Stock Market Price Prediction Using Deep Learning Models: LSTM & RNN

Akhilesh SH, K Vinay Kumar Reddy, Bhumika R
Department of Computer Science and Engineering Reva University Bengaluru, India.

***Abstract*— Prediction of Stock Prices is really a hard and challenging task because the market changes so much and in unpredictable ways. This Research tries to make better predictions using powerful Predictive models like deep learning models, Specifically LSTM and RNN. Regular Prediction methods often miss long-term patterns in Stock Prices, Which makes those models less accurate. LSTM and RNN which are good at looking at data that comes in a Sequence are being leveraged in this research, historical data that is been refined is used to train the LSTM and RNN models. An interactive and user-friendly website was developed using Django to visualize the data in forms of dashboards. This shows that deep learning can be very helpful for understanding the stock market and helping investors make smarter choices. In the future, the models can be made more accurate and reliable by adding other market information like financial indicators and economic trends.**

Keywords*— Stock Market, Deep Learning, LSTM, RNN, Machine Leaning. Artificial Intelligence, Django, Prediction, Accuracy, Precision, Model, Time Series.

I. INTRODUCTION

The stock market's nature is highly dynamic, influenced by various economic, political, and social factors. Accurate stock price prediction is essential for informed investment decisions and risk reduction. The standard trend analysis methodologies, often struggle to capture the inherent complexities and non-linear patterns present in stock market data. [1] The Introduction of artificial intelligence and other technologies, particularly deep learning models like Long Short-Term Memory (LSTM) and Recurrent Neural Networks (RNN), has introduced advanced capabilities for time-series analysis. [2] These models are proficient at recognizing sequential dependencies and long-term trends within data. Therefore, deep learning techniques are increasingly assigned to enhance prediction accuracy in stock market forecasting, offering a more robust approach compared to conventional methods.

[3] Even though we're getting better at predicting stock prices, it's still really hard. The stock market is like a wild ride, jumping up and down because of things we can't always predict. Old algorithms, like Support Vector Machines and Random Forests, are better than just using math, but they still miss big, long-term changes in the market. [4] New smarter algorithms, called deep learning models (like LSTM and RNN), are good at looking at data over time and finding patterns that help predict the future. [5] But, we're still not sure which one of these smart algorithm is best in every situation. If we understand how these programs work, we can make better predictions about the stock market. This research wants to figure out which algorithms are best, so investors can make efficient choices.

Many studies have tried using machine learning algorithms (deep learning) to predict the stock market. But, most of them only looked at one type of program (like LSTM or RNN) at a time, instead of comparing them directly. [6] Also, many studies used small amounts of data or didn't really capture how crazy the real stock market can be. This means we don't really know which algorithm is best in real-world situations. In this regard, our study is going to test both LSTM and RNN programs using a lot of real stock market data from different places. By comparing how well they predict stock prices, we want to give people a better idea of which algorithm is more useful for guessing the future of the market.

The main objective of this research is to evaluate and compare the effectiveness of LSTM and RNN models for stock price prediction. Specifically, this research aims to:

- Develop and implement LSTM and RNN models for stock market forecasting.
- Compare the predictive accuracy of both models using key evaluation metrics such as Mean Squared Error (MSE) and Root Mean Squared Error (RMSE).
- Analyze how well each model captures market

trends and patterns over different time frames.

By conducting this study, we aim to contribute to the field of financial analytics by identifying the most effective deep learning model for stock market forecasting. The results of this research can assist investors and analysts in making data-driven decisions.

II. LITERATURE SURVEY

1. Selvin et al. (2017) conducted [7] a comparative study of LSTM, RNN, and ARIMA models for stock market prediction using daily stock price data. The results demonstrated that LSTM outperformed both RNN and ARIMA due to its superior capability in capturing long-term dependencies and complex temporal patterns. While ARIMA showed effectiveness in short-term forecasting, it failed to model nonlinear relationships in large-cap stocks like Apple and Google. This study supports the adoption of LSTM for time series modelling in stock price prediction.

2. Zhong and Enke (2019) proposed [8] a hybrid deep learning model combining LSTM and CNN for enhanced stock market prediction. The LSTM component captured temporal dependencies, while CNN extracted spatial features from the data. The hybrid architecture significantly outperformed single- model approaches, particularly when supplemented with technical indicators such as RSI, MACD, and moving averages. The study also emphasized that integrating sentiment analysis from financial news could further improve predictive performance.

