TRIBHUVAN UNIVERSITY
INSTITUTE OF ENGINEERING
PULCHOWK CAMPUS

A
REPORT
ON
INTEGRATED ANALYSIS OF STOCK PRICE PREDICTION

SUBMITTED BY:
SAGAR SHARMA(PUL076BCT061)
SAILESH SHIWAKOTI (PUL076BCT063)
SAMIKSHYA CHAPAGAIN(PUL076BCT064)

SUBMITTED TO:
DEPARTMENT OF ELECTRONICS & COMPUTER ENGINEERING

November, 2023

Acknowledgements

We would like to extend our heartfelt gratitude to numerous individuals and organizations who have been instrumental in our academic and professional journey, especially in the field of data science and computer engineering. Their unwavering support, guidance, and contributions have profoundly shaped our path and enabled our growth and success.

Foremost, we express our sincere thanks to the Department of Electronics and Computer Engineering at Pulchowk Campus. The department's commitment to providing top-notch education and resources has played a pivotal role in our academic development. The dedicated professors and staff have not only shared their knowledge but also instilled in us a deep passion for learning and innovation.

Our journey in data science was greatly enriched by the "Applied Data Science" course offered by Samsung Innovative Campus. We acknowledge Samsung for providing this transformative program that significantly enhanced our data science skills and knowledge.

We extend our appreciation to Arun K. Timalsena, Suresh Pokharel, and Basanta Joshi for their exceptional teaching and mentorship in the fields of artificial intelligence and machine learning. Their guidance has been invaluable to us.

Lastly, we wish to express our gratitude to all our friends and family who have supported us throughout our academic and professional endeavors. Their unwavering encouragement and belief in our abilities have been a constant source of motivation.

Thank you to all those who have been a part of our journey and contributed to our growth. We are truly grateful for your support and mentorship.

Abstract

In an era defined by the relentless flow of information and the ever-evolving landscape of financial markets, the ability to make informed investment decisions has become paramount. This project embarks on a journey to harness the formidable power of deep learning, advanced machine learning, and predictive analytics to offer precise and insightful forecasts of stock prices for a specific company.

Our approach integrates a multifaceted data-driven methodology that incorporates historical share prices, technical indicators (e.g., Relative Strength Index - RSI), and key financial indicators (e.g., Price-to-Earnings ratio - P/E ratio). Additionally, we incorporate sentiment analysis of news related to the selected company and macroeconomic factors, such as the inflation rate.

By leveraging sophisticated algorithms ARIMA, SARIMAX and neural networks such as LSTM, our primary objective is to provide investors, financial analysts, and stakeholders with a cutting-edge tool for peering into the future performance of the chosen company's stock.

Through this innovative approach, we aim to empower investors with the knowledge and foresight needed to make well-informed investment decisions, optimize trading strategies, and cultivate a profound understanding of the intricate dynamics that underpin financial markets.

Contents

List of Abbreviations

MSE	Mean Square Error
RSI	Relative Strength Index
RNN	Recurrent Neural Network
LSTM	Long Short-Term Memory
RMSE	Root Mean Square Error
ARIMA	AutoRegressive Integrated Moving Average
SARIMA	Seasonal AutoRegressive Integrated Moving Average
P/E Ratio	Profit-Earn Ratio

1. Introduction

1.1 Background

This project centers around the utilization of machine learning and deep learning techniques to predict stock prices for a specific company. By harnessing advanced algorithms and predictive models, we aim to bridge the gap between historical financial data and future stock price trends.

Stock price prediction is crucial for investors, traders, and financial analysts as it provides valuable insights into market behavior and aids in making informed investment decisions. However, accurately forecasting stock prices is challenging due to the numerous factors influencing financial markets.

Despite these challenges, this project holds promise for applications like optimizing investment strategies, risk management, and financial market analysis. By integrating machine learning, data analysis, and financial expertise, it aims to empower stakeholders with predictive insights, facilitating more effective decision-making in stock trading and investment.

1.2 Objectives

The primary objectives of this project are as follows:

1. To develop ARIMA and SARIMA models and implement a Long Short-Term Memory (LSTM) neural network capable of generating accurate and coherent stock price predictions for a specific company, leveraging historical financial data and relevant market indicators.

2. To provide investors, traders, and financial analysts with a valuable tool for making informed decisions, optimizing trading strategies, and managing investment portfolios effectively.

3. To enhance the understanding of stock market dynamics and contribute to the field of financial forecasting through empirical research and data-driven insights.

1.3 Scope

The scope of this project includes:

1. Development and implementation of machine learning and deep learning techniques, including ARIMA, SARIMA, and LSTM models, to forecast stock prices for a specific company.
2. Analysis of the particular company's trend and historical analysis of stock price.
3. Utilization of diverse data sources, including financial, technical, macroeconomic, and financial news data, along with sentiment analysis.
4. Data normalization and input preparation using various techniques.
5. Hyperparameter tuning for model improvement.
6. Prediction of closing prices as the target variable.
7. Model evaluation and comparison using metrics such as RMSE, MAPE.

2. Literature Review

2.1 Related Work

In 1993, Pradhan[1] conducted an in-depth analysis of the Nepali stock market, focusing on the interrelationships between various financial metrics and liquidity, leverage, profitability, turnover, and interest coverage. This study was based on a dataset comprising 17 enterprises whose stocks were listed on the Stock Exchange Center and actively traded in the stock market.

Moving forward to 2004, Gurung [2] undertook a comprehensive examination of the growth trends and performance of the Nepal Stock Exchange (NEPSE). This investigation encompassed critical aspects such as the number of listed and traded companies, trading volumes, market capitalization, and the behavior of the NEPSE index. The study observed that market performance indicators displayed a lack of synchronization but exhibited erratic trends during the studied period.

Pun and Shahi [3] in 2018, ventured into the realm of stock price prediction by implementing Support Vector Regression and Artificial Neural Networks. Their study was structured around dividing NEPSE data into ten investment sectors, and the performance of each sector was meticulously evaluated using various assessment metrics.

In 2019, Saud and Shakya [4] contributed to stock price prediction accuracy through the examination of Gated Recurrent Unit (GRU) models under different optimization techniques, including momentum, RMSProp, and Adam. The study concluded that the GRU model with Adam optimization exhibited superior accuracy and consistency.

In 2020, Saud and Shakya [5] expanded their research to encompass a comparative analysis of deep learning techniques, including Vanilla RNN, Long Short-Term Memory (LSTM), and GRU, in predicting the next day's closing price. Their study considered variables such as fundamental stock data and various technical indicators for two commercial banks. The results indicated that GRU and LSTM outperformed Vanilla RNN in the context of stock price prediction.

