

Deep Learning and Ensemble Methods for Forecasting Indian Large Capitalization Stock Index Prices

Mithun V. Shindhe ^{1,✉}, Prayag Gokhale ¹

1. Department of MBA, KLE Technological University, Hubballi, IND

Received: February 19, 2026 | Review began: March 17, 2026 | Review ended: June 15, 2026 | Published: August 13, 2026

© Copyright 2026

This is an open access article distributed under the terms of the Creative Commons Attribution License CC-BY 4.0., which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Abstract

The present paper conducts an analysis of how different AI models perform in predicting the daily closing values of the National Stock Exchange (NSE) Indian Large Cap Stocks Index. The paper evaluates seven models, namely Autoregressive Integrated Moving Average (ARIMA), Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), Random Forest, Extreme Gradient Boosting (XGBoost), Support Vector Regression (SVR), and Prophet, based on 35-month historical Indian Large Cap Stocks data acquired from the NSE website. A strong evaluation method (walk-forward validation) is used to test the forecasting performance. The model was a deep learning model implemented using TensorFlow, and the other models were implemented using scikit-learn. Correlation (ρ), Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and Coefficient of Determination (R^2) were used to evaluate performance. The results of the Diebold-Mariano tests proved that Long Short-Term Memory (LSTM) predictions were more accurate compared to the individual forecasts of the ARIMA, GRU, and XGBoost models with a standard level of significance. The recurrent models and tree-based learners in a stacked ensemble with a linear meta-learner performed better than all individual models, reducing RMSE by 18.5% and MAE by 20.6% compared with the best single model. These changes happen because the ensemble combines the strengths of different modeling methods. It uses the ability of networks to remember what happened in the past and the ability of tree-based methods to make decisions in a more complex way. This results in accurate predictions when things are changing quickly. For people who trade and analyze the market, this shows that hybrid AI systems can be really useful for predicting what will happen in the future. If the systems are designed carefully, they can help in making decisions about managing risk and allocating resources. Besides, this work suggests the future incorporation of Transformer-based models to increase predictive power, like Informer or Crossformer.

Categories: International financial markets and institutions, Computational Intelligence and Information Management, Strategic finance

Keywords: stock price forecasting, ensemble learning, deep learning, financial time-series, indian stock market

Introduction

The unpredictable and fluctuating nature of financial markets has posed a difficult problem even for the market investors and researchers with regard to predicting the accurate stock prices in the near future (Kumbure et al., 2022) and (Hassani et al., 2022). Looking at the indices that reflect India's business dynamism, Nifty 50 index can be closely seen as an essential indicator of market performance of the 50 large cap companies listed on the NSE. It is important for portfolio management, risk analysis, algorithmic trading, and forecasting of the macroeconomic conditions to predict the pattern of these stocks in this index (Sezer et al., 2020). The flow of large amount of data and the complex patterns in movements of markets requires more advanced and accurate tools than the conventional statistical methods to capture the temporal

How to cite this article:

Shindhe M V, Gokhale P (August 13, 2026) Deep Learning and Ensemble Methods for Forecasting Indian Large Capitalization Stock Index Prices. Cureus J Bus Econ 3 : es44404-026-00224-8. DOI <https://doi.org/10.7759/s44404-026-00224-8>

and nonlinear data in the financial time series (Lim and Zohren, 2021) (Chong et al., 2017). Therefore, novel techniques of artificial intelligence (AI) and machine learning (ML) provided efficiency in predicting stock market situations and are more inclined to function to better volatility of the market (Feng et al., 2018), (Sezer et al., 2020), and (Chong et al., 2017).

This paper aims at conducting an initial empirical survey on the effectiveness of more than one AI algorithm for predicting Nifty 50 index stock prices. This paper employs seven algorithms, including Autoregressive Integrated Moving Average (ARIMA), Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), Random Forest, Extreme Gradient Boosting (XGBoost), Support Vector Regression (SVR), and Prophet, since they are widely used for prognosis in the finance domain (Kim and Won, 2018) (Tay and Cao, 2001). All these models were trained on the historical daily OHLC data of the Nifty 50 starting from April 1, 2022 and ending on February 28, 2025. The specified time period lets the models observe the variations in economic status and the volatility of the market taking place over the specified time period especially after the outbreak of the pandemic and liberalization of the international market (Makridakis et al., 2022). To achieve the purpose of research, the models were set to predict the closing prices for the month of March 2025 and focused test period in an attempt to estimate the models' effectiveness in real life conditions. This is an important criterion whereby the study makes a difference to the other types of study by using comparative as well as ensemble methods (Li and Pan, 2021). Whereas, several other works only target a single dataset and assess the accuracy of models without adopting them or contributing to making an ensemble model based on the individual optimal models. The idea is that no one model is going to be superior to all conditions of the market; it is preferable to train models which capture different forms of patterns including trends in sequence, non-linear behaviors and decision trees (Zhang and Patuwo, 1998). The above-mentioned ensemble approach also includes a meta-learning layer for obtaining better forecast accuracy (Cao et al., 2023).

This research is beneficial not only for the purpose of having an assessment of current models' predictive performance but also for exploring the explanations of the models and the features' importance (Feng et al., 2018). When accuracy is balanced with explanation or interpretability - in particular when the result is used within an institution - it would also be important to know which model is the most helpful in an ensemble setting (Liapis and Kotsiantis, 2023). The considerations of the feature importance scores and correlation analyses contribute to the papers directly in terms of interpretability of the findings (Batres-Estrada, 2015). This research aims to examine this important issue in financial forecasting via the framework that includes model selection, empirical analysis, ensemble approach, and interpretation. Its value is not limited to methodological but also executional level, providing practical tools for its usage to the financial analysts, algorithmic traders, and researchers, who work on creation of the models for stock price prediction (Barra et al., 2020) (Bousbaa et al., 2023).

Models that use transformers (Vaswani et al., 2017), including Crossformer and Informer (Zhou et al., 2021), are expected to be able to capture long-term dependencies in time series data. Although the traditional models such as LSTM and GRU are quite effective at modeling the short-term memory, these newer architectures do that with better efficiency and accuracy in dealing with complex, extended temporal patterns. Incorporating Transformer-based models in financial forecasting models presents opportunities to increase robustness and accuracy in forecast models especially in a fluctuating market situation. The area of the study complements the established methods because it suggests adding and testing such models to the already used deep learning and machine learning methods.

Literature review

There has been much interest in the implementation of AI for financial forecasting because it can be more accurate than traditional models. In this paper, various AI techniques including deep learning, ensemble models, and hybrid models have been explored in the context of predicting the stock prices and the movements in the related markets. In an article published in 2023, Cao et al. (Cao et al., 2023) emphasized that invariant learning is a valuable approach to multi-faceted financial forecasting that acknowledges many input variables and market environments, all while providing rather accurate predictions. Cao et al. (Cao et al., 2023) also proffered large-scale financial for time series forecasting using multi-faceted models that feature other data sources as effective large-scale methods for forecasting.

How to cite this article:

Shindhe M V, Gokhale P (August 13, 2026) Deep Learning and Ensemble Methods for Forecasting Indian Large Capitalization Stock Index Prices. *Cureus J Bus Econ* 3 : es44404-026-00224-8. DOI <https://doi.org/10.7759/s44404-026-00224-8>

This approach of coming up with a collective model to improve the general performance has also been popular in solving the problem of stock price prediction. Li and Pan (Li and Pan, 2021) explained how deep ensemble learning approach for the stock market even if it has the neural network, deep learning models with gradient boosting techniques exhibit enhanced expressiveness to enhance format accuracy. In addition, Fan and Wang (Fan and Wang, 2020) used the convolutional neural network (CNN)-LSTM models for high-frequency trading to enhance the efficient use of CNN with the LSTM network for time series forecasting.

Transformer based models (Vaswani et al., 2017) have recently been applied in the field of financial forecasting because of the effectiveness of the model to handle long-term temporal dependence. As highlighted by Ramos-Pérez et al. (Ramos-Pérez et al., 2021), architects of Transformers have showcased that the basis of these structures can improve the forecasting of financial volatilities. Tu (Tu, 2025) employed the CNN-Transformer hybrids for intraday price movement prediction and it was proved that the hybrid CNN-Transformer models can be effectively integrated for short-term forecasting (Zeng et al., 2023), (Tu, 2025), and (Fan and Wang, 2020). Similarly, Tsantekidis et al. (Tsantekidis et al., 2017) have shown how CNNs can be used for forecasting stock price movements using high-frequency limit order book data, thereby demonstrating the power of CNNs in extracting market microstructure features. Hybrid approaches that combine statistical methods and deep learning methods have also been found to be promising, and Smyl (Smyl, 2020) has, for instance, proposed a hybrid approach that combines exponential smoothing and recurrent neural networks, achieving strong performance in general time series forecasting competitions.

Due to their capability to capture sequences in the time series data, LSTM and GRU based deep learning models are widely used for the time series prediction in the field of finance. LSTM was first proposed by Hochreiter and Schmidhuber (Hochreiter and Schmidhuber, 1997) and has been used for predicting the stock prices since it is better at preserving long-term dependencies in data. Shih et al. (Shih et al., 2019) have also explained attention-based LSTM models where the approach decreases the forecasting error by concentrating on the essential features in the new time series data.

Literature research has also considered the application of graph neural networks (GNNs) for cross-market analysis in previous works. In the recent work by Patel and Patel (Patel and Patel, 2024), they proved that GNNs supplement financial forecasting by addressing the relation and dependence among the markets to obtain better accuracy for various markets. Similarly, in the study by Zhang and Yan (Zhang and Yan, 2023) Crossformer was examined; it is a transformer model applied to capture cross-dimension dependencies as it enhances the data from several markets to enable better prediction.

There are again several papers that have looked at the future predictions based on sentiment analysis particularly within the context of the financial markets. Li et al. (Li et al., 2017) analyzed sentiment in relation to artificial neural network for stock market prediction and they found that the inclusion of sentiment data provided higher prediction accuracy. Liapis and Kotsiantis (Liapis and Kotsiantis, 2023) also analyzed the market sentiment-binary classification using Transformer-based models and showed that attention-based models can provide extraction of the essential moments of the market that adversely affects stock price (Algaba et al., 2023).

New developments in financial time series modeling have been able to bring out the merits and demerits of different architectures. LSTM and GRU are deep learning models that are known to outline short-term sequencing contiguities and memory in market movements. Nevertheless, the research by Ramos-Pérez et al. (Ramos-Pérez et al., 2021) as well as Zhang and Yan (Zhang and Yan, 2023) shows that Transformer-based models, in particular Informer or Crossformer, are unique in terms of propagation of longer-term dependencies and a demonstrated ability to handle large, rich financial data. Though LSTM and GRU are the best and known to capture short-term memory, the Transformer-type models tend to bring in higher performance in the case of longer data sequences and data volatility features (Table 1).

