Stock Market Prediction using Hybrid Deep Learning with Feature Engineering Model

Tilottama Goswami
Department of IT
Vasavi College of Engineering
Hyderabad, India
tgoswami@staff.vce.ac.in

Anish Reddy Bemalked
Department of IT
Vasavi College of Engineering
Hyderabad, India
anishbemalked@gmail.com

Sree Ram Thanjavur
Department of IT
Vasavi College of Engineering
Hyderabad, India
sreeramthanjavur@gmail.com

Abhijit Sanyal
Department of DDIT
Novartis Healthcare Pvt. Ltd.
Hyderabad, India
abhijit.sap@gmail.com

Kovvur Ram Mohan Rao
Department of IT
Vasavi College of Engineering
Hyderabad, India
krmrao@staff.vce.ac.in

***Abstract*—Accurately predicting the closing prices of stock market indices remains a significant challenge due to the inherent volatility and non-linear characteristics of financial time series data. This paper proposes a novel Hybrid Deep Learning with Feature Engineering Model (HDLFE) that integrates Long Short-Term Memory (LSTM) layers and dense layers to enhance predictive accuracy. The proposed architecture consists of two stacked LSTM layers followed by two dense layers, with dropout layers strategically placed between them to effectively mitigate overfitting—one of the major challenges in existing stock prediction models. The HDLFE model is trained and rigorously evaluated using historical closing price datasets from three prominent global indices: Nifty 50 (India), S&P 500 (USA), and Nikkei 225 (Japan). To further improve model performance, comprehensive feature engineering techniques are applied, including normalization, which stabilizes training and enhances prediction quality. Empirical evaluations demonstrate outstanding results, achieving an average R-squared value of 0.995406, a Mean Squared Error (MSE) of 0.000233, and a Mean Absolute Error (MAE) of 0.01085 across the evaluated datasets. These results underscore the HDLFE model's ability to accurately capture complex market dynamics and temporal dependencies across diverse financial environments. The robustness of the model across different international indices confirms its effectiveness in handling multi-market temporal data. This paper offers valuable insights into advanced predictive modeling for financial forecasting and sets a foundation for future enhancements in deep learning-based stock market prediction.**

***Index Terms*—Stock market prediction, Machine Learning, Deep Learning, Hybrid Model, Feature Engineering**

I. Introduction

Stock market prediction is a crucial domain in financial analytics with high implications for investors, traders, financial institutions, and policymakers. The ability to accurately forecast the closing prices of stock indices can significantly impact investment strategies, risk management practices, and decision-making. Nevertheless, predicting stock markets remains extremely difficult due to the market's inherent volatility, non-linear trends, and sensitivity to myriad external factors ranging from political events and economic indicators to investor sentiment. As a result, traditional statistical approaches often fail to produce consistently reliable forecasts, leading researchers and practitioners to turn to advanced methods such as machine learning and deep learning.

In recent years, machine learning models, particularly deep learning approaches, have shown great promise in financial market forecasting. Among these approaches, LSTM networks have attracted considerable attention for their effectiveness in capturing temporal dependencies and modeling non-linear patterns in sequential data. An LSTM-based model is especially beneficial for stock market prediction because it addresses key shortcomings of traditional models, such as failing to remember long-term patterns and struggling with noisy, volatile datasets effectively [1], [2].

Moreover, recent studies have explored hybrid deep learning approaches that combine multiple architectures to improve prediction accuracy. For example, Alam et al. developed a hybrid model integrating LSTM and Deep Neural Networks (DNN) that achieved impressive results on several real-world stock datasets. Their model attained a high predictive performance with an average R^2 of 0.98606 and a mean absolute error (MAE) of 0.0210 across those datasets [3]. Likewise, Nareshsarathy and Enllawar found that augmenting an LSTM with technical indicators (e.g., moving averages) markedly improved prediction accuracy by capturing both short-term volatility and longer-term trends [4].

Despite these advances, stock index prediction still encounters several significant challenges, chief among them being model overfitting. Overfitting occurs when a model learns spurious noise or random fluctuations in the training data instead of the true underlying patterns, leading to poor generalization on unseen data [5]. This problem is particularly common in deep learning models because of their complex architectures and high capacity to memorize training examples. In addition, financial markets display pronounced seasonal trends, cyclic patterns, and varying levels of volatility, all of which contribute to highly non-linear and complex data characteristics [6]. If such seasonal patterns are not properly accounted for, they can greatly undermine a model's predictive performance.

