International Journal of Enhanced Research in Management & Computer Applications,
ISSN: 2319-7471, Volume 15 Issue 7, July 2026

Predictive Analytics for Stock Market Trends: A Comparative Study of SARIMA and LSTM Architectures for NIFTY Index Forecasting

Mansi[1], Amandeep[2]

[1]M.Sc. Computer Science[1], Artificial Intelligence and Data Science, GJUS&T
[2]Assistant Professor[2], Artificial Intelligence and Data Science, GJUS&T

ABSTARCT

Reliable forecasts of equity-index behaviour underpin risk management, portfolio allocation and market oversight. This paper presents a controlled comparison of a classical linear model, the Seasonal Autoregressive Integrated Moving Average model with exogenous regressors (SARIMAX), and a deep-learning sequence model, the Long Short-Term Memory (LSTM) network, applied to the daily closing price of the NIFTY 50 index of India's National Stock Exchange. A six-stage pipeline - data ingestion, preprocessing, exploratory diagnostics, parallel SARIMAX and LSTM branches, and a shared evaluation layer - was designed and implemented. An AIC/BIC grid search selected SARIMAX (5,1,2) for the univariate closing-price series, while a stacked LSTM network was trained on scaled multivariate Open-High-Low-Volume features. Both models were evaluated on an identical seventeen-day hold-out window immediately following the COVID-19 crash (15 April-8 May 2020) using RMSE, MAE and R2. LSTM achieved RMSE = 260.99, MAE = 190.96, R2 = 0.972, while SARIMAX returned RMSE = 361.17, MAE = 299.94 and a negative R2 of -1.959, performing worse than a naive mean forecast. A residual-correction hybrid SARIMA-LSTM architecture is proposed at the design level but left for empirical evaluation because of the compounding-error risk of its recursive multi-step structure. The results support the broader view that nonlinear, feature-rich deep-learning models tend to dominate linear statistical models during structural market disruption, while the confounding effects of the unequal input sets and the atypical test window are discussed explicitly as limitations.

***I*ndex Terms—SARIMA, SARIMAX, LSTM, time-series forecasting, NIFTY 50, stock market prediction, deep learning, Indian stock exchange.**

INTRODUCTION

Price information flows continuously in the equity markets as it is jointly shaped by macroeconomic data releases, investor sentiment, geopolitics, and algorithmic activity. The closing-price series resulting from the interactions of buyers and sellers may be a priori neither totally random nor fully determined. This is what makes short-horizon index forecasting attractive yet difficult. Furthermore, even a small improvement in forecasting ability can make a big difference in portfolio construction, hedging and systemic-risk monitoring [1].

Two broad methodological lines of research have come out of financial forecasting. The classical statistical models make up the first category, principally ARIMA and its seasonal extension SARIMA, which express future values as a linear combination of past values and past forecast errors after differencing removes non-stationarity [2][3]. Models ensure transparency and modest data requirements, but their linearity assumption sits uncomfortably with the regime shifts and feedback effects that characterise real markets [4]. The second line incorporates deep sequence models, particularly the LSTM network, which has input, output and forget gates. Using this enables it to retain or discard information across long sequences without the vanishing-gradient problem that befalls plain recurrent networks [5]. It is commonly reported that LSTM architectures outperform ARIMA-type models in volatile series, at the expense of interpretability, computational demand and data appetite [4][7].

A recent analysis sought to fill this gap by tracking a ‘buy and hold’ investment in three different scenarios: investing directly in the NIFTY 50 index, investing in an equivalent American index, and following NIFTY 50 signals in 2007. The gap mentioned in the preceding paragraph is addressed by designing, implementing, and evaluating a SARIMAX model and a multivariate LSTM network, both on NIFTY 50 closing prices under matched conditions. At the design level, a residual-correction hybrid architecture combining the two is proposed.

This work includes the following contributions: a reproducible and modular pipeline that separates data preparation, diagnostics and the two modelling branches behind a shared evaluation layer; an automated AIC/BIC order-selection procedure for SARIMAX that identifies and discards a numerically degenerate candidate order; evaluation of both models on an identical, unusually demanding hold-out window using four complementary error measures together with residual diagnostics; design and algorithmic specification of a hybrid SARIMA-LSTM extension, with the reasons for deferring its empirical evaluation made explicit.

The paper's remaining sections shall be organised accordingly. Section II provides a literature review. This is explained in Section III. The system architecture and modelling procedure are described in Section IV. Section V presents findings. Section VI concerns findings and limitations; Section VII concludes.

RELATED WORK

A. Business Marketplace and Future Pricing

A lot of literature regarding forecasting is framed in the light of the efficient market hypothesis (EMH), which states that prices reflect the available information to the full extent. Historical patterns do not have any predictive value [10][1].

Nonetheless, empirical studies on Indian indices note the presence of momentum effects, seasonality, and volatility clustering, which do not conform to strict efficiency [3][9] these phenomena are being reconciled by the Adaptive Market Hypothesis [11]. This type of time-varying partial inefficiency forms the theoretical backbone of our comparative exercise.

