**Predicting Stock Prices Using LSTM - Apple (NASDAQ) and Reliance Industries (NSE)**

## **Introduction**

The objective of this project is to predict the stock prices of Apple Inc. (AAPL) listed on NASDAQ and Reliance Industries Ltd. (RELIANCE) listed on the National Stock Exchange of India (NSE) using a Long Short-Term Memory (LSTM) model. Stock price prediction is a challenging task due to the volatile nature of financial markets.

LSTM, a type of Recurrent Neural Network (RNN), is well-suited for this task as it can capture the temporal dependencies in the sequential data of stock prices. The project involves collecting historical stock price data, preprocessing it to make it suitable for modeling, building an LSTM model, and evaluating its performance.

## Data Collection

The historical stock price data for Apple Inc. (AAPL) and Reliance Industries (RELIANCE) was collected from the Yahoo Finance API. The dataset includes daily data points for the past 5 years, covering the Open, High, Low, Close, and Volume for each trading day.

### Steps:

#### - Data Sources: Yahoo Finance API was used to download the historical data.

#### - Time Period: The data spans from 22.08.2022 to 22.08.2024, providing at least 5 years of daily stock prices.

#### - Data Points: Each record includes the following features:

- Open: The price at which the stock opened on a particular day.

- High: The highest price reached during the trading day.

- Low: The lowest price reached during the trading day.

- Close: The price at which the stock closed on a particular day.

- Volume: The number of shares traded during the day.

## Preprocessing

Before feeding the data into the LSTM model, several preprocessing steps were undertaken to clean, normalize, and structure the data appropriately.

Steps:

#### Data Cleaning:

**Missing Values:** Any missing values in the dataset were removed to ensure data integrity.

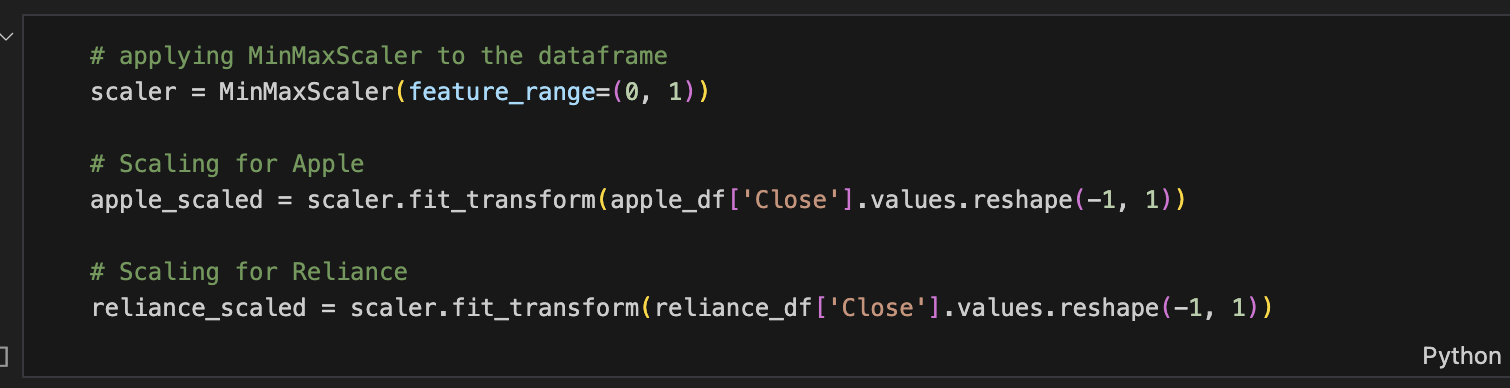
**Outliers:** Outliers were identified and handled to prevent skewed model training.

#### Feature Selection:

- The 'Close' price was selected as the primary feature for predicting future stock prices, as it reflects the most significant end-of-day price.

#### Normalization:

- The data was normalized using MinMaxScaler from the `sklearn.preprocessing` module to scale the 'Close' prices to a range of [0, 1]. This step is crucial as it helps in speeding up the convergence of the model during training.