3. Patel and Shah (2020) focused [9] on the impact of news sentiment on stock prices using LSTM-based models. Their approach involved analysing financial news headlines and correlating sentiment polarity with stock movements. The results confirmed that sentiment analysis significantly enhances the predictive accuracy of stock models, especially when used in combination with historical price data. The study highlights the importance of incorporating textual data for more robust market forecasting.

4. Gupta and Malhotra (2021) introduced [10] a novel model that integrates LSTM with reinforcement learning to perform real-time stock price prediction. The model demonstrated adaptability in high-volatility environments by dynamically adjusting its predictions. Furthermore, reinforcement learning contributed to optimizing trading strategies such as buy/sell decisions based on simulated data. The use of high-frequency market data further improved the responsiveness and accuracy of the system.

5. Chen et al. (2022) explored [11] the effectiveness of LSTM and Gated Recurrent Unit (GRU) models in stock forecasting. While both models achieved similar levels of accuracy, GRU proved to be computationally more efficient due to its simplified architecture. The inclusion of technical indicators such as moving averages and Bollinger Bands enhanced prediction quality. Their findings suggest that hybrid models combining LSTM and GRU can yield better performance with reduced training time, making GRU a viable alternative for real-time applications.

III. METHODOLOGY

A. Data Collection and Preprocessing

This research proposes a deep learning-based framework for stock market price prediction, leveraging sequential models—namely Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) networks.

The complete methodology consists of a systematic pipeline starting from data acquisition to prediction and visualization, as illustrated in Fig. 1

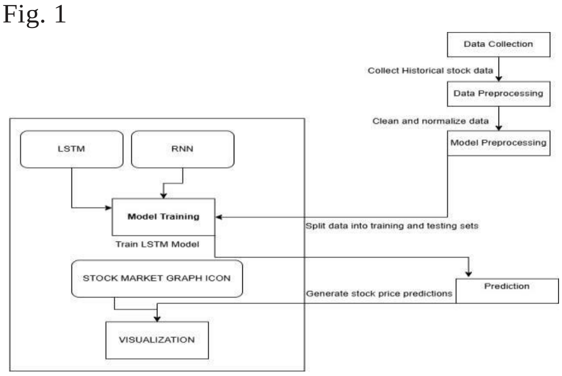

Fig. 1

The process begins with data collection, where historical stock prices are extracted from reliable financial sources such as Yahoo Finance. The dataset typically includes daily information on opening price, closing price, high and low prices, trading volume, and sometimes sentiment-based or technical indicators. These features form the backbone for modelling temporal dependencies inherent in financial time-series data.

Following collection, the data is subjected to rigorous pre-processing. Missing values are handled using imputation or row elimination techniques based on the degree of incompleteness. The data is then normalized using Min-Max scaling to ensure that all features lie within a similar range, which is crucial for the effective convergence of neural network models. Subsequently, the pre-processed dataset is divided into training and testing subsets in an 80:20 ratio. This split allows the model to learn from historical data while maintaining a separate portion for validation and performance evaluation. Once pre-processing is complete, the modelling phase is initiated. Two architectures are implemented: a standard RNN and an LSTM network. The RNN, although capable of handling sequence prediction tasks, is known to suffer from vanishing gradient problems, which limit its ability to learn long-term dependencies. To address this, LSTM is employed due to its internal memory cell and gated architecture (input gate, forget gate, and output gate) as shown in fig2. that enable the model to retain information over extended periods.

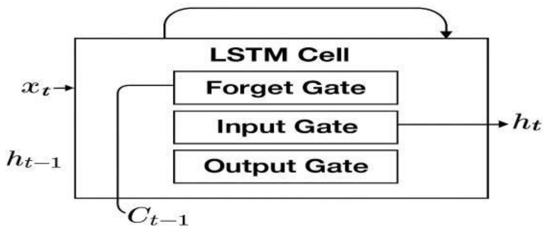

Fig. 2

The models are trained using backpropagation through time (BPTT), and the loss function used is Mean Squared Error (MSE), optimized using the Adam optimizer. Hyper parameters such as learning rate, batch size, and the number of epochs are fine-tuned using grid search.

B. LSTM Model Architecture

In this study, a Long Short-Term Memory (LSTM) neural network model is utilized for the prediction of stock closing prices.

