Predicting Stock Price Trends Using Historical Data: An Apple Inc. Case Study

Varshini Sherapur Basavarajappa
Vinisha Purushotham Recharla
Sai Nithin Krishna Souram
Dharanidhar Reddy Challa

March 20, 2025

Abstract

Stock market is suspicious and indicatable, very difficult to accurately forecast. Historical stock data of Apple Inc. (AAPL) is found to be used to forecast next-day closing price by applying advanced machine learning techniques with advanced features of the engineering. The research uses various technical indicators like Bollinger Bands, RSI, MACD, moving average and lagged closing price to capture both sentiment of market and temporal trend. Three separate models Linear Regression, Random Forest Regression and an LSTM network were trained and very thoroughly evaluated based on metrics such as RMSE, MAE, and the R^2 score. Even though deep learning methods are complicated, plus linear Regression model is nearly best for counting best Equation of it which is R^2 0.9802 and RMSE 2.7392. The chosen model is at present deployed in an environment for real time data integration by means of the marketplace, forming a fantastic tool for actual-time stock charges prediction, which may assist in decision making with investors and monetary consultants.

---

1 Introduction

Stock market activity produces quick unpredictable price fluctuations that bring about substantial financial profit or loss opportunities between regular movements. Financial analysts along with investors heavily depend on accurate stock price predictions to make properly-informed choices. The research targets Apple Inc. (AAPL) stock price prediction for the following day based on historical data acquired from Yahoo Finance. The system uses machine learning at its advanced stage while working alongside regression algorithms and deep learning methods. The methodology heavily relies on feature engineering which enables utilization of Bollinger Bands plus RSI and MACD with moving averages and lagged closing price elements to extract short-term patterns and market mood. This research uses historical stock data augmented with technical elements and past observation values to determine AAPL's future close-price performance. The study then evaluates the forecast precision between linear regression and random forest regression and long short-term memory networks and studies how the LSTM sliding window assists in analyzing time-series patterns in data sequences. The project evaluates different forecasting models by answering these questions while creating a practical forecasting system through real-time data acquisition from the *yfinance* library to enable online stock price predictions that support trading and investment decisions.

2 Related Works

In the current past, particularly with the utilization of big data, the application of artificial intelligence especially the machine learning has received quite attention in predicting stock prices. In earlier days different techniques were applied for analysis, including linear regression techniques which implied the linear model of the data. Such models are easy to understand and may achieve good results especially when used with good features selection. Hence, more sophisticated methods such as the Random Forest were invented to accommodate non-linear patterns into the properties of the financial data, although there are some shortcoming with it due to issues with problems between the hyperparameters and the sequential data in financial data.

In recent years, Recurrent Neural Networks (RNNs), especially, LSTM networks, were considered for the dependency of time series, which is necessary for forecasting. Researchers have adopted LSTM for integrating with

technical indicators and lagged variables and that leads to increase in computational time. However, the feature selection and engineering have become the new challenge in all the models. Despite the higher complexities of Random Forests and LSTM networks, this project extends from prior work to show that with appropriate optimization, Linear Regression could potentially prove more effective compared to these models in the case of Apple Inc's stock prices.

3 Preliminary/Background

Stock markets are fluctuating always in nature as they depend on various factors such as the performance of the companies or other economical events. Such quantitative information as historical stock data, for instance Apple Inc. (AAPL) from Yahoo Finance can be used in determining stock prices' trends and patterns to make estimation on its future behavior. Thus, data exploration and preprocessing were conducted in our work by converting the date strings into the correct format, completing missing data, and correctly identifying numerical data types. We also formed some other technical variables such as Bollinger Bands, RSI, MACD, Moving average 30 for current prices, lagged feature of closing price of the last 5 days. The first attempts to solve the problem using simple models, such as Linear Regression, appeared to be quite effective; therefore, it was decided to proceed to more complicated techniques like Random Forests and LSTM networks. These preliminary results indicate the importance of the features engineering and discussion about further model and suggest that decision-making process in time-critical financial environments is feasible and based on the robust foundation.