In a study conducted by Pokhrel et al.[6], a comparative analysis of three deep learning models—Long Short-term Memory (LSTM), Gated Recurrent Unit (GRU), and Convolutional Neural Network (CNN)—in predicting the next day's closing price of the Nepal Stock Exchange (NEPSE) index. The study used a set of sixteen predictors encompassing fundamental market data, macroeconomic data, technical indicators, and financial text data. The

LSTM model demonstrated superior predictive accuracy, as evidenced by standard assessment metrics such as Root Mean Square Error (RMSE)

In their work, Sun et al. [7] focused on pre-processing online financial text for sentiment classification using a natural language processing approach.

2.2 Related Theory

Time series analysis is a vital statistical technique employed for the examination of data collected over successive time intervals. This method plays a pivotal role in recognizing patterns, trends, and seasonality within the data, ultimately facilitating the generation of forecasts and predictions. Commonly utilized models in time series analysis encompass the ARIMA (AutoRegressive Integrated Moving Average) model, which is adept at capturing autocorrelation and stationarity in time series data. ARIMA is characterized by three principal components: AutoRegressive (AR), Integrated (I), and Moving Average (MA). It is particularly suitable for data exhibiting discernible trends and seasonality. To address seasonality explicitly, the Seasonal ARIMA (SARIMA) model extends ARIMA by introducing supplementary seasonal terms. SARIMA effectively handles time series data featuring repeating patterns. In scenarios where external variables influence the time series data, the SARIMAX (Seasonal ARIMA with Exogenous Variables) model can be employed to incorporate these exogenous factors into the analysis.

2.2.1 ARIMA: AutoRegressive Integrated Moving Average

ARIMA, which stands for AutoRegressive Integrated Moving Average, is a widely used time series forecasting method. It combines three key components: AutoRegressive (AR), Integrated (I), and Moving Average (MA) to model and forecast time series data.

Components of ARIMA

1. **AutoRegressive (AR):** The AR component models the relationship between the current value in the time series and its past values. It represents the "auto-regressive" nature of the data, meaning that the current value depends linearly on previous values.

2. **Integrated (I):** The I component represents the differencing of the time series data. It is used to make the data stationary, which means that the statistical properties (e.g., mean and variance) remain constant over time. The parameter 'd' determines the order of differencing.

3. **Moving Average (MA):** The MA component models the relationship between the current value and past white noise (error) terms. It captures short-term dependencies

in the data. The parameter 'q' determines the order of the MA process.

Box-Jenkins Methodology

ARIMA modeling often follows the Box-Jenkins methodology, which includes the following steps:

1. **Identification (I):** Determine the order of differencing 'd' needed to make the data stationary. Identify potential values for 'p' (AR order) and 'q' (MA order) based on autocorrelation and partial autocorrelation plots.

2. **Estimation (E):** Use maximum likelihood estimation to estimate the model parameters (p, d, q) and fit the ARIMA model to the data.

3. **Diagnostic Checking (C):** Evaluate the model's goodness of fit by examining residual plots and conducting statistical tests. Refine the model if necessary.

4. **Forecasting (F):** Once the model is validated, use it for forecasting future values of the time series.

The ARIMA(p, d, q) model can be represented by the following equation:

$$Y_t = c + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \ldots + \phi_p Y_{t-p} - \theta_1 e_{t-1} - \theta_2 e_{t-2} - \ldots - \theta_q e_{t-q} + e_t$$

Where: - Y_t is the value of the time series at time 't'. - c is a constant term. - $\phi_1, \phi_2, \ldots, \phi_p$ are the AR coefficients (parameters). - e_t is the white noise error at time 't'. - $\theta_1, \theta_2, \ldots, \theta_q$ are the MA coefficients (parameters).

This equation represents the relationship between the current value Y_t and its past values, as well as the past white noise errors.

2.2.2 SARIMA: Seasonal AutoRegressive Integrated Moving Average

SARIMA, or Seasonal AutoRegressive Integrated Moving Average, is an extension of the ARIMA model that includes seasonality components. It is a powerful time series forecasting method commonly used when data exhibits seasonal patterns.

Components of SARIMA

SARIMA combines the same three key components as ARIMA (AutoRegressive, Integrated, and Moving Average) with the addition of seasonal components:

1. **AutoRegressive (AR):** The AR component models the relationship between the current value in the time series and its past values. It represents the "auto-regressive" nature of the data.

2. **Integrated (I):** Similar to ARIMA, the I component represents differencing to make the data stationary.

3. **Moving Average (MA):** The MA component models short-term dependencies in the data.

4. **Seasonal AutoRegressive (SAR):** The SAR component accounts for seasonal patterns by considering the relationship between the current value and past values at seasonal lags.

5. **Seasonal Integrated (SI):** The SI component represents seasonal differencing, which is essential to remove seasonal trends.

6. **Seasonal Moving Average (SMA):** The SMA component models short-term dependencies in the seasonal component of the data.

2.2.3 RNN: Recurrent Neural Networks for Stock Price Prediction

Recurrent Neural Networks (RNNs) are a class of artificial neural networks well-suited for sequential data, making them valuable for time series forecasting tasks such as stock price prediction.

RNNs are designed to maintain a hidden state that captures information from previous time steps. The theory behind RNNs is based on the following principles:

1. Hidden State: RNNs maintain a hidden state h_t at each time step t. This hidden state can capture information from previous time steps and is updated at each new input.

2. Recurrent Connection: RNNs have recurrent connections, allowing information to flow from one time step to the next. This recurrent connection enables the network to model sequential dependencies.

3. Parameter Sharing: RNNs share the same set of weights across all time steps. This shared weight structure enables the network to learn patterns and relationships in sequential data.

The output of an RNN at each time step for stock price prediction is obtained by using a fully connected layer with linear activation to produce a continuous output value representing the predicted stock price at the next time step.

2.2.4 LSTM: Long Short-Term Memory Networks

LSTM, or Long Short-Term Memory, is a type of recurrent neural network (RNN) designed to address the vanishing gradient problem and capture long-range dependencies in sequential data.

- Cell State (C_t): LSTMs maintain a cell state that can carry information across long sequences. This cell state is regulated through a series of gates.
- Gates: LSTMs have three gates—forget gate (f_t), input gate (i_t), and output gate (o_t). These gates control the flow of information into and out of the cell state, allowing LSTMs to selectively remember or forget information.
- Hidden State (h_t): LSTMs also maintain a hidden state that is updated at each time step. The hidden state is influenced by the cell state and the current input.
- Activation Functions: LSTMs use activation functions such as sigmoid and hyperbolic tangent (tanh) to regulate the values of gates and hidden state.

