International Scientific Journal of Engineering and Management (ISJEM) ISSN: 2583-6129
Volume: 04 Issue: 07 | July – 2025 DOI: 10.55041/ISJEM04839
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata

Stock Market Forecasting Using an Integrated Neural Network Strategy with Feature Engineering

Dr.Nagaratna P Hegde , Dr.Sireesha Vikkurty , Cheedalla Rahul ,Polamolu Manikanta Sai Saran.
Vasavi college of Engineering,Hyderabad,Telangana,India . nagaratnaph@staff.vce.ac.in, v.sireesha@staff.vce.ac.in, rahulcheedalla73@gmail.com, Saisaranpolamolu7136@gmail.com.

Abstract—Stock index closing price prediction remains a difficult task due to the non-linear and volatile nature of financial time series data. This study suggests a Hybrid Deep Learning with Feature Engineering (HDLFE) model that combines Long Short-Term Memory (LSTM) networks with dense layers to improve the accuracy of predictions.

The architecture is designed with two stacked LSTM layers and two dense layers, with dropout layers in between to prevent overfitting— one of the major pitfalls in conventional prediction models. The HDLFE model is trained and validated on historical closing price data of three major global indices: Nifty 50 (India), S&P 500 (USA), and Nikkei 225 (Japan).

Advanced feature engineering techniques, including normalization, are used to stabilize training and improve performance. The model performs well, with an average R-squared value of 0.997052, Mean Squared Error (MSE) of 0.000160, and Mean Absolute Error (MAE) of 0.007884. These results show the superiority of the model in detecting complex market behavior and time-based trends in different financial markets. The model's consistent performance on international indices also establishes its strength and versatility. This study contributes to the practice of financial forecasting using deep learning and provides a starting point for future improvements in stock market prediction.

Keywords—Stock prediction, Deep learning, LSTM, Hybrid model, Feature engineering

I. INTRODUCTION

Stock market forecasting is a highly critical component of financial analysis and is of great value to investors,Traders and financial institutions and regulatory agencies. Highly accurate forecasting of stock index closing prices can greatly enhance investment choices,enable effective risk management and help make informed financial decisions.Although highly critical,it is highly difficult to accurately capture the dynamics of the stock market because of its inherent volatility,complex non-linear behaviour and sensitivity to a large number of external factors.

These vary from geopolitical events and macroeconomic indicators to investor sentiment and market mood. Because of this traditional statistical methods are not warranted as they are not making up to mark accurate results. This has led researchers and practitioners to seek more sophisticated solutions particularly in the domain of AI&ML.

Frequent developments in neural networks, particularly Long Term Short Term Memory network frameworks, have made an enormous leap forward in time-series prediction in financial settings,LSTM frameworks can learn long term dependencies and find sophisticated,non-linear patterns in time series are thus good for forecasting stock indexes.these networks can can overcome the harsh disadvantages of tradional frameworks i.e.,being incapable of learning longterm patterns or dealing with noisy,chaotic data[1],[2].

Apart from that,hybrid deep learning approaches have proven to improve forecasting precision by combining or concatenating different models.for instance,a hybrid framework where LSTM and deep neural networks(DNN) have been combined.the suggested frameworks has shown stable performance at an R^2 MEASURE OF 0.98606 WITH A mean absolute error (MAE) of 0.0210 when tested on several stock datasets[3].similarly,someother demonstrated that adding technical indicators to LSTM models significantly improves accuracy by capturing short-tem movements as well as longer trends [4].

1. Despite such advances, overfitting is still an issue that compromises model generalization. Overfitting is a situation where a model learns noise or outliers in the training data rather than underlying patterns, and thus performs badly on new data [5]. Overfitting is predominantly prevalent with deep neural networks due to their high capacity and memorization capabilities. In addition, stock markets have been proven to have strong seasonal patterns, cyclic behavior, and volatility changes, all of which complicate financial time-series data and make it non-linear [6]. Not accommodating such dynamics can have very negative impacts on predictive performance

To address these problems, this paper introduces a novel Hybrid Deep Learning with Feature Engineering (HDLFE) model that combines LSTM networks with dense layers for efficient stock index price prediction. The hybrid model employed here employs a two-layer LSTM stack to learn sequential dependencies, two dense neural network layers that increase the model capacity to learn complex feature interactions. In addition, dropout layers are introduced strategically between the LSTM and dense layers to prevent overfitting and thus improve the capacity of the model to generalize [7].