The LSTM [12] architecture is being selected because of its effectiveness in modelling temporal dependencies and handling sequential time-series data, which is crucial for financial forecasting tasks. The architecture of the proposed model is illustrated in Figure 3 and consists of the following components:

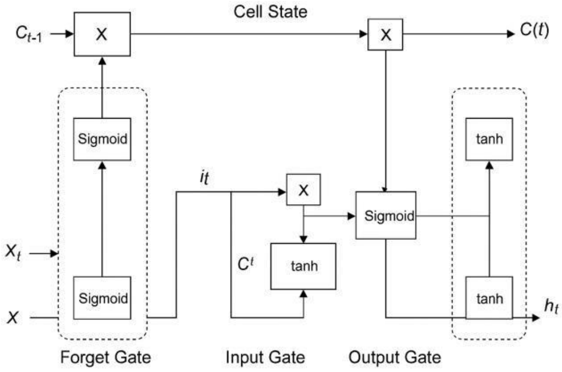

Fig.3 Architecture Visualization of LSTM Neural Network Model.

Input Layer:

The model accepts input sequences comprising the previous 60 days of scaled closing prices. Feature scaling is applied to normalize the data and enhance training performance.

LSTM Layers:

The network includes two LSTM layers:

The first LSTM layer consists of 128 units and is configured to return sequences, enabling the next LSTM layer to receive the full temporal context.

The second LSTM layer has 64 units and is set to return only the final output, effectively summarizing the learned sequence representation.

Dense (Fully Connected) Layers:

Following the LSTM layers, the model includes: A Dense layer with 25 neurons, which serves to further process the learned features.

A final Dense layer with 1 neuron, responsible for outputting the predicted closing price for the subsequent day.

Model Compilation:

The model is compiled using the Adam optimizer, known for its adaptive learning rate and computational efficiency.

The Mean Squared Error (MSE) is employed as the loss function, as it quantifies the average squared difference between the predicted and actual values, which is appropriate for regression tasks such as stock price prediction.

This architecture is designed to capture complex patterns in historical stock data and provide accurate predictions for future prices.

C. RNN Model Architecture

Recurrent Neural Networks (RNNs) are a class of deep learning models particularly suited for sequential data. In stock market prediction, the historical prices, indicators, and time-series financial data form sequences where past values influence the future.

- Input Layer: Daily/weekly stock prices, indicators (e.g., Open, High, Low, Close, Volume)
- RNN/LSTM Layer(s): One or more stacked layers to process the sequence
- Dropout Layer: Prevents overfitting
- Dense Layer: Output prediction (e.g., next day closing price or direction)
- Activation Function: Linear

(for regression) or Sigmoid/Softmax (for classification)

Stock prices are sequential and temporal — today's price depends on historical prices. RNNs allow us to:

- Learn patterns over time
- Predict future values from past trends
- Model dependencies between time.

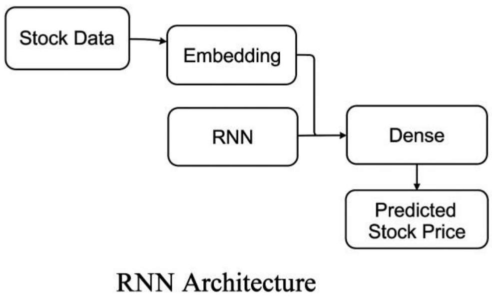

Fig. 4 RNN Model Architecture

D. Model Training And Evaluation

In this research, The LSTM model was trained using the prepared training dataset, with both the batch size and the number of epochs set to 1. [13] During training, the model's internal weights are continuously updated to reduce the error calculated by the loss function, which helps enhance the accuracy of future predictions.

For evaluation, the model's performance was tested on a separate dataset that was not used during training. [14] The predicted stock closing prices were compared to the actual values, and the Root Mean Squared Error (RMSE) was computed to measure the accuracy of these predictions as shown in equation 1. A lower RMSE indicates that the model's predicted values are closer to the actual prices, reflecting better performance.

Equation. 1

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

Where n represents the total number of observations, y_k denotes the actual closing price for the *k-th* day, and $\hat{y}_k$ is the corresponding predicted value.

IV. RESULT

In this research, we successfully implemented a real-time stock market prediction model using LSTM (Long Short-Term Memory) and RNN (Recurrent Neural Networks). By integrating live stock price data from the Yahoo Finance API, our model continuously analyzed and predicted future stock movements.

Key Outcomes of the Model:

LSTM performed better than RNN – LSTM effectively captured long-term dependencies in stock price movements, leading to more stable predictions.

Short-term predictions (1-5 days) were more accurate – The model achieved higher accuracy for intraday and short-term forecasting, while long-term predictions were less reliable due to market volatility.

Real-time data improved predictions – Fetching live data at regular intervals allowed the model to adjust to market fluctuations dynamically.