1. Forget Gate (f_t):

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

2. Input Gate (i_t):

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

3. Candidate Cell State ($\tilde{C}_t$):

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$

4. Update Cell State (C_t):

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t$$

5. Output Gate (o_t):

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$

6. Hidden State (h_t):

$$h_t = o_t \cdot \tanh(C_t)$$

Where: - f_t, i_t, and o_t are the values of the forget gate, input gate, and output gate, respectively. - $\tilde{C}_t$ represents the candidate cell state. - C_t is the updated cell state. - h_t is the hidden state. - x_t is the input at time t. - W_f, W_i, W_o, W_C are weight matrices, and b_f, b_i, b_o, b_C are bias vectors. - σ is the sigmoid activation function, and tanh is the hyperbolic tangent activation function.

3. Methodology

3.1 Data Source

Our stock market data was meticulously sourced from *Sharesansar*[1], a leading financial platform renowned for its comprehensive and reliable data on the Nepalese stock market. *Sharesansar* has established itself as a dependable source for real-time stock market information, offering a wide range of data points, including stock prices, trading volumes, financial reports, and market indices. The platform's reputation for accuracy and consistency in data delivery makes it an ideal choice for our research. Given our focus on Nabil Bank, one of Nepal's prominent financial institutions, *Sharesansar*'s data serves as an essential resource, aligning seamlessly with the objectives of our analysis.

3.2 Dataset Preparation

Our dataset was meticulously prepared, categorized into four distinct segments, each contributing to the holistic analysis of Nabil Bank's performance.

3.2.1 Fundamental Data

In the fundamental category, we collected essential metrics, including opening prices, daily lows, highs, and closing prices for Nabil Bank. These metrics serve as the core elements for understanding the stock's performance over time.

3.2.2 Macroeconomic Indicators

Our dataset includes macroeconomic data points, particularly remittance figures and the inflation rate, which are integral in assessing the broader economic context in Nepal and their impact on Nabil Bank's stock performance.

3.2.3 Technical Indicators

Technical indicators play a crucial role in our analysis. We included Moving Average Convergence Divergence (MACD) and Bollinger Bands bandwidth as indicators to analyze stock price trends and volatility. These tools help identify potential buy or sell signals, enhancing the technical perspective of our analysis.

[1] https://www.sharesansar.com/

3.2.4 Financial News Sentiment

To complement our quantitative analysis, we incorporated sentiment scores from financial news articles relevant to Nabil Bank. This qualitative dimension allows us to capture market sentiment and investor perception, providing a comprehensive view of factors influencing stock performance.

The dataset consists of data from 2018-05-01 to 2023-09-30.

3.3 Dataset Preprocessing

Data preprocessing is a vital step for ensuring a clean, complete, and well-structured stock market dataset, enhancing the reliability and validity of the analysis.

3.3.1 Feature Engineering

Feature engineering is a pivotal step in preparing our stock market dataset for analysis. It involves creating new variables that can help us capture critical patterns in the data. One of the central features we've engineered is the Bollinger Bands bandwidth, which plays a significant role in our analysis. The Bollinger Bands are constructed using moving averages and standard deviations, and the bandwidth represents the width between the upper and lower bands. This feature aids in identifying stock price volatility patterns, which are valuable for understanding market dynamics and making informed decisions.

3.3.2 Simple Moving Average

The 10-day Simple Moving Average (SMA) and 30-day SMA are widely used technical indicators in financial analysis. The 10-day SMA calculates the average of a stock's closing prices over the past 10 days, offering a short-term perspective and helping to reduce short-term price fluctuations. It's often used for identifying short-term trends and trading signals. In contrast, the 30-day SMA calculates the average over the last 30 days, providing a medium-term view that is smoother and less responsive to short-term volatility. This makes it suitable for spotting medium-term trends and making longer-term investment decisions. These moving averages are used to identify trends and crossovers, aiding traders and investors in their decision-making processes within the stock market.

3.3.3 Handling Missing Values

Managing missing data is a common challenge in dataset preparation. In our case, to maintain data continuity and ensure that our analysis isn't disrupted by gaps in the dataset, we've implemented a specific strategy. When we encountered missing values, we opted to fill them with the stock's value from the previous day. This approach allows us to maintain a seamless time series dataset, which is crucial for time-sensitive stock market analysis. It ensures that our analysis is based on a consistent and complete dataset.

3.3.4 Monthly Macroeconomic Data

Our dataset encompasses macroeconomic data with a monthly reporting frequency, such as remittances and inflation. Aligning this monthly data with our daily stock market dataset is a necessary step to effectively incorporate macroeconomic factors into our analysis. To achieve this, we've taken the approach of applying the monthly values to every day within the corresponding month.

3.3.5 Compound Sentiment Score

We added compound sentiment scores to the news we collected about the Nabil bank using VADER tool.

3.4 Exploratory Data Analysis (EDA)

Exploratory Data Analysis is a crucial step in time series analysis that helps you understand the underlying patterns, characteristics, and anomalies in time series data.

The target variable "Close Price" was plotted to view the trend and seasonality.

Figure 3.1: Original Close Price Data

The graph depicted both increasing and decreasing trends during different periods, and no clear seasonality was viewed.

The simple moving average method is used for smoothing out fluctuations in data to identify trends or patterns over time and 10-day and 30-day simple moving averages (SMA) for a given stock's closing prices was plotted accordingly.