#### Sequence Creation:

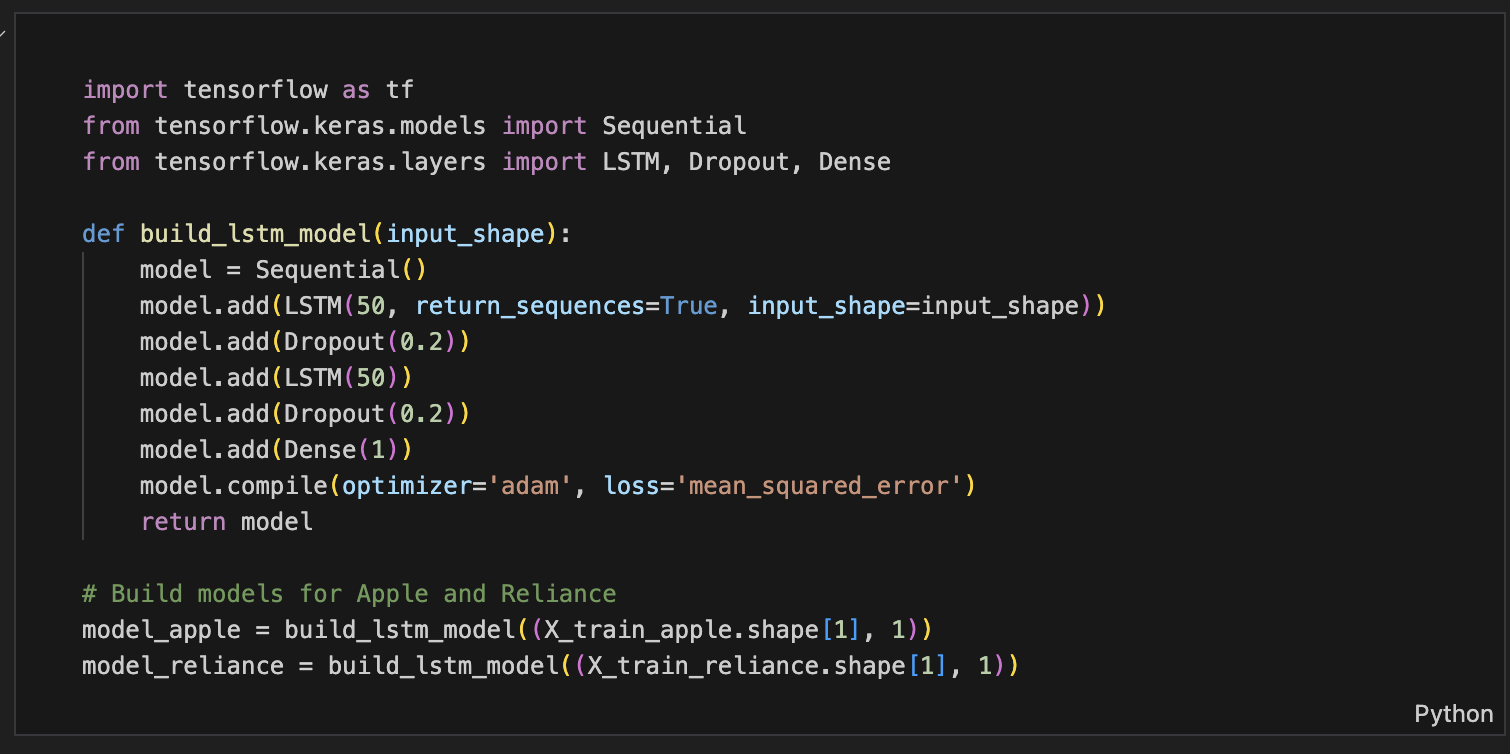
- To prepare the data for LSTM, it was converted into sequences of 60 days, where the model uses the past 60 days' prices to predict the next day's price.

## 

## 

## Model Building

The LSTM model was built using TensorFlow and Keras. The model architecture is designed to capture the sequential patterns in the stock price data and make predictions based on the learned patterns.



#### Model Architecture:

**LSTM Layers**:

- The model starts with an LSTM layer with 50 units, followed by a Dropout layer to prevent overfitting.

- A second LSTM layer with 50 units is added, followed by another Dropout layer.

#### Dense Layer:

- A Dense layer is used at the end to output the predicted stock price.