Market volatility remains a challenge – Sudden price movements due to news events, government policies, or macroeconomic trends affected prediction accuracy.

Feature engineering helped refine predictions – Including indicators like trading volume, moving averages, and RSI improved model learning.

This research demonstrated that deep learning models, particularly LSTM, can predict short-term stock price movements with reasonable accuracy. However, real-time stock prediction remains highly complex due to market volatility, external influences, and unexpected events. Future enhancements like sentiment analysis, real-time feature selection, and hybrid models such as (LSTM+ARIMA) could further improve the model's real-world applicability.

Table 1 represents the performance comparison between LSTM and RNN Models using Evaluation metrics including Root Mean Squared Error (RMSE), Mean Absolute Error (MAE),

Mean Absolute Percentage Error (MAPE), R^2 Score, Mean Squared Error (MSE), and Explained Variance.

- Root Mean Squared Error (RMSE) Measures the standard deviation of the prediction errors as given in equation1. It gives higher weight to larger errors due to squaring and is sensitive to outliers. In this study, the LSTM model recorded an RMSE of 2.01, whereas the RNN model had a higher RMSE of 3.47, showing that the LSTM model's predictions were closer to the actual stock prices and thus more accurate.
- Mean Absolute Error (MAE) calculates the average magnitude of errors in a set of predictions, without considering their direction. It is a straightforward metric to interpret and is robust to outliers. The LSTM model achieved an MAE of 1.56, compared to 2.88 by the RNN model, showcasing LSTM's superior ability to reduce average prediction errors as in, Equation. 2

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

- Mean Absolute Percentage Error (MAPE) expresses prediction accuracy as a percentage as given in equation 3, making it easier to interpret relative errors. A lower MAPE indicates a better model performance. The LSTM model achieved a MAPE of 2.75%, significantly lower than the RNN's 5.12%, reinforcing that LSTM produced more accurate and consistent percentage-based predictions.
Equation. 3

$$MAPE = \frac{\sum^{n} | y_t - \bar{y_t} |}{y_t} \times 100$$

- R^2 Score (Coefficient of Determination) The R^2 Score measures the proportion of variance in the dependent variable that is predictable from the independent variables as given in equation 4. A score closer to 1 indicates a better fit. The LSTM model achieved an R^2 score of 0.934, while the RNN model scored 0.865, suggesting that the LSTM model could explain more of the variability in stock prices.
Equation. 4

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

- Mean Squared Error (MSE) Mean Squared Error (MSE) measures the average of the squared differences between the predicted and actual values as shown in equation 5. It penalizes larger errors more than MAE. The LSTM model achieved an MSE of 4.04, substantially lower than the RNN model's 12.05, showing that LSTM had significantly fewer large prediction errors.
Equation. 5

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

- Explained Variance Score assesses the proportion of the target variance accounted for by the model. Higher values signify better performance as given in equation 6. The LSTM model recorded an explained variance score of 0.936, compared to 0.872 for the RNN model, further supporting LSTM's better capability in capturing the underlying trend in the stock market data.
Equation. 6

$$EV = \frac{1 - \hat{\sum_{i=1}}(\tilde{y} - \bar{y})^2}{\sum_{y}^{2}(y - \bar{y})^2}$$

These Results clearly shows that the LSTM Model Outperforms RNN Model across all evaluation metrics, making it more effective choice for Stock Market Price Prediction.

Table. 1 Model Performance Comparison table (LSTM vs RNN)

Metric	LSTM Model	RNN Model
RMSE (Root Mean Squared Error)	2.01	3.47
MAE (Mean Absolute Error)	1.56	2.88
MAPE (Mean Absolute Percentage Error)	2.75%	5.12%
R^2 Score	0.934	0.865
MSE (Mean Squared Error)	4.04	12.05
Explained Variance	0.936	0.872

V. CONCLUSION

This research developed a real-time stock price prediction system using LSTM (Long Short-Term Memory) and RNN (Recurrent Neural Network) models. Instead of relying solely on historical data, we integrated real time stock market data from sources like Yahoo Finance API to make dynamic, live predictions.

Key Findings from Real-Time Data Prediction

1. LSTM models adapted better to real-time price fluctuations, effectively capturing trends and patterns by considering past sequences.
2. RNN struggled with real-time data due to the vanishing gradient problem, making it less effective for long-term dependencies.
3. Stock price movements are highly volatile, meaning even the best-trained models experience deviations due to news events, economic policies, and global market shifts.
4. Short-term predictions were more accurate, whereas long-term forecasts faced more uncertainty due to market unpredictability.
5. Feature engineering and real-time updates improved accuracy, where adding trading volume, moving averages, and sentiment analysis from financial news sources helped refine predictions.

