Cureus Journal of Computer Science

Open Access Original Article

Modeling Coffee Price Volatility Using Time Series, Hybrid, and Deep Learning Approaches

Christina EA [1] , Hemalatha N [1] , Sathish Kumar Narayanan [2]

1. Information Science and Technology, St Aloysius Deemed to be University, Mangalore, IND 2. Business Process Compliance, EDGE Commercial, Dubai, ARE

Corresponding authors: Christina EA, christinaantony1609@gmail.com, Hemalatha N, hemasree71@gmail.com, Sathish Kumar Narayanan, auditorskn@gmail.com

Received 06/10/2025
Review began 06/29/2025
Review ended 08/24/2025
Published 08/27/2025

DOI:
https://doi.org/10.7759/s44389-025-07255-5

Abstract

Coffee prices are known for their unpredictability, shaped by different aspects such as weather, trade dynamics, and seasonal demand. In this study, several forecasting methods were applied to monthly coffee price data to find a model that balances accuracy with the ability to handle volatility. Techniques ranged from classic time series models to deep learning and hybrid approaches. The best performer was the long short-term memory model, which effectively captured complex, nonlinear patterns and volatility in coffee price movements, resulting in the lowest error rates among all models tested. While traditional statistical models captured trends, they often struggled to keep up with price variations. The long short-term memory model was trained for a maximum of 200 epochs; however, training effectively converged and was stopped early around the 55th epoch, as the loss stabilized at a low value. This early convergence further shows the model's efficiency in learning complex, nonlinear patterns from the coffee price data. This study would help the policy-makers, agriculturer, and traders.

Categories: Computer Vision, Machine Learning (ML), Deep Learning
Keywords: coffee prices, lstm, volatility hybrid, time series, epochs

Introduction

Coffee ranks among the most consumed beverages globally and holds significant weight in international trade, coming second only to crude oil in export volume [1]. In numerous coffee-producing nations, particularly those in Latin America, Africa, and Southeast Asia, coffee is more than a commodity; it is crucial in sustaining the livelihoods of many small-scale farmers and forms a foundational part of rural economies. Despite this importance, coffee prices are far from stable. They are subject to sharp and unpredictable swings due to a mix of weather disturbances, pest infestations, global market fluctuations, currency instability, and speculative trading activity [2,3]. These sudden changes create considerable difficulties for those involved in the supply chain, specifically farmers and exporters, who rely on predictable prices for budgeting, planting, and sales planning [4].

Accurate forecasting of coffee prices, therefore, is not merely of academic interest but a practical necessity. Price predictability can help mitigate risk, stabilize revenues, and support more informed decisions by farmers, traders, and policymakers operating in an increasingly uncertain global market environment. As agricultural markets become more exposed to external shocks, the demand for robust, adaptable forecasting models continues to grow.

In response to this need, the present study conducts a comprehensive comparative analysis of forecasting techniques applied to international coffee prices. The models considered include both conventional time series approaches commonly used for structured trend and seasonal data - and advanced deep learning architectures capable of capturing nonlinear patterns and complex dependencies in sequential data. By evaluating these models side by side under the same conditions, this research compares approaches to find out which models best capture the volatility and trend dynamics that characterize agricultural commodity markets.

The dataset used in this study comprises monthly international coffee prices measuring the period from 1973 to 2023. This data was publicly shared by M. Abdullah Sajid through an open-access repository [5]. Most of the previous research focuses on standalone models, making it harder for people to know which model to rely on when market conditions vary. Moreover, with the growing interest in deep learning and automated forecasting tools like long short-term memory (LSTM) and Prophet, there is a need to assess how these novel methods compare to traditional methods. This study examines how various forecasting models - including both traditional time series techniques and modern deep learning methods - perform when applied to monthly coffee price data. These include AutoRegressive Integrated Moving Average (ARIMA), Seasonal AutoRegressive Integrated Moving Average (SARIMA), Error, Trend, Seasonality (ETS), Prophet, Trigonometric, Box-Cox, ARMA errors, Trend, and Seasonal components (TBATS), LSTM, and SARIMA-Generalized AutoRegressive Conditional Heteroskedasticity (GARCH). Rather than assuming one model works best, the aim is to evaluate which models offer a good balance between accuracy and the ability to handle price swings. Our findings are intended to help farmers, traders, and policy makers who need practical tools to manage uncertainty in a highly volatile global market.

Materials And Methods

This section explains the data used in the study, preprocessing steps, forecasting methodology, and different forecasting models used in this work.

Dataset description

The dataset used in this study was obtained from a publicly accessible Kaggle repository [5]. It includes daily international price records for several agricultural commodities covering the period from 1973 to 2023. The dataset contains the following columns: Date, Coffee (USD), Wheat (USD), and Corn (USD). For the present analysis, only the Date and Coffee (USD) columns were selected. This choice narrowed the focus of the study to a univariate time series, concentrating on international coffee prices over time. The raw dataset comprised 12,540 daily observations, which were initially stored in CSV format and imported using the pandas library in Python.