4 Methodology

This project use multi-step and systematic strategy to forecast apple stock with historical. The approach is intended to get tractable elements out of raw financial datasets, to compare quite a few modeling technologies. The steps to check the suitable sleeping mattress ideas are described below:

4.1 Data Collection and Preprocessing

- Data Acquisition Stock data for Apple Inc. from a historical perspective was obtained from Yahoo Finance. This dataset includes daily records of primary pricing quotes (Open, High, Low, Close, Adjusted Close) and trading volume.
- Data Cleaning
  - The `Date` column has been parsed to `datetime` format.
  - Numeric columns (Open, High, Low, Close, Volume) were correctly cast as numerical data types.
  - Erroneous values were handled using forward-fill and backward-fill techniques.
  - The data were sorted in chronological order to preserve the inherent time sequence.
- Data Splitting

  A chronological split was adopted, where 80% of the data was used for the training set, while the remaining 20% was used for the testing set. This paradigm maintains temporal dependencies and simulates real-life forecasting scenarios.

4.2 Feature Engineering

- **Technical Indicators**

  Several technical measures were calculated to convey market attitude and volatility:
  - **Bollinger Bands:** Used for measuring the market deviation and consolidation from overbought/oversold points.
  - **Relative Strength Index (RSI):** Used to gauge the momentum and potential reversal points.
  - **MACD (Moving Average Convergence Divergence):** Serves as an indicator for trend changes.
  - **Moving Average:** A 30-day average was computed to smooth out short-term fluctuations.

- **Lag Features** To capture temporal dependencies, lagged versions of the closing price were created (e.g., `Close_lag_1` to `Close_lag_5`) to reflect recent historical trends in the prediction algorithm.

  Target Variable The forecast variable was set as the next day's closing price, achieved by shifting the `Close` column by one day.

4.3 Model Development

Three different modeling approaches were carried out and contrasted:

- **Linear Regression Concept:** A simple model assuming a linear relationship between the input features and the output variable.

  Mathematical Formulation:

  $$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_p x_p$$

  Estimation: The coefficients β are determined by minimizing the Sum of Squares of Errors (SSE).

- **Random Forest Regression Concept:** An ensemble method that builds multiple decision trees on non-overlapping subsamples of the original training set and aggregates the results to improve robustness.

  Implementation: Hyperparameters were tuned using `RandomizedSearchCV`, which searched parameters such as the number of trees, `max_depth`, and `min_samples_split`, among others.

  Aggregation: The final prediction is computed as the mean of all the predictions generated by the individual trees.

- **LSTM (Long Short-Term Memory) Network Concept:** A deep learning model designed to capture long-range dependencies in time series data, making it suitable for forecasting.

  Sliding Window Approach: A fixed-size window (e.g., 5 past steps) is used to create input sequences from the data.

  Architecture:

  - Two LSTM layers with dropout layers in between to prevent overfitting.
  - A final Dense layer to produce the prediction.

  Training: Backpropagation Through Time (BPTT) was applied to train the model by optimizing the Mean Squared Error (MSE), and early stopping was employed to mitigate overfitting.

4.4 Experimental Setup

- **Normalization** All features were scaled using `MinMaxScaler` to normalize the input data, ensuring consistent performance across different models.

- **Evaluation Metrics** Models were evaluated using the following metrics:

  Root Mean Squared Error (RMSE):

  $$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

  Mean Absolute Error (MAE):

  $$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

  R^2 Score:

  $$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

- **Model Comparison** Visual and quantitative comparisons (via error metrics and R^2) were performed to assess which model most accurately predicted next-day stock prices.

