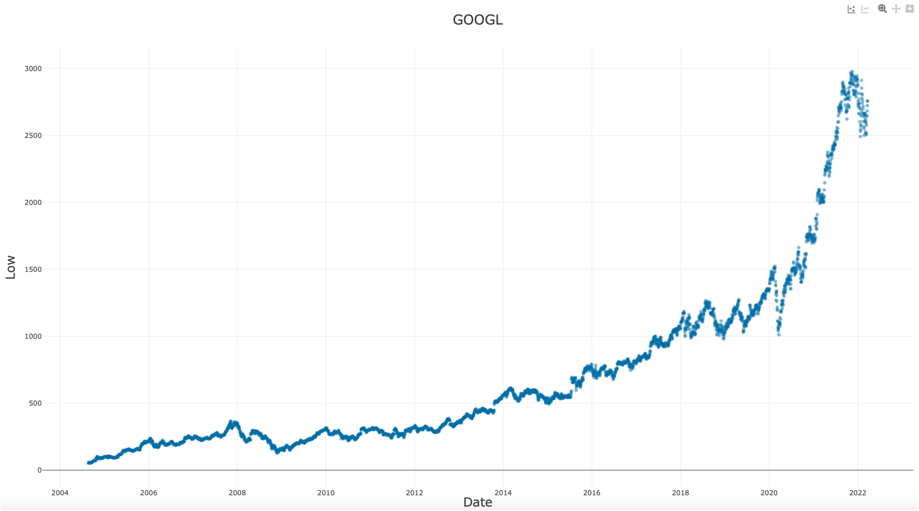

Figure 2: Low Prices vs. Date

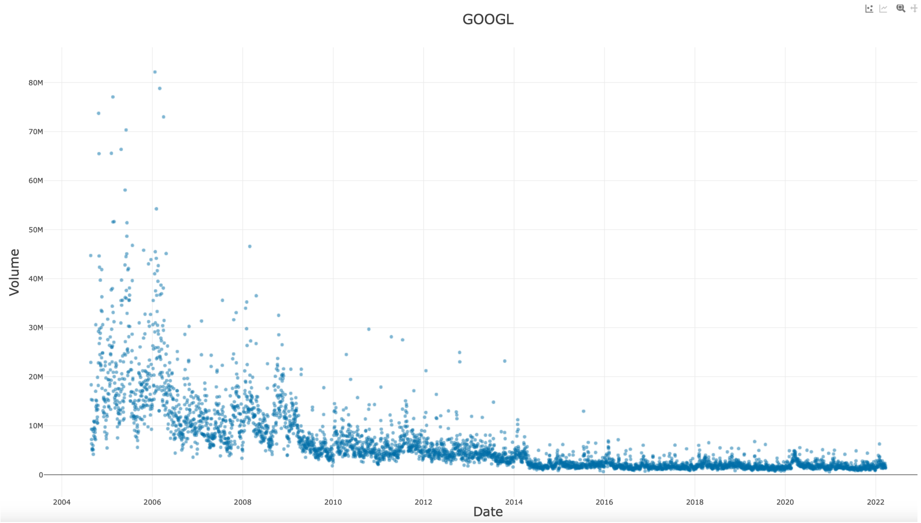

Figure 3: Volume vs. Date

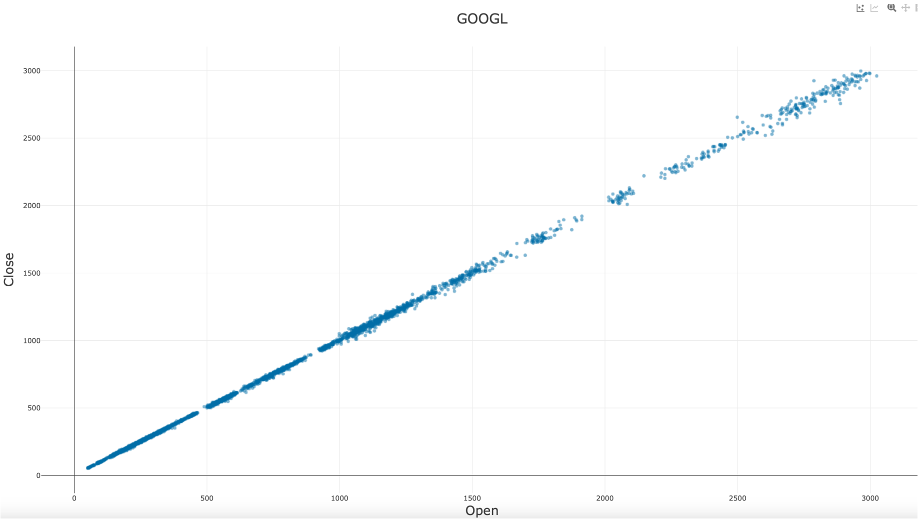

Figure 4: Close vs. Open Prices

  - Computationally efficient and interpretable.

- **Weaknesses:**
  - Assumes linear relationships, which may not capture complex patterns.
  - Limited in handling non-linear dependencies.
  - Requires careful parameter tuning for optimal performance (often difficult to do).