Preprocessing steps

After converting the Date column into a proper datetime format, the daily coffee price data was resampled to a monthly frequency by calculating the average price within each calendar month. This step helped smooth short-term fluctuations while retaining overall price trends, resulting in around 400 monthly observations. Missing or invalid dates were removed, and any remaining gaps were filled using linear interpolation to preserve continuity. The processed series was visually inspected using plots of rolling mean and standard deviation, and an additive seasonal decomposition was performed to observe long-term trends and seasonal components. Stationarity of the series was further checked using the Augmented Dickey-Fuller (ADF) and Kwiatkowski-Phillips-Schmidt-Shin (KPSS) tests, confirming its suitability for time series modeling.

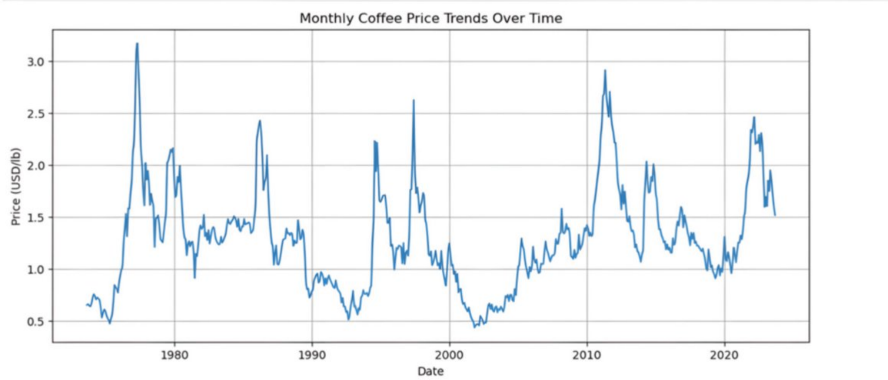

FIGURE 1: Monthly international coffee prices (1973–2023), showing substantial price volatility and multiple long-term cycles.

Figure *1* illustrates how coffee prices have changed on a monthly basis over several decades. Throughout the period, several sharp spikes are evident, particularly in the late 1970s and late 1990s, with a moderate rise around 2010. Sudden increases in coffee prices appear to have been affected by a range of factors, including adverse weather conditions, supply interruptions, and changes in global demand. After these sharp changes, prices usually fell again, though not in a consistent or easily predictable pattern. Overall, the pattern suggests that coffee prices tend to fluctuate over time, largely responding to shifts in broader global conditions.

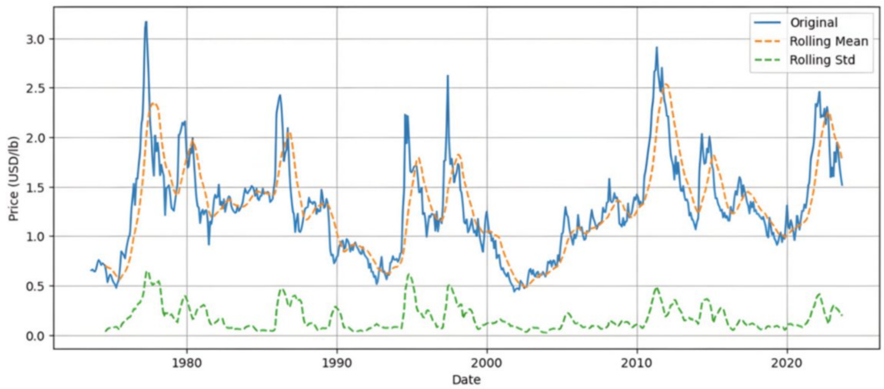

FIGURE 2: Rolling averages and standard deviations are shown alongside the initial price series.Visible shifts indicate non-stationarity in the coffee price data.

Figure 2 illustrates the monthly coffee prices along with their rolling mean and rolling standard deviation. The rolling mean clearly shows long-term trends by smoothing short-term variations, while the rolling standard deviation denotes how much the prices varied during different periods. The rolling mean closely follows the original price spikes, showing repeated cycles over time. Interestingly, the rolling standard deviation remains relatively steady, suggesting that although coffee prices shifted frequently, the overall level of volatility did not change drastically throughout the observed time span.

To determine whether the data was Stationarity, two standard tests were applied. The ADF test returned a p-value of approximately 0.00016, indicating that the null hypothesis of non-stationarity could be dismissed. The KPSS test, with a p-value of 0.1, did not provide sufficient evidence to reject the stationarity assumption. These results jointly supported the use of the original (non-differenced) series for model development. Additionally, a natural log transformation was applied to stabilize variance, a common preprocessing step in financial and commodity price modelling. The dataset was divided chronologically, reserving the final 24 months for out-of-sample testing, while the remaining portion was used for model training and validation.