4.5 Model Deployment

- **Selection:** Given that The Linear Regression model with an R^2 of 0.9802 and RMSE of 2.7392 was chosen for deployment because it traded off accuracy for interpretability.
- **Live Data Integration:** The selected model along with the feature scaler is saved and used in conjunction with the Python 'yfinance' library to enable live stock price predictions keeping track of the model all the time ie remaining dynamic in market.

This approach combines tight data protzessaging, automated feature Engineering and multi Model Evaluation in a single framework, which allows for accurately deportable predictive model, suitable for production usage.

5 Numerical Experiments

This section presents experiments comparing the performance of three models—Linear Regression, Random Forest Regression, and LSTM network—in forecasting the next-day closing prices of Apple Inc. (AAPL).

5.1 Model-Specific Experiments

- **Linear Regression:** A baseline model with engineered features.
  - **RMSE:** 2.7392, **MAE:** 2.1026, R^2**:** 0.9802

  Discussion: Excellent accuracy and strong predictive performance suggest that the features effectively capture price behavior.
- **Random Forest Regression:** Hyperparameter tuning using `RandomizedSearchCV`.
  - **Best Parameters:** `n_estimators = 50`, `min_samples_split = 5`, `max_depth = 30`, **RMSE:** 27.8556, **MAE:** 22.5621, R^2**:** -1.05

  Discussion: Poor performance due to Random Forest's inability to handle time-series dependencies effectively.
- **LSTM Network:** Constructed with a sliding window approach and two LSTM layers.
  - **RMSE:** 5.3071, **MAE:** 4.1859, R^2**:** 0.9242

  Discussion: While slightly less accurate than Linear Regression, LSTM demonstrated its ability to capture sequential data dependencies.

5.2 Comparative Analysis

Model	**RMSE**	**MAE**	R^2
Linear Regression	2.7392	2.1026	0.9802
Random Forest	27.8556	22.5621	-1.0510
LSTM	5.3071	4.1859	0.9242

Table 1: Comparative performance of different models.

5.3 Overall Findings

- **Linear Regression:** Best performer with high accuracy, indicating the effectiveness of engineered features.
- **Random Forest:** Poor performance due to inability to handle time-series dependencies.
- **LSTM Network:** Good at capturing sequential data but did not outperform Linear Regression in this study.

6 Visual Results and Analysis

The following set of figures demonstrates the essential aspects related to our data investigation along with model development and performance assessment. Each figure is accompanied by a brief description and key insights.

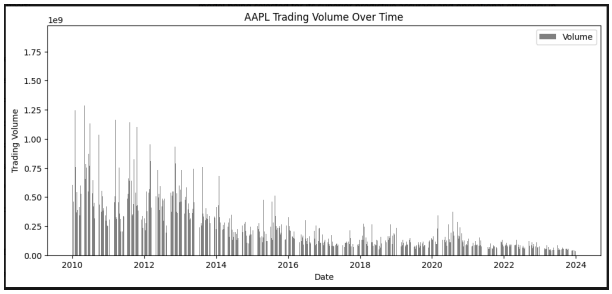

Figure 1: Historical Trading Volume Pattern of AAPL Stock

Description: The bar chart illustrates Apple's daily trading volumes between 2010 and 2024. The x-axis displays the trading dates while the y-axis shows the traded share numbers. The chart indicates that trading volume was higher in the earlier period and decreased steadily over time, reflecting market interest levels as well as economic and liquidity factors.