B. Statistical linear models.

ARIMA stands for Autoregressive Integrated Moving Average and is a forecasting algorithm used for time series prediction. Research on Indian markets cites reasonable short-horizon accuracy, which degrades over longer horizons or across structural breaks [3][12][1]. Some of the limitations that these have are that they can't capture nonlinear dependence, buggy code requiring heavy preprocessing and fixing of parameters that don't adapt to regime shifts and restriction without extending to SARIMAX [12][13][14]. SARIMA is still a standard benchmark on account of its transparent, well-understood coefficients.

C. LSTM Networks

The memory cell with gates of the LSTM network is designed to remove the vanishing-gradient problem and the requirement of stationarity transformations [5][6]. Studies in the Indian market, as quoted regarding their notable reduction in errors with respect to ARIMA. Sisodia et al. on the NIFTY 50 at RMSE being of near 60 with LSTM as against a value of 104 with ARIMA [9]. Similarly, Goswami et al. present a similar case on NIFTY Bank [8]. Architectural variants like bidirectional LSTM can enhance accuracy further, while the choice of LSTM, RNN and CNN sliding window models impacts results significantly. The use of heavy computation, sensitive to hyperparameters, risks overfitting in the absence of regularisation and a high loss of interpretability [7].

Table I. Summary of Key Studies in Financial Forecasting Literature

Study	Focus Area	Principal Finding
Ariyo et al. [2]	ARIMA, global indices	Effective short-term; degrades over longer horizons
Banerjee [3]	ARIMA, Indian indices	Good trend capture; weak across abrupt changes
Sirisha et al. [13]	SARIMA on NIFTY	Handles seasonality; limited during regime shifts
Daryl et al. [12]	SARIMA, structural breaks	Strong in stable periods; fails at structural breaks
Siami-Namini et al. [4]	LSTM vs. ARIMA	LSTM is consistently better; the largest gains under volatility
Sisodia et al. [9]	LSTM on NIFTY 50	RMSE ~60.2 (LSTM) vs. ~104.3 (ARIMA)
Goswami et al. [8]	LSTM on NIFTY Bank	RMSE ~70.4 (LSTM) vs. ~110.7 (ARIMA)

E. Research Gaps

This study is motivated by the following four gaps. First, there is limited comparative work that applies an identical protocol to both model families on Indian indices. Second, the published comparisons use inconsistent metrics and windows. Third, there is rare regime-stratified reporting. There is always a risk that aggregate errors mask regime-dependent behaviour. Fourth, there has been little explicit treatment of the accuracy-interpretability trade-off for Indian retail-investor and advisory contexts [7][10]. The current study deals with each of them directly while recognising in Sections VI-VII that complete resolution would require multi-regime, multi-index validation that goes beyond the present study.

METHODOLOGY

To properly evaluate the performance of a linear and a non-linear model, more than one accuracy measure is needed, since different penalty error types, and aggregate statistics can hide regime-dependent behaviour [13]. The framework developed herein aims at achieving four goals. First, characterising accuracy through various complementary error measures. Second, testing whether relative performance varies with market regime. Third, quantifying trade-offs between accuracy, interpretability and cost. Fourth, providing a reusable template for other NIFTY-family and other indices.

A. Mistake Associated Gauges

RMSE employs squaring of the errors before averaging, and as a result, large deviations will be disproportionately penalised [13]. MAE measures the average of the absolute difference between observed and predicted values. MAPE as a percentage of the actual number allows the error to be compared. The formula $R^2 = 1-[\Sigma(y_i-\hat{y}_i)^2/\Sigma(y_i-\bar{y})^2]$ measures how much variance is explained. If $R^2 < 0$, that means the model performs worse than a naive mean forecast. [13]

B. The criteria of directional accuracy and regime stratification.

The model of tomorrow's move must be different from the model of today. In other words, the forecast must indeed show a directional move, either up or down. Since aggregate error can mask regime-dependent behaviour, we also compute metrics separately on the stable/trending regime and the volatile/shock regime. This more rigorously tests whether SARIMA still remains competitive in calm markets (as suggested in the literature [4]) while LSTM dominates during turbulence (as suggested in the literature [13][11]). A model here is paired, and a comparison of forecast errors is made to determine whether a real quantity is more significant than zero. Thus, one determines whether the difference between forecasts is real or mere noise [13]. Further qualitative assessment of each model follows, which assesses interpretability and computational costs, such as training time, hardware needs and retraining overhead [6][7][10]. The framework is summarised in Table II.

TABLE II. SUMMARY OF THE PERFORMANCE-EVALUATION FRAMEWORK

Dimension	Measure(s)	Purpose
Accuracy (magnitude)	RMSE, MAE	Size of forecast errors
Accuracy (scale-free)	MAPE	Compare across price levels
Explained variance	R^2	Overall pattern captured
Directional performance	Directional accuracy (%)	Usefulness for trading signals
Regime sensitivity	Metrics per regime	Volatility dependence
Statistical validity	Paired error test	Rule out chance differences
Interpretability/cost	Qualitative scoring	Usability, deployment feasibility

SYSTEM ARCHITECTURE AND MODELLING METHODOLOGY

The SARIMAX and LSTM branches were developed using the same data and evaluation layer thanks to a modular pipeline approach in the system. As such, we can modify them both independently. Both branches process identical NIFTY data and are evaluated on the same hold-out window; thus, performance differences reflect a modelling choice rather than a data-handling choice; every intermediate output is inspectable, as per the interpretability aim of the work. Figure The six-layer architecture that this work results in is presented in 1: it comprises a data layer, preprocessing, exploratory analysis, a train/test split, 2 independent modelling branches and a shared evaluation layer.