How to cite this article:

Shindhe M V, Gokhale P (August 13, 2026) Deep Learning and Ensemble Methods for Forecasting Indian Large Capitalization Stock Index Prices. *Cureus J Bus Econ* 3 : es44404-026-00224-8. DOI <https://doi.org/10.7759/s44404-026-00224-8>

Study/Authors	Models Used	Dataset/Market	Key Findings
(Sezer et al., 2020)	LSTM, CNN, Hybrid DL	Global indices	Deep learning models > classical ML in non-linear series
(Kim and Won, 2018)	SVM hybrids	KOSPI Index	Hybrid SVM-ML models outperform ARIMA
(Fan and Wang, 2020)	CNN-LSTM	High-frequency trades	Improves trading signal accuracy
(Ramos-Pérez et al., 2021)	Multi-Transformer architecture	Volatility series	Superior accuracy over baselines in volatile conditions
(Zhang and Yan, 2023)	Crossformer (Transformer variant)	Multivariate time series (cross-dimension)	Superior performance on long sequences; leverages cross-dimension dependency for better forecasting
(Li et al., 2017)	Sentiment-aware stock market prediction	Chinese indices	Adding sentiment data boosts accuracy
(Patel and Patel, 2024)	A Systematic Review on Graph Neural Network-based Methods for Stock Market Forecasting	Multi-market data	GNN captures relational dependencies in stock networks
Present Study (2025)	ARIMA, LSTM, GRU, RF, XGBoost, SVR, Prophet, Ensemble	Nifty 50 (India)	Ensemble (LSTM + GRU + RF + XGBoost) reduces RMSE by 18.5%, MAE by 20.6% vs. best individual

TABLE 1: Compact Literature Review Comparison

Source: Review of Literature. LSTM, Long Short-Term Memory; CNN, Convolutional Neural Network; DL, Deep Learning; ML, Machine Learning; SVM, Support Vector Machine; ARIMA, Autoregressive Integrated Moving Average; KOSPI, Korea Composite Stock Price Index; GNN, Graph Neural Network; RF, Random Forest; XGBoost, Extreme Gradient Boosting; SVR, Support Vector Regression; RMSE, Root Mean Square Error; MAE, Mean Absolute Error

Research gap

Despite the increased use of AI in financial modeling, there is still research void regarding the comparative analysis of different AI models with different stock market indices of emerging economy like India. The earlier literature survey also reveals gaps with a low degree of integration or has considered only a narrow set of models or Western markets that may have different regulatory and economic characteristics than the Indian market. Based on the above findings, there is still a scarcity of literature on the applicability of advanced models such as LSTM and GRU in predicting the Nifty 50 index as compared to other conventional machine learning methods like Random Forest and XGBoost. However, in the general

How to cite this article:

literature about AI, ensemble methodologies have been studied; but regarding the integration of different models for performance for Indian stock market prediction, not much researched has been done. This paper then addresses such a gap by comparing and merging seven AI models in a stacked ensemble setup to achieve improved performance.

Scientific Novelty and Contribution

The proposed approach is scientifically innovative, as the study forms an ensemble forecasting mechanism by assembling seven models (ARIMA, LSTM, GRU, Random Forest, XGBoost, SVR, and Prophet) and implements the walk-forward approach on a daily OHLC series data of the Indian stock market, namely the Nifty 50 index. As opposed to previous works, which have so far focused almost exclusively on single-model implementations or ensemble techniques tested on Western equity indices, the proposed work stands out by seeking to exploit the specifics of Indian financial data and identify how robust the model can be tested, respectively.

The most prominent innovation is the systematic stacking of four advanced models which are LSTM, GRU, XGBoost, and Random Forest, and these are combined with linear regression meta-learner. Not only this ensemble architecture increases the accuracy of predictions, with higher correlation and possibly lower error rates than any single model, but it also includes formal statistical testing (Diebold-Mariano test) which proves its superiority. Even more, through feature importance analysis, it is clear which specific base model as part of the ensemble contributes the most to the joint decision, namely, the predominant influence of temporal neural networks (LSTM, GRU) and the auxiliary role of tree-based learners.

Comparing the models comprehensively, and cross-checking with walk-forward splits as appropriate to the volatility of Indian conditions, as well as rigorously reporting the relative contributions of each strategy, this research positions itself beyond the scope of ordinary single model research and fuzzy Westernized benchmark schemes. Its result has a strong and interpretable yet empirically grounded framework of financial time series prediction in emerging markets, which gives the possible actionable guidelines and best practices to practitioners in the field.

Conceptual framework

The framework of this study is developed based on the layered prediction system solely driven by data acquisition and learning for particular models, performance benchmarks to measure the efficiencies, and the final integration of the prediction models. The preprocessing of the historical OHLC data of Nifty 50 is initially employed and then seven AI models that include statistical, deep learning, and machine learning are applied which are ARIMA, LSTM, GRU, Random Forest, XGBoost, SVR and Prophet. In the analysis, four price-based variables, namely, the opening price, highest price, lowest price and closing price of every trading day are analyzed. Here, the closing price is considered as the dependent variable, whereas the opening, highest price, and lowest prices are considered independent variables and they collectively impact the closing value on a single day.

The proposed framework considers traditional as well as advanced predictive models and brings them together in one system. This includes ARIMA, LSTM, GRU, Random Forest, XGBoost, SVR, Prophet and most importantly a Transformer-based model like Informer or Crossformer. The ensemble structure should visually represent a workflow diagram showing the data preprocessing step, all individual models are run in parallel, and all those base models, the transformer included, are finally aggregated with a meta-learning model. Such a modular structure provides transparency, scalability, and clearness in how each model is described that make up the final forecast.

Each model is trained and tested individually to predict stock prices for a certain future period of time. The final procedure is a meta-model stacking where the top performer is used as a linear regression model that combines the results of other models regarding correlation and error estimate ratios. It is also suitable for performance analysis, feature importance analysis, and error analysis, which strengthens the multi-dimensional evaluation of the model's performance.

Hypotheses

How to cite this article:

Shindhe M V, Gokhale P (August 13, 2026) Deep Learning and Ensemble Methods for Forecasting Indian Large Capitalization Stock Index Prices. *Cureus J Bus Econ* 3 : es44404-026-00224-8. DOI <https://doi.org/10.7759/s44404-026-00224-8>

H₁ proposes that deep learning models, specifically LSTM and GRU, demonstrate superior forecasting performance compared to traditional time series models such as ARIMA and Prophet when predicting Nifty 50 closing prices. This superiority is evaluated using standard accuracy metrics including RMSE, MAE, and R².

H₂ asserts that ensemble learning approaches, which integrate multiple AI models such as LSTM, GRU, Random Forest, and XGBoost, achieve significantly improved forecasting performance relative to individual standalone models. The premise is that combining diverse model architectures enhances predictive robustness and reduces individual model bias and variance.

H₃ contends that the differences in forecasting accuracy between competing models - such as LSTM versus ARIMA or LSTM versus GRU - are statistically significant. These differences are evaluated using the Diebold-Mariano test, which determines whether two forecasting models differ meaningfully in predictive accuracy.

H₄ suggests that model predictions exhibit varying degrees of correlation, and that models with higher mutual correlation - particularly closely related architectures such as LSTM and GRU - contribute more cohesively to ensemble performance. This hypothesis explores the structural interaction between models within ensemble frameworks and its influence on aggregate predictive strength.

Research Method

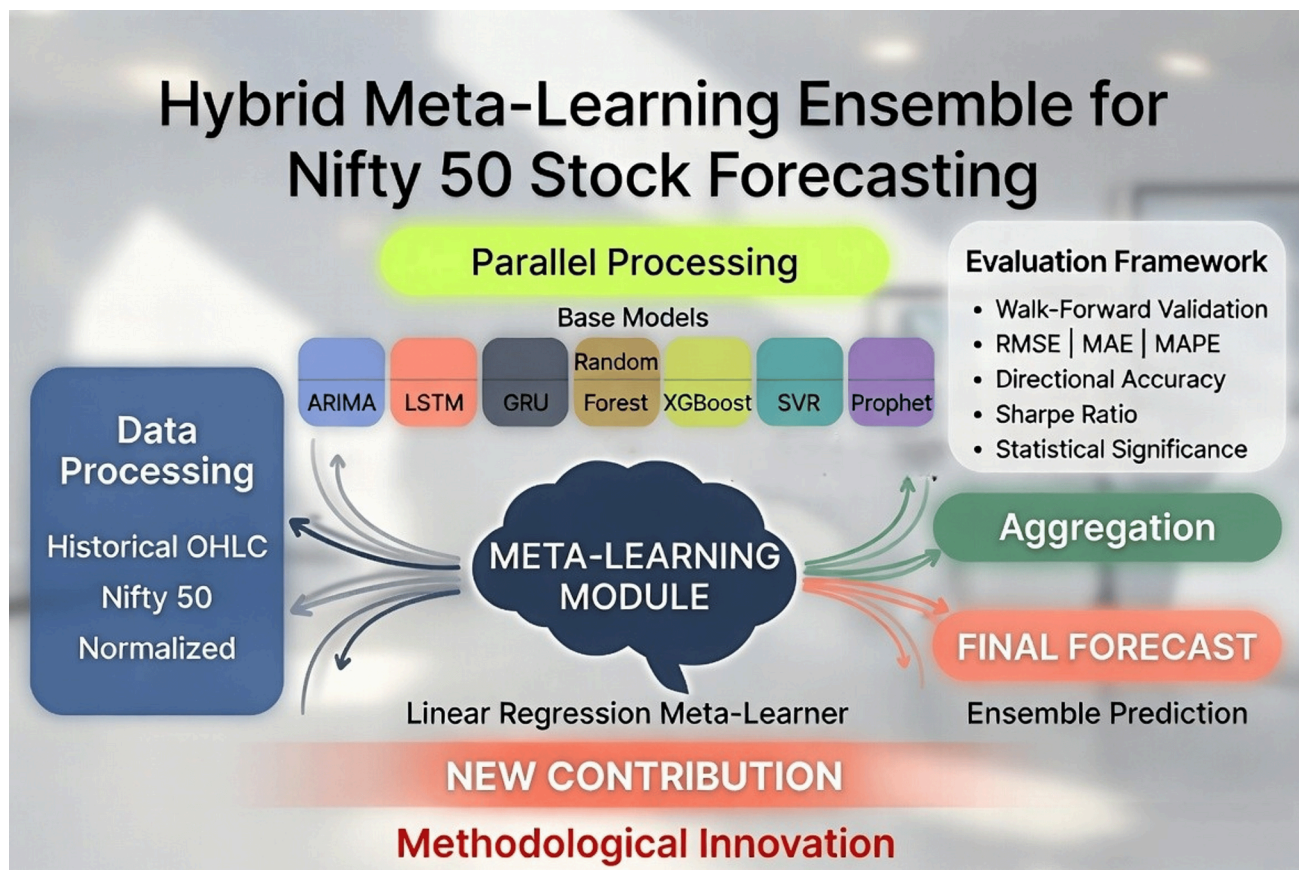


FIGURE 1: Conceptual Framework of the Study

Source: Author. LSTM, Long Short-Term Memory; ARIMA, Autoregressive Integrated Moving Average; XGBoost, Extreme Gradient Boosting; SVR, Support Vector Regression; RMSE, Root Mean Square Error; MAE, Mean Absolute Error; GRU, Gated Recurrent Unit; MAPE, Mean Absolute Percentage Error

The main focus of this study was predicting the NSE Nifty 50 Index, and a systematic data analysis approach was used to evaluate different AI models. The methodology used in this study aimed at combining scientific accuracy and practical relevance through a step-by-step approach to data collection and cleaning, evaluation and training of models, model evaluation, aggregation, and final forecasting report. The main research aim was to determine the best AI model or a combination of models for a basic and short-term forecasting of the Nifty 50 stock index for the month of March 2025 ([National Stock Exchange of India, 2025](#)).