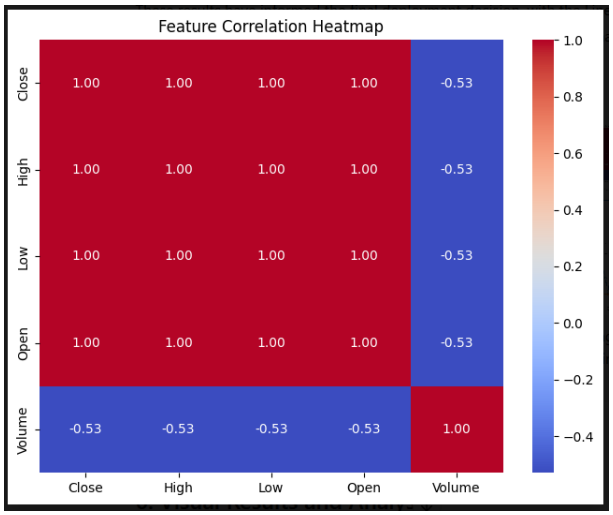

Figure 2: Feature Correlation Heatmap

Description: The heatmap shows the correlation relationships among the variables Close, High, Low, Open, and Volume. Positive correlations appear as bright red cells, while negative correlations appear as blue cells.

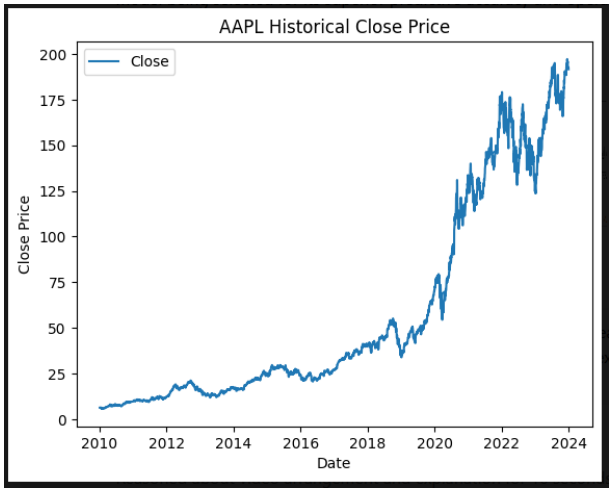

Figure 3: AAPL Historical Close Price (2010–2024)

Description: A line chart demonstrates Apple's closing stock price movement from 2010 to 2024. The x-axis displays trading dates and the y-axis represents the closing price in USD.

Key Insight: The chart indicates a general steady growth in Apple's stock price, with intermittent periods of market instability. Such patterns help analysts detect underlying market cycles and external factors that influence price fluctuations.

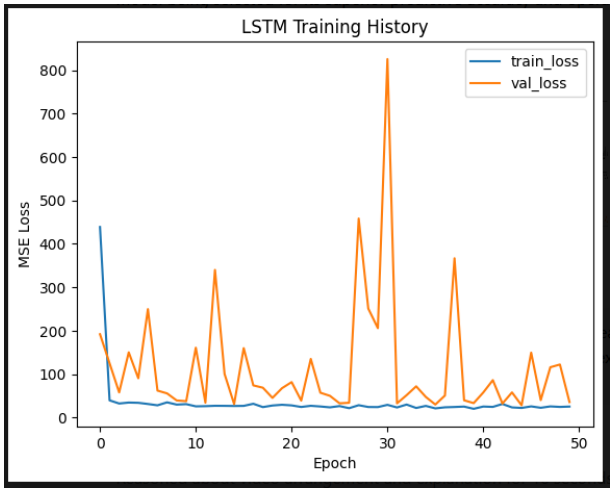

Figure 4: LSTM Training History

Description: The plot shows the training loss (blue line) and validation loss (orange line) of the LSTM model over each training epoch. The y-axis represents the mean squared error (MSE) loss, while the x-axis shows the number of training epochs.

Key Insight: Fluctuations in the validation loss suggest potential challenges in the model's ability to generalize to unseen data. The use of early stopping prevents overfitting once the validation loss stops improving.