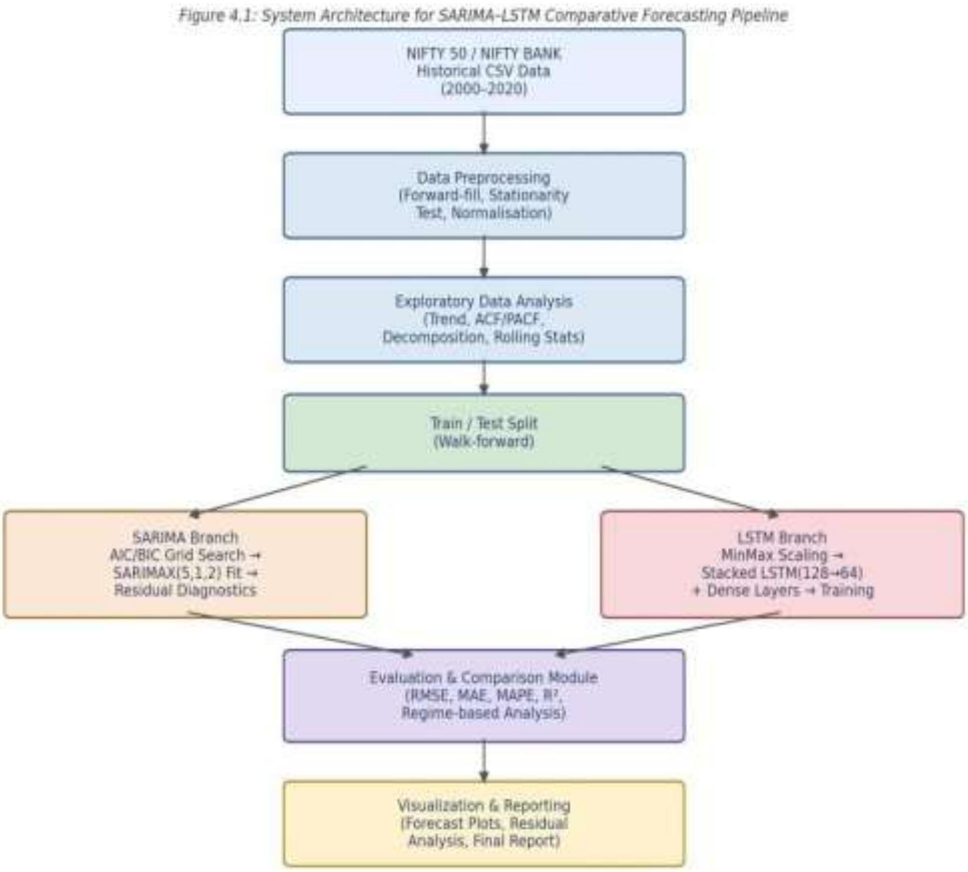

Fig. 1. System architecture of the SARIMAX-LSTM forecasting pipeline.

A. Preparing Data.

The input contains Nifty 50 and Nifty Bank daily data (2000 – 2020) – with Open, High, Low, Close, Volume, Turnover, PE, PB and Dividend Yield fields. NIFTY 50 serves as the main visualisation target; NIFTY Bank facilitates comparative insight. Values for volume and turnover were shifted forward. The closing-price series underwent an Augmented Dickey-Fuller (ADF) test to check for stationarity, with first-order differencing applied ahead of SARIMAX, while Open, High, Low and Volume were MinMax-scaled to [0,1] ahead of LSTM.

B. Exploratory Analysis and SARIMAX

Branch Before the modelling took place, the series was characterised by trend plots, a normalised NIFTY 50/NIFTY Bank comparison, rolling-average smoothing, seasonal decomposition and ACF/PACF analysis. The SARIMAX component is formalised in Algorithm 1: an exhaustive grid search over $p \in [0,5]$, $d \in [0,1]$, $q \in [0,2]$ scores each candidate by AIC/BIC on the training window; the best combination is refit on the full training data; standardised residual diagnostics and in-sample MAE are computed; and a confidence-bounded forecast is generated over the test window.

Algorithm 1: SARIMAX Order Selection and Forecasting

Input: Closing-price series {y_t}, t = 1..N
Candidate orders p in [0,5], d in [0,1], q in [0,2]
Output: Fitted SARIMA(p*,d*,q*) model, in-sample residuals,
17-day-ahead forecast with confidence interval

1. Split {y_t} into a training window (2018 to 15 Apr 2020) and a 17day hold-out test window (Section 3.6).
2. Run the Augmented Dickey-Fuller test on the training window; Record whether the non-stationarity null is rejected (Section 4.5).
3. best_score <- +infinity
4. for p in 0..5, d in 0..1, q in 0..2:
5. fit candidate <- SARIMAX (p, d, q) on training window 6. score <-
(AIC (candidate), BIC (candidate))