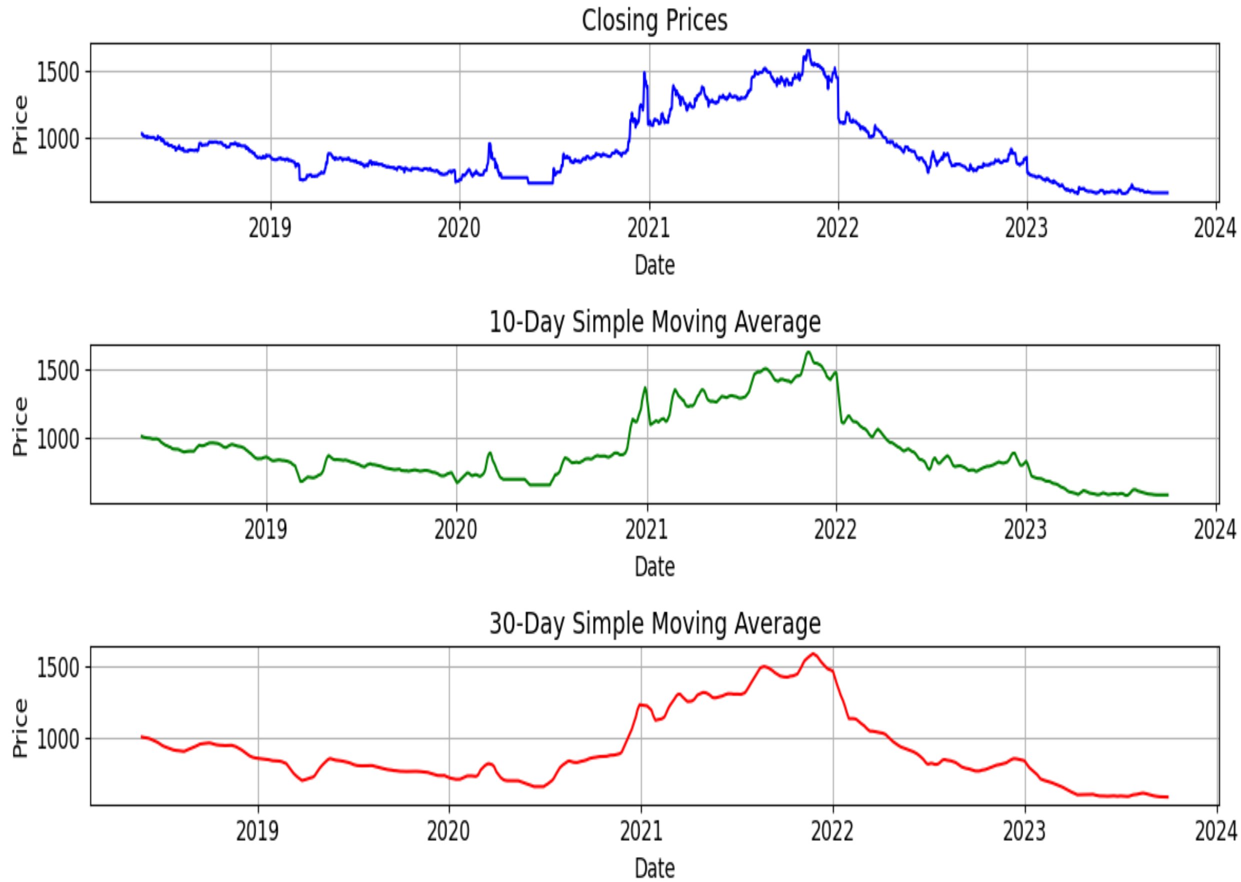

Figure 3.2: Simple Moving Average

Next, we analyze the distribution of stock prices using a histogram plot.

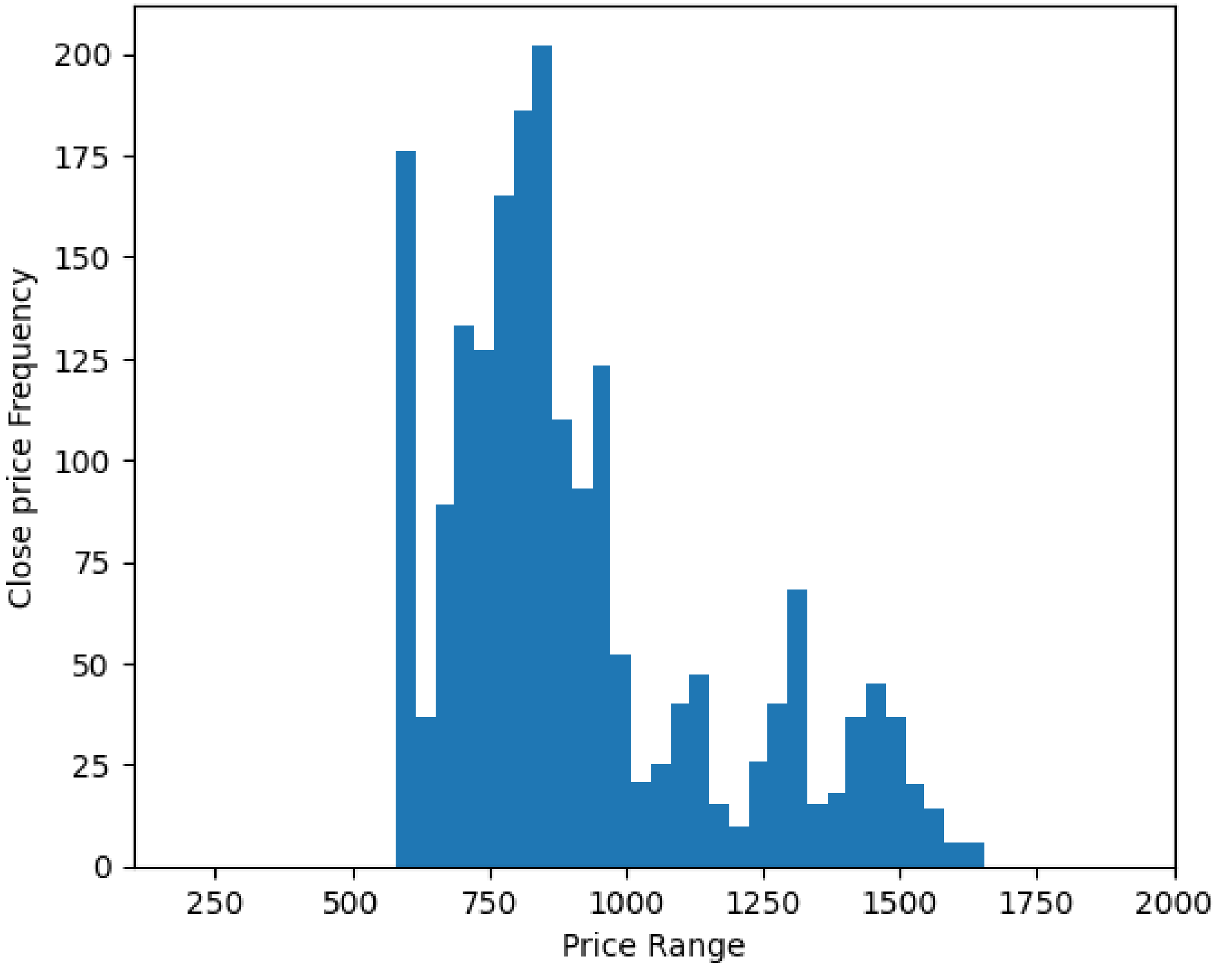

Figure 3.3: Stock Price Histogram

The histogram plot reveals that stock prices typically range from 500 to 1700 with a mean around 750.

We also use a boxplot to gain insights into the distribution of stock prices over different years.

