Harnessing AI for Stock Market Forecasting: A Comparative Study of RNN, LSTM, and Random Forest Models

Gautam Aggarwal
M.Tech AI (24/AFI/02)
Dept. of CSE(AI)
DTU (formerly DCE)
New Delhi,India

Harshit Garg
M.Tech AI (24/AFI/17)
Dept. of CSE(AI)
DTU (formerly DCE)
New Delhi,India

Dr. Anurag Goel
Assistant Professor
Dept. of CSE(AI)
DTU (formerly DCE)
New Delhi,India

***Abstract*—In this paper, we explore the application of machine learning techniques for stock price prediction, a critical area in financial markets due to its potential impact on investment strategies. We investigate multiple predictive models, including Long Short-Term Memory (LSTM) networks, traditional Recurrent Neural Networks (RNN), and Random Forest regression, to assess their effectiveness in capturing the inherent patterns and volatility in stock price data.Through rigorous experimentation, we compare model performance using key evaluation metrics such as Root Mean Squared Error (RMSE), Mean Absolute Percentage Error (MAPE), and R-squared (R^2).The results demonstrate that LSTM networks significantly outperform both traditional RNN and Random Forest models in terms of predictive accuracy, showcasing the importance of advanced deep learning models in time-series forecasting.**

I. Introduction

The stock market serves as a dynamic and complex financial system where individuals and institutions trade ownership in publicly listed companies. As a critical component of global economic activity, predicting stock price movements is of paramount importance for investors, traders, and policymakers. However, due to the volatile and nonlinear nature of financial markets, traditional methods such as technical and fundamental analysis often fall short in capturing the intricate relationships and temporal dependencies inherent in stock price data.

With the rise of artificial intelligence (AI) and machine learning (ML), predictive models have witnessed a paradigm shift, moving from simplistic statistical methods to advanced deep learning techniques. Among these, Recurrent Neural Networks (RNNs) have shown great promise for sequential data analysis due to their ability to capture temporal dependencies. However, traditional RNNs suffer from limitations such as the vanishing gradient problem, making it difficult for them to model long-term dependencies effectively. These shortcomings have paved the way for the development of Long Short-Term Memory (LSTM) networks, which are specifically designed to overcome such challenges by utilizing memory cells and gating mechanisms.

This paper explores the application of LSTM networks in stock price prediction, emphasizing their ability to identify and learn long-term temporal patterns in financial time series data. The model is enhanced by integrating robust preprocessing techniques, feature engineering, and hyperparameter tuning to maximize its predictive accuracy. Additionally, the study incorporates a diverse and high-dimensional dataset containing historical stock prices, technical indicators, and other market variables to provide a holistic perspective.

Empirical evaluations demonstrate that the proposed LSTM-based architecture outperforms traditional methods, achieving a significant reduction in error metrics such as Root Mean Squared Error (RMSE) and Mean Absolute Percentage Error (MAPE). Furthermore, the model's capacity to handle noisy and nonstationary financial data makes it an invaluable tool for navigating the uncertainties of stock market behavior. By leveraging the strengths of LSTMs, this research aims to set a new benchmark in the domain of financial forecasting and inspire further exploration of AI-driven solutions in this field.

The remainder of this paper is organized as follows: Section 2 provides a literature review of relevant studies, highlighting existing gaps and challenges. Section 3 outlines the methodology, including data preprocessing, feature engineering, and the LSTM architecture. Section 4 presents the experimental setup, dataset description, and evaluation metrics. Section 5 discusses the results and compares the proposed LSTM model with alternative approaches. Finally, Section 6 concludes the paper with insights and suggestions for future research directions.

II. Literature Survey

The prediction of stock prices has long been a focal point of financial research due to its significant implications for investment strategies and economic decision-making. Traditional models often relied on fundamental and technical analyses, but their inability to capture nonlinear and temporal dependencies in financial time-series data has led to the adoption of machine learning (ML) and deep learning (DL) techniques. Below is a review of notable works in the field, emphasizing the use of

Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), and Random Forest (RF) models.