7. if score improves on best_score:
8. best_score <- score; (p*, d*, q*) <- (p, d, q) 9. Refit SARIMA (p*, d*, q*) on the full training window -> SARIMA (5, 1, 2) (Section 4.7, step 2).
10. Compute standardised residuals of the refit model; Generate histogram vs. fitted normal density, Q-Q plot and correlogram; compute insample residual MAE (Section 4.7, step 3).
11. Generate a 17-step-ahead forecast, with confidence interval, over the test window.
12. return fitted model, residuals, forecast

C. Branch of LSTM.

The LSTM branch, whose architecture is fixed as in Algorithm 2, directly learns parameters from scaled OHLV features and does not use P/E, P/B and Dividend Yield, as these are not strictly price-only oriented. I rescaled the features using MinMaxScaler and reshaped them into a 3-D tensor format. I passed this into an LSTM (128, ReLU) → LSTM (64) → Dense (32) → Dense (1) network. I trained the network using Adam and MSE loss for 100 epochs with a batch size of 128. I turned shuffling off to maintain the chronological order in the training data.

Algorithm 2: LSTM Training and Forecasting

Input: Feature series Open_t, High_t, Low_t, Volume_t; target Close_t Output: Trained LSTM model; predictions for the test window

1. X <- [Open_t, High_t, Low_t, Volume_t] for all t
   (P/E, P/B, Dividend Yield excluded; Section 1.8, Section 4.8, step 1).
2. y <- Close_t
3. scaler_X, scaler_y <- MinMaxScaler fitted on the training portion of X, y
4. X_scaled <- scaler_X.transform(X); y_scaled <- scaler_y.transform(y) 5. Reshape X_scaled to a 3-D tensor of shape (samples, features, 1) (Section 4.8, step 2).
6. Build network:

LSTM layer (ReLU activation)
LSTM layer (ReLU activation)
Dense layer (ReLU activation) Dense output layer (linear activation) (Section 4.8, step 3).

7. Compile network with Adam optimiser, mean-squared-error loss. 8. Train for 100 epochs, batch size 128, shuffle = False for both training and test data, to preserve temporal order (Section 4.8, step 4).
9. y_pred_scaled <- model.predict(X_scaled for test window)
10. y_pred <- scaler_y.inverse_transform(y_pred_scaled)
11. return trained model, y_pred

D. Analysis of System

Tech Stack and Design Notes: Trees, RMSE, MAE, MAPE, R-squared for both, Comparison of residuals, Comparison of forecast-comparison plots, regime stratification, testing for significance of either model without going into either model. The pipeline handles data in pandas/NumPy, visualises with matplotlib/seaborn/plotly, runs stationarity testing, ACF/PACF, seasonal decomposition and SARIMAX in statsmodels, scales in scikit-learn, and LSTM in Keras, all done in Google Colab. The proposed design makes SARIMAX a univariate model and LSTM a multivariate model, reflecting the intrinsic strength of each model being independent. Also, because they are independent, the proposed residual correction hybrid can be formed by letting the SARIMAX residual series be the LSTM input according to the two-stage decomposition (TSD) strategy shown to be effective elsewhere [15][14]. This hybrid is specified only at the design level; its empirical evaluation is deferred to the future for the reasons laid out in Section VI.

RESULTS

This section presents the results obtained by running the pipeline of Section IV on the NIFTY 50 and NIFTY Bank datasets in the given order: exploratory findings; stationarity/correlation diagnostics; SARIMAX order selection and fitting; LSTM training; comparisons of accuracy and a stress test.

A) Findings and Stationarity Exploratory.

Between 2000 and 2020, the NIFTY 50 index closed between 854.20 and 12,362.30. The NIFTY Bank index closed between 743.70 and 32,443.85. The larger range of the NIFTY Bank index is quite indicative of the higher volatility in the banking sector when compared to the NIFTY 50. NIFTY Bank had a higher cumulative growth than NIFTY 50, but it had to afford higher volatility in order to achieve that. The ADF test on the raw closing-price series gave a statistic of -0.7656 and p = 0.8290 (5% critical value -2.8621; 5,035 observations), so the non-stationarity null cannot be rejected, confirming a random walk-like series requiring differencing. After differencing one of the series, the series fluctuated around zero with minimum and maximum values of −1135.20 and 708.40, respectively, which confirms the d = 1 condition. A seasonal decomposition with a 7-day window on the data from 2018 to 2020 showed that the trend component closely followed the price with a small seasonal component in the weekly domain that had a nonzero value, and finally, the residual component that captured the issue in March 2020. According to the ACF/PACF graphs, the

ACF was showing a slow decay, which indicates a persistent series, while the PACF is cutting off sharply after lag 1 and then with a smaller spike further (Fig. 2). This indicates a proposal to check over an AR of up to the order of 5.

