# Stock Market Predictions Using LSTM


## Abstract

This project aims to predict stock prices using Long Short-Term Memory (LSTM) networks, a type of Recurrent Neural Network (RNN) capable of learning long-term dependencies in sequential data. By training the LSTM model on historical stock price data, the system learns temporal trends and attempts to forecast future stock prices with reasonable accuracy. This predictive capability is useful for financial decision-making and risk assessment in trading systems.


## Introduction

Stock price prediction is a challenging task due to the volatile and non-linear nature of financial markets. Traditional methods often fall short in capturing the complex patterns hidden in time-series data. Deep learning models, especially LSTMs, offer a robust solution by effectively handling sequential dependencies. This project implements an LSTM-based model trained on stock data to forecast future closing prices, demonstrating the use of AI in financial forecasting.

## Tools Used

- Python: Programming language for data handling and model development
- Pandas & NumPy: Data preprocessing and manipulation
- Matplotlib & Seaborn: Visualization of trends and model predictions
- TensorFlow / Keras: Building and training the LSTM model
- scikit-learn: Data scaling and performance evaluation
- Jupyter Notebook: Interactive development and testing environment

# Stock Market Predictions Using LSTM

## Steps Involved in Building the Project

1. Data Collection

- Obtained historical stock prices from Yahoo Finance using the yfinance library.
- Focused on key features such as Open, High, Low, Close, and Volume.

2. Data Preprocessing

- Normalized the data using MinMaxScaler to bring all values to a similar scale.
- Converted the time-series data into supervised learning format using sliding window approach.

3. Model Building

- Designed an LSTM network with input, LSTM, dropout, and dense layers.
- Compiled the model using adam optimizer and mean squared error loss function.

4. Training the Model

- Trained the LSTM model on a portion of the dataset.
- Used early stopping and validation split to prevent overfitting.

5. Evaluation

- Tested the model on unseen data and visualized predicted vs actual values.
- Calculated RMSE (Root Mean Squared Error) for quantitative evaluation.

6. Forecasting

- Deployed the model for multi-day future predictions using the most recent data.
- Plotted future trend predictions against actual values to validate performance.

## Conclusion

The LSTM-based stock market prediction system demonstrates the effectiveness of deep learning in time-series forecasting. While it cannot guarantee precise future prices due to market volatility, it provides valuable trend insights that can aid in investment strategies. Future improvements could include integrating external factors like news sentiment, macroeconomic indicators, and multiple stock indices to enhance accuracy.