To tackle these challenges, this paper introduces a novel Hybrid Deep Learning with Feature Engineering (HDLFE) model that combines LSTM networks with dense layers for effective stock index price forecasting. The proposed hybrid architecture employs a two-layer LSTM stack to capture sequential dependencies, followed by two dense neural network layers that broaden the model's ability to learn complex feature interactions. Additionally, dropout layers are strategically inserted between the LSTM and dense components to reduce overfitting risk and thereby improve the model's generalization capabilities [7].

Furthermore, we emphasize the critical role of feature engineering, specifically normalization, in improving the predictive performance of our proposed model. Normalization is utilized to standardize the stock indices data within a consistent range, significantly accelerating model convergence and improving prediction stability. Previous studies have consistently demonstrated the importance of proper data pre-processing to enhance the efficacy of predictive models [8].

Our approach significantly differentiates itself by undergoing comprehensive validation across three prominent global stock indices: the Nifty 50 (a float-weighted index comprising the 50 largest publicly traded companies in India), the S&P 500 (which tracks the performance of 500 leading companies listed in the United States), and the Nikkei 225 (a price-weighted index representing 225 top-tier companies listed on the Tokyo Stock Exchange in Japan). These indices represent diverse economic environments and offer a robust testing ground to evaluate our model's ability to generalize across different markets. In extensive empirical testing, our HDLFE model demonstrated superior predictive accuracy with an average R-squared value of 0.995406, a Mean Squared Error (MSE) of 0.000233, and a Mean Absolute Error (MAE) of 0.01085, clearly indicating its effectiveness and reliability in capturing complex financial trends and seasonalities.

The rest of this paper is systematically structured to ensure comprehensive coverage and clarity. Section II provides a detailed analysis of related works and positions our study within existing literature. Section III describes the methodology extensively, covering dataset formation, data pre-processing, and model development. Section IV offers an extensive performance analysis, demonstrating the efficacy of the proposed HDLFE model. Section V outlines future research directions and potential improvements. Finally, Section VI concludes the paper by summarizing key findings and contributions, followed by the acknowledgment and a comprehensive compilation of references.

II. Related Work

Stock market prediction has significantly evolved from basic statistical methods to sophisticated hybrid deep learning techniques due to inherent complexities like volatility and non-linearity. Initially, statistical models such as Linear Regression, Moving Averages, and Auto-Regressive (AR) models dominated financial forecasting. However, these approaches often assumed stationary, linear data, severely limiting their effectiveness given the dynamic nature of financial markets. To overcome these limitations, advanced statistical methods like the Auto-Regressive Integrated Moving Average (ARIMA) emerged. ARIMA effectively captured temporal dependencies but struggled to model non-linear patterns inherent in financial data, requiring manual parameter tuning and stationary data assumptions [9]. Models like Generalized Auto-Regressive Conditional Heteroskedasticity (GARCH) addressed volatility clustering effectively but were inflexible due to rigid functional assumptions, limiting adaptability to sudden market shifts [10]. The limitations of statistical approaches led researchers toward machine learning (ML) methods, including Support Vector Machines (SVM) and Random Forest (RF). ML methods improved flexibility and accuracy by identifying patterns directly from data without stringent assumptions. However, these approaches required intensive feature engineering and struggled to adequately model sequential time-series data. For instance, using SVM with social media sentiment analysis offered enhanced accuracy but suffered from noise and inconsistencies within social data sources [11].

Subsequently, deep learning (DL) methods, particularly LSTM networks, emerged prominently due to their ability to learn long-term sequential patterns. Zhang et al. demonstrated significant accuracy improvements using attention-based multi-input LSTM networks [2]. Sisodia et al. similarly leveraged LSTM's capabilities to achieve over 83% accuracy on the Nifty50 dataset, highlighting the model's effectiveness in capturing complex market dynamics [7]. LSTM's suitability for trend prediction tasks in time-series forecasting has also been validated in classification-based approaches [13].