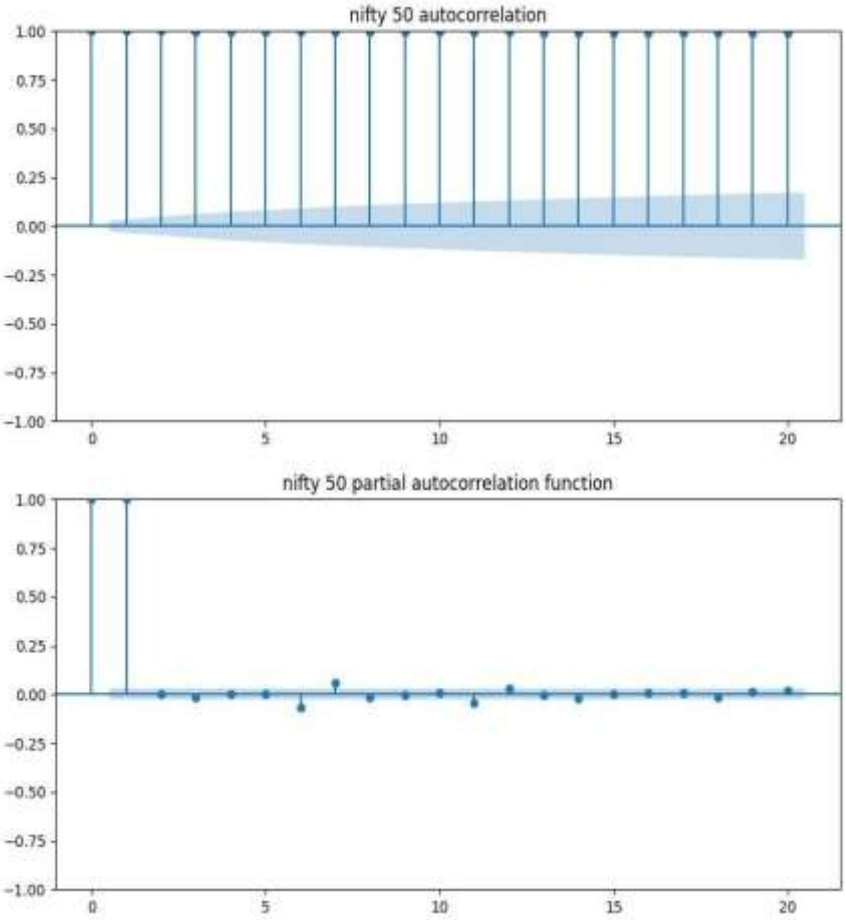

Fig. 2. Autocorrelation (top) and partial autocorrelation (bottom) of the NIFTY 50 closing price, 20 lags.

B) Selection of SARIMAX order and fitting

The grid search ran 36 candidates, for p∈ {0...,5}, d∈ {0,1}, q∈ {0,1,2}; the results can be seen in Table III, which lists the five with the lowest AIC. The automatic choice of top candidate (3,1,1) reported a suspiciously low AIC (~10.0), which is characteristic of a degenerate fit that has failed to converge, so it was rejected as a numerical artefact. The next best candidate (5,1,2) changed slightly from the surrounding candidates, and this model was adopted as the operative specification with fixed intercepts. (BIC = 7,061.33; AIC = 7,026.69) Using maximum likelihood estimation on the training partition (2018-15 Apr 2020), the model produced a mean absolute in-sample residual of 102.14 index points. Residual diagnostics reveal near-zero autocorrelation with a pronounced volatility cluster around the March-April 2020 shock and heavier-than-normal tails.

TABLE III. TOP FIVE SARIMAX CANDIDATE ORDERS RANKED BY AIC

Rank	P	d	q	AIC	BIC
1*	3	1	1	~10.0 (artefact)	~31.65 (artefact)
2	5	1	2	7,026.69	7,061.33
3	3	1	2	7,038.55	7,064.53
4	5	1	1	7,057.31	7,087.61
5	4	0	2	7,061.41	7,091.73

*The rank-1 (3,1,1) result was excluded as a numerical artefact; SARIMAX (5,1,2) was adopted.

C) Predicting Outcomes.

Figure 3 exhibits the SARIMAX prediction and its 95% confidence interval over the test period. While the point forecast flattens, the interval widens rapidly, which is a characteristic of ARIMA-family models for near-random-walk processes. The actual price continues to move directionally through the recovery, while the forecast stays fairly flat.

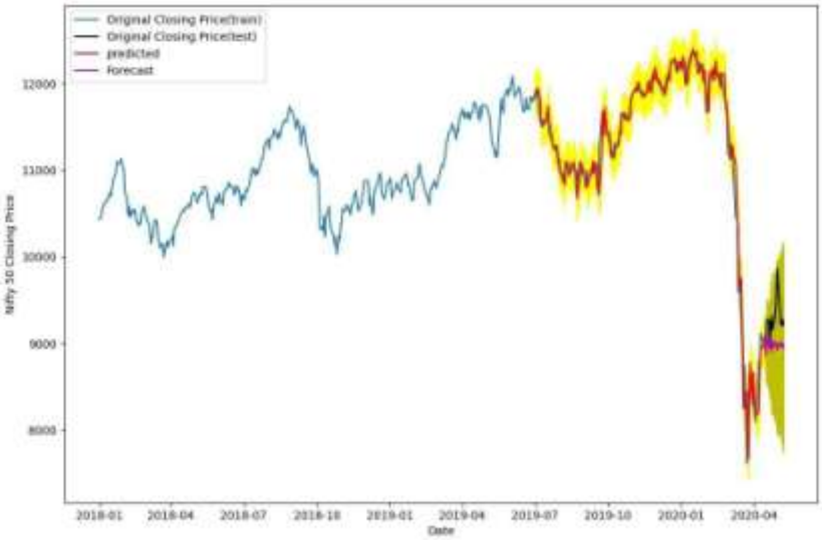

Fig. 3. SARIMAX in-sample prediction and out-of-sample forecast with 95% confidence interval.

