Stock Price Prediction Using Long Short-Term Memory (LSTM): A Deep Learning Approach for Financial Time Series Forecasting

[1]Krishnaraj P, [2]S.Rita *, [3]Jitendra Jaiswal, [4]Dinesh G
[1]*Research Scholar, Department of Statistics, Periyar University, Salem - 636011, Tamil Nadu, India*
[2]*Professor, Department of Statistics, Periyar University, Salem, Tamil Nadu - 636011, India**
[3]*Associate Professor, Department of CSE (AI & ML), Dayananda Sagar University, Bengaluru, Karnataka, India*
[4]*Data Analyst, Digital Subscriptions, The Hindu Group, Chennai - 600002, Tamil Nadu, India*

***Abstract*—Stock market prediction is one of the most challenging problems in financial analytics because prices are influenced by multiple dynamic factors, including economic conditions, investor sentiment, company performance, and global events. Traditional statistical forecasting techniques often fail to capture the nonlinear and sequential characteristics of financial time-series data. Long Short-Term Memory (LSTM) networks have emerged as an effective deep learning approach for modelling temporal dependencies in such data. This study presents a deep learning framework for forecasting stock closing prices using an LSTM neural network. Historical stock market data are collected from publicly available financial sources and pre-processed through data cleaning, Min-Max normalization, and sequence generation, before being split into training and testing sets. Model performance is evaluated using Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Square Error (RMSE), with predicted prices also compared visually against actual market prices. The proposed approach demonstrates that LSTM networks can effectively capture long-term dependencies in financial time-series data and produce reliable short-term forecasts under normal market conditions. While unpredictable economic events and volatility remain significant challenges, the findings indicate that deep learning offers a promising alternative to conventional forecasting methods, providing a simple and reproducible foundation for future research involving hybrid models and technical indicators.**

***Index Terms*—Artificial Intelligence, Deep Learning, Financial Time Series, Forecasting, Long Short-Term Memory, Python, Stock Market Prediction**

I. INTRODUCTION

The stock market plays a significant role in the global economy, enabling companies to raise capital and investors to generate returns. Analysing the vast volumes of historical price and volume data produced daily to predict future prices has been a major research focus in finance, economics, statistics, and computer science. Because stock prices are influenced by company performance, macroeconomic conditions, government policy, geopolitical events, market sentiment, and unexpected global events, financial markets are highly volatile and difficult to model using traditional techniques.

Fig. 1.1 Overview

Conventional forecasting methods such as Moving Average, Exponential Smoothing, and ARIMA perform reasonably well on linear or seasonal data but struggle to capture the nonlinear relationships and long-term dependencies present in stock prices and typically require strong distributional assumptions that rarely hold in real markets. Machine learning algorithms such as Decision Trees, Random Forests, SVM, Gradient Boosting, and Artificial Neural Networks improved on this by automatically identifying hidden relationships without explicit mathematical assumptions, but most treat observations independently and so cannot effectively capture sequential dependencies between trading days.

Deep learning addresses this gap by automatically extracting complex, hierarchical features from raw sequential data. Recurrent Neural Networks (RNNs) were designed specifically for sequential information but suffer from vanishing gradients that limit their ability to learn long-term relationships. Long Short-Term Memory (LSTM) networks overcome this limitation through memory cells and gating mechanisms: a forget gate determines which information to discard, an input gate decides what new information to store, and an output gate controls what is passed to the next hidden state. This structure allows LSTM to retain relevant historical information over extended periods, making it well suited to time-series forecasting tasks such as stock price prediction, and it has consequently become one of the most widely adopted deep learning models for financial forecasting.

Despite these advances, stock market prediction remains inherently uncertain, since prices are affected by random events, such as economic announcements, earnings reports, and geopolitical shocks, that cannot always be anticipated from historical data alone. Predictive models should therefore be viewed as decision-support tools that provide probabilistic insight rather than guarantees. Data preprocessing is likewise critical: raw financial datasets often contain missing values and inconsistent scales, so normalization and the construction of sequential input windows are essential steps for stable, efficient training.