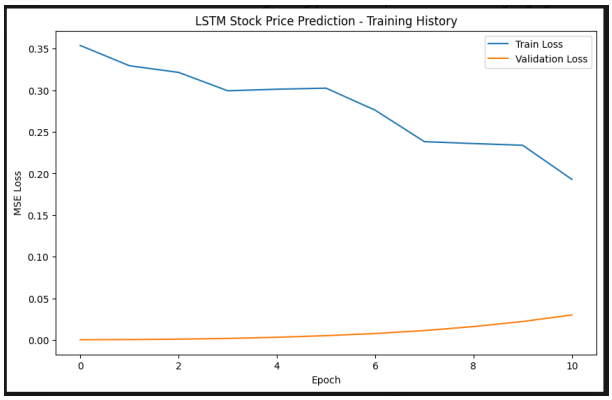

Figure 5: LSTM Stock Price Prediction Training History

Description: This plot shows the training and validation losses of the LSTM model as a function of the number of epochs. The y-axis represents the mean squared error (MSE) loss, and the x-axis represents the epoch number.

Key Insight: The gradually decreasing training loss, while the validation loss remains relatively steady, indicates that the model is learning the underlying patterns without overfitting. In some cases, a slight increase in validation loss signals that more fine-tuning, particularly in terms of hyperparameters, is required.

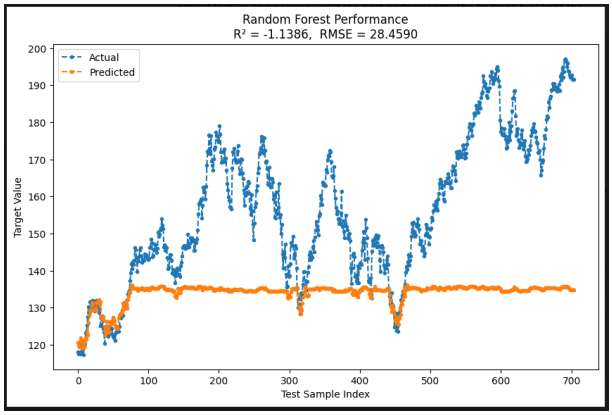

Figure 7: Random Forest Performance

Description: This plot includes a scatter plot of actual vs. predicted values of the Random Forest model on the test set, along with R^2 and Root Mean Squared Error (RMSE). The horizontal axis represents the test sample index, and the vertical axis corresponds to the target value (closing price).

Key Insight: The significant deviation of the estimated values from the actual values, along with a negative R^2 and high RMSE, clearly indicates that the Random Forest model failed to generalize well on this time-series data.

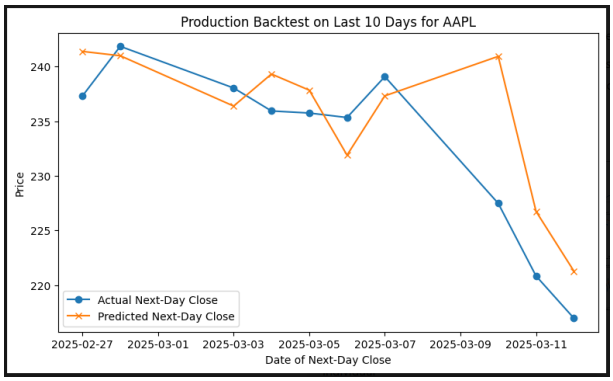

Figure 6: Backtest of the Production for AAPL for the Last 10 Days

Description: The blue line represents the actual next-day close, while the orange line represents the predicted next-day close for a 10-day look-back window. The x-axis shows the date of the close, and the y-axis represents the stock price in USD.

Key Insight: This visualization illustrates how the model would perform in near-real conditions, demonstrating its short-term predictive capability. The deviations between the actual and predicted lines indicate dates where the model is less accurate, pointing to areas for improvement.