The LSTM part has 4,976 training and 86 test sequences, each comprising four features, obtained using scaling and reshaping. The training loss decreased from over 3.6×10^7 to a stable plateau in 10-15 epochs. The trained network achieved a test RMSE of 260.99; Fig. As Figure 4 shows, unlike SARIMAX, LSTM prediction follows both the level and the short-term direction of the actual series, including recovery post-shock.

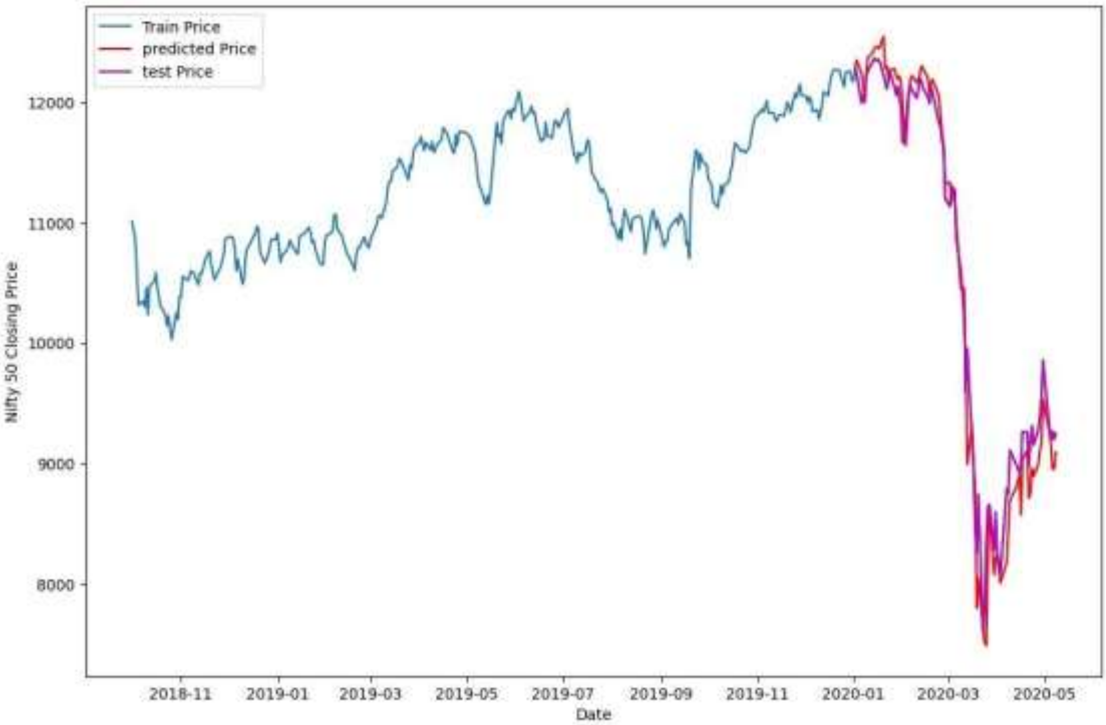

Fig. 4. LSTM predicted closing price versus actual closing price, October 2018 onward.

D) Comparative Performance

Table IV summarises quantitative performance on the identical test period. LSTM outperforms SARIMAX on every metric: RMSE is about 28% lower and MAE about 36% lower. Most notably, LSTM achieves $R^2 = 0.972$ (97.2% of variance explained) against SARIMAX's $R^2 = -1.959$, meaning the SARIMAX forecast performed worse than a naive mean forecast - a direct expression of the flattening behaviour in Fig. 3.

TABLE IV. COMPARATIVE FORECASTING PERFORMANCE ON THE NIFTY 50 TEST PERIOD

Model	RMSE	MAE	R^2
SARIMAX (5,1,2)	361.17	299.94	−1.959
LSTM (128→64→32→1)	260.99	190.96	0.972

SARIMAX residuals grow steadily more negative as the horizon extends, indicating accumulating bias, while LSTM residuals stay closer to zero and are more stable; the corresponding error distributions confirm lower bias and variance for LSTM. Fig. 5 consolidates the comparison: the LSTM curve tracks the actual price far more closely than SARIMAX, which diverges as the horizon lengthens.