4.4.2 Prophet by Meta

- **Strengths:**
  - Robust to missing data and capable of handling outliers.
  - Automatically detects and incorporates seasonal effects and trends.
  - Easy to implement and interpret with minimal parameter tuning.
  - Suitable for time series with strong seasonal components.
- **Weaknesses:**
  - May not perform well with data that lacks strong seasonal patterns.
  - Can be less accurate for very short-term forecasts compared to ARIMA.
  - Requires a large amount of historical data for optimal performance.

4.4.3 LSTM (Long Short-Term Memory)

- **Strengths:**
  - Capable of capturing complex patterns and long-term dependencies.
  - Effective in modeling non-linear relationships in time series data.
  - Suitable for datasets with significant temporal dynamics.
- **Weaknesses:**
  - Computationally intensive and requires significant training time.
  - Requires a large amount of data and computational resources.
  - More challenging to interpret and tune compared to ARIMA and Prophet.

4.5 Model Selection

Based on the comparison of strengths and weaknesses, the Prophet model was selected for the following reasons:

- **Handling Seasonality:** Prophet effectively captures seasonal patterns and can incorporate holiday and current trend effects, which are relevant for stock price data.
- **Long-Term and Short-Term Forecasting:** Prophet is capable of providing both short-term and long-term forecasts, making it suitable for our project's dual objectives.
- **Robustness to Missing Data:** Prophet's ability to handle missing data and outliers enhances its applicability to real-world financial data (where not everything may be readily available).
- **Ease of Use and Interpretability:** The model is relatively easy to implement and interpret, with minimal parameter tuning required as per sources online.

While LSTM offers the potential to capture more complex patterns, the computational resources required and the project's need for both short-term and long-term forecasting led to the selection of Prophet as the most appropriate model. Prophet balances the trade-offs between ease of use, robustness, and the ability to handle complex seasonality and trend patterns effectively, which is exactly what I am looking for.

4.6 Model Evaluation

In order to evaluate models my models performance I plan metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE). This step ensures that the chosen model provides accurate and reliable forecasts. By comparing these metrics across different models, I can determine which one performs best on our dataset and provides the most reliable predictions.

5 Model Evaluation Metrics

To evaluate the models' performance, I will use the following metrics:

- **Mean Absolute Error (MAE)**:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{1}$$

  - **Purpose:** Measures the average magnitude of errors.
  - **Insight:** Lower MAE indicates predictions closer to actual values.
  - **Strength:** Robust to outliers, treating all errors equally.

- **Mean Squared Error (MSE)**:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{2}$$

  - **Purpose:** Highlights larger errors due to squaring.
  - **Insight:** Lower MSE indicates smaller average squared errors.
  - **Strength:** Useful for identifying significant deviations.

- **Root Mean Squared Error (RMSE)**:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{3}$$

  - **Purpose:** Provides error magnitude in the same units as data.
  - **Insight:** Lower RMSE signifies close predictions with less variance.
  - **Strength:** Sensitive to large errors, highlighting models with significant outliers.

By comparing these metrics across different models, I can determine which one performs best on our dataset and provides the most reliable predictions. Each metric offers a unique perspective on model performance:

- **MAE:** Best for understanding average error magnitude.
- **MSE:** Highlights models with significant deviations.
- **RMSE:** Combines error magnitude and variance for an overall comprehensive view.

5.1 Generative AI Insights and Machine Learning/Neural Networks

My plan is to integrate the open source OpenAI's GPT-4o API to generate narrative insights that provide context and explanations for the trends and forecasts. The implementation involves feeding the forecasted data into the GPT-4 API, which will generate comprehensive narratives based on the observed patterns. These narratives will be automatically updated as new data is processed, ensuring that real-time updated/explanations are provided. Additionally, I will develop a user interface that displays these narratives alongside the visualized forecasts, making it easy for users to interpret the results and make informed decisions. This integration not only enhances the analytical depth of the project but also makes it more accessible and user-friendly. This qualitative layer will help users understand the factors driving stock price movement——highlighting potential causes and implications of observed trends.

Moreover, the machine learning and neural network models will be updated in real-time to stay aligned with the latest data. This will be achieved through continuous retraining of the models with newly available stock price data. This approach ensures that the models remain accurate and reflective of the most current market trends, providing users with up-to-date and reliable forecasts (which can be fed into the generative AI model to provide rationale for these current updates).

6 Changes After Proposal

Overall, the final project does follow closely the initial proposal with minor adjustments to the models used and the revamped potentials/ideas of including generative AI insights. The initial plan I had in mind was to use ARIMA models but it was found that Prophet would be better after doing research. Some bottlenecks included handling the large dataset and exploring trends that are present in such a large file Additionally, the seasonal time series components were a bit tricky to understand at first, however with research (ie. De Liveria et al.'s study) I was able to understand their basic principles and capabilities to really enhance the project.

7 Conclusion

This project demonstrates the potential application of advanced forecasting techniques and generative AI to analyze Google stock trends. Future work could involve exploring more sophisticated models and expanding the machine learning capabilities that are leveraged. By combining quantitative and qualitative approaches, this project offers a comprehensive tool for understanding and predicting stock price movements, providing valuable insights for investors and analysts alike. I believe that this paper introduced me nicely to the theoretical framework behind creating such a project, and as I gain more expertise I hope to bring this idea to fruition in the near future.