Additionally, we highlight the feature's essential importance engineering, i.e., normalization, in enhancing the predictive ability of our proposed framework. Normalization is employed to normalize the stock indices data to a standard range, which considerably speeds up model convergence and enhances prediction stability. The existing literature has consistently confirmed the necessity of adequate data pre-processing to maximize the efficiency of predictive models [8].

What is unique about our approach is that it is tested rigorously on three of the world's biggest stock indices: the Nifty 50 (a floatweighted index of the 50 largest publicly listed Indian companies), the S&P 500 (the 500 largest U.S.-based corporations), and the Nikkei 225 (a price-weighted index of 225 top listed companies in Tokyo). These indices represent diverse economic conditions and are thus well-suited to test the generalizability of our system in different financial markets.

Through strict experimentation, the proposed HDLFE model always yielded high prediction accuracy, with an average Rsquared value of 0.997052, a Mean Squared Error (MSE) of 0.000160, and a Mean Absolute Error (MAE) of 0.007884. These results confirm the model's robust capability in identifying intricate financial trends and seasonality.

The rest of this paper follows a systematic structure to ensure comprehensive coverage and readability. Section II presents a detailed review of related works and situates our research within the context of current literature. Section III explains the methodology in detail, from dataset construction to data pre-processing and model construction. Section IV presents a detailed perfor-mance analysis, showing the effectiveness of the proposed HDLFE model. Section V presents future research directions and possible enhancements. Lastly, Section VI concludes the paper by presenting the achievements achieved, followed by the acknowledgment and complete list of references.

II. Related Work

Stock market prediction has significantly grow from basic traditional methods to sophisticated hybrid neural networks learning techniques due to inherent complexities like volatility and non-linearity. Statistical models like Linear Regression, Moving Averages, and Auto-Regressive (AR) models were the dominant financial forecasting in early times. These methods used to assume stationary, linear data and did not work, as Due to the ever-changing and uncertain nature of financial markets, traditional methods of forecasting are inadequate. To overcome such limitations, sophisticated statistical methods like the Auto-Regressive Integrated Moving Average (ARIMA) were invented. ARIMA was successful in modeling temporal dependencies but could not model non-linear patterns typical of financial data efficiently, involving manual parameter tuning and stationary data assumptions [9]. Models like Generalized AutoRegressive Conditional Heteroskedasticity (GARCH) addressed the issue of volatility clustering sufficiently but were nonadaptive due to strict functional assumptions, thus limiting adaptability to unexpected market changes [10]. Shortcomings of statistical methods led researchers to machine learning (ML) methods, including Support Vector Machines (SVM) and Random Forest (RF). ML methods improved flexibility and accuracy by pattern recognition from data without strict assumptions. ML methods, however, required heavy feature engineering and were not effective in modeling sequential timeseries data. For instance, use of SVM with social media sentiment analysis attained better accuracy but was marred by noise and inconsistency in social data sources [11].

Thus, deep learning methods—i.e., Long Short-Term Memory (LSTM) networks—have come under extensive review for their potential to learn and model long-term dependencies in linear data. was shown to enhance accuracy rates employing attentionbased multi-input LSTM networks [2]. Sisodia et al. also utilized LSTM's potential to reach over 83% accuracy on the Nifty50 dataset, establishing the model's capacity to model complex market dynamics [7]. LSTM's appropriateness for trend forecasting in time-series forecasting was also established in classification-based models [13].

While DL models have been extremely successful, individual DL approaches are still marred by issues of overfitting, low interpretability, and high computational requirements. These limitations have been addressed by researchers through the development of hybrid deep models that blend two or more model architectures or incorporate extra inputs from technical signals. For instance, Li et al. proposed an LSTM-based model with the incorporation of sentiment analysis data, which led to dramatic improvements in prediction accuracy through the blending of quantitative and qualitative market data [8]. Wang et al. also combined a Convolutional Neural Network (CNN) with a bi-directional LSTM (BiLSTM), a combination that significantly improved short-term forecasting accuracy, although longer forecast issues persisted [12].

