SOC PROJECT REPORT

Stock Price Prediction

Jovita Bhasin (24B0939)
Mentor - Anuj Yadav

Project Objective

This project aims to predict daily stock closing prices for a given stock (e.g., Apple Inc. - AAPL) using historical price data. The model forecasts the next day's closing price based on a sequence of past prices.

Why?

- Stock price prediction helps traders and investors identify potential market movements.
- LSTM networks are particularly effective for time-series forecasting due to their ability to:
  - Learn long-term dependencies in sequential data
  - Handle non-linear patterns in financial markets
  - Maintain memory of important past trends while processing new data

Data Used

Source

- Yahoo Finance API (*yfinance* Python library)
- Ticker: AAPL (Apple Inc.), GOOG (Google)

Data Structure

- Time Period: January 1, 2010 - December 31, 2024
- Features Used:
  - Close price (primary feature)
  - Date index (datetime format)

Preprocessing

1. **Normalization:** MinMaxScaler (0 to 1) applied to closing prices
2. **Sequence Creation:** 60-day windows for training (X = past 60 days, y = next day)
3. **Train-Test Split:** 80% training, 20% testing

Feature Selection

Why Only Closing Price?

- Closing price reflects final market consensus for the day

- Most widely reported and analyzed price point in financial markets
- Avoids dealing with multiple price points (open/high/low) that may contain noise
- Most technical indicators (RSI, MACD, etc) are traditionally calculated using closing prices
- Focuses on the most economically significant price point

Potential Future Features

- Volume traded
- Moving averages
- RSI, MACD (technical indicators)
- Sentiment analysis from news

Model Used

Why LSTM?

1. **Handling Sequential Data:** Stock prices are time-series data, where past values influence future movements. Unlike traditional machine learning models that treat each data point independently, LSTMs (Long Short-Term Memory networks) are designed to recognize patterns in sequences, making them ideal for financial forecasting.

2. **Memory of Long-Term Dependencies:** Traditional Recurrent Neural Networks (RNNs) suffer from the vanishing gradient problem, making it difficult to learn long-term patterns. LSTMs use memory cells and gating mechanisms (input, forget, and output gates) to retain important information over long periods, which is crucial for stock trends that evolve over weeks or months.

3. **Capturing Non-Linear Patterns:** Stock markets exhibit complex, non-linear behavior due to market sentiment, news events, economic indicators etc. LSTMs can model these intricate relationships better than linear models (e.g., ARIMA) or shallow neural networks.

4. **Robustness to Noise:** Financial data is noisy. LSTMs can learn meaningful trends while filtering out short-term noise. They can also adapt to sudden market shifts.

5. **Stateful Predictions:** Unlike models that predict each day independently (e.g. linear regression), LSTMs maintain an internal state (hidden state) that

evolves over time and use past predictions to influence future ones (useful for multi-day forecasting)

LSTM Architecture

Model: Sequential

Layer (type)	Output Shape	Param #
lstm (LSTM)	(None, 60, 50)	10400
dropout (Dropout)	(None, 60, 50)	0
lstm_1 (LSTM)	(None, 60, 50)	20200
dropout_1 (Dropout)	(None, 60, 50)	0
lstm_2 (LSTM)	(None, 50)	20200
dropout_2 (Dropout)	(None, 50)	0
dense (Dense)	(None, 1)	51

Total params: 50,851
Trainable params: 50,851
Non-trainable params: 0

Key Hyperparameters

- **Time steps:** 60 days lookback
- **LSTM layers:** 3 layers (50 units each)
- **Dropout:** 20% for regularization
- **Optimizer:** Adam
- **Loss:** Mean Squared Error (MSE)
- **Epochs:** 25
- **Batch size:** 32

Results

The model was trained on Apple Inc. (AAPL) and Google (GOOG) (each twice) stock closing prices using 60-day historical windows to predict the next day's closing price. The model performance was evaluated using mse, mae and rmse metrics.

For AAPL:

Metric	Value (I)	Value (II)
MSE	61.24	47.39
MAE	6.03	5.42
RMSE	7.83	6.88

For GOOG:

Metric	Value (I)	Value (II)
MSE	62.59	36.19
MAE	6.49	4.77
RMSE	7.91	6.02

Visualization Highlights:

- **Loss Plot**: MSE decreased steadily during training, with no signs of overfitting.
- **Predicted vs Actual**: The model's predictions closely follow actual closing prices, especially during stable periods. Some lag is visible during high volatility phases, which is expected.

Evaluation

Strengths

- Accurately predicts short-term price trends using only past closing prices.
- The model generalizes well to unseen data as shown by low MAE and RMSE.
- LSTM architecture is well-suited for sequential financial data.

Weaknesses & Limitations

- Sudden price spikes (news, earnings, etc.) are not captured well since the model only uses historical closing prices.
- Only one feature (closing price) was used, which limits model depth.
- The model predicts only one day ahead; performance for longer horizons is untested.

Conclusion

The project successfully implemented an LSTM-based model for next-day stock price prediction using historical closing prices. The model achieved good accuracy, with low error rates on the test data. The results demonstrate the strength of LSTM networks in handling time series data for financial forecasting.

Future Scope

1. **Add More Features**:
    - Volume, open, high, low prices
    - Technical indicators (RSI, MACD, Moving Averages)
2. **Improve Model Robustness**:
    - Try deeper networks or GRU/Transformer models
    - Tune hyperparameters further
3. **Explore Multi-Step Forecasting**:
    - Predict stock prices for the next 3, 5, or 7 days
4. **Use Additional Stocks**:
    - Apply the same model to other stocks or index data for broader insight
5. **Backtest a Strategy**:
    - Build a simple trading simulation based on model output to evaluate financial impact