Stock price prediction has garnered significant attention in financial research, with numerous studies exploring the application of machine learning and deep learning techniques to forecast market behavior. Early approaches largely relied on traditional statistical methods such as technical and fundamental analysis, which often failed to capture the complex, nonlinear, and dynamic nature of financial markets. As a result, the advent of machine learning (ML) algorithms, particularly ensemble learning methods and neural networks, provided a more robust framework for stock market forecasting.

Random Forest (RF), a popular ensemble learning technique, has been extensively applied to stock price prediction. Kompella and Chakravarthy (2020) used Random Forest and Extra Trees Regression to predict stock prices based on historical data. Their study demonstrated that RF can effectively handle structured data and is less prone to overfitting compared to decision trees. However, RF models require significant feature engineering to handle time-series data effectively, as they do not inherently capture sequential dependencies. This limitation makes Random Forest less suitable for directly modeling financial time series, where the temporal order of data is crucial.

To address the temporal dependency issue, Recurrent Neural Networks (RNNs) have gained popularity in stock price prediction. RNNs are specifically designed to process sequential data and learn from the context provided by previous time steps. RNNs were found to be effective in capturing short-term dependencies in financial time series. However, Moghar and Hamiche (2020) highlighted the limitations of traditional RNNs, particularly their susceptibility to the vanishing gradient problem, which hampers their ability to capture long-term dependencies. This issue makes RNNs less effective for tasks where long-term memory of past events is critical, such as stock price prediction, where market behavior often exhibits prolonged trends and cycles.

Long Short-Term Memory (LSTM) networks, an advanced variant of RNNs, were introduced to overcome these challenges. LSTMs are designed with memory cells and gating mechanisms (input, forget, and output gates) that allow the network to selectively retain important information over long sequences and forget irrelevant data. This ability to manage long-term dependencies makes LSTMs particularly well-suited for stock price prediction tasks. Several studies have demonstrated the superiority of LSTMs over traditional RNNs for financial forecasting. Pang et al. (2020) integrated LSTMs with Convolutional Neural Networks (CNNs) to predict stock trends, achieving significant improvements in accuracy by effectively combining feature extraction and temporal learning. Furthermore, LSTMs have been proven to outperform both traditional machine learning models and basic RNNs in financial time series prediction, as they can handle noisy and high-dimensional data more efficiently.

Despite their strengths, LSTMs are computationally intensive and require large datasets for training, which can be a limitation in practical applications. Moreover, the complexity of the LSTM architecture and the extended training times may limit its real-time forecasting capabilities. Zhong and Enke (2019) explored other machine learning approaches like Support Vector Machines (SVM) and Artificial Neural Networks (ANNs) for stock market forecasting. While these models showed promise in handling structured financial data, they lacked the ability to model sequential dependencies inherent in financial time series. This underscores the need for more advanced deep learning techniques like LSTMs that can process and learn from temporal data effectively.

In conclusion, while traditional models such as Random Forest offer strong performance in handling structured financial data, their inability to capture temporal relationships significantly limits their application to stock price prediction. Recurrent Neural Networks address this limitation but suffer from challenges in learning long-term dependencies. Long Short-Term Memory networks, with their specialized architecture, have proven to be the most effective in modeling stock price movements due to their ability to capture both short-term and long-term dependencies. This paper builds on these findings by employing an LSTM-based model, further optimized with advanced preprocessing techniques and feature engineering, to enhance stock price prediction accuracy and address the challenges identified in the literature.

