Stock Price Prediction with Random Forest Regressor vs. Long Short-Term Memory Network

Siddharth Salapaka

***Abstract*— This paper presents an analysis of stock price prediction using two machine learning models: Random Forest Regressor and Long Short-Term Memory (LSTM) networks. The models are applied to both daily and hourly stock data, leveraging technical indicators such as Simple Moving Average (SMA) and Relative Strength Index (RSI). The study discusses data preprocessing, model architecture, performance evaluation using Mean Squared Error (MSE), and an in-depth analysis of the model predictions on UPST stock.**

I. Introduction

Stock market prediction is a challenging problem due to the stochastic nature of financial markets. This study explores the application of machine learning models to predict stock prices using historical data and technical indicators. The models analyzed are:

- Random Forest Regressor (RF)
- Long Short-Term Memory (LSTM) network

II. Data Collection and Preprocessing

Stock data for the UPST ticker was obtained from Yahoo Finance using the `yfinance` library. Two datasets were used:

1) Daily stock prices from January, 2019 to February, 2025.
2) Hourly stock prices for the past 240 days (as of February 24th, 2025).

Missing values were handled using forward fill, and data was sorted chronologically. Features extracted include:

- Closing price
- Simple Moving Average (SMA) with a window of 20 periods
- Relative Strength Index (RSI) with a window of 14 periods

III. Feature Engineering

The SMA and RSI metrics were computed using the following equations:

$$\text{SMA}_t = \frac{1}{N} \sum_{i=0}^{N-1} P_{t-i}$$

$$\text{RSI}_t = 100 - \frac{100}{1 + \frac{\text{Average Gain}}{\text{Average Loss}}}$$

where P_t represents the closing price at time t, and N is the moving window size.

Lag features were created by shifting the closing price forward by one period, setting it as the target variable.

IV. Random Forest Regressor

Random Forest is an ensemble learning technique that constructs multiple decision trees and aggregates their outputs. The model is trained using Time Series Cross-Validation with 10 splits.

The model's prediction is computed as:

$$\hat{y} = \frac{1}{T} \sum_{i=1}^{T} f_i(X)$$

where $f_i(X)$ is the prediction from the i^{th} tree, and T is the number of trees.

V. Long Short-Term Memory (LSTM) Network

LSTMs are a type of recurrent neural network designed to capture long-term dependencies. The LSTM architecture used includes:

- Two LSTM layers with 50 units each
- Dropout layers with a rate of 0.2 to prevent overfitting
- A fully connected dense layer with one neuron for regression

The network is trained using the Adam optimizer and Mean Squared Error (MSE) loss function.

The LSTM recurrence is defined by:

$$\begin{aligned}
f_t &= \sigma(W_f x_t + U_f h_{t-1} + b_f) \\
i_t &= \sigma(W_i x_t + U_i h_{t-1} + b_i) \\
\tilde{C}t &= \tanh(W_C x_t + U_C ht - 1 + b_C) \\
C_t &= f_t \odot C_{t-1} + i_t \odot \tilde{C}t \\
o_t &= \sigma(W_o x_t + U_o ht - 1 + b_o) \\
h_t &= o_t \odot \tanh(C_t)
\end{aligned}$$

where f_t, i_t, and o_t are the forget, input, and output gates respectively.

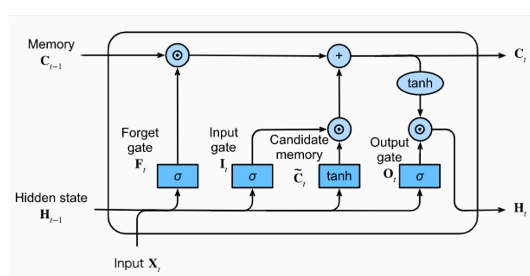

Fig. 1. LSTM Network Architecture

VI. Performance Evaluation

Model performance was evaluated using Mean Squared Error (MSE):

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$$

where y_i and $\hat{y}_i$ are the actual and predicted stock prices.

VII. Model Predictions and Analysis for UPST

Figures 2, 3, 4, and 5 present the prediction results for UPST stock using the Random Forest and LSTM models, respectively.

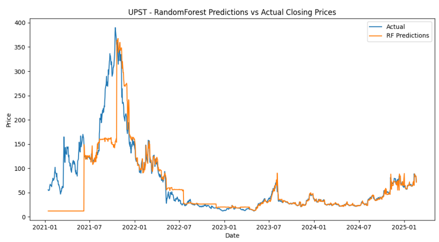

Fig. 2. Random Forest Predictions vs Actual Prices for UPST (Daily Data)

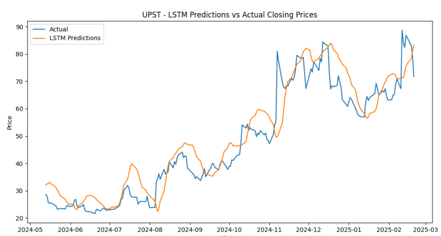

Fig. 3. LSTM Predictions vs Actual Prices for UPST (Daily Data)

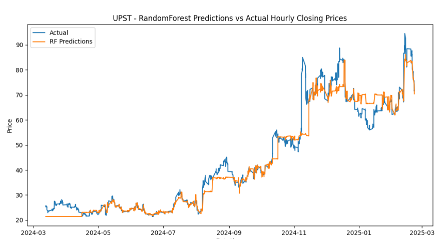

Fig. 4. Random Forest Predictions vs Actual Prices for UPST (Hourly Data)

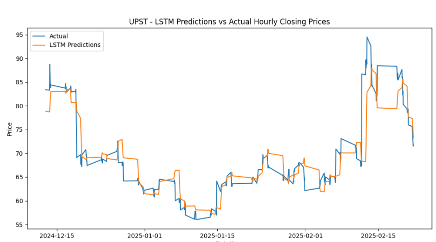

Fig. 5. LSTM Predictions vs Actual Prices for UPST (Hourly Data)

VIII. Comparison of Hourly vs. Daily Data

- Random Forest performs better on daily data and hourly data during periods with less variability, but fails to track sharp variations in the stock price.
- LSTM captures short-term variations in hourly data but exhibits some lag in predicting sharp price spikes, but it still tracks steep ups and downs better than Random Forest.
- Random Forest seems to be preferable for intraday trading strategies, whereas LSTM is more stable for long-term investment as it captures overall trends of the stock better.

IX. Conclusion

The comparison between LSTM and Random Forest models highlights their strengths and limitations across different time resolutions. The LSTM model is advantageous for long-term investment decisions due to its ability to capture overarching trends with stability. Conversely, the Random Forest model is better suited for short-term trading strategies on stable stocks, as it effectively captures smaller price movements. Note that neither model performs well with extreme volatility in a short period of time, although LSTM adapts quicker than Random Forest.

Future research should focus on hybrid modeling approaches that integrate both techniques to leverage their strengths. Additionally, incorporating alternative data sources, such as news sentiment and macroeconomic indicators, could further improve prediction accuracy and robustness.