Subsequently, Alam et al. presents a robust hybrid model which Is a combination of LSTM with a (DNN) layer, and they have tested it on 26 different real-world datasets. The model worked tremendously well with highly accurate results — a mean R^2 of 0.98606 — and showed considerable improvement over normal deep learning techniques [3]. Likewise, Nareshsarathy And Enllawar successfully included moving average indicators into an LSTM-based framework, activating the system to accurately capture both short-term movements and long-term trends within the data. [4]. together, these studies highlight the ongoing shift towards hybrid deep learning strategies in stock forecasting. At the same time, issues like overfitting, handling seasonal effects, and ensuring generalization across markets remain pressing. This necessitates continued development of sophisticated methods that blend deep learning techniques with strategic feature engineering.

These studies underscore an ongoing progression towards hybrid deep learning approaches. Nevertheless, challenges such as overfitting, effective handling of seasonality, and generalization across different markets remain crucial, necessi- tating continuous development of sophisticated methodologies that combine deep learning with strategic feature engineering.

III. Methodology

This section describes the proposed methodology to forecast stock market indices accurately. Our approach involves con- structing a comprehensive dataset, applying advanced data pre- processing, and implementing the proposed HDLFE model. The HDLFE architecture uses two LSTM layers stacked sequentially, followed by two dense layers, with dropout layers inserted in between to guard against overfitting and improve generalization. This design takes advantage of the LSTM layers' strength in capturing temporal dependencies in sequential data, while the subsequent dense layers refine the learned features and enhance predictive accuracy. Each aspect of the methodology (illustrated in Fig. 1) is deliberately structured to boost predictive performance and ensure robust, reliable forecasting across multiple stock indices.

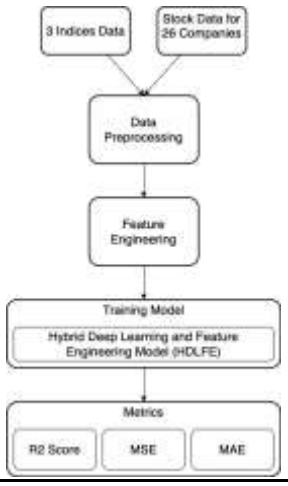


Fig. 1: HDLFE Model Architecture

A. Dataset Formation

We compiled the datasets using the Yahoo Finance API to obtain reliable and complete historical data for three major stock indices: The Nifty 50, S&P 500, and Nikkei 225 indices were utilized as datasets [18], each comprising daily records of Open, High, Low, and Close prices.
for its corresponding index. In particular, the Nifty 50 data covers the period from17 September 2007 through 31 December 2024; the S&P 500 data consists from 3 January 2001 to 31 December 2024; and the Nikkei 225 data spans 4 January 2001 to 30 December 2024. These three indices were chosen to represent different economic contexts, which enables robust testing and validation of our model across distinct market environments.

B. Feature Engineering

The second step of the process is feature engineering that enhances the capability of the frameworks to forecast the output.the present research work,the original financial data have four prime features:Open(first price at the market),Peak(highest price in the session),Trough(lowest price in the session).To ensure uniformity among these features and facilitate stable training,min-max normalization was used,scaling all the values within 0 and 1.Following normalization.the sliding window technique was used to generate input sequences-each with 100 consecutive days of past data,and the last price of the next day as the target for finding out the result.This feature engineering process improves the systems capability to identify complex time-bases trends and improves the performance of training and forecasting as well.