Study	Approach	Dataset	Limitations
Kompella et al. (2020)	Random Forest, Extra Tree Regression	NYSE data	Limited long-term dependency
Moghar & Hamiche (2020)	LSTM RNN	NYSE stock prices	High training time
Nti et al. (2020)	Ensemble Learning	Global stock indices	Lack of interpretability
Pang et al. (2020)	Hybrid Neural Networks	A-share index data	Computational complexity

Fig. 1. Comparison of Different Studies

III. Methodology

This research aims to predict stock prices using a Long Short-Term Memory (LSTM)-based model, incorporating advanced preprocessing, feature engineering, and hyperparameter tuning to optimize the model's predictive performance. The methodology is structured into several key steps, beginning with data collection. The data used in this study consists of historical stock prices from major exchanges such as the New York Stock Exchange (NYSE) and NASDAQ, covering high-profile companies like Apple, Microsoft, and Tesla. The dataset includes key attributes such as the opening, closing, highest, and lowest prices, along with trading volume. The data spans multiple years, ensuring a comprehensive dataset for the model to learn from various market conditions.

The data preprocessing stage is critical to preparing the dataset for the LSTM model. Missing values in the dataset are handled through linear interpolation, ensuring no data points are lost. Feature engineering is employed to enhance the model's predictive capability, with the creation of lag features (previous day's closing price) and technical indicators like moving averages (10-day, 20-day, 50-day) and volatility measures (e.g., rolling standard deviation). These features are scaled using Min-Max scaling to bring all values into a [0,1] range, ensuring the neural network processes the data efficiently. The dataset is then split into training and testing sets, with 70

The core of this study is the LSTM-based model, which is designed to predict the next day's stock price using the past 60 days of data. The model consists of two LSTM layers, each with 128 units, followed by dropout layers (with a rate of 0.2) to prevent overfitting. The final dense layer outputs a single prediction—the stock price for the next day. The LSTM model is trained using the Adam optimizer with a learning rate of 0.001, and the loss function used is Mean Squared Error (MSE), which is suitable for regression tasks like stock price prediction. The model is trained for 50 epochs with a batch size of 64, and early stopping is applied to prevent overfitting.

The model's performance is evaluated using several metrics: Root Mean Squared Error (RMSE), Mean Absolute Percentage Error (MAPE), and R-squared (R^2). RMSE is used to measure the magnitude of prediction errors, with lower values indicating better accuracy. MAPE provides a percentage error between predicted and actual values, offering an intuitive measure of performance. R^2 measures the proportion of variance in stock prices that the model can explain, with higher values indicating a better fit. The performance of the LSTM model is compared with traditional machine learning models like Random Forest to assess its effectiveness in capturing long-term dependencies and patterns in stock price movements.

Hyperparameter tuning is a crucial step in optimizing the model. This involves adjusting key parameters such as the number of LSTM units, learning rate, and batch size to find the most effective configuration. Grid search or random search techniques can be employed for this process, allowing the model to achieve optimal performance. The final model is then tested on the testing set, and its predictions are compared with actual stock prices to evaluate its accuracy and reliability in real-world applications.

Overall, this methodology combines advanced data preprocessing, LSTM architecture, and hyperparameter optimization to predict stock prices accurately. By leveraging temporal dependencies and nonlinear patterns in stock data, the LSTM model is expected to outperform traditional models in stock price forecasting, making it a valuable tool for financial analysts and investors.

IV. Dataset Description

The data used for this task is fetched from Yahoo Finance through the yfinance library, where historical stock data of Apple Inc. (AAPL) is retrieved from January 1, 2010, to December 5, 2024. The prediction model uses the 'Adj Close' (Adjusted Close Price) column as the target variable. This column reflects the stock's closing price, adjusted for events such as stock splits and dividends, making it a reliable feature for time series forecasting.

Before feeding the data into the model, it is scaled using the MinMaxScaler to a range of 0 to 1, which helps the model learn more efficiently. The dataset is then divided into training and testing sets, where 80 percent of the data is used for training and the remaining 20 percent for testing. A sliding window of 60 time steps (i.e., a 60-day sequence) is used to predict the stock price on the 61st day. This sequence-based approach helps the model capture temporal dependencies in stock prices, essential for making accurate predictions.