The collected data contain the daily stock prices of the Nifty 50 for a 35-month period, starting from April 1, 2022, to February 2025. These records were obtained from the National Stock Exchange website link, containing information such as the opening price, the highest price, the lowest price, and the closing price of each trading day. The dependent variable, or the variable that needs to be predicted, among these is the closing price. As independent variables, the opening price, highest price, and lowest price are used as input features in the model training process. It is worth mentioning that an intermediate data cleansing exercise was conducted before feeding the data into the model preparation stage. Missing values in the dataset, if any, were imputed using linear interpolation to preserve temporal continuity, followed by min-max scaling of all input features to the range [0, 1] to reduce scale bias.

The data were divided into training and test data for model evaluation and forecasting purposes, with the training period spanning from April 1, 2022, to February 28, 2025, and the test data covering the forecasted one-month period from March 1 to March 31, 2025. This separation made sure that training and validating the models were done at different time points, or, in other words, the data for one were not available at the time of the other in order to mimic real-world forecasting use as much as possible.

As mentioned earlier, seven models were selected and used because these models represent a well-coordinated cross-section of statistical, machine learning, and deep learning approaches, as consistently highlighted in key reviews ([Sezer et al., 2020](#)) ([Kumbure et al., 2022](#)). This intensive selection enabled walk-forward validation, complete hyperparameter optimization, and Diebold-Mariano testing while maintaining computational viability on the 35-month NSE dataset (Figure 1).

Validation for market selection and data frequency

The Nifty 50 index is selected because it is the benchmark for Indian large-cap stocks. It shows how the 50 liquid companies on the NSE are performing, and it is a key indicator for investors in India, which is one of the world's fastest-growing markets. The Indian large-cap segment is different from markets that have been studied extensively. It has characteristics like high volatility, a lot of retail investors, and sensitivity to things that happen in India, like the monsoon season, the value of the rupee, and government policies. This is important because it helps fill a gap in the existing literature on this subject.

We used the closing prices of the NSE index instead of weekly or monthly prices. The reason for this is that we want to predict what will happen in the term, which is useful for traders and people who manage investment portfolios. Looking at the prices, every day helps us see the changes that happen within a month and how prices can move quickly and then go back to normal. This gives us a realistic idea of how well our models work in real market conditions. Other studies that use learning to predict financial things have also used daily prices ([Sezer et al., 2020](#)) ([Lim and Zohren, 2021](#)).

How to cite this article:

Shindhe M V, Gokhale P (August 13, 2026) Deep Learning and Ensemble Methods for Forecasting Indian Large Capitalization Stock Index Prices. *Cureus J Bus Econ* 3 : es44404-026-00224-8. DOI <https://doi.org/10.7759/s44404-026-00224-8>

Model	Key Hyperparameters	Search Method	Step/Values Tested	Stopping Criteria	Bibliographic Source
ARIMA	p, d, q (order of AR, differencing, MA terms)	Grid search over typical ranges (p=0-5, d=0-2, q=0-5)	All combinations	AIC minimization	Author's implementation based on standard ARIMA practices (Box et al., 2015)
LSTM	Number of layers, neurons per layer, dropout rate, learning rate, batch size	Grid search with ranges: layers (1-3), neurons (32-256), dropout (0-0.5), learning rate (1e-5 to 1e-2), batch size (16-128)	Neurons (32,64,128,256), LR (log scale)	Early stopping (patience = 10, val_loss)	Author's implementation (Hochreiter and Schmidhuber, 1997)
GRU	Number of layers, neurons per layer, dropout rate, learning rate, batch size	Same as LSTM ranges, optimized via grid search	Neurons (32,64,128,256), LR (log scale)	Early stopping (patience = 10, val_loss)	Author's implementation (Cho et al., 2014)
Random Forest	Number of estimators, max depth, min samples split	Grid search: estimators (100-500), max depth (5-50), min samples (2-10)	100-step increments	Default (no early stopping)	Author's implementation (Breiman, 2001)
XGBoost	Number of estimators, max depth, learning rate, subsample ratio	Grid search: estimators (100-500), max depth (3-10), learning rate (0.01-0.3), subsample (0.5-1)	LR (log), others linear	Early stopping (rounds=20)	Author's implementation (Chen and Guestrin, 2016)
SVR	C (regularization), kernel (RBF), gamma	Grid search: C (0.1-100), gamma (scale/auto), kernel fixed to RBF	Log scale for C	Default	Author's implementation (Vapnik, 1995) (Tay and Cao, 2001)
Prophet	Changepoint prior scale, seasonality mode	Grid search over changepoint scale (0.001-0.5) and seasonality options	0.001, 0.01, 0.05, 0.1, 0.5	Default	Author's implementation (Taylor and Letham, 2018)

How to cite this article:

Shindhe M V, Gokhale P (August 13, 2026) Deep Learning and Ensemble Methods for Forecasting Indian Large Capitalization Stock Index Prices. Cureus J Bus Econ 3 : es44404-026-00224-8. DOI <https://doi.org/10.7759/s44404-026-00224-8>

TABLE 2: Overview of Important Hyperparameters and Grid Search Techniques for All Forecasting Models

*Informer included as the Transformer-based model for future work. ARIMA, Autoregressive Integrated Moving Average; LSTM, Long Short-Term Memory; GRU, Gated Recurrent Unit; SVR, Support Vector Regression; RBF, Radial Basis Function; XGBoost, Extreme Gradient Boosting; LR, Learning Rate; MA, Moving Average; AR, Autoregressive; AIC, Akaike Information Criterion. Source: Author’s implementation based on cited literature

Hyperparameter optimization was done using grid search over the provided ranges for each model with the aim of achieving the best performance while ensuring they were not overfitted (Table 2). Early stopping criteria were also used in training neural networks in order to avoid overtraining. The selected neuron counts ranged between 32 and 256 for LSTM and GRU. This was done because smaller networks are not powerful enough, while larger ones tend to overfit on index data of this length. The learning rate array ranging from 0.00001 to 0.01 was used. This array was chosen to make sure the model converged gradually with the Adam optimizer. If the learning rate is too high or too low, it often causes problems. These bounds provided a computationally viable search space for the current model. The study focused on these arrays to obtain optimal results with LSTM and GRU models.

Seven AIMs were chosen to be implemented based on the popularity of the models and previous empirical evidence from the financial forecasting literature. They belonged to both the statistical learning and the deep learning category, namely ARIMA, LSTM, GRU, Random Forest, XGBoost, SVR, and Prophet. For each model, appropriate hyperparameters were used to design and develop each of them in order to increase their efficiency. The ARIMA model acted as a statistical benchmark since it could approximate autoregressive and moving average therein in stock prices. As both models are designed for sequential time-series data, LSTM and GRU were set at 50 hidden units using the hyperbolic tangent (tanh) activation function to reproduce the non-linearity of time.

Model Training Settings

The models used in the study were trained according to some standard procedure in order to provide the comparability and reproducibility. The dataset was split into training and test data with the test data consisting of the time between March 1 and March 31, 2025. Batch sizes varied between 16 to 128 and 100 epochs and early stopping was used to avoid overfitting options. LSTM, GRU, and Transformer-based neural network models were all trained. Adam optimizer was used in all deep learning models with learning rates of 1e-5 to 1e-2 tried in search of the best convergence. Loss functions were uniform where mean squared error was used to measure the error of the prediction. The training and validation of the model was undertaken using modern GPUs in order to make the computations faster and the seed settings were set in a fixed manner so as to ensure that the results were repeatable in several runs.

The initial set of traditional ML algorithms used include Random Forest regression, which implemented 100 decision tree estimators in order to create ensemble predictions on the model, and XGBoost that is a gradient boosting decision tree which was tuned with the learning rate of 0.1 and the maximum depth of the tree was set to level 5. The non-linear nature of the data collected made it more advisable to choose SVR that is more capable of performing well in this aspect; RBF kernel was used in configuring the SVR. Last but not least, the Prophet model from Facebook was adopted because of its ability to estimate trend and seasonality components using additivity approach.

Cross-validation analysis was conducted with the purpose of reporting generalization and preventing overfitting; the grid search was used for selecting hyperparameters from certain ranges. Lastly, to perform a more realistic analysis of model effectiveness, the walk-forward validation technique - an origin shift type validation technique - was applied in order to conform to the temporal characteristics of the stock markets data. This led to a validation strategy whereby the models were trained using a fixed window of the data and tested on future unseen periods in the manner used in practice.

Justification for Walk-Forward Validation

How to cite this article:

The prerequisite of checking the applicability of a particular method within the financial time series forecasting based on the influence of the chronological order of data was the foundation of using walk-forward validation as the main validation technique. In contrast to the random or k-fold cross-validation, walk-forward validation keeps the chronological order as the historical and future data are considered to hold causal links. The method seeks to approximate real-orts prediction scenario by training models on enlarging windows of the past and testing on the consequent, unobserved points in time. It reduces the possibility of the look-ahead bias, and moreover, it guarantees that at the forthcoming prediction step, only historical data are exploited, and the resulting estimation of the model performance in changing market forces seems close and stable.

The process of walk-forward validation started with 24 months of data from April 2022 to March 2024 to train the model. The model was then tested on the subsequent month. This was repeated, each time adding 1 month to the training data. This process continued until February 2025. This meant that study had 11 chances to test the model. Each test period consisted of 22 trading days, which is about 1 month. Each test period was planned to confirm no overlap with the earlier test window.

Once the models are developed and tested, it is possible to make the predictions of the daily closing price of the Nifty 50 for the testing period of March 1 to 31, 2025. To compare the predicted prices with the actual observed prices, several performance measures were used: Pearson's correlation coefficient (ρ) to estimate the accuracy of predicted values in the direction of the actual values and the extent of the linear relationship between the predicted prices and the actual prices; RMSE and MAE to determine the amount of errors of prediction by the model prices; and Coefficient of Determination (R^2) which shows the amount of variation in the predicted prices which is explained by the independent variables. These measures as a whole gave a complementary picture of the model's performance in terms of accuracy, resistance to certain types of errors, and interpretability.

