# Stock Price Prediction Using LSTM & XGBoost

## Objective

The objective of this project is to develop a predictive model to forecast the next day's closing price of a stock using historical financial data. The models used in this analysis are:

1. **LSTM (Long Short-Term Memory)**: A deep learning model designed to handle sequential time-series data.
2. **XGBoost (Extreme Gradient Boosting)**: A powerful machine learning algorithm that performs well on structured datasets.

## 1. Data Collection

### Source

- The dataset is obtained from **Yahoo Finance** using the yfinance Python library.
- The stock chosen for this state is **Apple Inc. (AAPL)**.
- The dataset spans from **January 1, 2020, to January 1, 2024**.

### Selected Features

The dataset includes the following essential features:

- **Open Price**
- **High Price**
- **Low Price**
- **Close Price** (Target variable)
- **Volume**

Additionally, compute several **technical indicators** to enhance model performance.

## 2. Feature Engineering

To improve model accuracy, additional **technical indicators** were computed:

1. **Relative Strength Index (RSI):** Measures stock momentum and potential overbought/oversold conditions.
2. **Moving Average Convergence Divergence (MACD):** Identifies trend direction and momentum strength.
3. **Bollinger Bands:** Captures volatility by providing upper and lower price bands.
4. **Simple Moving Averages (SMA_50, SMA_200):** Helps in trend detection and smoothing fluctuations.

These indicators were calculated using the **ta** (Technical Analysis) library and merged into the dataset.

**3. Data Preprocessing**

**Steps:**

- **Missing Values Handling:** Removed NaN values to ensure data consistency.
- **Feature Scaling:** Applied Min-Max Scaling to normalize numerical features.
- **Data Splitting:**
    - **Training Set:** 80% of the dataset
    - **Testing Set:** 20% of the dataset
- **Feature Engineering:**
    - Used **past 60 days of stock data** to predict the **next day's closing price**.
    - Converted sequential time-series data into a **tabular format** for XGBoost and a **sequential format** for LSTM.

**4. Model Development**

**Model 1: LSTM (Long Short-Term Memory)**

LSTM is a type of recurrent neural network (RNN) that is well suited for time-series forecasting. The model was trained with the following parameters:

- **Epochs:** 20
- **Batch Size:** 32
- **Optimizer:** Adam
- **Loss Function:** Mean Squared Error (MSE)

**Model 2: XGBoost (Extreme Gradient Boosting)**

XGBoost is a decision tree-based ensemble learning algorithm known for its efficiency and high performance in structured datasets. The model was trained with the following parameters:

- **Number of Estimators:** 100
- **Learning Rate:** 0.05
- **Objective Function:** reg:squarederror (optimized for regression tasks)
- **Evaluation Metric:** RMSE (Root Mean Squared Error)

**Training Process**

- Both models were trained on **80% of the dataset** and validated on the remaining **20%**.
- Predictions were made for the test set, followed by performance evaluation.

