Assignment 1 : Stock Price Prediction Report

Model Architecture and Training Details

Model Architecture: The model consists of two Long Short-Term Memory (LSTM) layers, followed by Dense layers. This architecture is designed to capture temporal dependencies and trends in the stock price data.

Libraries Used:

- **yfinance**: For downloading stock data from Yahoo Finance.
- **pandas**: For data manipulation and analysis.
- **numpy**: For numerical operations.
- **matplotlib**: For plotting and visualisation.
- **scikit-learn**: For data scaling.
- **tensorflow.keras**: For building and training the LSTM model.

Data Source: Stock data for Apple Inc. was sourced from Yahoo Finance using the `yfinance` library. The dataset includes daily closing prices from the date the stock was listed up to the latest available data.

Epochs: 7

Batch Size: 1

Loss Function: Mean Squared Error (MSE)

Data Preparation and Model Training

The stock data was scaled using the MinMaxScaler from `scikit-learn` to normalise the values between 0 and 1. This step is crucial for improving the performance of the LSTM model, which is sensitive to the scale of the input data.

Training data was prepared by using a window of 60 previous days to predict the next day's stock price. The LSTM model was then trained on this data over 7 epochs with a batch size of 1. The Mean Squared Error (MSE) was used as the loss function to measure the performance of the model during training.

Model Performance and Predictions

The model was evaluated by predicting stock prices for the upcoming 1 week, 1 fortnight, and 1 month. These predictions were then compared with the actual stock prices over these periods.

Comparison of Predicted Prices vs. Actual Prices

The comparison between the predicted and actual stock prices revealed that, while there were slight differences in the predicted and actual prices, the overall trend was captured accurately by the model. The graphs of predicted vs. actual prices for 1 month show that the model's predictions follow the same trend as the actual prices, albeit with minor vertical offsets.

Reasoning for Differences

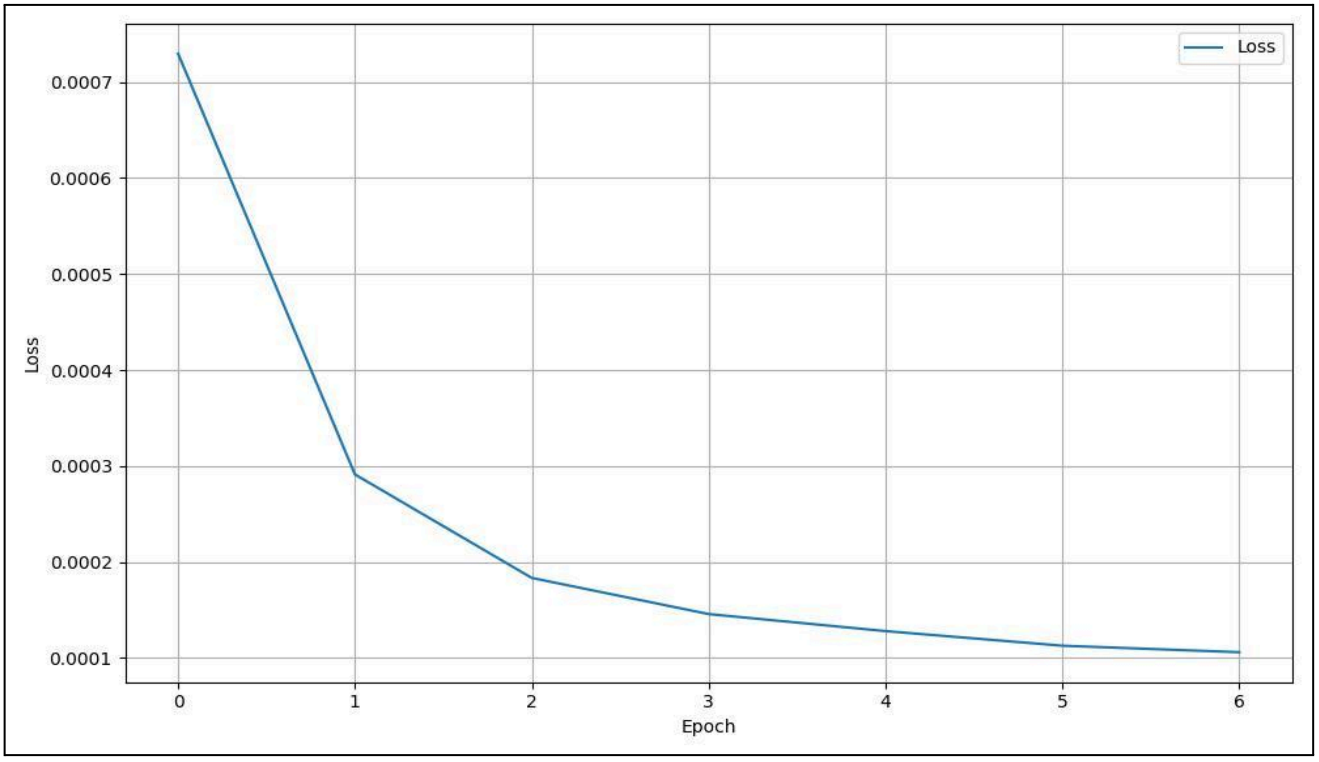
The slight differences between the predicted and actual prices can be attributed to the following factors:

1. **Historical Data Window**: The model uses the previous 60 days of stock prices to predict the future prices. This fixed window size may not always capture all the relevant patterns and trends, especially if there are significant market changes beyond this period.
2. **Model Complexity**: The current model uses a relatively simple architecture with two LSTM layers. More complex architectures or additional features might improve the accuracy of predictions.

Loss Function Plot

The plot of the loss function against epochs shows a decreasing trend, indicating that the model is learning and improving its predictions over time.

Loss Vs No. of epochs Graph

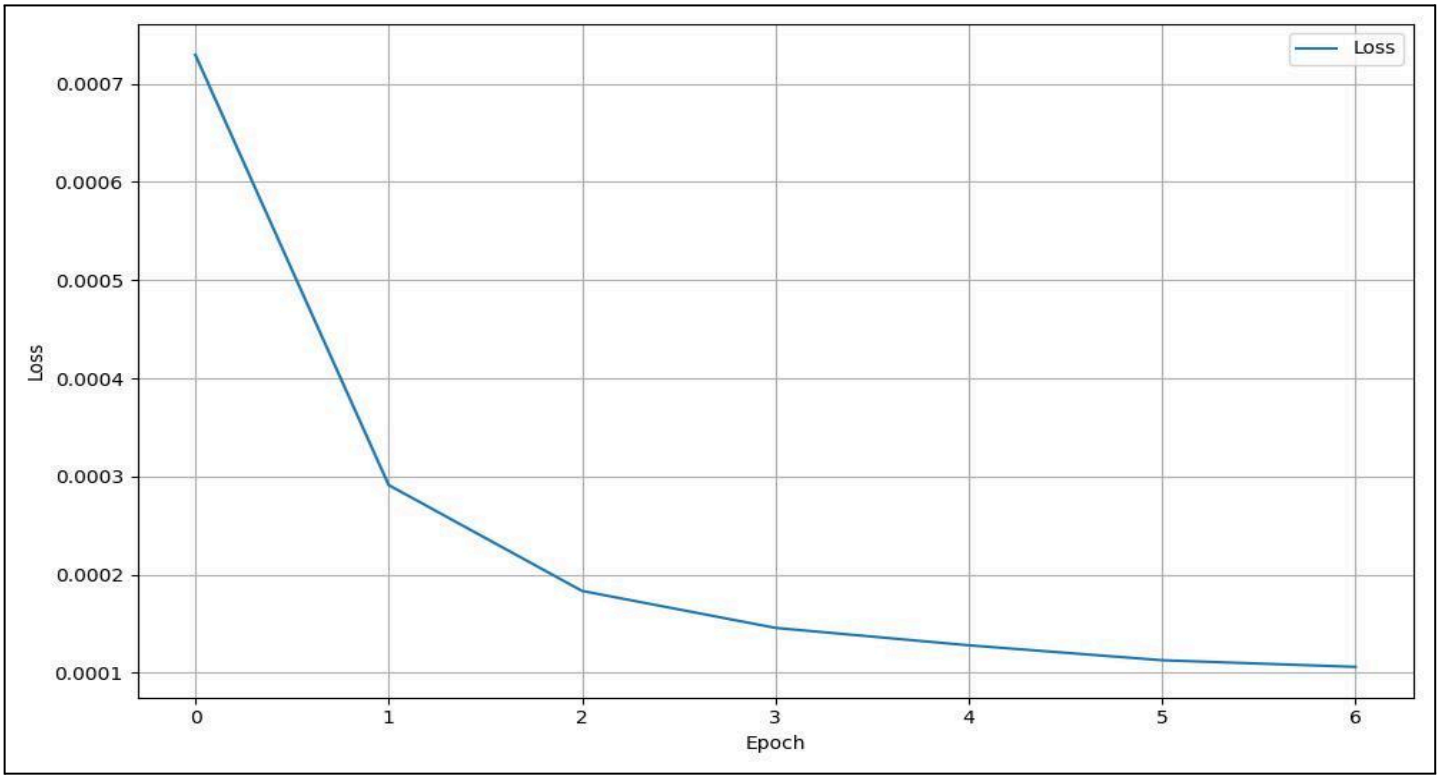


Conclusion

Despite some differences in the predicted and actual prices, the overall trend alignment indicates the model's effectiveness. Future improvements could involve optimizing the historical data window and experimenting with more complex model architectures.