Although deep learning (DL) models have achieved considerable success, standalone DL approaches still suffer from issues like overfitting, limited interpretability, and high computational complexity. Researchers have responded to these shortcomings by developing hybrid deep learning models that integrate multiple model architectures or incorporate additional inputs from technical indicators. For instance, Li et al. proposed an LSTM-based model augmented with sentiment analysis data, which led to notable improvements in prediction accuracy by including both quantitative and qualitative market information [8]. Similarly, Wang et al. combined a Convolutional Neural Network (CNN) with a bi-directional LSTM (BiLSTM), a fusion that substantially improved short-term forecasting accuracy, though challenges remained for longer-term predictions [12].

More recently, Alam et al. introduced a robust hybrid model that combines LSTM with a Deep Neural Network (DNN) layer, and they validated it on 26 different real-world datasets. This model achieved very high accuracy — an average R^2 of 0.98606 — and showed substantial improvements over standard deep learning approaches [3]. Likewise, Nareshsarathy and Enllawar successfully incorporated moving average indicators into an LSTM model, which allowed their system to effectively capture short-term fluctuations as well as long-term trends in the data [4]. Collectively, these studies highlight the ongoing shift towards hybrid deep learning strategies in stock prediction. 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, necessitating continuous development of sophisticated methodologies that combine deep learning with strategic feature engineering.

III. Methodology

This section describes the proposed methodology to predict stock market indices accurately. Our approach involves constructing a comprehensive dataset, applying advanced data preprocessing, 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.

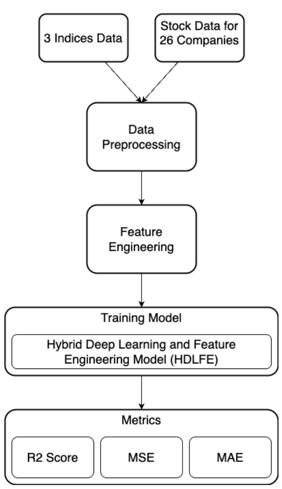

Fig. 1: HDLFE Model Architecture

A. Dataset Formation

We compiled the datasets using the Yahoo Finance API to obtain reliable and complete historical data for three major stock indices: Nifty 50, S&P 500, and Nikkei 225 [18]. Each dataset consists of the daily Open, High, Low, and Close prices for its respective index. In particular, the Nifty 50 data covers the period from 17 September 2007 through 31 December 2024; the S&P 500 data ranges 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

Effective feature engineering plays a pivotal role in enhancing model performance. The raw financial data utilized in this paper includes four key attributes: Open (the price at which the stock market opens), High (the highest price reached during each trading session), Low (the lowest price during each trading session), and Close (the final trading price at stock market close). To standardize the data and ensure stable model convergence, these values were normalized using Min-Max scaling, transforming them into a common range between 0 and 1. Following normalization, input sequences were generated using a sliding window approach, where each window consists of 100 consecutive days of historical data used as input, and the closing price of the following day serves as the target prediction [3]. This feature engineering strategy significantly enhances the model's capacity to capture complex temporal dependencies and contributes to improved training efficiency and predictive accuracy.

C. Model Development

The proposed HDLFE model is composed of a sequential architecture combining LSTM layers with fully connected dense layers to capture both temporal dependencies and complex feature relationships in stock market data. The model begins with an LSTM layer consisting of 64 units, followed by a dropout layer with a rate of 0.2 to mitigate overfitting. A second LSTM layer with 32 units is then applied to extract deeper temporal patterns, followed by another dropout layer with the same rate.

The output from the LSTM layers is passed through a dense layer with 32 neurons using Rectified Linear Unit (ReLU) activation, enabling non-linear transformation of the extracted features. The final output layer is a single-node dense layer with linear activation to perform one-step-ahead regression for predicting the next day's closing price.

This architecture maintains a balance between complexity and generalization by incorporating dropout layers after each LSTM block. These dropout layers randomly deactivate a fraction of neurons during training, which helps prevent overfitting by ensuring the model does not rely too heavily on specific neuron activations. As a result, the network learns more robust and generalizable representations of the input data. The detailed configuration of each layer, including its output shape, activation function, and regularization components, is outlined in Table 1.