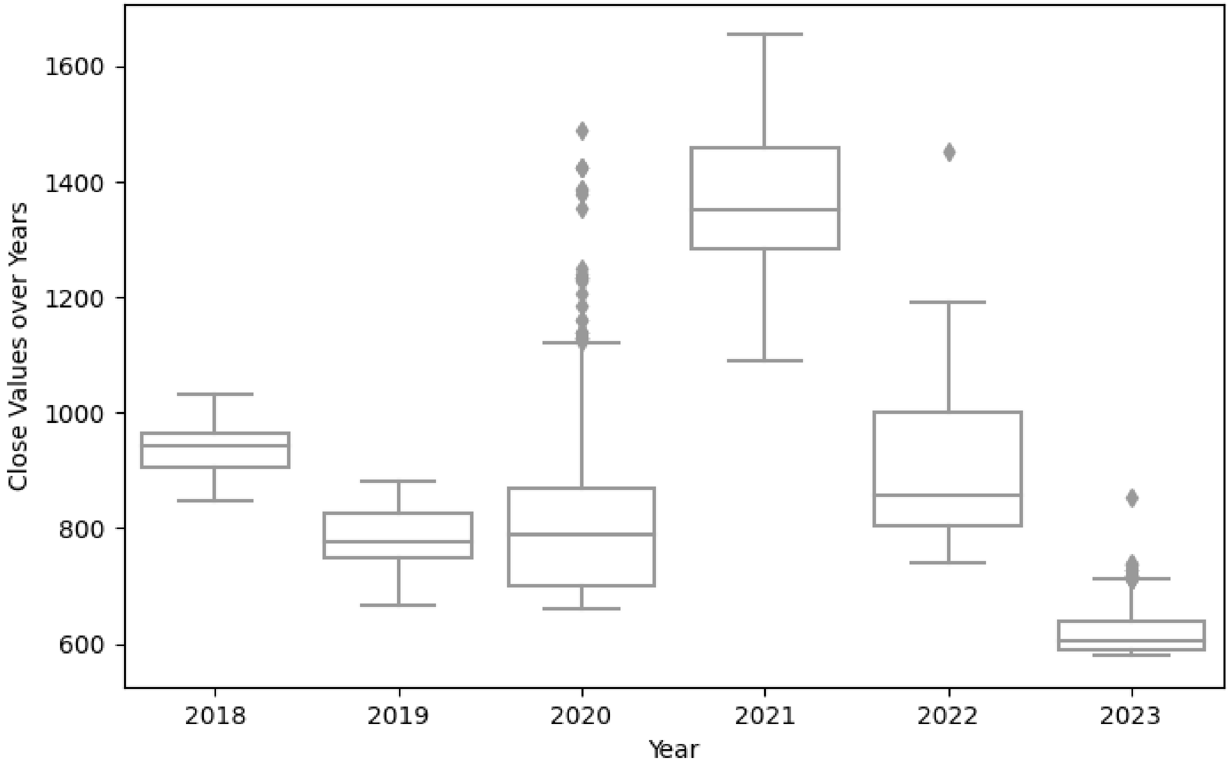

Figure 3.4: Stock Price Boxplot Analysis

In the boxplot, it's evident that the year 2020 had many observations greater than the Q3 values. Towards the end of 2020, the stock price started increasing significantly, leading to a peak price higher than that of other years.

A heatmap was generated to view the correlation between the different features in the dataset.

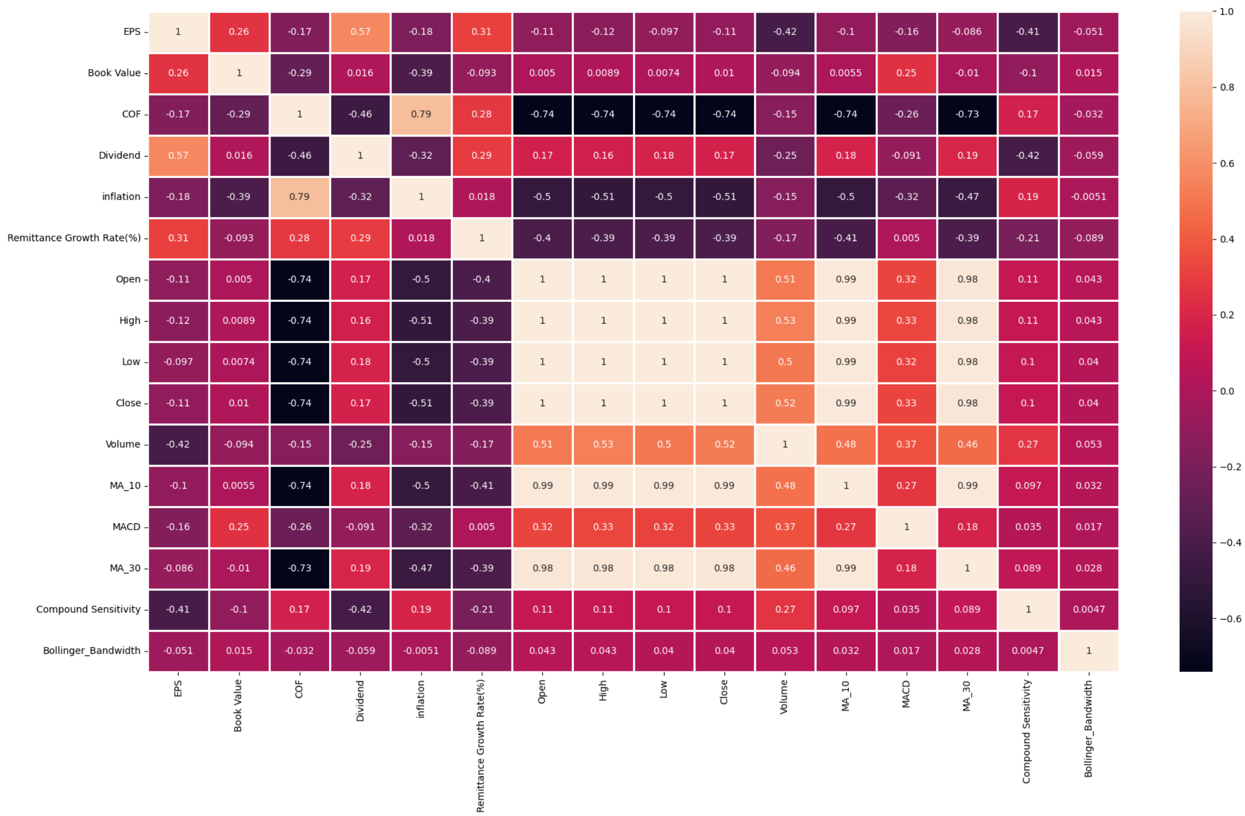

Figure 3.5: Heatmap Representation

3.5 Model Development

Two key models were used for development: the traditional ARIMA model and the deep learning LSTM model. ARIMA was employed for its ability to analyze historical trends, while LSTM was chosen to capture complex, non-linear relationships in stock market data. The inclusion of both models reflects a comprehensive and balanced approach to stock market analysis, utilizing the strengths of classical statistical methods and cutting-edge deep learning techniques.

3.6 Predictions

In the final stages of the project, the focus shifted to fine-tuning the models and making accurate predictions. This involved a thorough optimization process to enhance model performance. Fine-tuning encompassed parameter adjustments, hyperparameter optimization, and extensive model validation to ensure the best possible fit to the data.

4. Results

In this section, we present the results of our analysis, including the performance metrics of different models used for predicting stock prices.