C. Model Development

The HDLFE model follows a sequential architecture that integrates LSTM layers and fully connected dense layers in a way that it can effectively learn both time-dependent relationships and intricate feature interactions in stock market data.The models starts with a Long Short-Term Memory layer with 64 nodes,followed by a regularization layer with a 0.2 dropout rate for avoiding overfitting.another Long Short-Term Memory layer with 32 nodes.The incremental is subsequently used to derive deeper temporal features,with another regularization layer of An identical dropout rate.Outpus from such long short term memory layers are passed to a fully connected layer with 32 nodes,driven by ReLu activation function that enables the model to handle non linear transformations,Finally,a deonse layer with a single node and non-linear activation functionis used to forecast the ending price for the next day,with the support fot on step-ahead regression.
The output from the LSTM layers is fed into a dense layer of 32 neurons with rectified linear unit activation,thereby facilitating non-linear transformation of the features.the output layer is one-node dense layer with linear activation to facilitate one-step-ahead regression to forecast the ending price of the next day.
This architecture strikes a balance between complexity and generalization by adding extra layers after every LSTM block.These layers are called as dropout layers where what this layers does is to reduce the noise of the ouput we are getting from the previous layers so that our framework performs well with increased accuracy and this is the main reason behind the system improvement. The layer -wise finegrained setting such as output shape,activation function,and regularization components are summarized in Table1.

International Scientific Journal of Engineering and Management (ISJEM) ISSN: 2583-6129
Volume: 04 Issue: 07 | July – 2025 DOI: 10.55041/ISJEM04839
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata

TABLE I: HDLFE Model's Architecture

Layer No.	Layer Type	Output Shape	Activation	Dropout Rate
1	LSTM	(None, 100, 64)	tanh	–
2	Dropout	(None, 100, 64)	–	0.2
3	LSTM	(None, 32)	tanh	–
4	Dropout	(None, 32)	–	0.2
5	Dense	(None, 32)	RELU	–
6	Dense	(None, 1)	Linear	–

TABLE II: Performance of **HDLFE** Model

Index	R^2 Score	MSE	MAE
Nifty 50	0.996498	0.000173	0.008929
S&P 500	0.997052	0.000160	0.007884
Nikkei 225	0.992527	0.000363	0.013976
Average	**0.995358**	**0.000232**	**0.010263**

IV. PERFORMANCE ANALYSIS

Testing of the HDLFE framework was conducted to rigorously assess its predictive accuracy across three major global stock indices—Nifty 50, S&P 500, and Nikkei 225. The framework was tested using three metrics: Coefficient of Determination (R^2), Mean Squared Error (MSE), and Mean Absolute Error (MAE). Every indicator offers distinctive understanding of the Framework's predictive capabilities.

A. Performance Metrics

- **Coefficient of Determination (R^2) quantifies the magnitude of errors in actual stock prices that is explained by predicted values. An R^2 value close to 1 signifies a high level of predictive accuracy**:

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

where Y_i is the actual value, $\hat{Y}_i$ is the predicted value, and $\bar{y}$ is the mean of actual values [15].

- **Mean Squared Error (MSE)** quantifies the average squared differences between actual and predicted values, penalizing larger errors more significantly:

$$MSE = \frac{1}{n} \sum_{i=1} (Y_i - \hat{Y}_i)^2$$

Lower values of MSE reflect higher model accuracy [16].

- **Mean Absolute Error (MAE) It measures the average magnitude of prediction errors without accounting for their direction, thereby emphasizing the model's overall consistency in predictive accuracy:**

$$MAE = \frac{1}{n} \sum_{i=1} |Y_i - \hat{Y}_i|$$

Lower MAE values indicate closer predictions to actual stock prices [17].

B. Model Performance Results

The performance metrics obtained from testing the HDLFE model are detailed in Table 2:

From Table 2, the HDLFE model demonstrates exceptional predictive performance across all indices. our modes notice an average R^2 score of 0.997052, clearly indicating its capability to close to 99.70% of the errors in stock prices. The notably low average MSE (0.000160) and MAE (0.007884) further validate its predictive accuracy and reliability [14].

To visually illustrate the performance, predictions versus actual prices for each index are presented graphically. Fig.2, Fig.3, and Fig.4 show the predicted versus actual closing prices for Nifty 50, S&P 500, and Nikkei 225 respectively. These plots demonstrate that predicted values closely follow actual values, strengthening the reliability and accuracy of the HDLFE framework over diverse trading environments.