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	0.995926	0.00023	0.010639
S&P 500	0.996597	0.000189	0.009407
Nikkei 225	0.993696	0.000281	0.012503
Average	**0.995406**	**0.000233**	**0.01085**

IV. Performance Analysis

A comprehensive evaluation of the HDLFE model was conducted to rigorously assess its predictive accuracy across three major global stock indices—Nifty 50, S&P 500, and Nikkei 225. The model's performance was evaluated using three critical metrics: Coefficient of Determination (R^2), Mean Squared Error (MSE), and Mean Absolute Error (MAE). Each metric provides distinctive insights into the model's predictive capabilities.

A. *Performance Metrics*

- **Coefficient of Determination (R^2)** measures the proportion of variance in actual stock prices explained by the predicted values. An R^2 close to 1 indicates high 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:

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

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

- **Mean Absolute Error (MAE)** measures the average magnitude of errors in predictions without considering their direction, emphasizing consistent predictive accuracy:

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

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. The model achieved an average R^2 score of 0.995406, clearly indicating its capability to explain approximately 99.54% of the variance in stock prices. The notably low average MSE (0.000233) and MAE (0.01085) 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, underscoring the robustness and accuracy of the HDLFE model across diverse market conditions.

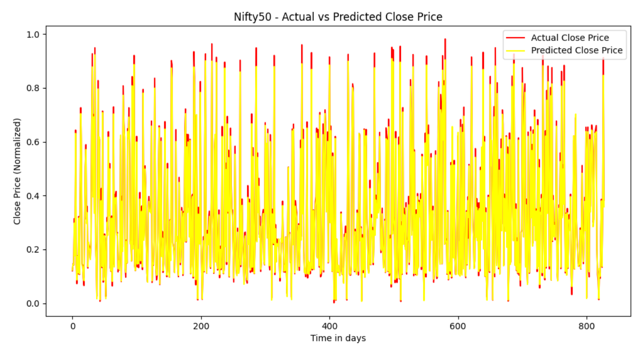

Fig. 2: Actual vs Predicted Closing Prices for Nifty 50

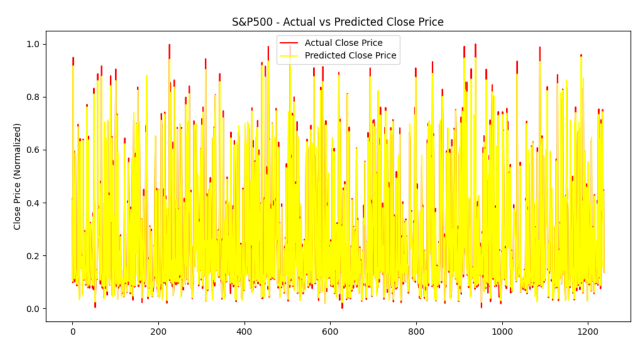

Fig. 3: Actual vs Predicted Closing Prices for S&P 500

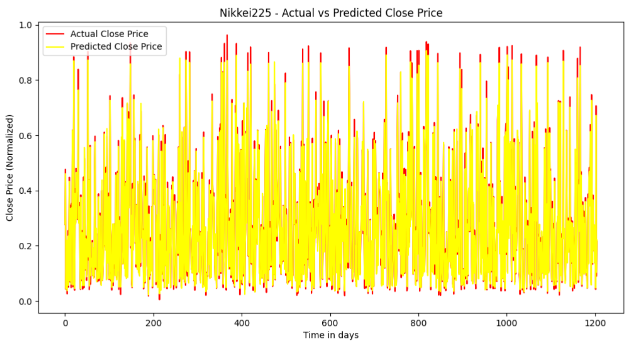

Fig. 4: Actual vs Predicted Closing Prices for Nikkei 225

C. 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 a comprehensive comparison between the LSTM-DNN model and the HDLFE model, using key performance metrics: R² 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]	0.98607	0.001147	0.02100
HDLFE	**0.987281**	**0.000560**	**0.014141**

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 R² scores.

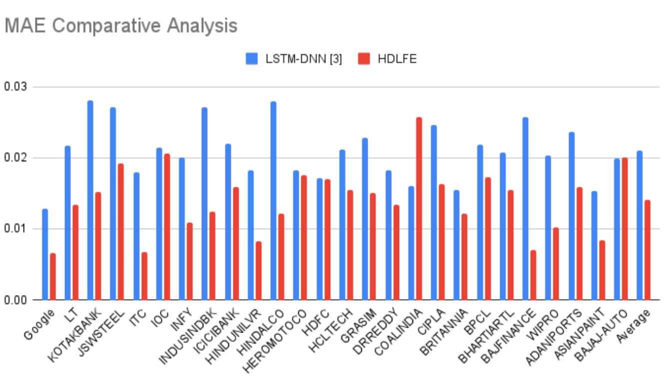

Fig. 5: Comparative MAE results across 26 datasets selected.