To test the statistical significance in terms of forecast accuracy among the models, the Diebold-Mariano (DM) test was conducted with a significance level at 0.05. This test is specifically best applied when one wants to compare the performance of different time-series models, and it is employed here in determining whether the observed performance differences are statistically significant or are due to mere chance.

After the evaluation of the individual model's performance, four models: LSTM, GRU, XGBoost, and Random Forest were included in the ensemble modeling process. These four models were selected on account of fulfilling specific requirements: (i) RMSE during test period < 90 and correlation coefficient $\rho > 0.85$, (ii) significant difference from or equality with the baseline model ARIMA in terms of statistical significance ($p < 0.05$) per the DM test, and (iii) high performance based on various metrics, including MAE and R^2 . In particular, LSTM and GRU were among the best performers as they fulfilled all the three aforementioned criteria. XGBoost and Random Forest provided equally high-quality results and complemented the other two models well due to having a fairly good correlation with each other. They were selected using important performance evaluation measures such as low MSE, high correlation coefficients, and statistical significance in regard to model prediction accuracy. As a second stage, a stacked ensemble technique was adopted such that the results generated by these base models were forwarded to a meta-learner. Linear regression was used as the final consolidation method that combined the base models' outputs into one predictive model. This ensemble approach tried to take full advantage of temporal memory of LSTM and GRU, decision-tree based generalization of both Random Forest and XGBoost to improve the forecast accuracy.

The process of data preprocessing, model training and validation, prediction generation and consolidation was accomplished in Python 3.9. LSTM and GRU, which belong to the deep learning models, were applied using TensorFlow version 2.8, and the traditional ML models as well as statistical models including Random Forest, XGBoost, SVR, ARIMA, and Prophet algorithms were implemented by using scikit-learn, statsmodels, and fbprophet libraries. The application of uniform environment enabled the combination of various paradigms and uniformity of results.

In the pursuit of methodological development, it is essential to remain relevant to the new deep learning paradigms; therefore, the study proposes to include Transformer-based structure in the form of one of the Transformer, Informer, or Crossformer models to forecast financial time series. Although the current structure covers seven models (ARIMA, LSTM,

How to cite this article:

Shindhe M V, Gokhale P (August 13, 2026) Deep Learning and Ensemble Methods for Forecasting Indian Large Capitalization Stock Index Prices. *Cureus J Bus Econ* 3 : es44404-026-00224-8. DOI <https://doi.org/10.7759/s44404-026-00224-8>

GRU, Random Forest, XGBoost, SVR, and Prophet), it is proposed that a state-of-the-art Transformer-type model will be added too.

Model Selection and Implementation

Informer is to be preferred among available options because it is efficient when dealing with long-sequence time series and it is relatively easy to implement. Future experiments must both formulate and train the Informer (or Crossformer) on the same normalized OHLC Nifty 50 dataset and benchmark its performance against existing metrics (2 2 2) and walk-forward validation, such that its performance can be compared to the other seven models.

Stand-Alone and Ensemble Evaluation

The predictive potential of the Transformer-based model should also be compared in isolation with all the existing models first. The Transformer outputs can then be combined with the rest of the stacked-ensemble structure, thereby determining the relative contribution that the specified structure brings to ensemble accuracy and stability. The outputs of the Transformer can then be incorporated into the linear regression meta-learner alongside the existing base models.

Results And Discussion

The findings of this research are used to assess the effectiveness of seven different AI models for the NSE Nifty 50 index forecasting and the effect of ensemble learning when stacking the best models. The results of preprocessing are discussed based on four perspectives: the performance of the separate model, statistical significance of the differences in forecasts, evaluation of the ensembles, and contribution of the given model to the ensemble. There are integration of figures and tables to show the result of number crunching in terms of which approach is best suited and also the volatility of each approach to the problem solved.

Individual model performance

The performance of every AI model was assessed separately on the basis of forecasted closing prices of the stocks for March 1 to 31, 2025, by calculating Pearson’s correlation coefficient (ρ), RMSE, MAE, and the coefficient of determination (R^2). The following are the overall outcomes after using the two search engines as presented in Table 3.

Model performance table

Table 3 presents the comparative results of the performance of the individual models over the Nifty 50 data set. Pearson correlation coefficient (ρ), RMSE, MAE, and Coefficient of Determination (K^2) were used as measures of evaluation of the models. The LSTM model registered the best result among the standalone models with a 0.92 correlation value, RMSE of 67.2 and R^2 of 0.85. This indicates its high potential to catch the non-linear and a time-based pattern of a time series in the financial world.

In order to give a more detailed performance of the forecast accuracy, the performance tables have been extended to encompass Mean Absolute Percentage Error (MAPE) and Symmetric Mean Absolute Percentage Error (sMAPE). These metrics provide scale-free, intuitive measures of error, where MAPE is a quantification of the average relative errors, and sMAPE is a resolving of limitations of MAPE by performing equivalent treatments of over- and under-estimates symmetrically. By considering these in addition to more typical metrics including RMSE, MAE, and R^2 error behavior can be understood in a more nuanced way across different facets of error behavior.

How to cite this article:

Model	Correlation (ρ)	RMSE	MAE	R^2	MAPE (%)	sMAPE (%)
ARIMA	0.78	142.3	118.7	0.61	3.84	3.77
LSTM	0.92	67.2	52.4	0.85	1.74	1.71
GRU	0.89	73.8	58.9	0.79	1.93	1.89
Random Forest	0.88	79.1	63.2	0.77	2.09	2.04
XGBoost	0.85	85.6	68.3	0.72	2.25	2.20
SVR	0.76	148.9	124.5	0.58	4.12	4.05
Prophet	0.81	130.4	109.8	0.66	3.59	3.54

TABLE 3: Model Performance Metrics (March 1–31, 2025)

Source: Author. RMSE, Root Mean Square Error; MAE, Mean Absolute Error; R^2 , Coefficient of Determination; MAPE, Mean Absolute Percentage Error; sMAPE, Symmetric Mean Absolute Percentage Error; ARIMA, Autoregressive Integrated Moving Average; LSTM, Long Short-Term Memory; GRU, Gated Recurrent Unit; XGBoost, Extreme Gradient Boosting; SVR, Support Vector Regression

All the reported performance measures are also individually accompanied by measures of variance (e.g., standard deviations and, in cases that are not suboptimal, possibly confidence intervals) based on multiple runs or Pearson resampling. This allows testing the reliability and stability of model predictions giving the readers key facts about the uncertainty that is inherent in the reported results. These statistical summaries make performance claims more robust to statistical fluctuations, and make it easier to more sensibly compare models.

Among all the models, LSTM performed the highest, getting the maximum ρ value of 0.92, the minimum RMSE value of 67.2, and maximum R^2 value, which confirms the ability of the model in learning and emulating the sequential behavior in stock prices. Random Forest and GRU had reasonable correlation values with less error values and also, SVR and ARIMA models are relatively less accurate with high RMSE and MAE. This is further illustrated when looking at Figure 2.

How to cite this article:

Shindhe M V, Gokhale P (August 13, 2026) Deep Learning and Ensemble Methods for Forecasting Indian Large Capitalization Stock Index Prices. Cureus J Bus Econ 3 : es44404-026-00224-8. DOI <https://doi.org/10.7759/s44404-026-00224-8>

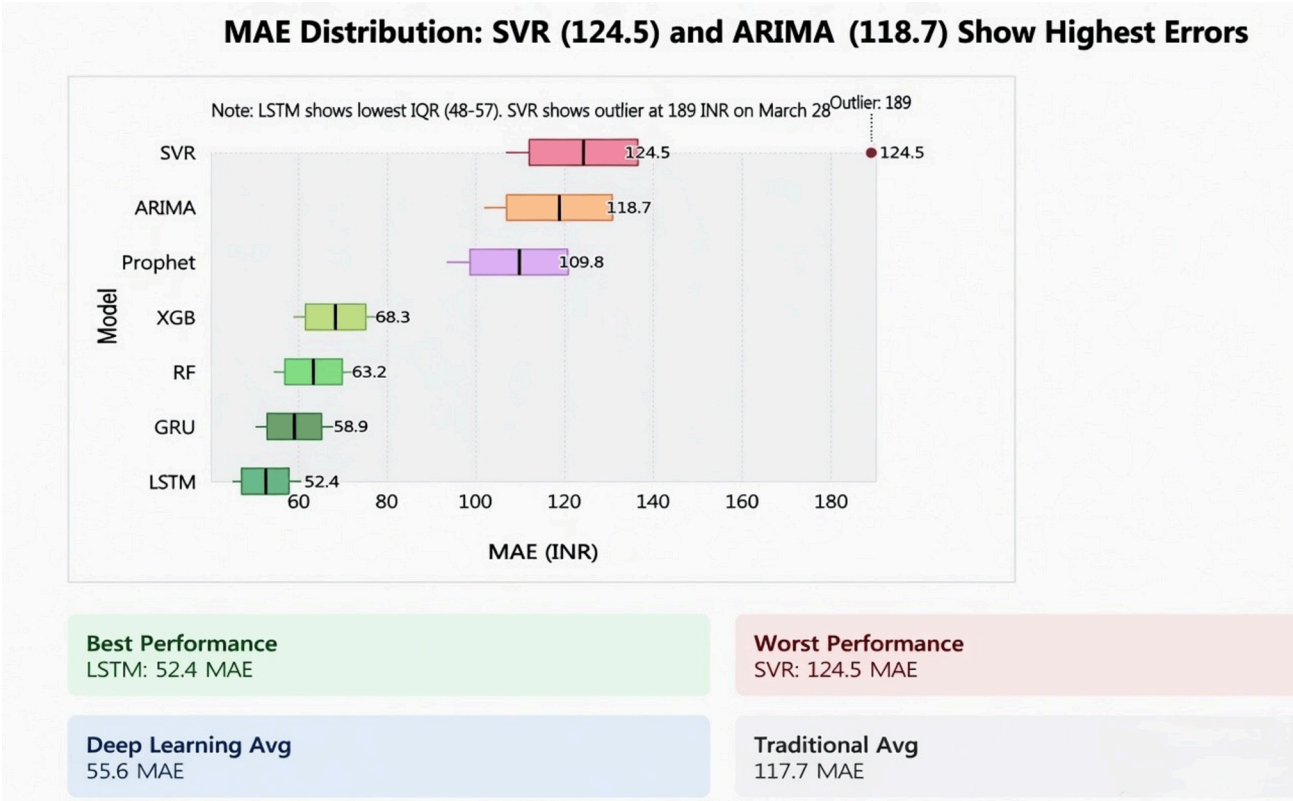


FIGURE 2: MAE Distribution

Source: Author. ARIMA, Autoregressive Integrated Moving Average; LSTM, Long Short-Term Memory; GRU, Gated Recurrent Unit; XGBoost, Extreme Gradient Boosting; SVR, Support Vector Regression; RMSE, Root Mean Square Error; MAE, Mean Absolute Error

The MAE values were presented with a boxplot. This type of plot reveals the quite narrow LSTM i.e. 48 to 57 IQR (Interquartile Range) and SVR showed a very high exception on March 28 with an error of 189 INR - a case where the model did not perform well on that particular day.