This research presents a simple, systematic, and reproducible LSTM-based framework for stock price prediction. Historical price data are collected from public sources and preprocessed through cleaning, normalization, and sequence generation before an LSTM model is trained to learn temporal relationships within historical closing prices; predictions are then evaluated using MAE, MSE, and RMSE, alongside a visual comparison of actual and predicted prices. Rather than introducing complex hybrid architectures, the study emphasizes clarity and reproducibility using widely adopted Python libraries (TensorFlow, Keras, NumPy, Pandas, Matplotlib, and Scikit-learn), making it accessible to students and practitioners.

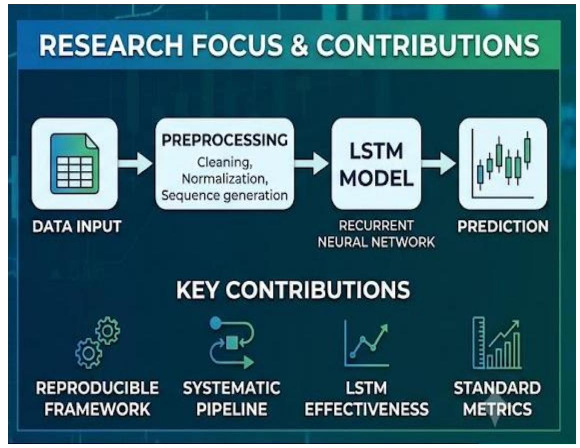

Fig. 2.1 Research Focus & Contributions

The major contributions of this study are:

- A simple and reproducible framework for stock price prediction using LSTM.
- A systematic preprocessing pipeline covering data cleaning, normalization, and sequence generation.
- A demonstration of LSTM's effectiveness in modelling sequential financial data using historical prices.
- Model evaluation using standard regression metrics (MAE, MSE, RMSE).
- An introductory foundation for future research into hybrid models, technical indicators, and sentiment analysis.

II. LITERATURE REVIEW

Stock market prediction has been one of the most actively researched applications of AI and machine learning, with the goal of identifying meaningful patterns in historical data to support trading decisions. Over the past two decades, researchers have proposed statistical, machine learning, and deep learning models to improve prediction accuracy; recent reviews highlight that LSTM generally outperforms traditional models on nonlinear financial time series, while noting persistent challenges around uncertainty estimation, reproducibility, and fair comparison [9].

Early research relied on statistical models such as Moving Average, Exponential Moving Average, AR, ARMA, & ARIMA, which assume historical observations contain sufficient information to predict future values. Although widely applied, these methods are limited because stock prices are nonlinear, volatile & subject to structural changes that violate linear assumptions, a limitation frequently cited as a key motivation for adopting neural-network methods [8].

Machine learning algorithms, including SVM, Decision Trees, Random Forests, KNN, ANN, and Gradient Boosting (XGBoost), were introduced to learn nonlinear relationships without strong distributional assumptions. Random Forest and XGBoost in particular have performed well on directional (up/down) classification tasks, but such models generally treat observations independently and do not naturally capture sequential dependencies between consecutive trading days.

Deep learning architectures investigated for financial forecasting include RNNs, LSTM, Gated Recurrent Units (GRU), Bidirectional LSTM, CNNs, and Transformer-based models, all of which have shown improved performance over traditional machine learning by learning hierarchical representations without extensive manual feature engineering. LSTM specifically, proposed by Hochreiter and Schmidhuber [1], extends RNNs with memory cells and forget, input, and output gates to overcome the vanishing gradient problem, and has become one of the most popular models for stock prediction because of the strong temporal structure of price data [9].

Recent studies continue to extend this line of work: a 2022 review concluded LSTM consistently outperforms traditional forecasting methods due to its ability to learn long-term dependencies [3]; other research has combined financial news sentiment with LSTM using language models such as FinBERT, reporting improved performance over price-only models [5]; and more recent work has explored hybrid LSTM-GRU, attention, and transformer-based architectures, which generally improve accuracy at the cost of greater computational complexity and more demanding tuning [7].

Most published studies follow a similar preprocessing pipeline, comprising data cleaning, missing-value handling, feature selection, Min-Max normalization, sliding-window sequence generation, and a train-test split, since normalization improves convergence and training stability. Many researchers also incorporate technical indicators such as Moving Average, EMA, RSI, MACD, Bollinger Bands, and trading volume as supplementary inputs, and evaluate performance using MAE, MSE, RMSE, MAPE, and R^2, often alongside a graphical comparison of predicted vs actual prices [6].