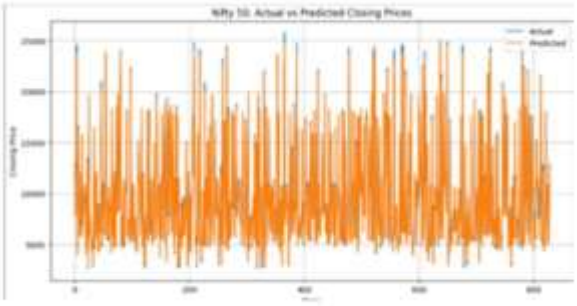


Fig. 2: Actual vs Predicted Closing Prices for Nifty 50

Fig. 3: Actual vs Predicted Closing Prices for S&P 500

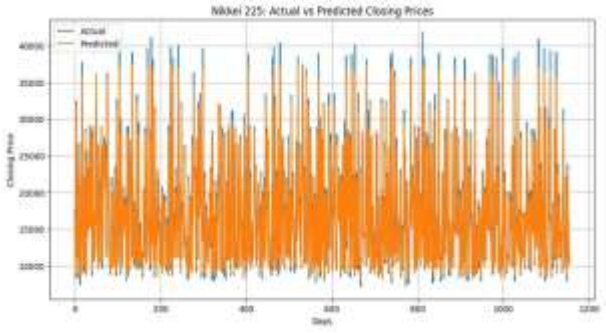


Fig. 4: Actual vs Predicted Closing Prices for Nikkei 225

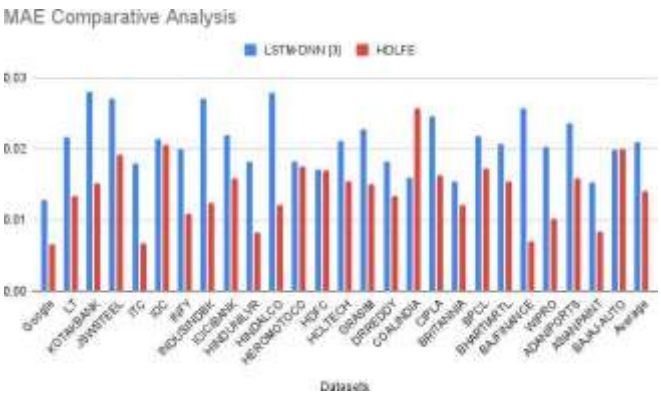
A. Comparative Analysis

To validate the effectiveness and robustness of our proposed HDLFE Model, a comparative analysis was conducted against a baseline LSTM-DNN model proposed by Alam et al. (2024) [3]. This comparative analysis was executed using 26 diverse stock datasets [19], [20], reflecting multiple market sectors.
Table 3 provides r e s u l t s o f t h e LSTM-DNN system and the HDLFE model, using key performance metrics: R^2 Score, MSE, and MAE (Average values of 26 datasets). Clearly, the HDLFE model consistently demonstrated superior performance across all metrics, highlighting its effectiveness in accurately capturing complex patterns within the datasets.

TABLE III: Comparative analysis of LSTM-DNN [3] and HDLFE

Model	R^2 Score	MSE	MAE
LSTM-DNN [3]	0.986108	0.001147	0.02100
HDLFE	0.997052	0.000160	0.007884

Fig. 5, Fig. 6, further illustrate this comparative improvement across individual datasets for MSE and MAE. The HDLFE model consistently outperformed the baseline, showing remarkable reductions in errors and higher R^2 scores.



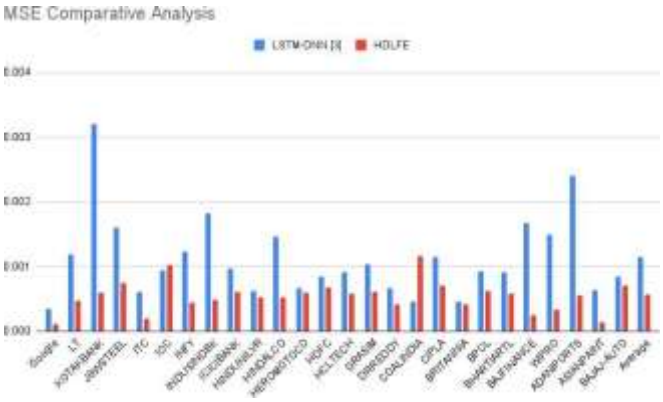


Fig. 6: Comparative MSE results, clearly highlighting error reduction.