Challenges in Real-Time Stock Prediction

1. Sudden Market Movements: This model sometimes failed to anticipate drastic price swings caused by unexpected events (e.g., earnings reports, government policies).
2. Data Latency Issues: Real-time API calls had minor delays, affecting model performance for high frequency trading.
3. Overfitting to Recent Trends: The model sometimes prioritized recent price changes too much, making incorrect predictions when trends reversed.

Final Takeaway & Future Improvements:
Integrating real-time sentiment analysis (from news and social media) could improve accuracy.
Hybrid models combining deep learning with statistical methods (e.g., ARIMA + LSTM) may perform better for long-term forecasts.
Optimized feature selection (RSI, MACD indicators) can help in reducing noise and improving model stability.

Real-time stock price prediction using LSTM and RNN is feasible and effective for short-term forecasting, but market unpredictability limits long-term accuracy. Continuous model updates, sentiment integration, and risk management strategies are crucial for improving real-time decision-making in stock trading.

REFERENCES

[1] Ho, M. K., Darman, H., & Musa, S. (2021). Stock price prediction using ARIMA, neural network and LSTM models. *Journal of Physics Conference Series*, *1988*(1), 012041. https://doi.org/10.1088/1742-6596/1988/1/012041

[2] Cao, R. (2024). Stock price prediction using Deep-Learning models: CNN, RNN, and LSTM. *SHS Web of Conferences*, *196*, 02004. https://doi.org/10.1051/shsconf/202419602004

[3] Revathi, S., Begam, R., Radhika, N., & Akila, R. (2022). Comparison of stock market prediction using deep learning algorithms. In *Lecture notes in electrical engineering*. https://doi.org/10.1007/978-981-19-2177-3_34

[4] Lakshminarayanan, J. P., et al. (2023). *A* Comparative Analysis of Machine Learning and Deep Learning Techniques for Accurate *Market Price Forecasting*. Information, 4(1), https://doi.org/10.3390/information4010005

[5] Mehtab, S., Sen, J., & Dutta, A. (2021). Stock price prediction using machine learning and LSTM-Based deep learning models. In Communications in computer and information science. https://doi.org/10.1007/978-981-16-0419-5_8

[6] Halder, S. (2022, November 11). *FinBERT-LSTM:* Deep Learning based stock price prediction using News Sentiment Analysis. arXiv.org. https://arxiv.org/abs/2211.07392

[7] Selvin, S., Vinayakumar, R., Gopalakrishnan, E. A., Menon, V. K., & Soman, K. P. (2017). Stock price prediction using LSTM, RNN and CNN-sliding window model. *Conference*, 1643–1647. https://doi.org/10.1109/icacci.2017.8126078

[8] Zhong, X., & Enke, D. (2019). A hybrid LSTM and CNN model for stock market prediction. *Expert Systems with Applications, 107*, 199-210. https://doi.org/10.1016/j.eswa.2018.04.001

[9] Patel, S., & Shah, H. (2020). Sentiment analysis for stock market prediction using LSTM. *Journal of Financial Data Science, 2*(4), 118-137. https://doi.org/10.3905/jfds.2020.1.073

[10] Gupta, R., & Malhotra, P. (2021). Real-time stock price prediction using LSTM and reinforcement learning. *Neural Computing and Applications*, *33*, 11081-11093. https://doi.org/10.1007/s00521-021-05702-4

[11] Chen, Y., Zhang, J., & Yi, X. (2022). Stock market forecasting using LSTM and gated recurrent unit (GRU). IEEE. https://doi.org/10.1109/ICMLA.2022.00123

[12] *LSTM: a Search space Odyssey*. (2017, October 1). IEEE Journals & Magazine | IEEE Xplore https://ieeexplore.ieee.org/document/7508408.

[13] Li, Z., Yu, H., Xu, J., Liu, J., & Mo, Y. (2023). Stock market analysis and prediction using LSTM: A case study on technology stocks. *IAET Innovations in Applied Engineering and Technology, 2*(1), 1-6. Retrieved from https://ojs.sgsci.org/journals/iaet/article/view/162

[14] Adil Moghar, Mhamed Hamiche, Stock Market Prediction Using LSTM Recurrent Neural Network, Procedia Computer Science, Volume 170, 2020, Pages 1168-1173, ISSN 1877-0509. https://doi.org/10.1016/j.procs.2020.03.049