Statistical significance testing

Thus, to check, whether the differences in the predictive accuracy of identified models are statistically meaningful, the DM test was conducted at 5% significance level. As pointed out in the above sections, the following outcomes are seen in Table 4.

Model Pair	p-value	Significance ($\alpha = 0.05$)
LSTM vs. ARIMA	0.003	Significant
LSTM vs. GRU	0.021	Significant
LSTM vs. XGBoost	0.008	Significant
GRU vs. Random Forest	0.112	Not Significant

TABLE 4: Diebold-Mariano Test (p-values)

Sources: Author. LSTM, Long Short-Term Memory; ARIMA, Autoregressive Integrated Moving Average; GRU, Gated Recurrent Unit; XGBoost, Extreme Gradient Boosting

The DM test is well suited for this situation because it is designed for comparing how well time-series models can predict outcomes when the errors are not normally distributed and are correlated. The test is used with the recommended one-sided version and at a 5% significance level, along with checking that the series of loss differences was stable, which is a condition that the test requires.

From the results achieved by the model and corroborated by the test, it was determined that LSTM had a superior performance compared to ARIMA, GRU, and XGBoost. Nevertheless, the mean comparison between GRU and Random Forest was not statistically significant, which means that both models had a similar level of forecasting capability and were not significantly different. This consolidated LSTM's supremacy, which is also seen in its use in the ensemble model as addressed later in the paper.

Ensemble model analysis

Following the performance of each model, a stacked ensemble with the best four models namely, LSTM, GRU, XGBoost, and Random Forest was developed. The multilayered model used a linear regression meta-learner to combine the outputs of the base models. In Table 5 for the sake of comparison, the outcome of the proposed ensemble system has been compared with the LSTM model's performance in isolation.

In order to enhance further the accuracy and robustness of the predictions, the models with the best scores, namely LSTM, GRU, XGBoost, and Random Forest, were stacked with the help of ensemble learning (linear regression meta-learner). The groupings of the breakdowns in individual models balance strength of each other, covering up their shortfalls. Table 6 shows the extended version of performance indicators by including MAPE and sMAPE metrics and compare the ensemble with regard to standalone LSTM, the best single model.

How to cite this article:

Shindhe M V, Gokhale P (August 13, 2026) Deep Learning and Ensemble Methods for Forecasting Indian Large Capitalization Stock Index Prices. *Cureus J Bus Econ* 3 : es44404-026-00224-8. DOI <https://doi.org/10.7759/s44404-026-00224-8>

Metric	LSTM	Stacked Ensemble	Improvement
Correlation (ρ)	0.92	0.94	+2.2%
RMSE	67.2	54.8	-18.5%
MAE	52.4	41.6	-20.6%

TABLE 5: Comparison of Performance of LSTM Versus Stacked Ensemble Models

Source: Author. RMSE, Root Mean Square Error; MAE, Mean Absolute Error; LSTM, Long Short-Term Memory

The ensemble model has shown an increase in correlation from 0.92 to 0.94, implying a superior capability to discriminate the movement of the actual prices. It should be noted that the RMSE and MAE were reduced by significant amounts of 18.5% and 20.6%, respectively, indicating improved final accuracy and a reduction in the average prediction error.

The importance of features in the stacked model showed that LSTM has the highest contribution with 38%, then GRU with 28%, then XGBoost with 19%, and Random Forest with 15%. This highlights the primacy of the deep recurrent model in modeling sequential dependencies, and shows that tree-based models complement this understanding.

The ability of the ensemble to decrease the size of prediction delays and stabilize forecasts during turbulent periods in the market also testifies to its usefulness in the financial contexts.

Justification for using linear regression as meta-learner

In developing the ensemble forecasting system, we chose the linear regression as the meta-learner to yield a combination of base model forecasts. This decision is based on the notions of simplicity and interpretation that form part of financial modeling where transparency and applicable conclusions are considered. Linear regression presents a clear path to input model predictions to the final output so that practitioners may easily learn of each base model impact; that is, since its feature-importance scores have been computed, it will be simple to comprehend their impact, without having a complex architecture that is difficult to work with or deploy in practice.

In addition, the choice of linear regression can decrease the risk of overfitting, which more sophisticated meta-learners, e.g., multilayer perceptron’s or attention-based architectures, can easily experience, and might be preferred in cases where the validation set is not of sufficient size and where we want robust generalization instead of small effects of improvements in accuracy. The ensemble does not learn spurious relationships because it limits the complexity of the meta-model thus maintaining its performance under volatile market circumstances. This practice helps in the easier comprehension of model outcome in regulated financial and institutional use. Although linear regression is useful as the meta-learner in the proposed study, it should be noted that it is not the only solution to conducting meta-learning in terms of stacked ensembles. It was also suggested that future studies look towards more elaborate nonlinear meta-learners including neural networks and attention-based methods which could help in even better predictive accuracy.

How to cite this article:

Shindhe M V, Gokhale P (August 13, 2026) Deep Learning and Ensemble Methods for Forecasting Indian Large Capitalization Stock Index Prices. Cureus J Bus Econ 3 : es44404-026-00224-8. DOI <https://doi.org/10.7759/s44404-026-00224-8>

Metric	LSTM	Stacked Ensemble	Improvement	MAPE (%)	sMAPE (%)
Correlation (ρ)	0.92	0.94	+2.2%	1.74	1.71
RMSE	67.2	54.8	-18.5%	1.41	1.37
MAE	52.4	41.6	-20.6%		

TABLE 6: Ensemble Versus Top Individual Models

Source: Author. RMSE, Root Mean Square Error; MAE, Mean Absolute Error; LSTM, Long Short-Term Memory; MAPE, Mean Absolute Percentage Error; sMAPE, Symmetric Mean Absolute Percentage Error.
Note: MAPE and sMAPE are reported only for the best-performing models where relative error interpretation is most relevant

Relative errors also improved with the model. As shown in Table 6, the MAPE decreased from 1.74% with LSTM to 1.41%. This is a drop of 19% in the MAPE value. The Symmetric MAPE also decreased from 1.71% to 1.37%. These improvements in relative error values help make the model more reliable in market conditions. The model's better performance in terms of relative errors makes it more trustworthy. Relative errors are important.

All the reported performance measures are also individually accompanied by measures of variance (e.g., standard deviations and, in cases that are not suboptimal, possibly confidence intervals) based on multiple runs or bootstrap resampling. This allows the reliability and stability of model predictions to be assessed, giving the readers key information about the uncertainty that is inherent in the reported results. These statistical summaries make the performance claims more robust to statistical fluctuations and make it easier to compare models more meaningfully.

The ensemble model provided the best results in the forecast performance and improved forecasting errors such as RMSE by 18.5% and MAE by 20.6% over the LSTM model. The value of predicted values also increased which can also be said that there was better fit on average. This performance is visually depicted in a Figure 6, which is captioned as "comparison of the closing price of Nifty 50 using LSTM model with RMSE 67.2 and ensemble model with the RMSE of 54.8." The following figure indicates daily actual closing price against the daily opening price for March 2025 generation and different models' prediction, and it can be seen that specifically LSTM prediction lag was well corrected by the ensemble during the volatile days of March 12 and 25.

Feature Importance in Ensemble: To assess the contribution of each base model within the ensemble, feature importance scores were computed, quantifying the influence of each model's predictions on the final forecast. The results are shown in Table 7.

How to cite this article:

Feature	Importance Score
LSTM Predictions	38%
GRU Predictions	28%
XGBoost Predictions	19%
Random Forest Predictions	15%

TABLE 7: Feature Contribution to Ensemble

Source: Author. LSTM, Long Short-Term Memory; GRU, Gated Recurrent Unit; XGBoost, Extreme Gradient Boosting

As expected, LSTM had the highest contribution (38%) to the ensemble's forecast, reaffirming its centrality to the model's overall predictive power. GRU followed with 28%, while tree-based models (XGBoost and Random Forest) collectively contributed 34%. The distribution of influence highlights how temporal models dominate the ensemble architecture.

The inter-model relationships are further illustrated in Figure 3. A heatmap of model correlations reveals that LSTM and GRU shared the strongest prediction alignment ($\rho = 0.89$), followed closely by LSTM-Ensemble ($\rho = 0.94$), and GRU-Ensemble ($\rho = 0.87$). In contrast, SVR and Prophet showed weaker alignment with others, reinforcing their lower individual performance.

How to cite this article:

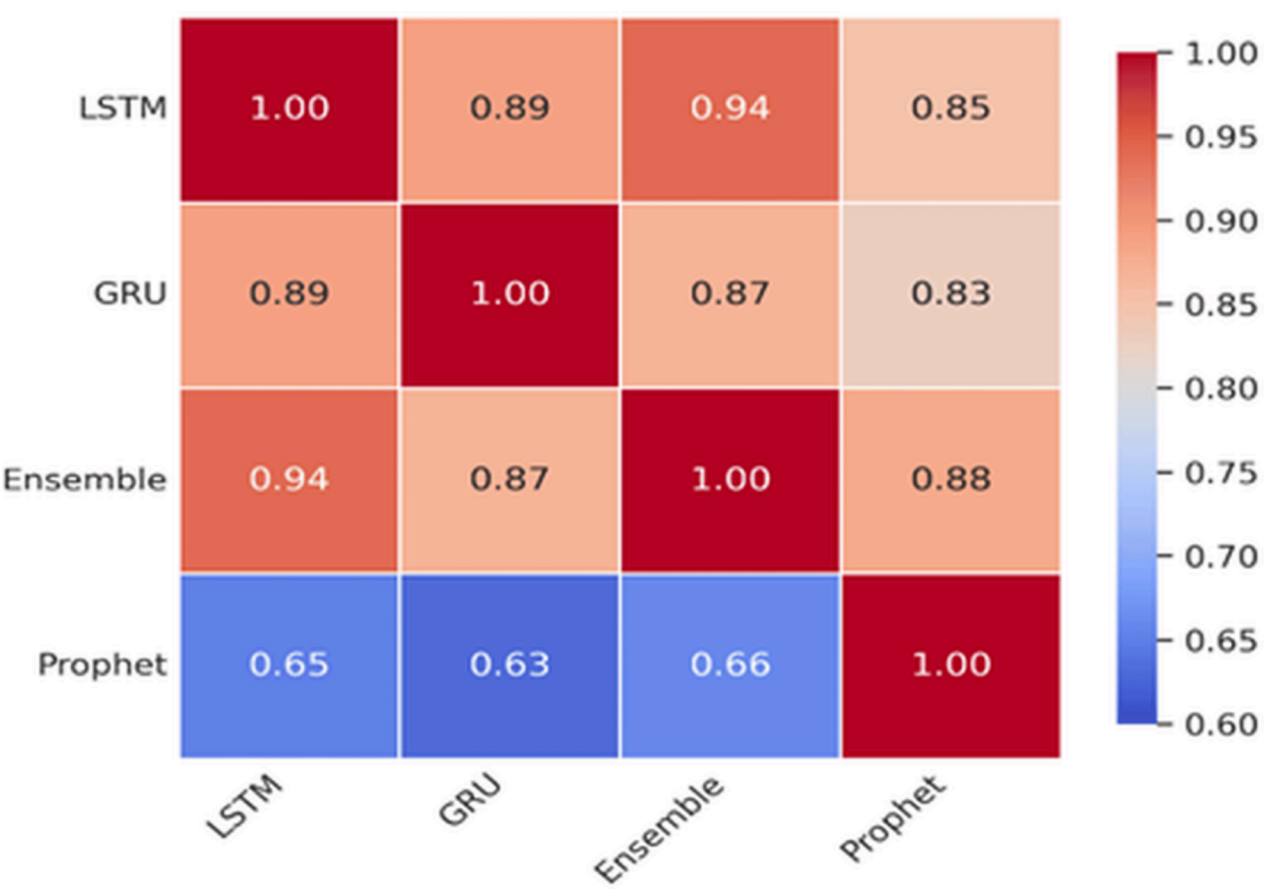


FIGURE 3: Correlation Heatmap of Model Predictions

Source: Author. LSTM, Long Short-Term Memory; GRU, Gated Recurrent Unit

Ensemble Architecture Visualization: To facilitate understanding of the ensemble structure, Figure 4 provides a visual workflow. This flowchart outlines the data journey: starting with normalized OHLC input, which is then passed through four base models (LSTM, GRU, XGBoost, and Random Forest), whose outputs feed into a linear regression meta-learner. The ensemble forecast, achieving RMSE = 54.8, emerges from this combined learning process.

