Program

import numpy as np

import pandas as pd

import yfinance as yf

from sklearn.preprocessing import MinMaxScaler

from tensorflow.keras.models import Sequential

from tensorflow.keras.layers import Dense, LSTM

import matplotlib.pyplot as plt

# Parameters

ticker = "AAPL"

start\_date = "2015-01-01"

end\_date = "2024-12-31"

# Download stock data

df = yf.download(ticker, start=start\_date, end=end\_date)

data = df["Close"].values.reshape(-1, 1)

# Normalize the data

scaler = MinMaxScaler(feature\_range=(0, 1))

scaled\_data = scaler.fit\_transform(data)

# Create training and test datasets

train\_size = int(len(scaled\_data) \* 0.8)

train\_data = scaled\_data[:train\_size]

test\_data = scaled\_data[train\_size:]

# Convert data into sequences

def create\_dataset(dataset, time\_step=60):

x, y = [], []

for i in range(time\_step, len(dataset)):

x.append(dataset[i - time\_step:i, 0])

y.append(dataset[i, 0])

return np.array(x), np.array(y)

time\_step = 60

X\_train, y\_train = create\_dataset(train\_data, time\_step)

X\_test, y\_test = create\_dataset(test\_data, time\_step)

# Reshape input to be [samples, time steps, features]

X\_train = X\_train.reshape(X\_train.shape[0], time\_step, 1)

X\_test = X\_test.reshape(X\_test.shape[0], time\_step, 1)

# Build LSTM model

model = Sequential([

LSTM(50, return\_sequences=True, input\_shape=(time\_step, 1)),

LSTM(50),

Dense(1)

])

model.compile(loss='mean\_squared\_error', optimizer='adam')

model.fit(X\_train, y\_train, epochs=10, batch\_size=32, verbose=1)

# Predict

predicted\_stock\_price = model.predict(X\_test)

predicted\_stock\_price = scaler.inverse\_transform(predicted\_stock\_price)

real\_stock\_price = scaler.inverse\_transform(y\_test.reshape(-1, 1))

# Plot results

plt.figure(figsize=(12, 6))

plt.plot(real\_stock\_price, label="Actual Price")

plt.plot(predicted\_stock\_price, label="Predicted Price")

plt.title(f"{ticker} Stock Price Prediction")

plt.xlabel("Time")

plt.ylabel("Price")

plt.legend()

plt.show()

Output

* : The code generates a plot comparing actual and predicted stock prices, showing how well the LSTM model captures the stock's trends.[Read Medium articles with AI](https://readmedium.com/predictive-analysis-of-apple-stock-price-using-lstm-models-b7625aa78e43?utm_source=chatgpt.com)
* **Performance Metrics**: While the code doesn't explicitly calculate performance metrics, typical evaluations include:
  + **Root Mean Squared Error (RMSE)**: A lower RMSE indicates better predictive accuracy.[ResearchGate+6GitHub+6Medium+6](https://github.com/jovanthompsonmds/Market-Price-Prediction-Via-LSTM-Recurrent-Neural-Network?utm_source=chatgpt.com)
  + **R² Score**: An R² score closer to 1 suggests that the model explains a high proportion of the variance in the stock prices.

**🔍 Model Performance Insights**

Comparative analyses of LSTM models for AAPL stock prediction have yielded varying results:

* **Rohan Kumar's Study**: Utilizing 60-day sequences, the LSTM model demonstrated strong performance in capturing stock price trends, though it faced challenges with recent market volatility.
* **Dr. Temesgen Deressa's Comparison**: In a broader study, the LSTM model achieved an R² of 0.94, indicating excellent performance in capturing long-term trends. [Directory of Open Access Journals](https://doaj.org/article/cad60d91a0664e3ebafa9a10afb1f4b5?utm_source=chatgpt.com)
* **Li-HSIN Hung's Analysis**: Incorporating additional indicators, the LSTM model effectively predicted stock prices, with visualizations showing close alignment between predicted and actual values. [GitHub](https://github.com/jovanthompsonmds/Market-Price-Prediction-Via-LSTM-Recurrent-Neural-Network?utm_source=chatgpt.com)

**🧠 Considerations for Improvement**

* **Model Complexity**: Incorporating more features (e.g., technical indicators) and tuning hyperparameters could enhance model performance.
* **Evaluation Metrics**: Explicitly calculating metrics like RMSE and R² can provide a clearer assessment of model accuracy.[GitHub](https://github.com/jovanthompsonmds/Market-Price-Prediction-Via-LSTM-Recurrent-Neural-Network?utm_source=chatgpt.com)
* **Data Splitting**: Ensuring proper splitting of data into training and testing sets is crucial for unbiased evaluation.