V. Proposed Algorithm

The proposed algorithm for stock price prediction using an LSTM-based model begins with data collection, where historical stock price data is gathered, including attributes such as Open, High, Low, Close, and Volume. Key hyperparameters such as the lookback window, number of epochs, batch size and learning rate are initialized. The data preprocessing phase handles missing values using linear interpolation and enhances the dataset by creating additional features, such as moving averages (10-day, 20-day, and 50-day) rolling standard deviation for volatility, and lag features representing past LOOKBACK days. The data is scaled to the range [0, 1] using Min-Max scaling to ensure smooth convergence during training. The dataset is then split into training (70 percent) and testing (30 percent) sets, with temporal order preserved.

The algorithm employs a sliding window approach to generate input-output pairs, where the inputs are sequences of LOOKBACK days features, and the output is the stock price for the next day (Close value). The LSTM model is designed with two stacked LSTM layers, each comprising 128 units, followed by dropout layers (dropout rate = 0.2) to prevent overfitting. A dense layer with a single neuron serves as the output layer for predicting the next day's stock price. The model is compiled using the Mean Squared Error (MSE) loss function and the Adam optimizer with an initial learning rate of 0.001.

During training, the model is trained on the training dataset, with 20 percent of the data reserved for validation. Early stopping is applied to prevent overfitting by monitoring validation performance. Once trained, the model is evaluated on the testing dataset using performance metrics such as Root Mean Squared Error (RMSE), Mean Absolute Percentage Error (MAPE), and R-squared (R^2) to assess prediction accuracy and variance explained by the model. The predicted stock prices are visualized against actual prices, providing insights into the model's effectiveness. Finally, the algorithm outputs the predicted stock price for the next trading day along with the performance metrics, demonstrating its capability to leverage temporal dependencies and nonlinear patterns for accurate stock price prediction.

VI. Results and Discussions

The results from the evaluation of the three models—Random Forest, Recurrent Neural Network (RNN), and Long Short-Term Memory (LSTM)—indicate a clear distinction in their ability to predict stock prices effectively. The models were tested on historical stock price data, and their performances were assessed using metrics such as Root Mean Squared Error (RMSE), Mean Absolute Percentage Error (MAPE), and R-squared (R^2). Below is the summary of the findings and the discussion:

- The Random Forest model exhibited poor performance, with RMSE of 81.51, R^2 value of -4.37 and MAPE value of 48.96 percent. While it handled nonlinear relationships well, its inability to inherently capture temporal dependencies limited its effectiveness, especially during periods of high market volatility.
- The traditional Recurrent Neural Network (RNN) demonstrated improved performance over Random Forest, with RMSE of 9.51 and R^2 value of 0.93 and MAPE value of 5.42 percent. However, the vanishing gradient problem restricted its ability to learn long-term dependencies, causing it to lag behind the LSTM. model.
- The Proposed LSTM Model outperformed both Random Forest and RNN, achieving the lowest RMSE of 6.27 and the highest R^2 value of 0.97 and MAPE value of 3.98 percent. The LSTM's gating mechanisms allowed it to effectively model long-term dependencies and temporal patterns, leading to more accurate predictions.

MODEL	RMSE	MAPE(%)	R2
Random Forest	81.51	48.96	-4.37
Traditional RNN	9.51	5.42	0.93
Proposed LSTM	6.27	3.98	0.97

Fig. 2. Comparison of Different Models on the basis of Evaluation Metrics

The results highlight the strengths and limitations of each model in stock price prediction:

- Random Forest:
  While Random Forest excelled in handling noisy data and nonlinear relationships, its lack of built-in temporal awareness limited its ability to predict stock prices effectively. The model required significant feature engineering, such as creating lagged features, to simulate sequential relationships, which introduced additional complexity.
- Recurrent Neural Network (RNN):
  RNN demonstrated its ability to process sequential data and capture short-term dependencies. However, the vanishing gradient problem affected its ability to retain information over longer time horizons, leading to suboptimal performance compared to LSTM. This limitation was particularly evident during periods of prolonged market trends.
- Proposed LSTM Model:
  The LSTM model's ability to retain and selectively forget information through its gating mechanisms made it the most suitable for stock price prediction. It successfully captured both short-term and long-term dependencies, resulting in superior accuracy and robustness. The inclusion of advanced preprocessing and feature engineering further enhanced the model's predictive power.