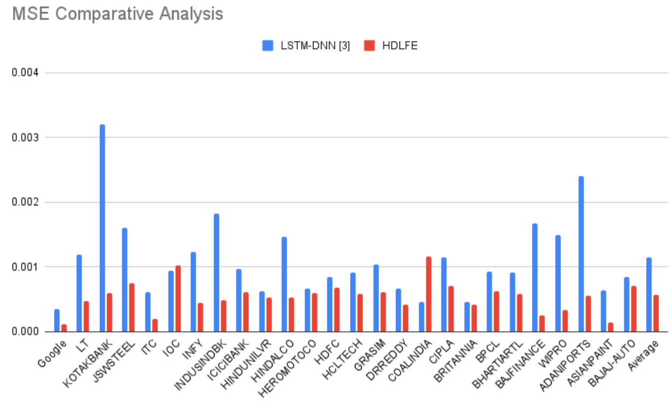

Fig. 6: Comparative MSE results, clearly highlighting error reduction.

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**

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.

V. Limitations

While the proposed HDLFE model exhibits strong predictive capabilities across multiple stock indices, few limitations need to be acknowledged to ensure transparency and guide future enhancements. First, although the model has been evaluated on three major indices Nifty 50, S&P 500, and Nikkei 225 these represent highly liquid, data-rich markets. The generalizability of the HDLFE architecture to smaller or emerging markets, which often have limited historical data and reduced trading volume, remains an open challenge.The model relies on historical closing price data and omits external market-influencing variables such as macroeconomic indicators, financial news sentiment, and geopolitical events. These external factors can significantly affect stock price movements, especially in turbulent market conditions. Their absence may restrict the contextual responsiveness of the model in real-world deployments. While dropout layers have been introduced to mitigate overfitting, the risk remains, particularly with longer training sequences or high-capacity neural components.

The financial implications—such as potential trading returns, drawdown control, or portfolio optimization—have not been quantified in the present scope. This would add the analysis and applicability in operational financial environments.

VI. Future Work

Future enhancements to the HDLFE model will focus on improving its real-world applicability through real-time prediction capabilities and broader market adaptability. One key direction is the integration of real-time data streams from financial APIs to enable instant forecasting of stock prices. This will allow the model to react promptly to changing market conditions and provide up-to-date predictions. To support real-time deployment, it is essential to ensure that the HDLFE model can generate low-latency predictions and operate efficiently in live environments. This includes enabling rapid inference to accommodate high-frequency trading scenarios and integrating seamlessly with real-time dashboard systems for continuous monitoring and decision-making. The model's adaptability to smaller or less liquid markets will also be explored. These markets often suffer from limited historical data and higher volatility. Techniques such as transfer learning and data augmentation will be investigated to improve performance in such scenarios. Another important direction is incorporating real-time financial news and sudden market movement indicators, such as breaking geopolitical events or major economic announcements. Including these external signals will help the model better capture context and improve its responsiveness during volatile periods.

VII. Conclusion

This paper proposed a HDLFE Model combining LSTM and dense layers for stock market prediction. Through rigorous experimentation on datasets from Nifty 50, S&P 500, and Nikkei 225 indices, the model achieved outstanding results, outperforming conventional deep learning models in terms of R^2, MSE, and MAE. The incorporation of dropout layers and normalization techniques significantly reduced overfitting and improved model generalization. Furthermore, comparative analyses across 26 datasets affirmed the HDLFE model's robustness and superior predictive capability. While limitations remain, the findings demonstrate strong potential for the HDLFE framework in real-world financial forecasting and lay the groundwork for future enhancements.

Acknowledgment

The authors would like to thank Vasavi College of Engineering, Hyderabad for supporting the work.

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.

[4] S. Nareshsarathy and A. Enllawar, "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.

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

[6] 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.

[7] 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.

[8] 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.

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

[10] 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.

[11] 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.

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

[13] 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.

[14] 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/anish-bemalked/HDLFE

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

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

[17] 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.

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

[19] 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].

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