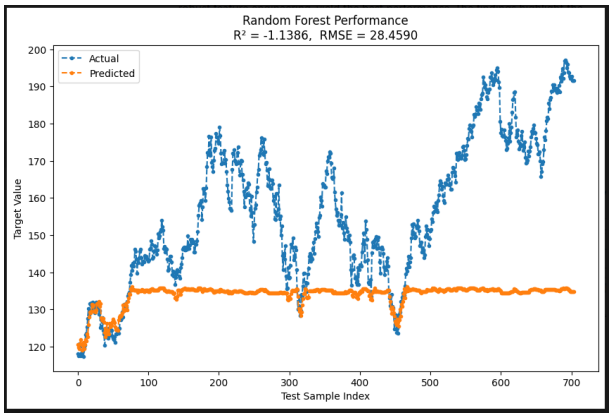

Figure 8: Actual vs. Random Forest & LSTM (Sliding Window) on the Test Set

Description: The blue line represents the actual closing price, while the orange line represents the LSTM model's predicted price for the test set. The x-axis shows the date, and the y-axis indicates the closing price.

Key Insight: The figure demonstrates that LSTM predictions are better at identifying the overall trends in the actual prices compared to Random Forest predictions. This supports the use of sequential models for time-series data, particularly when there are dependencies from preceding periods, such as with stock prices.

7 Conclusion

This project has successfully demonstrated the use of machine learning models for forecasting the next-day closing price of Apple Inc. (AAPL). We utilized Linear Regression, Random Forest Regression, and Long Short-Term Memory (LSTM) networks to evaluate their effectiveness in predicting stock prices.

7.1 Key Findings

1. **Linear Regression:** The most suitable analysis method was the Linear Regression model, with an R^2 value of 0.9802 and a Root Mean Squared Error (RMSE) of 2.7392. These results demonstrate that the engineered features and the model's ease of interpretation contributed to its strong performance in forecasting stock prices.
2. **Random Forest Regression:** Despite being an effective ensemble method, Random Forest yielded poor results for time-series forecasting. It returned a negative R^2 value and a high RMSE score. This performance might be attributed to the model's tendency to perform better with cross-sectional data, as opposed to time-series data, which requires handling dependencies and temporal trends in stock prices.
3. **LSTM Networks:** Although LSTM networks did not outperform Linear Regression in terms of accuracy, they were useful in identifying long-term trends, crucial for capturing temporal patterns and dependencies in large, complex datasets. Their ability to recognize these trends positions LSTMs as a valuable tool for more advanced time-series forecasting tasks.

7.2 Practical Implications:

The results indicate that Linear Regression models, when combined with efficient feature selection, are more suitable and accurate for financial forecasting tasks, especially when the dataset is relatively simple. More complicated models like LSTM could prove to be advantageous when dealing with real-time, growing data or datasets with non-linear patterns and high-frequency features.

7.3 Future Work:

- **Implication:** Future work can focus on identifying additional features, such as macroeconomic indicators or sentiment analysis of social media (e.g., tweets), to enhance forecasting performance.
- **Ensemble Methods:** Combining simple models, like Linear Regression, with more complex models such as LSTM may improve the robustness of the forecasting system.
- **Real-time Model Updates:** Continuously training the model with new data and updating the model to reflect current market trends will be a significant advantage for real-time forecasting.

References

[1] Yahoo Finance. Apple Inc. (AAPL) stock historical prices data, 2025. Available at: `https://finance.yahoo.com/quote/AAPL`

[2] Roussi, R. & GitHub Community. yfinance discussion #1084, 2025. Available at: `https://github.com/ranaroussi/yfinance/issues/1084`

[3] Brownlee, J. (2016). *Machine Learning Mastery with Python: Understand Your Data, Create Accurate Models, and Work Projects End-to-End*. Machine Learning Mastery.

[4] Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction (2nd ed.)*. Springer.

[5] Zhang, Y., & Wang, J. (2019). Stock price prediction using machine learning: A survey. *International Journal of Computer Applications*, 182(28), 40-46.

[6] Olson, D., & Delen, D. (2008). *Advanced Data Mining Techniques*. Springer.

[7] Heaton, J. (2017). *Introduction to Neural Networks with Keras*. Heaton Research, Inc.

[8] Chollet, F. (2017). *Deep Learning with Python*. Manning Publications.