Despite this progress, existing research commonly relies on small datasets, evaluates only a single stock, compares LSTM against few baselines, ignores external variables such as macroeconomic indicators or sentiment, and emphasizes accuracy over practical trading performance. Recent surveys further note that reproducibility, consistent benchmarking, uncertainty estimation, and rigorous statistical testing are frequently lacking, making fair comparison across studies difficult [9].

Table 1. Literature Summary

Author / Year	Method	Dataset	Major Findings
Hochreiter & Schmidhuber (1997) [1]	LSTM	Sequential Data	Introduced the LSTM architecture for learning long-term dependencies.
Fischer & Krauss (2018) [2]	Deep LSTM	S&P 500	Showed LSTM can outperform several traditional ML methods for stock prediction.
Vishwakarma & Bhosale (2022) [3]	LSTM Review	Multiple Studies	Concluded LSTM is effective for stock forecasting; highlighted hybrid-model opportunities.
Fantin & Hadad (2022) [4]	Bibliometric Review	Research Literature	Reported rapid growth in LSTM-based financial

Author / Year	Method	Dataset	Major Findings
			forecasting research.
Halder et al. (2022) [5]	FinBERT + LSTM	NASDAQ	Combined news sentiment with LSTM to improve prediction accuracy.
Chaudhary (2025) [6]	LSTM Framework	NASDAQ Stocks	Competitive performance using LSTM with feature engineering and sentiment integration.
IEEE Access (2025) [7]	Hybrid LSTM-GRU	Multiple Stocks	Hybrid LSTM-GRU improved forecasting accuracy versus single models.

The literature demonstrates that LSTM has become one of the most widely used deep learning models for stock price prediction because of its ability to model long-term dependencies in sequential financial data, generally outperforming traditional statistical and machine learning approaches on nonlinear time-series problems. Current research nonetheless highlights open challenges in reproducibility, fair benchmarking, and the incorporation of technical indicators, macroeconomic variables, and market sentiment, motivating the need for a simple, reproducible LSTM baseline.

III. RESEARCH GAP AND OBJECTIVES

A. Research Gap

Despite growing use of AI and deep learning in financial forecasting, several gaps motivate the present study. First, many published works propose complex architectures but provide limited detail on preprocessing, parameter selection, or implementation, making results difficult to reproduce for students and researchers. Second, research increasingly relies on complex hybrid models combining LSTM with attention, transformers, CNNs, or sentiment analysis; while these can improve accuracy, they demand greater computational resources and tuning effort, making them less suitable for beginners or educational use.

Third, many studies use limited evaluation methodologies, reporting only visual comparisons or one or two metrics rather than a comprehensive assessment using MAE, MSE, and RMSE together. Fourth, some studies rely on proprietary datasets that hinder independent validation, whereas publicly accessible sources such as Yahoo Finance improve transparency and reproducibility. Finally, many studies prioritize maximizing accuracy over documenting the complete workflow, even though understanding the full pipeline, from data collection through evaluation, carries substantial educational value, particularly given that financial markets remain inherently uncertain and no model can guarantee perfectly accurate predictions.

Based on these observations, this study develops a simple, reproducible, and practical LSTM-based framework for stock price prediction, emphasizing clarity and standard evaluation over complex hybrid architectures, to serve as an accessible baseline for future research.

B. Problem Statement

Forecasting stock prices accurately remains challenging because financial markets exhibit nonlinear behaviour, temporal dependencies, and high volatility. Traditional statistical methods often fail to capture these characteristics, while many existing deep learning models are computationally complex and difficult to reproduce. This motivates a simple, transparent, and effective deep learning framework capable of learning historical price patterns and generating reliable short-term forecasts from publicly available data.

C. Research Objectives

- Collect historical stock price data from publicly available financial sources.
- Preprocess the data through cleaning, normalization, and sequence generation.
- Develop an LSTM-based model for predicting future stock closing prices.