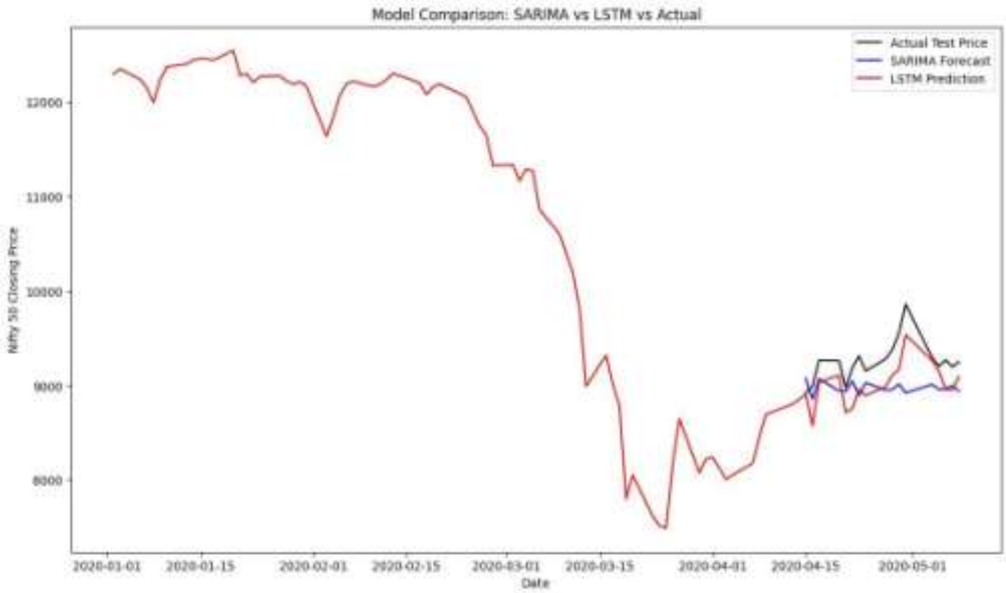

Fig. 5. Combined comparison of actual NIFTY 50 closing price, SARIMAX forecast and LSTM prediction.

E. Stress-Testing Results

A synthetic +5% shock applied to the scaled test-set inputs, without retraining, produced a shocked LSTM prediction uniformly higher than the normal prediction while preserving the same overall shape - evidence that the network responds to input magnitude in a stable, directionally sensible way rather than producing erratic output, supporting its suitability for exploratory what-if analysis alongside point forecasting.

DISCUSSION

Given the considerations used in this study, the results consistently indicate that the NIFTY 50 closing-price series exhibits a high degree of persistence. Therefore, it should be characterised as close to a non-stationary process. In addition, the short-horizon behaviour of this series appears to be better characterised by a nonlinear, multi-feature model than by a linear model fitted to the univariate closing price alone. Thus, this outcome confirms erroneous one-dimensional estimates. Finally, the results seem to be over the window and in this entire study. Although SARIMAX, backed by a robust AIC/BIC selection and reasonable training-data diagnostics, exhibited adequate training performance, it poorly generalised out of sample. The test window closely followed an extremely low-frequency shock that a linear model estimated on pre-shock data was unable to foresee. In contrast, LSTM leveraged same-day OHLV knowledge and nonlinear representation to track the recovery much more closely.

There are several caveats to this conclusion. Due to the intentionally turbulent nature of the evaluation window, this difference is likely to be larger than one would observe in calmer periods and should not be taken to be making a general claim to LSTMs' superiority. The asymmetrical inputs confound the comparison because SARIMAX used only the closing price while LSTM used OHLV, and therefore part of the advantage of LSTM could simply be due to its richer feature set rather than its architecture. This could be tested more fairly by using a multivariate SARIMAX with the same regressors. The forecasting and evaluation were limited only to NIFTY 50; a single train/test split was used and not a rolling-origin validation; further, the metrics used measure price-level accuracy, as opposed to profitability or risk-adjusted return, which would have required a dedicated back-testing framework.

The trade-off between interpretability and accuracy continues to be relevant in practice. In particular, SARIMAX's interpretable coefficients can be explained to non-technical market stakeholders, which is relevant for retail investors, fiduciary advisors and regulators. But with LSTM, there is no comparable degree of transparency, even though LSTM is more accurate in this instance. This is arguably a more actionable conclusion than simply ranking predictive accuracy. The proposed residual-correction hybrid was not tested empirically because the recursive, multi-step design can compound small residual errors that may arise early in the forecast period. Analysing this properly requires a dedicated error-propagation analysis and is left for future work.

CONCLUSION AND FUTURE WORK

The study outlined, constructed and analysed a module-wise pipeline contrasting SARIMAX and multivariate LSTM networks on NIFTY 50 closing prices under stipulated conditions, and earmarked a residual-correction hybrid architecture for future study. We can draw three conclusions. Nifty 50 is confirmed to be non-stationary. It is also highly persistent and hence adheres to weak-form efficiency. Nifty 50 requires differencing and not level modelling. The model SARIMAX (5,1,2), despite being a robust choice, generalised to poor performance on the test window of COVID-19 with a negative R-squared. LSTM performed overwhelmingly well against SARIMAX on every metric with residuals that were less biased, smaller, and more stable. Empirically, SARIMAX (5,1,2) recorded a test RMSE of 361.17, MAE of 299.94, and R^2 of −1.959, while the LSTM network achieved a test RMSE of 260.99, MAE of 190.96, and R^2 of 0.972 — a roughly 28% reduction in RMSE and 36% reduction in MAE over SARIMAX. The ADF test further confirmed non-stationarity in the NIFTY 50 series (statistic = −0.7656, p = 0.829), validating the need for differencing before autoregressive modelling. Further, its impulse response to a synthetic input shock was stable and directionally sensible.