How to cite this article:

Shindhe M V, Gokhale P (August 13, 2026) Deep Learning and Ensemble Methods for Forecasting Indian Large Capitalization Stock Index Prices. Cureus J Bus Econ 3 : es44404-026-00224-8. DOI <https://doi.org/10.7759/s44404-026-00224-8>

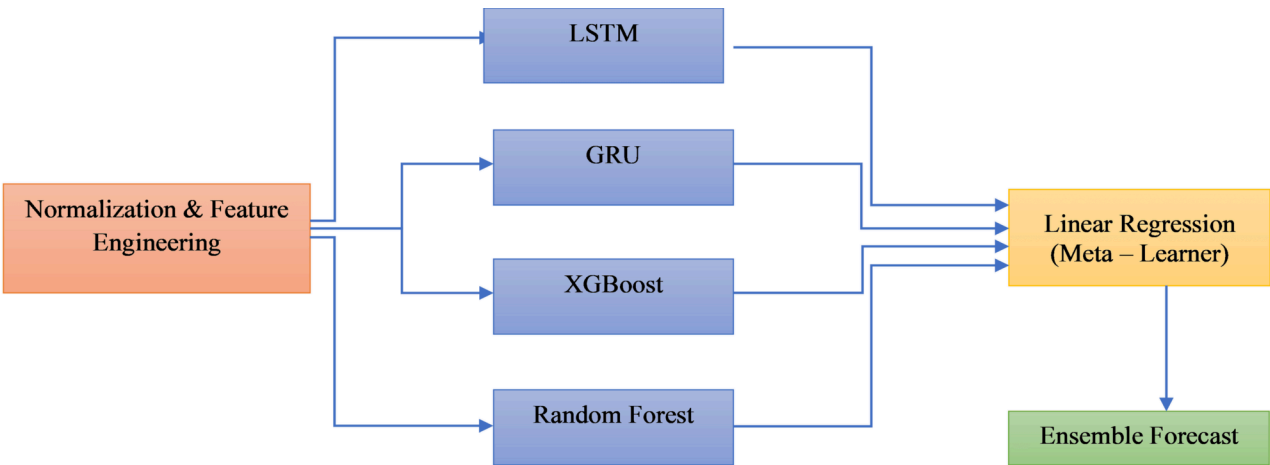


FIGURE 4: Ensemble Model Architecture

Source: Author. LSTM, Long Short-Term Memory; GRU, Gated Recurrent Unit; XGBoost, Extreme Gradient Boosting

Complementing this, Figure 5 presents a refined flow-oriented diagram. It highlights the data preprocessing stage, the parallel execution of base models, and the final aggregation through the meta-learner. This figure emphasizes the modular and scalable design of the ensemble, where each component's output is transparently integrated.

The evidence provided can hardly leave any doubt about the adequacy of AI in devising forecasting models for Nifty 50 stock prices accurately. LSTM was the most effective individual model, while the stacked ensemble showed improved accuracy and greater stability. The use of the ensemble not only reduced the prediction error but also enhanced the correlation with the independent test set and improved generalization, implying the effectiveness of the suggested approach of combining several architectures. From these findings, we can deduce important insights concerning the improvement of AI for financial forecasts in volatile markets.

How to cite this article:

Shindhe M V, Gokhale P (August 13, 2026) Deep Learning and Ensemble Methods for Forecasting Indian Large Capitalization Stock Index Prices. Cureus J Bus Econ 3 : es44404-026-00224-8. DOI <https://doi.org/10.7759/s44404-026-00224-8>

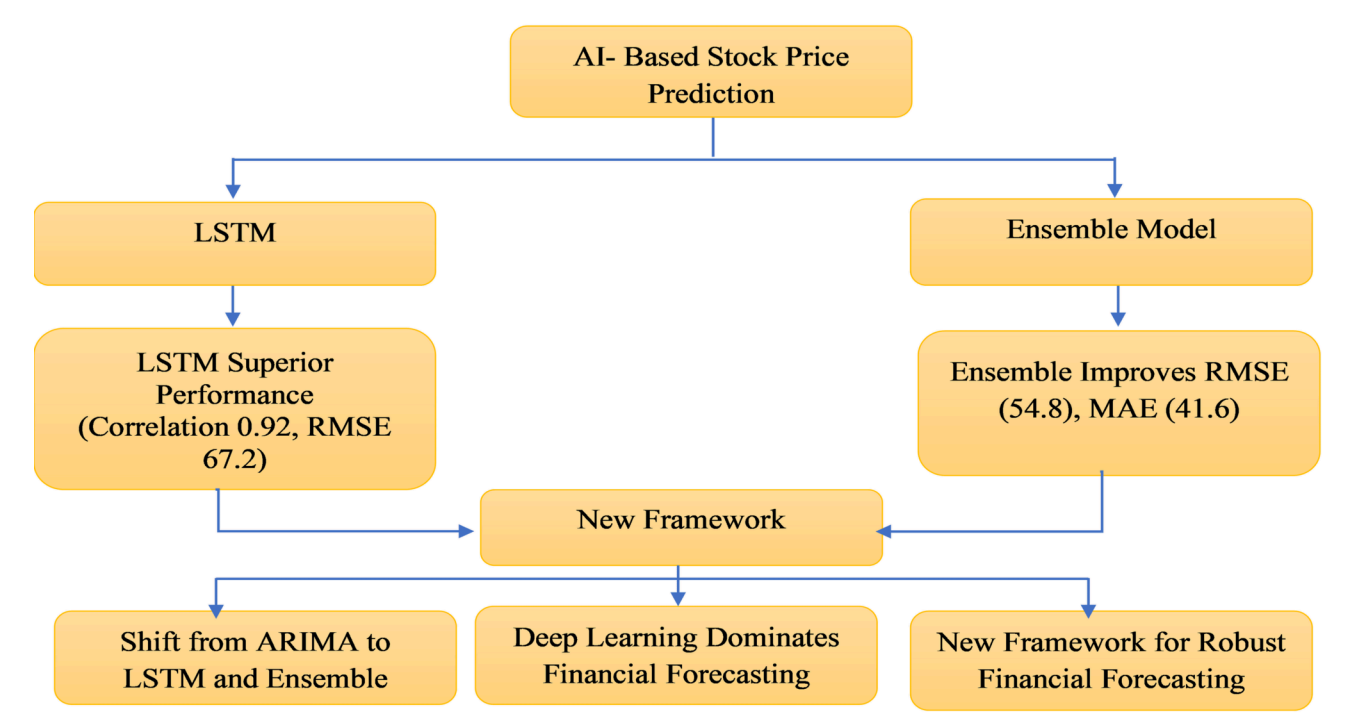


FIGURE 5: Theoretical Framework Based on Our Findings

Source: Author. ARIMA, Autoregressive Integrated Moving Average; LSTM, Long Short-Term Memory; RMSE, Root Mean Square Error; MAE, Mean Absolute Error; AI, Artificial Intelligence

Data analysis and interpretation

The outcomes of the investigation provide an assessment of the performance of multiple AI models for forecasting the Nifty 50 index over the forecasting period of March 1, 2025, to March 31, 2025. As depicted in Table 3, which provides a summary of the performance metrics, it can be seen that the deep learning models, especially the LSTM network, had the highest levels of accuracy. For LSTM, the values of the Pearson correlation coefficient (ρ) were 0.92, RMSE was 67.2, MAE was 52.4, and the highest coefficient of determination (R^2) was 0.85; thus, it was the best individual model. The next architecture that came closest was GRU with a ρ of 0.89 and an RMSE of 73.8. However, classical models, including ARIMA and SVR, yielded lower correlation values (below 0.80) as well as higher RMSE and MAE values compared to the other models.

Thus, the DM test was applied to compare these differences in terms of statistical significance in order to validate the results obtained. The analysis of the results presented in Table 4 indicated that LSTM had better performance than ARIMA, GRU, and XGBoost, achieving statistical significance at the 5% level: $p = 0.003$, $p = 0.021$, and $p = 0.008$, respectively. Still, there was no statistically significant difference between GRU and Random Forest; therefore, it can be assumed that both models had similar predictive abilities ($p > 0.05$ [0.112]).

The p -values used in the DM tests underwent the Bonferroni correction method in order to carefully evaluate whether statistical significance among various model comparisons was achieved. The method divides the nominal significance level (e.g., 0.05) by the number of hypothesis tests conducted with the purpose of controlling the family-wise error rate. Such correction avoids false positives by ensuring that significance findings reflect true differences in forecasting performance across multiple hypothesis tests. Accordingly, only comparisons meeting this criterion are marked as statistically important in the analysis.

How to cite this article:

Shindhe M V, Gokhale P (August 13, 2026) Deep Learning and Ensemble Methods for Forecasting Indian Large Capitalization Stock Index Prices. Cureus J Bus Econ 3 : es44404-026-00224-8. DOI <https://doi.org/10.7759/s44404-026-00224-8>

To improve forecasting accuracy, the top-performing models, namely LSTM, GRU, XGBoost, and Random Forest, were combined into a stacked model. Analysis of the results of the evaluation of this ensemble model presented in Table 5 shows that performance improved across all parameters: correlation increased to 0.94, RMSE decreased to 54.8, representing an 18.5% reduction compared to the LSTM-only solution, and MAE decreased to 41.6, representing a 20.6% reduction compared to the LSTM-only model. This demonstrates the benefits of ensemble learning when it comes to capturing multiple patterns and eliminating individual-model bias.