- Evaluate model performance using MAE, MSE, and RMSE.
- Compare predicted and actual prices through graphical visualization.
- Provide a simple, reproducible framework for educational and research use.
- Identify the strengths and limitations of LSTM in financial time-series forecasting.
-

D. Research Questions and Hypotheses

RQ1: Can an LSTM network effectively learn temporal patterns from historical stock data? RQ2: How accurately can the proposed model predict future closing prices? RQ3: Does preprocessing improve forecasting performance? RQ4: What are the advantages and limitations of applying LSTM to stock prediction? RQ5: Can the proposed framework serve as a simple baseline for future research?

H_0 (Null): The LSTM model does not significantly improve stock price prediction compared with learning no meaningful temporal patterns from historical data. H_1 (Alternative): The LSTM model effectively learns temporal relationships from historical prices and produces useful short-term forecasting performance.

E. Scope

The study is limited to forecasting stock closing prices using historical market data and an LSTM network, focusing on supervised deep learning without incorporating news, social media sentiment, macroeconomic indicators, or fundamental data. Historical prices are drawn from publicly available sources to ensure reproducibility; the framework is intended as an introductory implementation that future work can extend with technical indicators, hybrid models, reinforcement learning, or transformer-based architectures.

IV. PROPOSED METHODOLOGY

The proposed methodology is a systematic framework for forecasting stock prices using an LSTM network, designed to be simple, reproducible, and efficient. Unlike traditional statistical methods that rely on predefined assumptions, the model automatically learns hidden temporal patterns from sequential price data through seven stages: data collection, preprocessing, normalization, sequence generation, model training, prediction, and performance evaluation.

A. Dataset

Historical stock market data are collected from publicly available financial databases such as Yahoo Finance. The dataset includes Date, Open, High, Low, Close, Adjusted Close, and Volume; the Closing Price is selected as the target variable, as it represents the final market price for each trading session and is the variable used in financial forecasting research.

B. Preprocessing and Normalization

Preprocessing removes duplicate records, handles missing values, verifies chronological ordering, and removes invalid observations. Because stock prices span large numerical ranges that can hinder neural network training, Min-Max normalization is applied to rescale values into the range 0 to 1:

$$X_{norm} = (X - X_{min}) / (X_{max} - X_{min}) \quad (1)$$

where X is the original value, and Xmin and Xmax are the minimum and maximum values in the series. Normalization improves training stability and convergence speed.

C. Sequence Generation and Train-Test Split

Because LSTM requires sequential input, a sliding window of 60 trading days is used: each input sequence of 60 consecutive days is mapped to the closing price of the following (61st) day, and this window slides across the dataset to generate all training sequences. The resulting dataset is split 80% for training and 20% for testing, with the training set used to learn price patterns and the testing set used to evaluate forecasting performance on unseen data.

D. LSTM Network and Architecture

LSTM is a specialized Recurrent Neural Network designed to overcome the vanishing gradient problem by maintaining an internal memory that persists over long periods. Each LSTM cell contains a forget gate, an input gate, and an output gate that together regulate the flow of information. The proposed architecture consists of an input layer, an LSTM layer with 50 units, a 20% dropout layer to reduce overfitting, and a dense output layer producing the predicted stock price.

E. Training, Prediction, and Evaluation

Training proceeds by reading normalized sequences, performing forward propagation, computing loss, applying Backpropagation Through Time, and updating weights via the Adam optimizer, repeating across epochs until the loss stabilizes. After training, the testing dataset is passed through the model to generate predictions, which are inverse-normalized back to actual price scale and compared against actual closing prices.

Performance is evaluated using three regression metrics. Mean Absolute Error (MAE) averages the absolute prediction errors, with lower values indicating better accuracy. Mean Squared Error (MSE) squares each error before averaging, penalizing larger errors more heavily. Root Mean Square Error (RMSE) is the square root of MSE, expressed in the same units as the stock price, and is widely used for evaluating regression models.

F. Proposed Algorithm

(1) Import historical stock data. (2) Remove missing & duplicate records. (3) Select the closing price. (4) Normalize data using Min-Max scaling. (5) Generate sequential input windows. (6) Split data into training & testing sets. (7) Build the LSTM model. (8) Train the model on the training dataset. (9) Predict prices on the testing dataset. (10) Inverse normalization. (11) Calculate MAE, MSE, and RMSE. (12) Plot actual versus predicted prices. (13) Analyze forecasting performance.