4.1 ARIMA and SARIMAX

In this section, we delve into the analysis using ARIMA (AutoRegressive Integrated Moving Average) and SARIMAX (Seasonal ARIMA with Exogenous variables) models. We aim to understand the underlying patterns in stock price data, assess stationarity, and make predictions.

4.1.1 Original Price Data

To start our analysis, we first look at the original stock price data. Figure 1 illustrates the trend of Nabil Bank's stock prices over time.

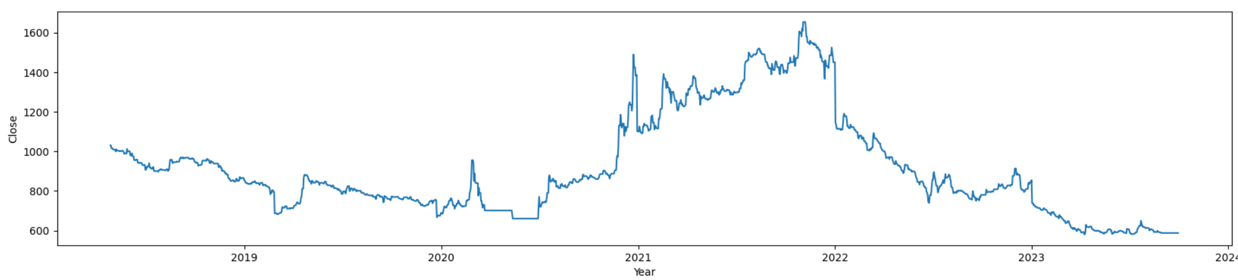

Figure 4.1: Original Stock Price Data

In Figure 1, we observe peaks and valleys in the data. Notably, there is a peak in the third portion of the data, and clear valleys are visible. Certain constant data points indicate missing prices during COVID-related market disruptions.

4.1.2 Testing for Stationarity

Before building ARIMA and SARIMAX models, it's essential to test the stationarity of the stock price data. We employ the Augmented Dickey-Fuller test.

Augmented Dickey-Fuller Test Results:

- Test Statistic: -1.320689
- p-value: 0.619655
- Lags: 7
- Observations: 1975
- Critical Value (1%): -3.433665
- Critical Value (5%): -2.863005
- Critical Value (10%): -2.567550

The test statistic (-1.320689) exceeds the critical values, indicating that the stock price data is not stationary and requires differencing.

After differencing the data (d=1), we observe stationarity. However, we further test with d=2, d=3, and d=4 to determine if additional differencing improves model performance.

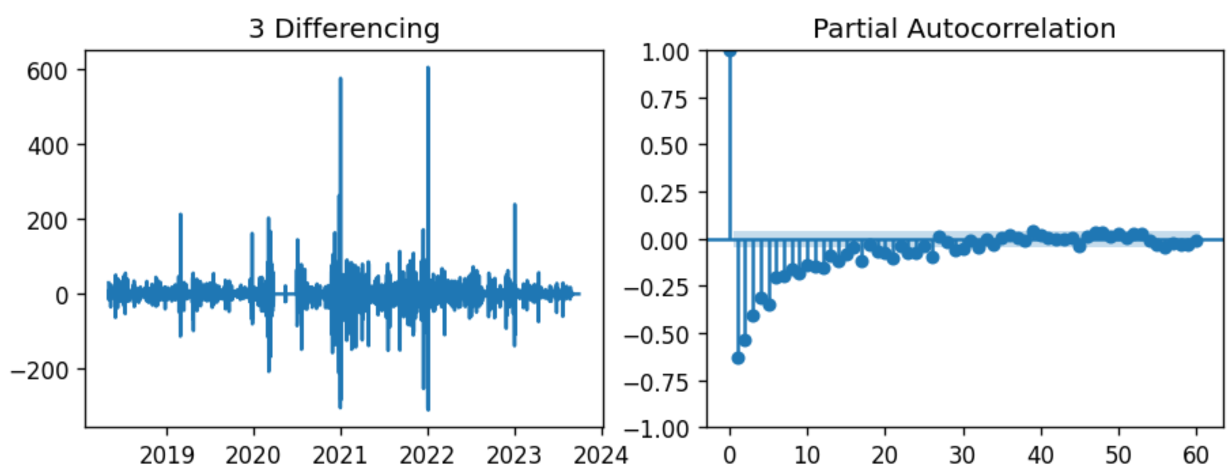

Figure 4.2: Differenced Data and Auto-correlation Plot

4.1.3 Model Testing and Prediction

We employ various parameter combinations for ARIMA and SARIMAX models to make predictions. The following figures illustrate the results of our model testing and predictions:

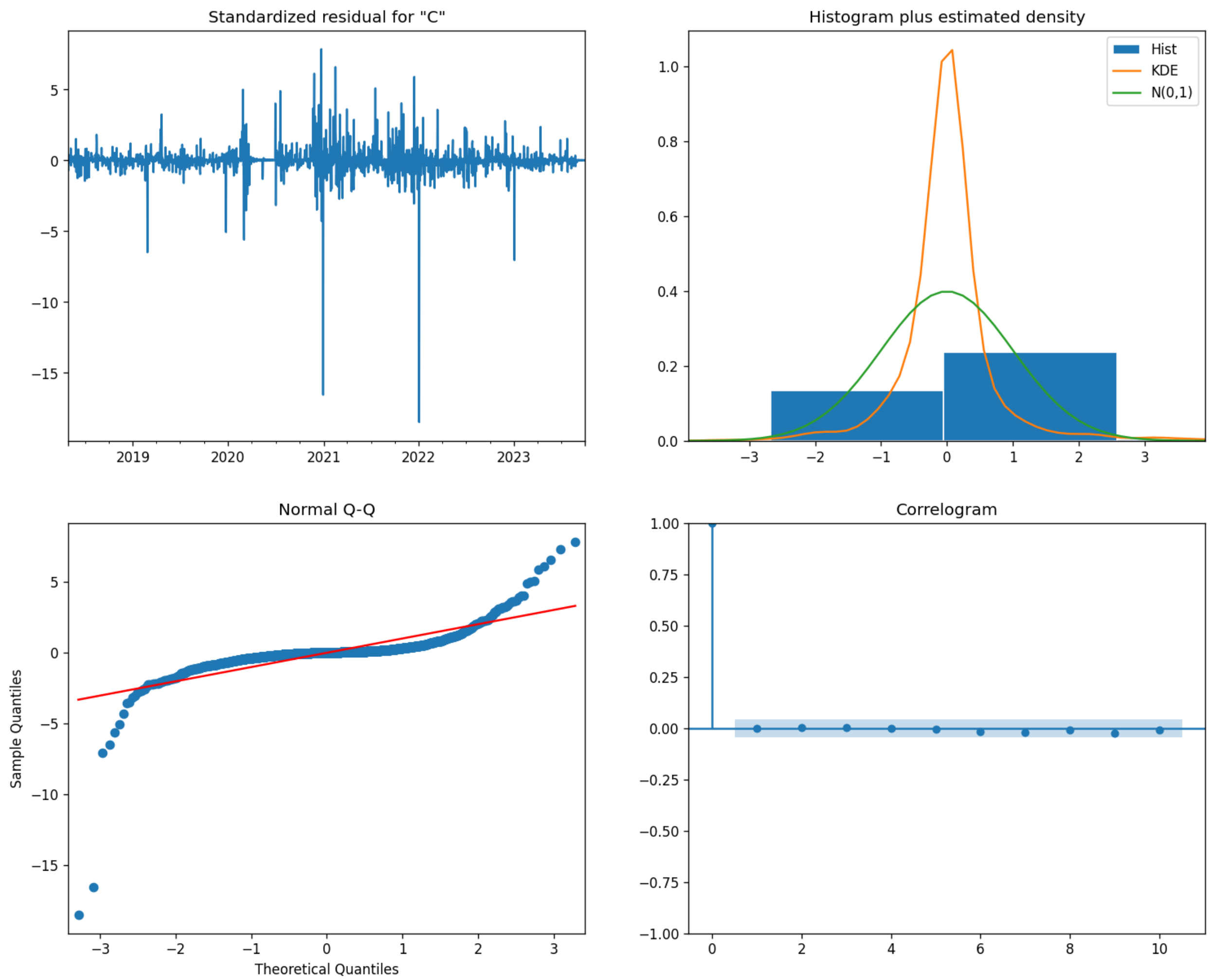

Figure 4.3: Residual Plot

Figure 4.4: Stock Price Forecasting

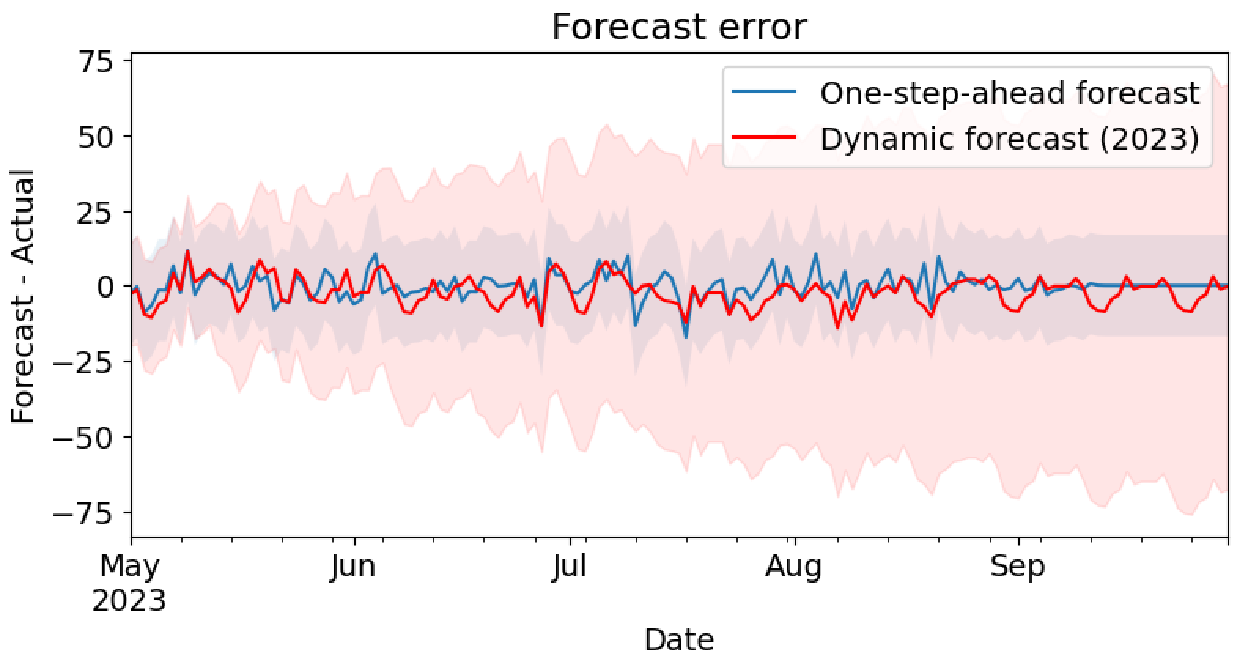

Figure 4.5: Forecast Error

4.2 Long Short Term Memory(LSTM)

In this section, we delve into the analysis using LSTM. LSTM models are a type of Recurrent Neural Network (RNN) that can be used for time series analysis and prediction. They are particularly useful for retaining and capturing complex patterns in the sequential data. By utilizing different input features, the model learns to identify the patterns and trends in the stock market data over time.

4.2.1 LSTM Model

The LSTM model was developed and tested for different hyperparameters to find the best performance metrics. The LSTM model has 8 LSTM units, a dropout rate of 0.1, a regularization rate of 0.001, and was trained for 120 epochs with a batch size of 5.

4.2.2 Training and Validation

The dataset was divided as train,validation and test set. 65% of the data was used for training, 15% for the validation and 20% was used as the testing data.

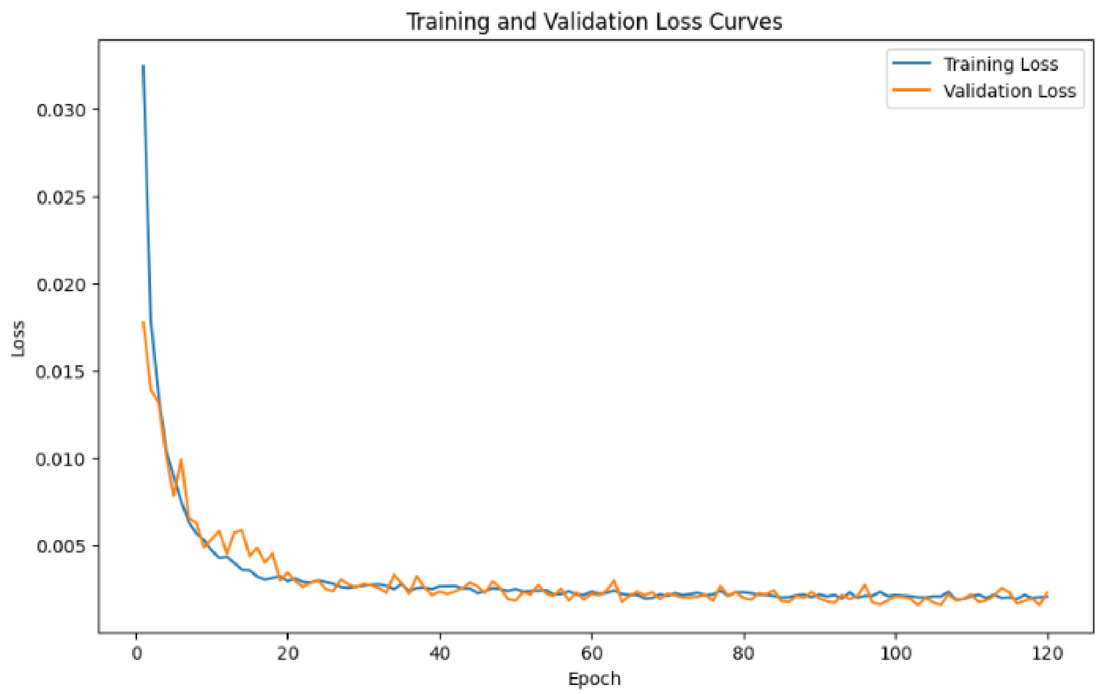

Figure 4.6: Training and Validation Loss

4.2.3 Prediction Graph

The predictions from the model was analyzed with the actual test data for the above mentioned hyperparameters.