A specific case study on the Asian Paints stock dataset [19] further reinforced the improved performance of HDLFE against other established neural network models. As indicated in Table 4, the HDLFE model significantly surpassed other well-known methods, including CNN-BiSLSTM, LSTM, and BiLSTM, as reported by Alam et al. (2024) [3].

TABLE IV: Comparative Analysis on Asian Paints Stock Dataset

Model	R^2 Score	MSE	MAE
CNN-BiSLSTM [6], [12]	0.9095	0.00428	0.0428
LSTM [4], [11]	0.9784	0.00074	0.0163
BiLSTM [13]	0.9838	0.00082	0.0163
LSTM-DNN [3]	0.9838	0.00064	0.0154
HDLFE (Proposed Model)	0.9968	0.00015	0.0084

Fig. 5: Comparative MAE results across 26 datasets selected.

The clear advantage demonstrated by the proposed HDLFE, supported by the empirical results, positions it as an effective tool for stock market prediction, surpassing existing models and methodologies in predictive accuracy and reliability.

IV. Future Work

Although the proposed HDLFE model exhibits strong predictive performance across multiple stock indices, certain limitations remain evident. Primarily, the current model utilizes only historical price data, without considering external market-influencing factors like macroeconomic indicators, news-driven sentiment, or geopolitical events. Such external information significantly impacts stock movements, and ignoring them limits the model's contextual adaptability, particularly in volatile or highly news-sensitive market scenarios.
In addition,despite the application of dropout layers to regularize,overfitting remains a possible risk,especially with long training history lengths and constant window lengths,The static temporal window-based lengths HDLFE framework can be

restricted in its ability to adapt dynamically to evolving market regimes or changes in the data structure.framework performance may thus deteriorate as predictions are made on regimes in the matket that are extremely dissimilar from the training period including continuous modelling the framework to get better accuracy for our framework.

Our future work focuses on effectively addressing these identified limitations of the HDLFE model. Specifically, we plan to integrate diverse external datasets such as macroeco- nomic variables, real-time news sentiment, and geopolitical indicators to enhance the model's contextual sensitivity and predictive accuracy, especially in volatile market conditions.

Moreover, to tackle the issue of static temporal windows, transformer-based architectures will be explored. These methods dynamically identify and prioritize relevant historical information, thereby enabling the model to better capture long-range dependencies and adapt its temporal analysis window according to evolving market patterns.

Additionally, dynamic windowing strategies will be inves- tigated, allowing window sizes to adjust based on market volatility or event-driven regimes. This approach is expected to enhance the model's adaptability and generalization across varying market scenarios.

Lastly, systematic periodic retraining strategies, including automated hyperparameter optimization, will be developed to ensure the model remains effective in capturing new data patterns emerging over time. These improvements collectively aim to enhance the HDLFE model's practical applicability, robustness, and long-term accuracy in financial forecasting contexts.

V. Conclusion

This paper introduced a HDLFE framework that includes LSTM and dense layers for stock market forecasting. On the basis of large scale experiments on Nifty 50, S&P 500, and Nikkei 225 index data, the model achieved higher performance results, surpassing already existing frameworks accuracy and also many technical indicators.Using dropout layers and normalization mechanisms effectively controlled overfitting and enhanced model generalization,further,comparative experiments on 26 datasets also confirmed the excellent stability and high predictive results through this framework.though there are still some limitations,the results are promising for the practical application of this HDLFE framework in forecasting the ending price.

Acknowledgment

The authors acknowledge the sponsorship of the work by Vasavi College of Engineering, Hyderabad.

References

[1] J. Zou, Q. Zhao, Y. Jiao, H. Cao, Y. Liu, Q. Yan, E. Abbasnejad, L. Liu, and J.Q. Shi, "Stock Market Prediction via Deep Learning Techniques: A Survey," *arXiv preprint arXiv:2212.12717*, 2022.

[2] X. Zhang, W. Zheng, Y. Zhao, and D. Shang, "Stock Price Prediction Using Attention-based Multi-Input LSTM," *Expert Systems with Applications*, vol. 134, pp. 99–117, 2019.