#### Compilation:

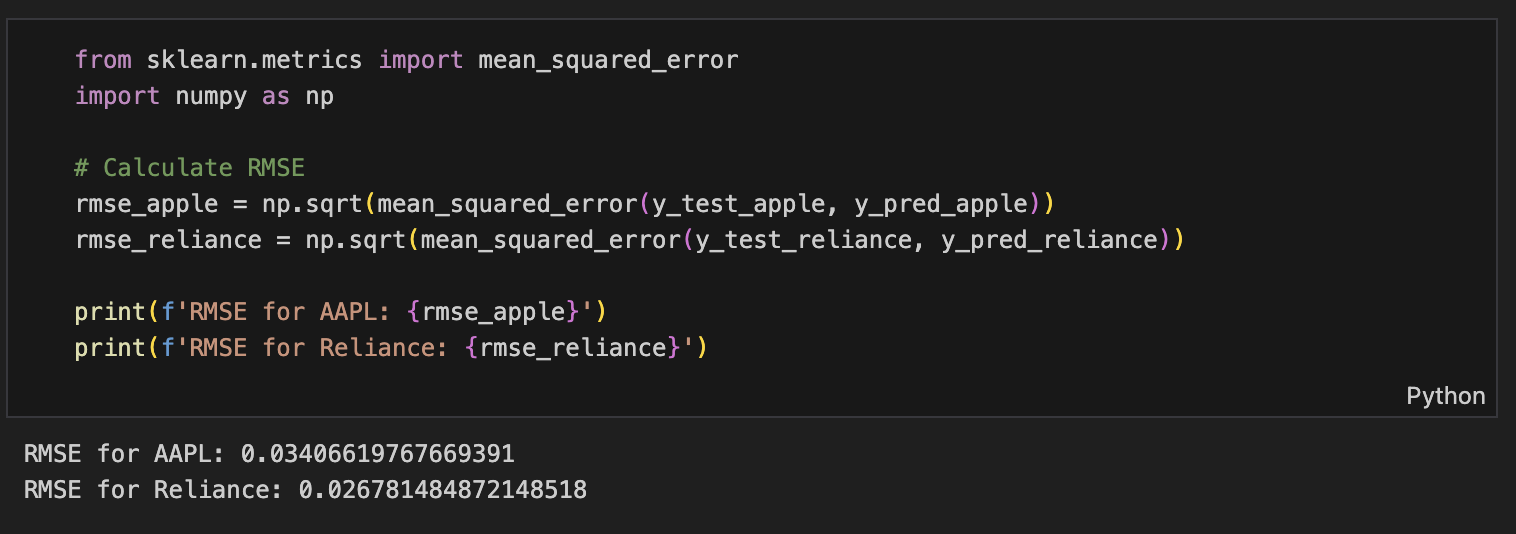
- The model is compiled using the Adam optimizer and Mean Squared Error (MSE) as the loss function.

#### Training:

- The models were trained for 100 epochs with a validation split to monitor the performance on unseen data.

## Evaluation

The model's performance was evaluated using the Root Mean Squared Error (RMSE) metric, which measures the difference between the predicted and actual stock prices.



#### Evaluation Metrics:

**RMSE Calculation**:

- The RMSE values for the Apple and Reliance models were calculated using the following formula:

#### Results

**Apple Model:**

- The RMSE for the Apple model was 0.03406619767669391, indicating average performance.