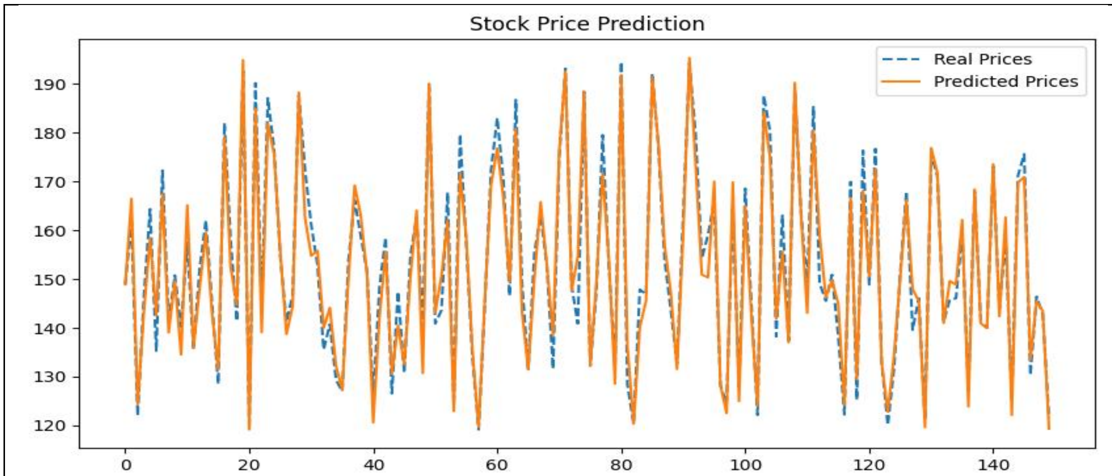
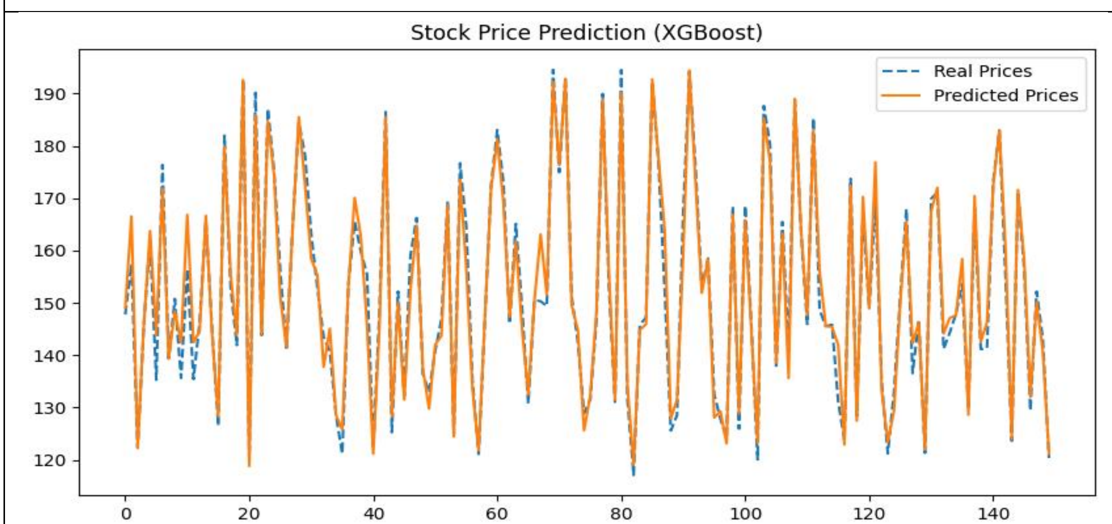
**5. Model Evaluation & Comparison**

To assess the models' performance, the following metrics were used:

| Model | RMSE (Root Mean Squared Error) | MAPE (Mean Absolute Percentage Error) |
|---|---|---|
| **LSTM** | **4.13** | **2.12%** |
| **XGBoost** | **3.60** | **1.72%** |

**Evaluation Interpretation:**

- **LSTM RMSE of 4.13** vs. **XGBoost RMSE of 3.60**: XGBoost achieves lower prediction error.
- **LSTM MAPE of 2.12%** vs. **XGBoost MAPE of 1.72%:** XGBoost achieves slightly higher accuracy in percentage terms.
- **Prediction Plots:**
    - LSTM's predictions fluctuate more compared to XGBoost.
    - XGBoost's predictions closely follow the actual stock price movements, indicating better performance.

| | |
|---|---|
|   | LSTM |
|   | XGBoost |

**Key Insights from Comparison**

- **LSTM is better suited for long-term sequential dependencies**, but in this case, stock prices have short-term variations that favor XGBoost.
- **XGBoost slightly outperforms LSTM in both RMSE and MAPE,** indicating that a tree-based ensemble model works better for short-term stock price prediction.
- **LSTM requires more computational power and training time**, whereas XGBoost is **faster and easier to interpret.**

**6. Findings:**

- Both models perform well, but XGBoost **slightly outperforms** LSTM in stock price prediction accuracy.
- LSTM requires **more data and longer training times** to be effective.
- Feature engineering, including **RSI, MACD, Bollinger Bands, and Moving Averages**, played a crucial role in improving predictions.

**7. Conclusion**

This project successfully compares **LSTM** and **XGBoost** for stock price prediction. While LSTM models are powerful for time-series tasks, XGBoost achieved **slightly better accuracy**, **lower error rates, and faster training times**.