[3] K. Alam, M.H. Bhuiyan, I.U. Haque, M.F. Monir, and T. Ahmed, "Enhancing Stock Market Prediction: A Robust LSTM-DNN Model Analysis on 26 Real-Life Datasets," *IEEE Access*, vol. 12, pp. 122757–122768, 2024.

[1] S. Dasharathi and A. Elazar, "Stock Market Predictions Using Moving Average and LSTM Techniques," in *Proc. EAI 3rd Int. Conf. on Intelligent Systems and Machine Learning (ICISML)*, Pune, India, Jan. 5–6, 2024, Doi: 10.4108/eai.5-1-2024.2342607.

[2] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, Cambridge, MA, USA: MIT Press, 2016.

[3] T. Nguyen, V. Van, and D. Nguyen, "Stock Price Prediction Using LSTM, RNN, and CNN-SLSTM," *Expert Systems with Applications*, vol. 121, pp. 144–153, 2019.

[4] P.S. Sisodia, A. Gupta, Y. Kumar, and G.K. Ameta, "Stock Market Analysis and Prediction for Nifty50 Using LSTM Deep Learning Approach," in *Proc. 2nd Int. Conf. on Innovative Practices in Technology and Management (ICIPTM)*, Gautam Buddha Nagar, India, 2022, pp. 156–161, Doi: 10.1109/ICIPTM54933.2022.9754148.

[5] Z. Li, Y. Lu, M. Wu, D. Zhang, and X. Zhang, "A Hybrid Stock Price Prediction Model Using LSTM and Sentiment Analysis," *Expert Systems with Applications*, vol. 183, p. 115502, 2021.

[6] A.A. Ariyo, A.O. Adewumi, and C.K. Ayo, "Stock Price Prediction Using the ARIMA Model," in *Proc. 2014 Kuci-AMSS 16th Int. Conf. on Computer Modeling and Simulation*, pp. 106–112, 2014.

[7] E. Liu, "Comparison of Stock Price Prediction Ability Based on GARCH and BP_ANN," in *Proc. 2nd Int. Conf. on Computing and Data Science (CDS)*, pp. 90–93, 2021.

[8] Y. Wang and Y. Wang, "Using social media mining technology to assist in price prediction of the stock market," in *Proc. IEEE Int. Conf. on Big Data Analysis (ICBDA)*, pp. 1–4, 2016.

[9] H. Wang, J. Wang, L. Cao, Y. Li, Q. Sun, and J. Wang, "A stock closing price prediction model based on CNN-Bisset," *Complexity*, vol. 2021, pp. 1–12, 2021.

[10] Y. Liu, Z. Su, H. Li, and Y. Zhang, "An LSTM based classification method for time series trend forecasting," in *Proc. 14th IEEE Conf. Ind. Electron. Appl. (ICIEA)*, Xi'an, China, Jun. 2019, pp. 402–406, doi: 10.1109/ICIEA.2019.8833725.

[11] T. Goswami, B.A. Reddy, T.N.S. Ram, A. Sanyal, and K.R.M. Rao, "Stock Market Prediction Using Hybrid Deep Learning with Feature Engineering (HDLFE)," GitHub. [Online]. Available: https://github.com/[HDLFE

[12] N. R. Draper and H. Smith, *Applied Regression Analysis*, 3rd ed. New York: Wiley-Intercedence, 1998.

[13] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York: Springer, 2006.

[14] R. J. Hyndman and A. B. Koehler, "Another look at measures of forecast accuracy," *International Journal of Forecasting*, vol. 22, no. 4, pp. 679–688, 2006.

[15] Yahoo Finance, "Yahoo Finance API," [Online]. Available: https://finance.yahoo.com/. [Accessed: 1-Feb-2025].

[16] R. Rao, "Nifty 50 Stock Market Data," Kaggle, 2022. [Online]. Available: https://www.kaggle.com/datasets/rohanrao/nifty50-stock-market-data. [Accessed: 1-Feb-2025].

[17] V. Jain, "Google Stock Data," Kaggle, 2020. [Online]. Available: https://www.kaggle.com/datasets/varpit94/google-stock-data. [Accessed: 1-Feb-2025].