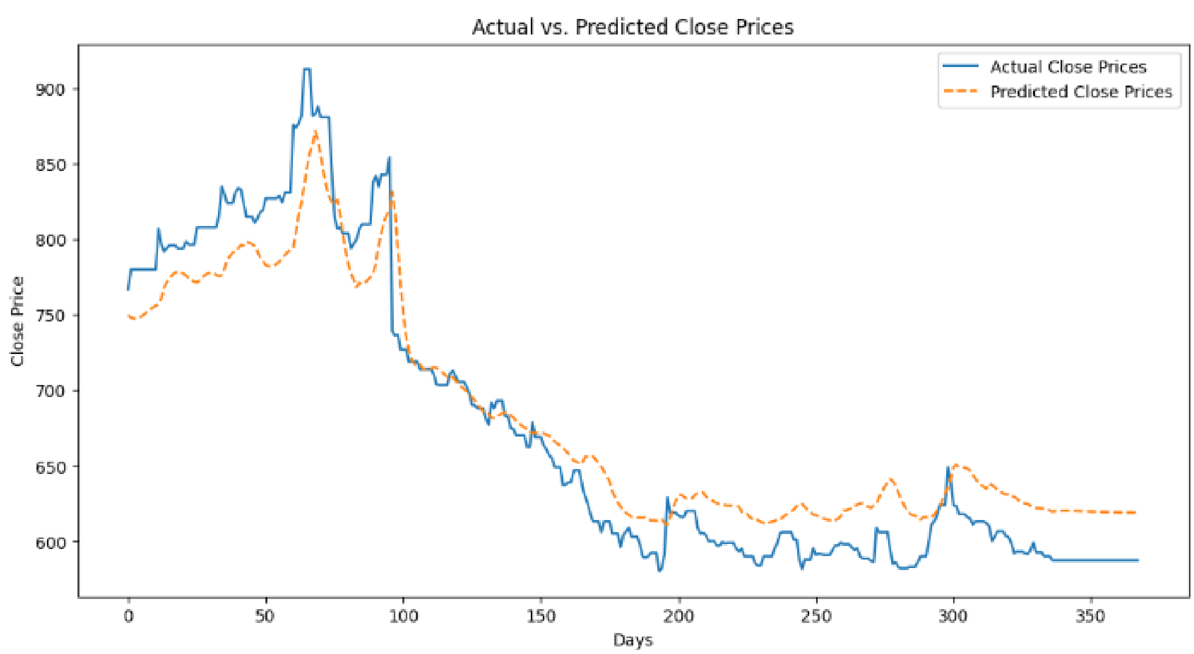

Figure 4.7: Actual vs Predicted

4.2.4 Model Performance Metrics

Model	MSE	RMSE	MAPE (%)
ARIMA	1700	170	40
SARIMAX	1000	100	30
LSTM	-	29	4

Table 4.1: Performance Metrics of Different Models

5. Future Enhancements

For future enhancements , there are several promising avenues to explore. First, expanding the range of input features to include alternative data sources, sentiment analysis of financial news, or social media sentiment can provide richer context for predictive modeling. Furthermore, the analysis can benefit from exploring advanced machine learning and deep learning models beyond ARIMA and LSTM. Techniques like hybrid models, ensemble methods, or reinforcement learning can be considered to improve prediction accuracy. Additionally, real-time data integration can enhance the project's scalability and responsiveness to rapidly changing market conditions. Lastly, implementing interpretability and explainability techniques can offer valuable insights into model decisions and build trust among users and investors. These future enhancements will ensure the Stock Market Analysis project remains dynamic and competitive in the ever-evolving financial landscape.

6. Conclusion

In conclusion, our analysis using ARIMA, SARIMAX, RNN, and LSTM models demonstrated the effectiveness of these models in predicting stock prices. The results indicate that the LSTM model outperformed the other models in terms of lower error metrics, including MSE, RMSE, and MAPE. The use of compound sentiment scores and feature engineering, such as Bollinger Bands bandwidth, contributed to the overall accuracy of the predictions. Overall, our models nicely predicted stock prices, providing valuable insights for investors and financial analysts.

References

[1] Prof Pradhan and Radhe Shyam. Stock market behavior in a small capital market: A case of nepal. *Radhe Shyam and Pradhan, Prof. Dr. Radhe Shyam, Stock Market Behavior in a Small Capital Market: A Case of Nepal (June 9, 1993). Pradhan, RS*, pages 20–32, 1993.

[2] Jas Bahadur Gurung. Growth and performance of securities market in nepal. *The journal of Nepalese business studies*, 1(1):85–92, 2004.

[3] Top Bahadur Pun and Tej Bahadur Shahi. Nepal stock exchange prediction using support vector regression and neural networks. In *2018 Second International Conference on Advances in Electronics, Computers and Communications (ICAECC)*, pages 1–6, 2018.

[4] Arjun Singh Saud and Subarna Shakya. Analysis of look back period for stock price prediction with rnn variants: A case study on banking sector of nepse. *Procedia Computer Science*, 167:788–798, 2020. International Conference on Computational Intelligence and Data Science.

[5] Arjun Singh Saud and Subarna Shakya. Analysis of gradient descent optimization techniques with gated recurrent unit for stock price prediction: A case study on banking sector of nepal stock exchange. *Journal of Institute of Science and Technology*, 24(2):17–21, Dec. 2019.

[6] Nawa Raj Pokhrel, Keshab Raj Dahal, Ramchandra Rimal, Hum Nath Bhandari, Rajendra K.C. Khatri, Binod Rimal, and William Edward Hahn. Predicting nepse index price using deep learning models. *Machine Learning with Applications*, 9:100385, 2022.

[7] Fan Sun, Ammar Belatreche, Sonya Coleman, T.M. Mcginnity, and Yuhua Li. Pre-processing online financial text for sentiment classification: A natural language processing approach. 03 2014.