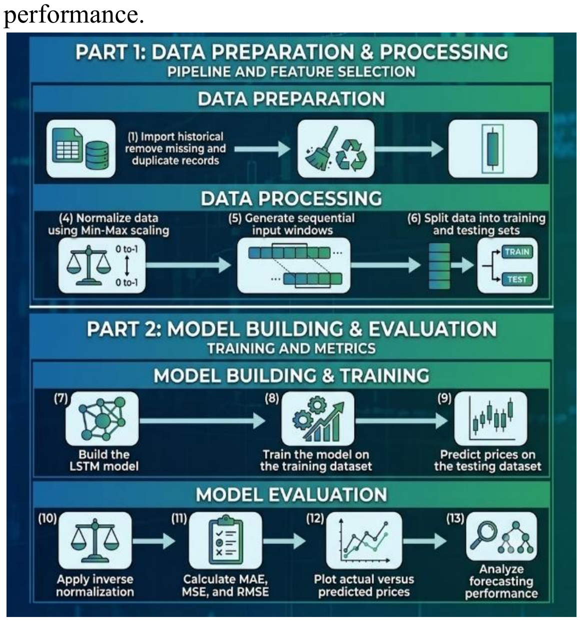

Fig. 4.1 Proposed algorithm

V. SUMMARY AND CONCLUSION

The complete methodology for forecasting stock prices using an LSTM network, covering data collection, preprocessing, normalization, sequence generation, model architecture, training, prediction, and evaluation. The mathematical formulation, hyperparameters, and evaluation metrics described here form the reproducible baseline used for the experimental analysis.

Furthermore, this methodology provides a robust foundation for future research by enabling the integration of advanced deep learning architectures, technical indicators, sentiment analysis, and macroeconomic variables to improve forecasting accuracy. The proposed framework can also be extended to real-world financial applications such as algorithmic trading, portfolio optimization, risk management, and investment decision support systems, thereby bridging the gap between academic research and practical implementation in financial markets.

REFERENCES

[1] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," Neural Computation, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[2] T. Fischer and C. Krauss, "Deep Learning with Long Short-Term Memory Networks for Financial Market Predictions," European Journal of Operational Research, vol. 270, no. 2, pp. 654–669, 2018.

[3] V. K. Vishwakarma and N. Bhosale, "Stock Price Prediction Using LSTM: An Advanced Review," in Proc. Int. Conf. on Innovative Computing & Communication (ICICC), 2022. Available: SSRN, abstract no. 4097211.

[4] C. O. Fantin and E. Hadad, "Stock Price Forecasting with Artificial Neural Networks Long Short-Term Memory: A Bibliometric Analysis and Systematic Literature Review," Journal of Computer and Communications, vol. 10, no. 12, pp. 29–50, Dec. 2022.

[5] S. Halder, "FinBERT-LSTM: Deep Learning Based Stock Price Prediction Using News Sentiment Analysis," arXiv preprint arXiv:2211.07392, Nov. 2022.

[6] R. Chaudhary, "Advanced Stock Market Prediction Using Long Short-Term Memory Networks: A Comprehensive Deep Learning Framework," arXiv preprint arXiv:2505.05325, May 2025.

[7] A. Farhadi, A. Zamanifar, A. Alipour, A. Taheri, and M. Asadolahi, "A Hybrid LSTM-GRU Model for Stock Price Prediction," IEEE Access, vol. 13, pp. 117594–117618, 2025.

[8] W. Jiang, "Applications of Deep Learning in Stock Market Prediction: Recent Progress," Expert Systems with Applications, vol. 184, Art. no. 115537, Dec. 2021.

[9] Z. Xingyu, "A Review of the Application of LSTM Model in Stock Price Prediction," Development Economics of China, vol. 9, no. 8, pp. 140–145, 2025.

[10] Huang, W. Forecasting Stock Market Movement Direction with Support Vector Machine. Computers & Operations Research.

[11] Moody, J. Learning to Trade via Direct Reinforcement. IEEE Transactions on Neural Networks.