Contributions include a reusable layered framework separating data preparation, diagnostic and independent modelling branches; a criterion-driven SARIMAX order-selection procedure that expressly excludes a degenerate candidate; a like-for-like comparison through multiple complementary metrics; and a stress-testing module extending the framework beyond point forecasting. The limitations noted in section VI apply to these assessments: the failure of the test window to be representative, asymmetric input sets, single-index scope, no rolling-origin validation, and no profitability metrics.

Future work should evaluate both models across multiple regimes (low-volatility, trending, crisis) to test whether the LSTM advantage persists; extend SARIMAX to a multivariate or vector-autoregressive form using the same regressors as LSTM; apply the framework to NIFTY Bank and other indices; adopt rolling-origin validation with confidence intervals; explore attention-based models, gradient-boosted trees and further ensembling; and, building directly on this design, empirically evaluate the proposed hybrid SARIMA-LSTM architecture alongside a backtesting layer that reports risk-adjusted trading performance rather than statistical accuracy alone. To conclude, during the particular demanding window studied, the multivariate LSTM network significantly outperformed the univariate SARIMAX model in forecasting NIFTY 50. Further, this is an evidence-based result constrained by the data and features used and not a blanket assertion about deep learning superiority. Also, we offer this as an extensible template for future researchers to make a similar comparative study on Indian and other emerging market indices.

REFERENCES

1. P. Mondal, L. Shit, and S. Goswami, "Study of effectiveness of time series modeling (ARIMA) in forecasting stock prices," International Journal of Computer Science, Engineering and Applications, vol. 4, no. 2, pp. 13-29, 2014.
2. A. Ariyo, A. O. Adewumi, and C. K. Ayo, "Stock price prediction using the ARIMA model," in Proc. 2014 UKSim-AMSS 16th Int. Conf. Computer Modelling and Simulation (UKSim), Cambridge, U.K., 2014, pp. 106-112.
3. D. Banerjee, "Forecasting of Indian stock market using time-series ARIMA model," in Proc. 2014 2nd Int. Conf. Business and Information Management (ICBIM), 2014, pp. 131-135.
4. S. Siami-Namini, N. Tavakoli, and A. Siami Namin, "A comparison of ARIMA and LSTM in forecasting time series," in Proc. 2018 17th IEEE Int. Conf. Machine Learning and Applications (ICMLA), Orlando, FL, USA, 2018, pp. 1394-1401.
5. D. M. Q. Nelson, A. C. M. Pereira, and R. A. de Oliveira, "Stock market's price movement prediction with LSTM neural networks," in Proc. 2017 Int. Joint Conf. Neural Networks (IJCNN), Anchorage, AK, USA, 2017, pp. 1419-1426.
6. M. A. I. Sunny, M. M. S. Maswood, and A. G. Alharbi, "Deep learning-based stock price prediction using LSTM and bi-directional LSTM model," in Proc. 2020 2nd Novel Intelligent and Leading Emerging Sciences Conf. (NILES), 2020, pp. 87-92.
7. M. Pirani, P. Thakkar, P. Jivrani, M. H. Bohara, and D. Garg, "A comparative analysis of ARIMA, GRU, LSTM and BiLSTM on financial time series forecasting," in Proc. 2022 IEEE Int. Conf. Distributed Computing and Electrical Circuits and Electronics (ICDCECE), 2022, pp. 1-6.
8. S. Goswami, A. K. Sagar, P. Nand, and O. I. Khalaf, "Time series analysis using stacked LSTM model for Indian stock market," in Proc. 2022 IEEE IAS Global Conf. Emerging Technologies (GlobConET), 2022, pp. 399-405.

9. 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. 2022 2nd Int. Conf. Innovative Practices in Technology and Management (ICIPTM), vol. 2, Gautam Buddha Nagar, India, 2022, pp. 156-161.
10. J. Patel, S. Shah, P. Thakkar, and K. Kotecha, "Predicting stock market index using fusion of machine learning techniques," Expert Systems with Applications, vol. 42, no. 4, pp. 2162-2172, 2015.
11. A. Chatterjee, H. Bhowmick, and J. Sen, "Stock price prediction using time series, econometric, machine learning, and deep learning models," in Proc. IEEE Int. Conf. MysuruCon 2021, Hassan, Karnataka, India, 2021, pp. 289-296.
12. Daryl, A. Winata, S. Kumara, and D. Suhartono, "Predicting stock market prices using time series SARIMA," in Proc. 2021 1st Int. Conf. Computer Science and Artificial Intelligence (ICCSAI), vol. 1, Jakarta, Indonesia, 2021, pp. 92-99.
13. U. M. Sirisha, M. C. Belavagi, and G. Attigeri, "Profit prediction using ARIMA, SARIMA and LSTM models in time series forecasting: A comparison," IEEE Access, vol. 10, pp. 124715-124727, 2022.
14. H. Yu, L. J. Ming, R. Sumei, and Z. Shuping, "A hybrid model for financial time series forecasting - integration of EWT, ARIMA with the improved ABC optimised ELM," IEEE Access, vol. 8, pp. 84501-84518, 2020.
15. S. A. Dwivedi, A. Attry, D. Parekh, and K. Singla, "Analysis and forecasting of time-series data using S-ARIMA, CNN and LSTM," in Proc. 2021 Int. Conf. Computing, Communication, and Intelligent Systems (ICCCIS), 2021, pp. 131-136.