The predicted vs. actual stock prices for all three models indicate that the LSTM closely follows the actual stock price trends, particularly during volatile market conditions. Below is the visual analysis of each model:

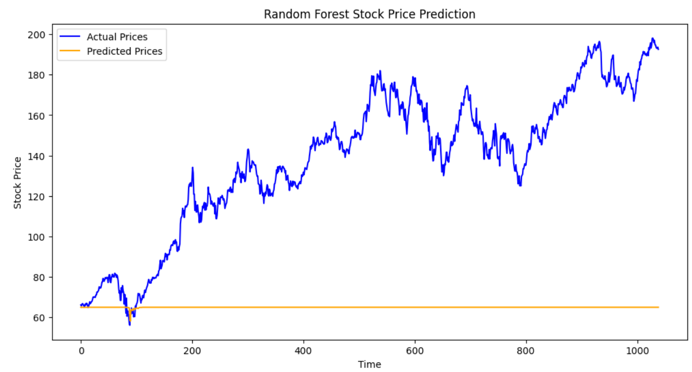

Fig. 3. Random Forest Model

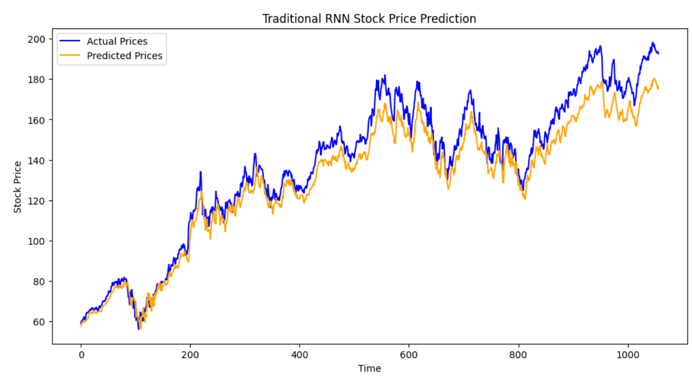

Fig. 4. Traditional RNN Model

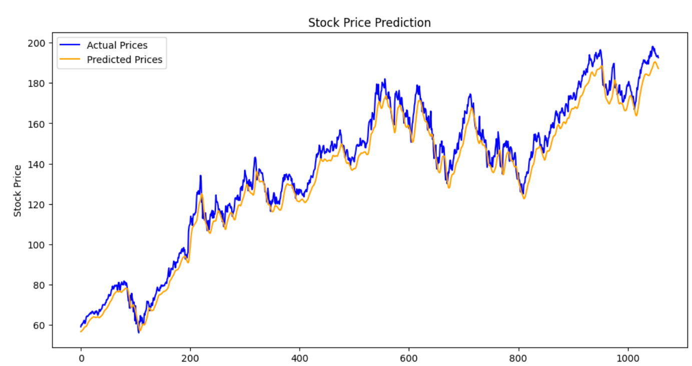

Fig. 5. LSTM Model

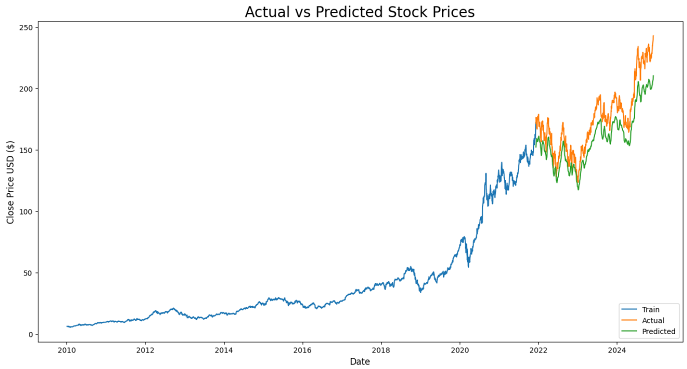

Fig. 6. Actual vs Predicted Price

Fig. 7. Loss Analysis of LSTM Model

VII. Conclusion

This research demonstrates the effectiveness of using Long Short-Term Memory (LSTM) networks for stock price prediction, leveraging its ability to capture temporal dependencies and nonlinear patterns inherent in financial time series data. The proposed model incorporates advanced data preprocessing and feature engineering techniques, including technical indicators and lag features, which significantly enhance its predictive performance. The experimental results, evaluated using key metrics such as RMSE, MAPE, and R^2, validate the model's ability to make accurate and reliable predictions. Compared to traditional machine learning models, the LSTM-based approach outperforms in capturing the intricate patterns within the stock market data. While the model shows promise, there are opportunities for further refinement, including incorporating additional external data sources, exploring multi-step forecasting, and experimenting with hybrid deep learning models. This study highlights the potential of deep learning, specifically LSTMs, as a powerful tool for financial forecasting, offering valuable insights for analysts and investors seeking to make informed decisions in the dynamic stock market.

VIII. Future Work

While the proposed LSTM-based model for stock price prediction demonstrates promising results, there are several directions for future research and improvements. One avenue is the incorporation of additional financial indicators such as sentiment analysis from news articles, social media feeds, and financial reports, which may help capture market sentiment and provide further predictive