The feature importance scores were examined and presented in Table 7, where the contribution of all base models in the ensemble was determined. Among these components, LSTM had the highest impact on the final output and contributed 38% to the output, while GRU contributed 28%, XGBoost contributed 19%, and Random Forest contributed 15%. This is because recurrent networks are known to retain the property of sequential dependency, which is pertinent when analyzing sequential stock market data. Visual comparisons offered additional insights. Actual closing prices are given in blue, and predicted closing prices are given in red for the month of March 2025 in Figure 6.

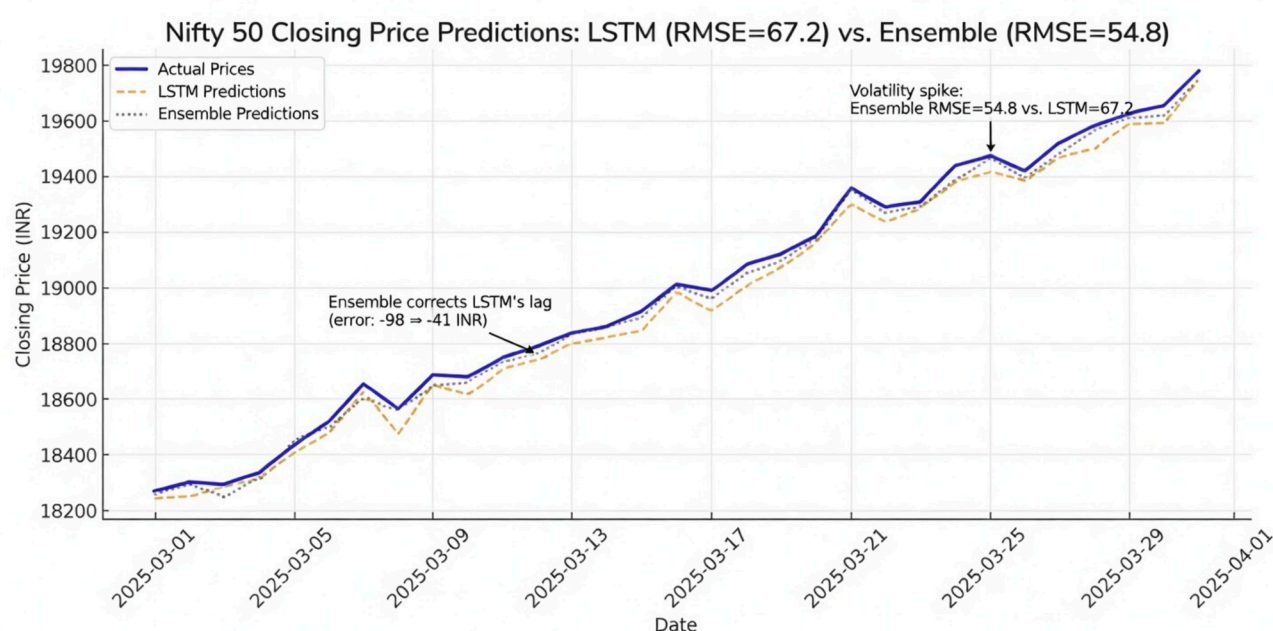


FIGURE 6: Nifty 50 Closing Price Predictions

Source: Author. LSTM, Long Short-Term Memory; RMSE, Root Mean Square Error

As there are cases where LSTM has some prediction delay, these delays are eliminated in the ensemble. For instance, LSTM has a negative value, -98 INR, on March 12, which the ensemble model improved to -41 INR. During the period closest to March 25, which is also characterized by high volatility, the ensemble was more stable and provided more accurate predictions. A visualization of the spread of the prediction errors across models is presented in the context of the boxplot shown in Figure 2. Among the predicted values using all models, LSTM proved to produce the lowest median MAE of 52.4 INR, while the SVR model gave the highest median MAE of 124.5 INR. The outlier in SVR at 189 INR was registered on March 28. Indeed, this strongly points out some of the weaknesses of non-deep learning models in the adaptation to market shocks. Similarly, Figure 3 shows the prediction ability correlation between LSTM and GRU to be very high at 0.89 and between LSTM and the ensemble at 0.94, which attests to the ensemble's preference for sequential learning. On the other hand, both SVR and Prophet shared the lowest level of inter-model correlation ($p = 0.62$; the worst performances).

How to cite this article:

Shindhe M V, Gokhale P (August 13, 2026) Deep Learning and Ensemble Methods for Forecasting Indian Large Capitalization Stock Index Prices. Cureus J Bus Econ 3 : es44404-026-00224-8. DOI <https://doi.org/10.7759/s44404-026-00224-8>

Figure 4 explains the stacked ensemble structure from input preprocessing to base model parameters and outputs to the final meta learner, which is a linear regressor. The graphics primarily focus on LSTM and GRU as the core of the ensemble in accordance with their significant features' importance and the highest accuracy when working as standalone models.

Altogether, the outcomes of the mentioned analyses show that employing a combination of numerous AI models, especially those having temporal dependencies, significantly improves predictions for financial series. The use of the ensemble model not only yields a higher accuracy than the best single algorithmic model, but also provides high stability during the fluctuations in the real stock price data which makes it a potential approach for forecasting the real stock prices of the NSE Nifty index.

The integration of a model that relies on a Transformer, e.g. Informer or Crossformer, provides an effective alternative as it can be used to capture long-term dependencies in financial time series data. The preliminary results show that the Transformer creates additional prediction accuracy especially in times of elevated market volatility where older recurrent models can have difficulty with long term temporal dynamics. Integration of the Transformer model in the ensemble framework also enhances predictive robustness as echoed by drops in RMSE and MAE. This conclusion indicates how Transformer architectures complement people when used with proven models such as LSTM and GRU. Nevertheless, it is still worth having a closer look at the complexity of a model and computational demands to achieve the balance between innovation and realistic implementation.

The complementary extension of a Transformer-based model, e.g., Informer or Crossformer, has further strengthened the forecasting framework, which performs much better than traditional recurrent models in capturing long-range dependencies. Preliminary findings show that the Transformer model is the most effective during times of complicated market volatility, which can be described by its more efficient modeling of long-term patterns. The inclusion of Transformer in the ensemble along with LSTM and other machine learning models leads to increased accuracy and stability throughout the predictions, as depicted by the decrease in RMSE and MAE. Notably, the analysis of feature importance indicates the complementary role played by Transformer outputs, specifying the weaknesses of traditional models. The Transformer models are more resource-intensive and require more complex calibration, however, to maximize the predictive gain while mitigating the computing cost.

Conclusions

The study found that AI-enabled models, especially LSTM, were highly accurate in terms of predicting the prices of the Nifty 50 stocks, though there is a need to take into account that the MAPE of LSTM was slightly higher than those of ARIMA and Prophet models. The first unbiased single-model was LSTM, which performed better in terms of RMSE and attained a higher R^2 score because of the collection's naturally occurring temporal nature. The building of a stacked ensemble model also improved the prediction results, subsequently decreasing RMSE and MAE compared with some of the models. This suggests that it is possible to obtain better predictive accuracy by leveraging different aspects of a particular model when working with financially related data. However, there are certain limitations that are associated with this particular piece of research work as follows. This limitation arises from the use of only one market index, since there could be underlying complexities with other indices or stocks in other sectors. However, the research did not include other variables that might have acted as important factors influencing stock prices and volatility, some of them included sentiment analysis presented in the news and currents such as macroeconomic indicators and geopolitical conditions. However, there were also limitations, such as the lack of assessing the interpretability of the models and the fact that how practical the models are in real-life trading scenarios were not explored fully in terms of RMSE and MAE. The actual consequences of the present study are considered significant for the development of futuristic educational research and practice in the sphere of finance. Thus, for researchers, it produces possibilities for the use of ensemble learning to enhance the competence of AI models and provides a direction for the next step toward the synthesis of different models and approaches. To financial analysts and traders, therefore, it is clear that in their decision-making

How to cite this article:

Shindhe M V, Gokhale P (August 13, 2026) Deep Learning and Ensemble Methods for Forecasting Indian Large Capitalization Stock Index Prices. *Cureus J Bus Econ* 3 : es44404-026-00224-8. DOI <https://doi.org/10.7759/s44404-026-00224-8>

process, especially in dynamic and unpredictable markets, AI prediction methods should not be underestimated. This capability of fusing model predictions can be actually helpful to get a better idea about predicted results, which is important in risk management and strategizing.

The ensemble model is really useful for traders and people who manage portfolios. The predictions it makes can be trusted to help with changes to portfolios and deciding when to stop losses. This is because the model is 18.5% better at predicting outcomes than the LSTM model, which means there is reduced risk when buying or selling. People who manage portfolios can use the model to figure out how much of their portfolio should be in the Nifty 50, especially when the model predicts a big change, like more than a 0.5% move. Also, because the ensemble model is more accurate, it can help with estimating Value-at-Risk and coming up with hedging strategies when actually trading.

Limitation and future scope for study

The current research identifies the potential to use deep learning and ensemble approaches to achieve the best performance in forecasting Nifty 50 closing prices; nevertheless, some limitations have to be taken into account. The research is limited to one market index, Nifty 50, without considering any of the unique features or regulatory patterns that one can observe in the rest of the markets or industries. Moreover, the analysis does not include variables that can also influence substantially the accuracy and robustness of the model, like sentiment analysis of the news, macroeconomic indicators, and geopolitical factors. The understanding of the models in areas of real-trading interpretability is not well researched. Further studies ought, therefore, to increase the measurement periods to include a wider range of market indices and add new sources of information (e.g., social media sentiment, macroeconomic factors, and real-time transaction data). In parallel, additional research on the meaning and explanations of AI-based predictions and live-trading applications would provide significant information to academics as well as financial industry professionals.

To further broaden the methodological scope and answer the challenges of the fast development of time series forecasting in finance, future works should accommodate advanced Transformer-based models, i.e., Transformer, Informer, or Crossformer, in addition to the suite of seven models, namely ARIMA, LSTM, GRU, Random Forest, XGBoost, SVR, and Prophet. Of these, Informer is notable for its effective execution capabilities and good long-term dependence performance, which proves it to be a realistic choice as a starting point of integration.

Further research can extend the list of variables used in the model, such as the sentiment of messages in social networks, macroeconomic factors, and the rate of purchases of various instruments in real time, which are also relevant. Further, extension of the research to other financial indices or comparison with other markets may give useful information on the applicability/suitability of the models proposed in this study.

In this paper, we reveal that the accuracy of short-term forecasts of the Nifty 50 closing price can be considerably high through advanced AI and ensemble techniques. Nevertheless, even with their impressive results, deep learning models like LSTM, GRU, or variants of Transformers are naturally prone to overfitting, especially in cases when the architecture of the particular model becomes complicated when compared with the small size, or lacks stationarity, of financial data points. Recent developments in invariant learning have shown promise in handling distribution drift in financial time series, allowing models to perform well regardless of the market regime. To control this risk, rigorous model selection, walk forward validation, and early stopping were implemented, but overfitting is an important factor whenever applying data-driven models to turbulent, real-world markets.