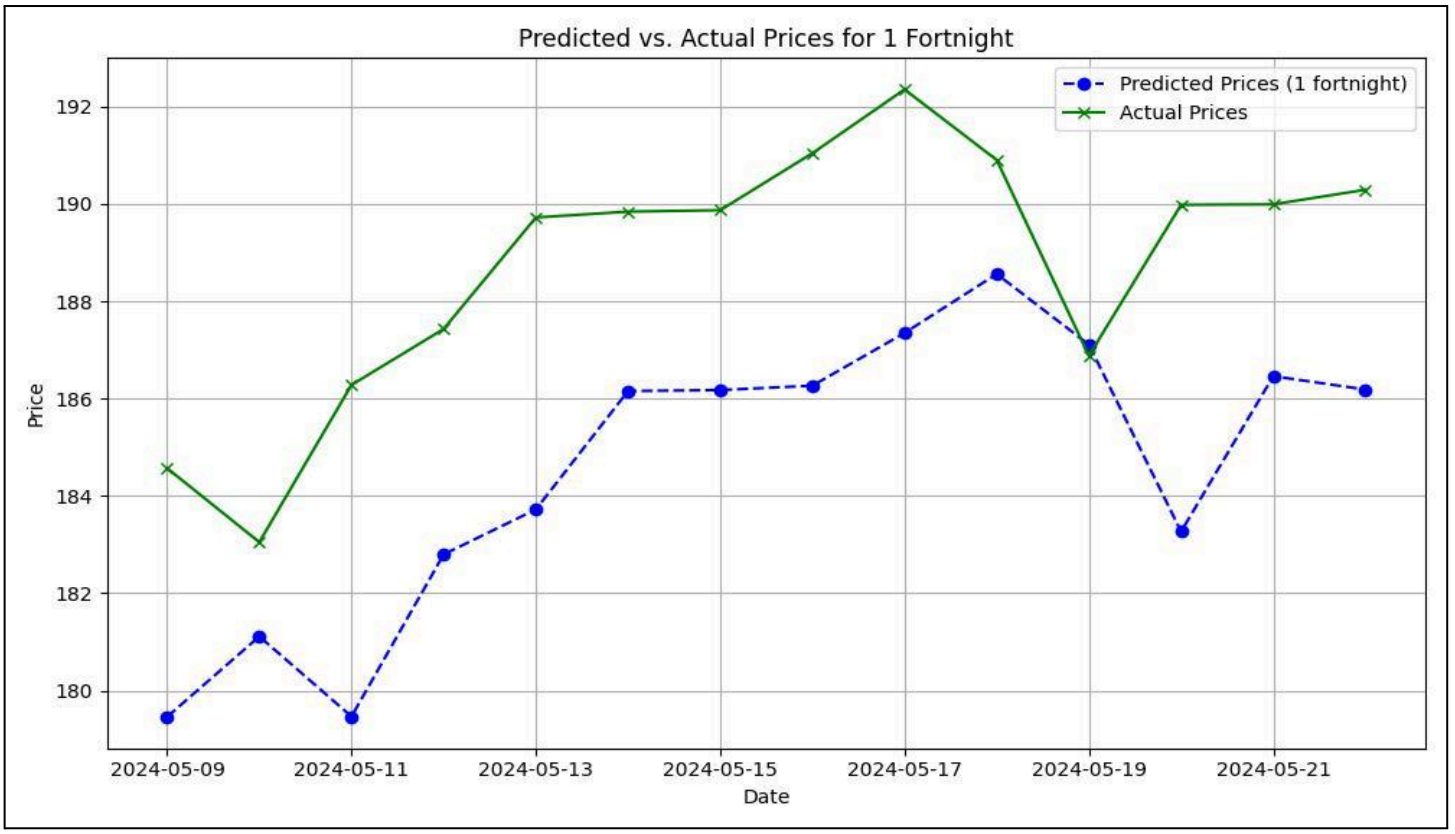
Plots

- **Loss Function Plot**: This plot shows the decrease in the loss function over the training epochs, demonstrating the model's learning process.



- **Predicted vs. Actual Prices for 1 Week**: This plot compares the predicted and actual stock prices for a one-week period.

- **Predicted vs. Actual Prices for 1 Fortnight**: This plot compares the predicted and actual stock prices for a two-week period.



- **Predicted vs. Actual Prices for 1 Month**: This plot compares the predicted and actual stock prices for a one-month period.