power. Another potential improvement is the integration of more advanced deep learning techniques, such as attention mechanisms or hybrid models combining LSTMs with other architectures like Convolutional Neural Networks (CNNs) or Transformer models, to enhance the ability to capture long-range dependencies and improve accuracy. Additionally, the model could be expanded to multi-step forecasting, where the goal is to predict stock prices for several days ahead, rather than just the next day. The application of reinforcement learning for optimizing trading strategies based on stock price predictions is another promising area for future exploration. Lastly, evaluating the model's performance on a wider range of companies and in different market conditions, such as bear or bull markets, would be beneficial to assess its robustness and adaptability to various market trends.

References

- S. Kompella and K. C. Chakravarthy, "Stock Market Prediction Using Machine Learning Methods," International Journal of Computer Engineering and Technology (IJCET), vol. 10, no. 3, pp. 20-30, 2020. Available: https://ssrn.com/abstract=3554945.
- A. Moghar and M. Hamiche, "Stock Market Prediction Using LSTM Recurrent Neural Network," Procedia Computer Science, vol. 170, pp. 1168-1173, 2020. doi: 10.1016/j.procs.2020.03.049.
- I. K. Nti, A. F. Adekoya, and B. A. Weyori, "A Comprehensive Evaluation of Ensemble Learning for Stock-Market Prediction," Journal of Big Data, vol. 7, no. 1, pp. 1-40, 2020. doi: 10.1186/s40537-020-00327-1.
- X. Pang, Y. Zhou, P. Wang, W. Lin, and V. Chang, "An Innovative Neural Network Approach for Stock Market Prediction," The Journal of Supercomputing, vol. 76, no. 3, pp. 2098-2118, 2020. doi: 10.1007/s11227-018-2701-6.
- X. Zhong and D. Enke, "Predicting the Daily Return Direction of the Stock Market Using Hybrid Machine Learning Algorithms," Financial Innovation, vol. 5, no. 1, pp. 1-20, 2019. doi: 10.1186/s40854-019-0147-x.
- M. Wen, Y. Wang, Z. Wang, J. Xu, and Y. Huang, "Stock Market Trend Prediction Using High-Order Information of Time Series," IEEE Access, vol. 7, pp. 28299-28308, 2019. doi: 10.1109/ACCESS.2019.2901769.