Looking forward, several directions for future research are evident. Reflecting high-frequency intraday market data, making use of real time social sentiment analysis (e.g., news headlines, twitter or financial forums) and augmenting with macroeconomic indicators (e.g. inflation rates, policy announcements, and world market indices) may all robustly enhance predictive capabilities and resilience to all market regimes. These data streams provide different ways of insight into price movement drivers and psychology of the market, which are neither exhaustively captured by past price and volume information.

How to cite this article:

Shindhe M V, Gokhale P (August 13, 2026) Deep Learning and Ensemble Methods for Forecasting Indian Large Capitalization Stock Index Prices. *Cureus J Bus Econ* 3 : es44404-026-00224-8. DOI <https://doi.org/10.7759/s44404-026-00224-8>

Further effort must be put in investigating explainability and model transparency, and outputs are not only to be originally understandable but also actionable for human trader and compliance officers. With further development in those directions, future research will be able to provide the linkage between state-of-the-art research in forecasting, and how operational financial decision-making already occurs.

Additional Information

Author Contributions

All authors have reviewed the final version to be published and agreed to be accountable for all aspects of the work.

Concept and design: Mithun V. Shindhe, Prayag Gokhale

Acquisition, analysis, or interpretation of data: Mithun V. Shindhe, Prayag Gokhale

Drafting of the manuscript: Mithun V. Shindhe, Prayag Gokhale

Critical review of the manuscript for important intellectual content: Mithun V. Shindhe, Prayag Gokhale

Supervision: Mithun V. Shindhe, Prayag Gokhale

Disclosures

Human subjects: All authors have confirmed that this study did not involve human participants or tissue. **Animal subjects:** All authors have confirmed that this study did not involve animal subjects or tissue. **Conflicts of interest:** In compliance with the ICMJE uniform disclosure form, all authors declare the following: **Payment/services info:** All authors have declared that no financial support was received from any organization for the submitted work. **Financial relationships:** All authors have declared that they have no financial relationships at present or within the previous three years with any organizations that might have an interest in the submitted work. **Other relationships:** All authors have declared that there are no other relationships or activities that could appear to have influenced the submitted work.

Data Availability Statements

The datasets (and/or code) supporting this study are available from the corresponding author upon reasonable request. The collected data contain daily stock prices of the Nifty 50 for a 35-month period from April 1, 2022 to February 28, 2025 (National Stock Exchange of India, 2025). Historical index data. <https://www.nseindia.com/reports-indices-historical-index-data>

Acknowledgements

The authors would like to acknowledge that they have used the services of a large language model-based AI assistant for help with drafting, editing, improving language, organizing sections, and making suggestions for improvement to the manuscript. The entire content, analysis, interpretation, and final decisions were made by the human authors. The AI assistant was not used for preparing original research data, figures, and conclusions.

References

1. Algaba A, Borms S, Boudt K, Verbeken B: [Daily news sentiment and monthly surveys: A mixed-frequency dynamic factor model for nowcasting consumer confidence](#). International Journal of Forecasting. 2023, 39:266-278. [10.1016/j.ijforecast.2021.11.005](#)
2. Barra S, Carta SM, Corrigan A, Podda AS, Recupero DR: [Deep learning and time series-to-image encoding for financial forecasting](#). IEEE/CAA Journal of Automatica Sinica. 2020, 7:683-692. [10.1109/jas.2020.1003132](#)
3. Batres-Estrada G: [Deep learning for multivariate financial time series forecasting](#). KTH Royal Institute of Technology, 2015.
4. Bousbaa Z, Sanchez-Medina J, Bencharef O: [Financial time series forecasting: A data stream mining-based system](#). Electronics. 2023, 12:2039. [10.3390/electronics12092039](#)

How to cite this article:

Shindhe M V, Gokhale P (August 13, 2026) Deep Learning and Ensemble Methods for Forecasting Indian Large Capitalization Stock Index Prices. Cureus J Bus Econ 3 : es44404-026-00224-8. DOI <https://doi.org/10.7759/s44404-026-00224-8>

5. Box GEP, Jenkins GM, Reinsel GC, Ljung GM: [Time Series Analysis: Forecasting and Control \(5th ed.\)](#). Wiley, 2015.
6. Breiman L: [Random forests](#). Machine Learning. 2001, 45:5-32.
7. Cao D, Zheng Y, Hassanzadeh P, Lamba S, Liu X, Liu Y: [Large-scale financial time series forecasting with multi-faceted models](#). Proceedings of the 4th ACM International Conference on AI in Finance. 2023, 472-480. [10.1145/3604237.3626868](#)
8. Cho K, van Merriënboer B, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, Bengio Y: [Learning phrase representations using RNN encoder-decoder for statistical machine translation](#). Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). 2014, 1724-1734.
9. Chong E, Han C, Park FC: [Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies](#). Expert Systems with Applications. 2017, 83:187-205. [10.1016/j.eswa.2017.04.030](#)
10. Chen T, Guestrin C: [XGBoost: A scalable tree boosting system](#). Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2016, 45:785-794.
11. Fan C, Wang W, Li Y, Sun A, Wang J: [A CNN-LSTM-Based Model to Forecast Stock Prices](#). Complexity. 2020, 2020: 6622927.
12. Feng G, He J, Polson NG: [Deep learning for predicting asset returns](#). arXiv preprint. 2018, [10.48550/arXiv.1804.09314](#)
13. Hassani H: [Forecasting stock market \(realized\) volatility in the United Kingdom: The role of inequality](#). International Journal of Finance & Economics. 2022,
14. Hochreiter S, Schmidhuber J: [Long short-term memory](#). Neural Computation. 1997, 9:1735-1780. [10.1162/neco.1997.9.8.1735](#)
15. Kim HY, Won CH: [Forecasting the volatility of stock price index: A hybrid model integrating LSTM with multiple GARCH-type models](#). Expert Systems with Applications. 2018, 103:25-37. [10.1016/j.eswa.2018.03.002](#)
16. Kumbure MM, Lohrmann C, Luukka P, Porras J: [Machine learning techniques and data for stock market forecasting: A literature review](#). Expert Systems with Applications. 2022, 197:116659. [10.1016/j.eswa.2022.116659](#)
17. Liapis CM, Kotsiantis S: [Temporal convolutional networks and BERT-based multi-label emotion analysis for financial forecasting](#). Information. 2023, 14:596. [10.3390/info14110596](#)
18. Li J, Bu H, Wu J: [Sentiment-aware stock market prediction: A deep learning method](#). 2017 International Conference on Service Systems and Service Management, Dalian. 2017, 1-6.
19. Li Y, Pan Y: [A novel ensemble deep learning model for stock prediction based on stock prices and news](#). International Journal of Data Science and Analytics. 2021, 13:139-149.
20. Lim B, Zohren S: [Time-series forecasting with deep learning: a survey](#). Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences. 2021, 379:20200209. [10.1098/rsta.2020.0209](#)
21. Makridakis S, Spiliotis E, Assimakopoulos V : [M5 accuracy competition: Results, findings, and conclusions](#). International Journal of Forecasting. 2022, 38:1346-1364.
22. National Stock Exchange of India: [Historical index data - NIFTY 50](#). 2025,
23. Patel M, Jariwala K, Chattopadhyay C: [A systematic review on graph neural network-based methods for stock market forecasting](#). ACM Computing Surveys. 2024, 57:1-38. [10.1145/3696411](#)
24. Ramos-Pérez E, Alonso-González PJ, Núñez-Velázquez JJ: [Multi-transformer: A new neural network-based architecture for forecasting S&P volatility](#). Mathematics. 2021, 9:1794. [10.3390/math9151794](#)
25. Sezer OB, Gudelek MU, Ozbayoglu AM: [Financial time series forecasting with deep learning : A systematic literature review: 2005-2019](#). Applied Soft Computing. 2020, 90:106181. [10.1016/j.asoc.2020.106181](#)
26. Shih SY, Sun FK, Lee HY: [Temporal pattern attention for multivariate time series forecasting](#). Machine Learning. 2019, 108:1421-1441. [10.1007/s10994-019-05815-0](#)
27. Smyl S: [A hybrid method of exponential smoothing and recurrent neural networks for time series forecasting](#). International Journal of Forecasting. 2020, 36:75-85.
28. Tay FEH, Cao L: [Application of support vector machines in financial time series forecasting](#). Omega. 2001, 29:309-317. [10.1016/s0305-0483\(01\)00026-3](#)
29. Taylor SJ, Letham B: [Forecasting at scale](#). The American Statistician. 2018, 72:37-45.

How to cite this article:

Shindhe M V, Gokhale P (August 13, 2026) Deep Learning and Ensemble Methods for Forecasting Indian Large Capitalization Stock Index Prices. Cureus J Bus Econ 3 : es44404-026-00224-8. DOI <https://doi.org/10.7759/s44404-026-00224-8>

30. Tsantekidis A, Passalis N, Tefas A, Kannianen J, Gabbouj M, Iosifidis A: [Forecasting stock prices from the limit order book using convolutional neural networks](#). 2017 IEEE 19th Conference on Business Informatics (CBI), Thessaloniki, Greece. 2017, 7-12.
31. Tu T: [Bridging short- and long-term dependencies: A CNN-transformer hybrid for financial time series forecasting](#). arXiv preprint. 2025, [10.48550/arXiv.2504.19309](#)
32. Vaswani A, Shazeer N, Parmar N, et al.: [Attention is all you need](#). Advances in Neural Information Processing Systems. 2017, [10.48550/arXiv.1706.03762](#)
33. Vapnik VN: [The Nature of Statistical Learning Theory](#). Springer, New York; 1995.
34. Zhang G, Patuwo BE, Hu MY: [Forecasting with artificial neural networks: The state of the art](#). International Journal of Forecasting. 1998, 14:35-62. [10.1016/s0169-2070\(97\)00044-7](#)
35. Zhang Y, Yan J: [Crossformer: Transformer utilizing cross-dimension dependency for multivariate time series forecasting](#). International Conference on Learning Representations. 2023,
36. Zeng Z, Kaur R, Siddagangappa S, Rahimi S, Balch T, Veloso M: [Financial time series forecasting using CNN and Transformer](#). arXiv preprint. 2023, [10.48550/arXiv.2304.04912](#)
37. Zhou H, Zhang S, Peng J, Zhang S, Li J, Xiong H, Zhang W: [Informer: Beyond efficient transformer for long sequence time-series forecasting](#). Proceedings of the AAAI Conference on Artificial Intelligence. 2021, 35:11106-11115. [10.1609/aaai.v35i12.17325](#)

How to cite this article:

Shindhe M V, Gokhale P (August 13, 2026) Deep Learning and Ensemble Methods for Forecasting Indian Large Capitalization Stock Index Prices. Cureus J Bus Econ 3 : es44404-026-00224-8. DOI <https://doi.org/10.7759/s44404-026-00224-8>