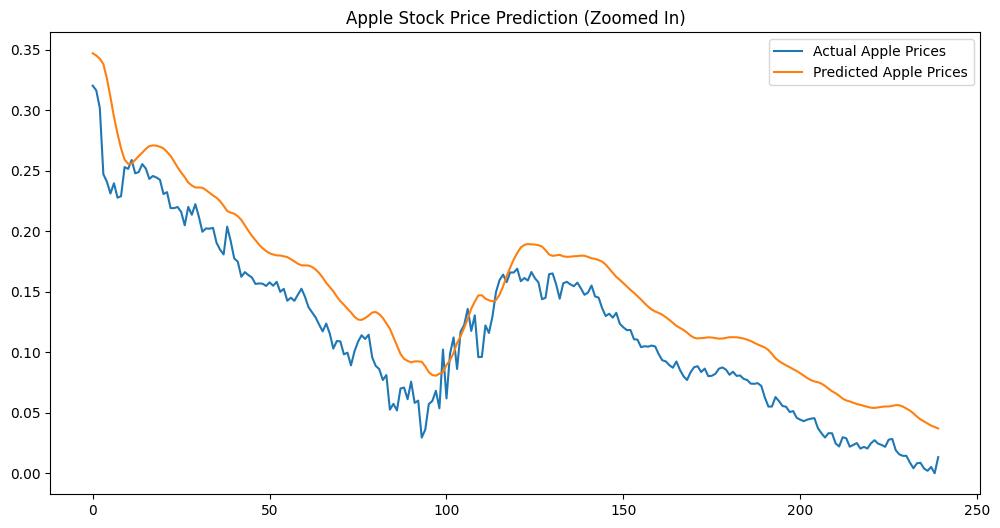
**Reliance Model:**

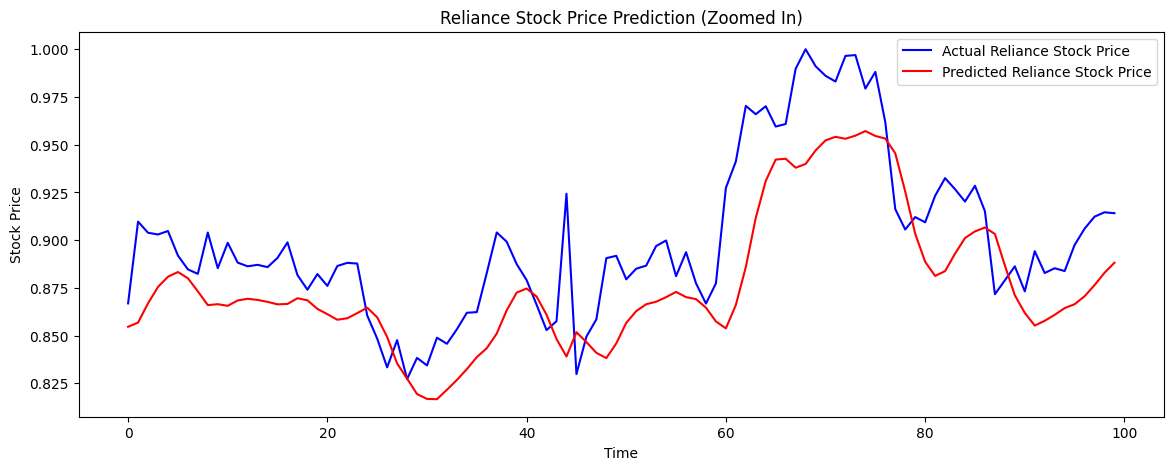
- The RMSE for the Reliance model was 0.026781484872148518, indicating average performance.

#### Model Performance:

**Visualizations:**

- The predicted vs. actual stock prices were plotted to visually assess the model's performance.





#### Challenges faced

## Ensuring the accuracy and completeness of historical stock data was a primary challenge, as stock market data is prone to noise and missing values, which can skew model performance. Despite efforts to clean the data and use reliable sources, some inconsistencies persisted. Feature selection and engineering also presented difficulties; choosing the right features and creating meaningful sequences for the LSTM model required careful consideration. Additionally, tuning the LSTM model’s hyperparameters—such as the number of layers, units, and dropout rates—was complex and involved extensive experimentation. Interpreting RMSE values in the context of stock price prediction added another layer of challenge, as understanding their significance amidst market volatility was crucial. Finally, the inherent volatility of stock prices, influenced by external factors beyond historical data, impacted the model’s predictive accuracy.

#### Potential Improvements

To enhance the model’s accuracy, incorporating additional features like technical indicators (e.g., Moving Averages or Relative Strength Index) could provide deeper insights into market trends that are not captured by historical prices alone. Exploring advanced model architectures, such as Bidirectional LSTMs or Gated Recurrent Units (GRUs), might improve performance by better capturing temporal dependencies. More extensive hyperparameter tuning using grid search or cross-validation could optimize model performance further. Ensemble methods, combining predictions from multiple models, could enhance accuracy by leveraging the strengths of various approaches. Applying additional regularization techniques, such as L1/L2 regularization, could prevent overfitting. Finally, deploying the model for real-time testing would assess its robustness under live market conditions and enable practical applications, providing valuable insights into its effectiveness and adaptability.

## Conclusion

This project successfully demonstrated the application of LSTM models to predict stock prices for Apple Inc. and Reliance Industries. The LSTM model effectively captured the temporal dependencies in the stock price data, although its performance varied between the two stocks. The RMSE values provided insights into the accuracy of the predictions, and the visualizations offered a clear comparison between the predicted and actual prices.