[12] Analytics Vidhya. A Complete Guide to Regression. Available at: https://www.analyticsvidhya.com/blog/2015/08/comprehensive-guide-regression/

[13] Microsoft Azure. Machine Learning Algorithm Choice. Available at: https://azure.microsoft.com/enin/documentation/articles/machine-learning-algorithm-choice/

[14] Perwej, Yusuf, and Perwej, Asif. Prediction of the Bombay Stock Exchange (BSE) Market Returns Using Artificial Neural Network and Genetic Algorithm. Scientific Research.

[15] Kannan, K. Senthamarai, Sekar, P. Sailapathi, Sathik, M. Mohamed, and Arumugam, P. Financial Stock Market Forecast Using Data Mining Techniques. Proceedings of the International MultiConference of Engineers and Computer Scientists (IMECS 2010), Hong Kong, March 17–19, 2010. ISSN: 2078-0966.

[16] Iqbal, Zahid, Ilyas, R., Shahzad, W., Mahmood, Z., and Anjum, I. Efficient Machine Learning Techniques for Stock Market Prediction. International Journal of Engineering Research and Applications (IJERA), Vol. 3, No. 6, November–December 2013, pp. 855–867.

[17] Mittermayer, Marc-André. Forecasting Intraday Stock Price Trends with Text Mining Techniques. Proceedings of the Hawaii International Conference on System Sciences (HICSS), 2004.

[18] A. Graves, "Long Short-Term Memory," in Supervised Sequence Labelling with Recurrent Neural Networks, Springer, 2012, pp. 37-45.

[19] M. Nelson, A. Pereira and R. de Oliveira, "Stock market's price movement prediction with LSTM neural networks," in 2017. International Joint Conference on Neural Networks (IJCNN), 2017, pp. 1419-1426.

[20] S. Selvin, R. Vinayakumar, E. A. Gopalakrishnan, V. K. Menon and K. P. Soman, "Stock price prediction using LSTM, RNN and CNN-sliding window model," in 2017 ICACCI, pp. 1643-1647.

[21] J. Patel, S. Shah, P. Thakkar and K. Kotecha, "Predicting stock and stock price index movement using machine learning techniques, " Expert Systems with Applications, vol. 42, no. 1, pp. 259-268, 2015.

[22] K. Kim, "Financial time series forecasting using support vector machines," Neurocomputing, vol. 55, pp. 307-319, 2003.

[23] X. Ding, Y. Zhang, T. Liu and J. Duan, "Deep learning for event-driven stock prediction," in IJCAI, 2015, pp. 2327-2333.

[24] Q. Lin, Y. Chen, S. Chen and S. Liu, "Deep learning for stock market prediction: A survey," IEEE Access, vol. 9, pp. 123-140, 2021.

[25] Z. Hu, Y. Zhao and M. Khushi, "A survey of forex and stock price prediction using deep learning," Applied System Innovation, vol. 4, no. 1, pp. 1-19, 2021.

[26] B. Rout, S. K. Dash and S. Dash, "A hybrid model for stock market prediction using machine learning techniques," Procedia Computer Science, vol. 132, pp. 1351-1362, 2018.

[27] H. Mo, "Application of LSTM neural network in stock prediction," Journal of Physics: Conference Series, vol. 1237, 2019.

[28] P. K. Babu and N. R. Mohan, "Stock price prediction using deep learning models," International Journal of Engineering and Advanced Technology, vol. 8, no. 6, pp. 234-239, 2019.

[29] J. Brownlee, "Introduction to Time Series Forecasting with Python," Machine Learning Mastery, 2017.

[30] F. Chollet, Deep Learning with Python, Manning Publications, 2018.
[31] I. Goodfellow, Y. Bengio and A. Courville, Deep Learning, MIT Press, 2016.
[32] S. Siami-Namini, N. Tavakoli and A. S. Namin, "A comparison of ARIMA and LSTM in forecasting time series," in 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA), pp. 1394-1401.
[33] R. Akita, A. Yoshihara, T. Matsubara and K. Uehara, "Deep learning for stock prediction using numerical and textual information," in 2016 IEEE/ACIS ICIS, pp. 1-6.
[34] H. Bao, X. Yue and Y. Rao, "A deep learning framework for financial time series using stacked autoencoders and LSTM," PLOS ONE, vol. 12, no. 7, 2017.