Forecasting methodology

This study primarily aimed to evaluate how different forecasting models performed on same dataset to determine their effectiveness in predicting volatile coffee prices. Each model was trained using the training portion of the dataset and then used to generate predictions for the 24 month test window.

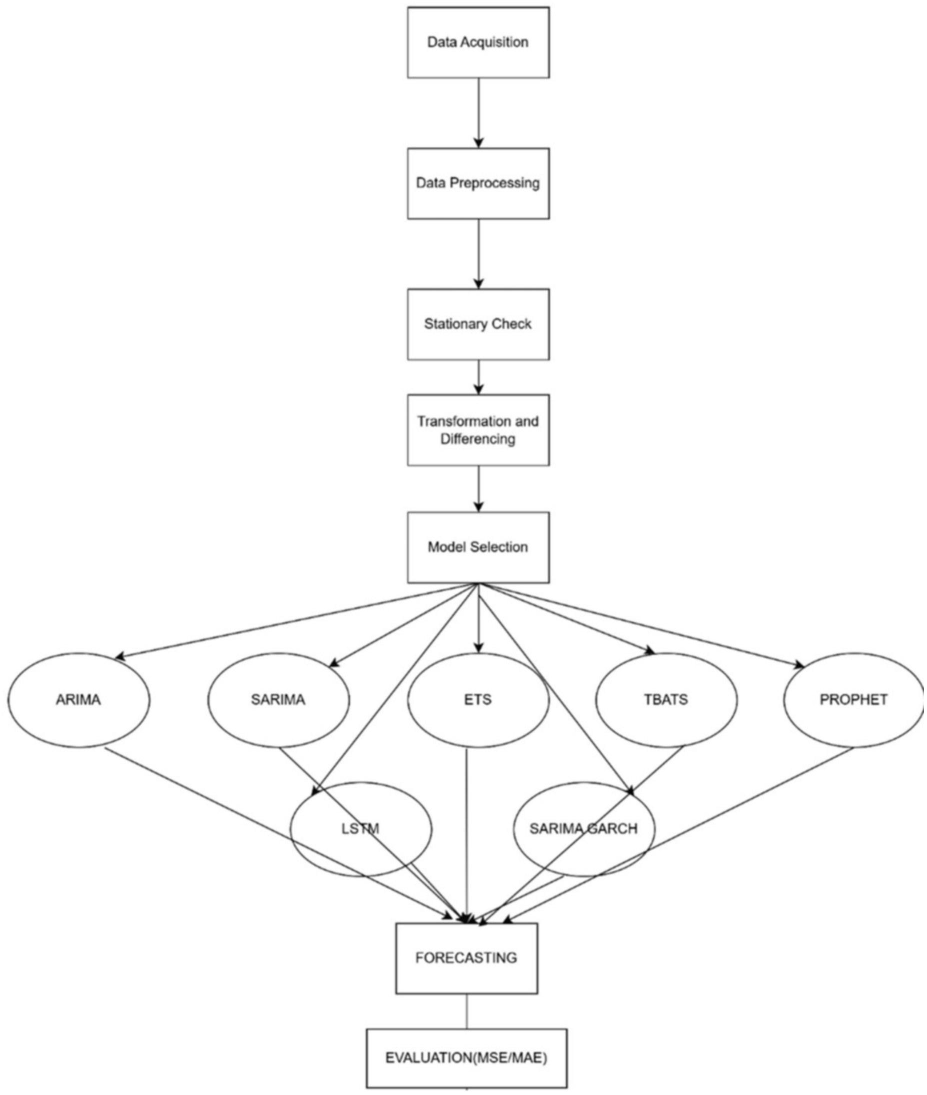

FIGURE 3: Flowchart of forecasting.

ARIMA, AutoRegressive Integrated Moving Average; SARIMA, Seasonal AutoRegressive Integrated Moving Average; ETS, Error, Trend, Seasonality; TBATS, Trigonometric, Box-Cox, ARMA errors, Trend, and Seasonal components; LSTM, Long Short-Term Memory; GARCH, Generalized AutoRegressive Conditional Heteroskedasticity

Figure *3* illustrates the complete forecasting pipeline implemented in this study. The process begins with data acquisition, where historical daily commodity prices were retrieved from a publicly available Kaggle dataset. This is followed by data preprocessing, which included converting date formats, selecting relevant variables, resampling the series to monthly frequency, and handling missing values through interpolation. A stationarity check was then carried out to determine whether differencing or transformation was needed. For models that require a stationary input, appropriate transformations such as differencing or logarithmic scaling were applied.

Once the data was prepared, the model selection phase involved applying six forecasting models: ARIMA, SARIMA, ETS, TBATS, Prophet, and a hybrid SARIMA-GARCH model. In parallel, an LSTM-based neural network was used to model complex, non-linear relationships. Each model was trained on an identical training dataset and tasked with forecasting the final 24 months of the time series. The final stage involved forecast generation and evaluation. Model forecasts were compared against observed values from real data using by applying common evaluation measures to assess predictive accuracy across all methods.

Forecasting models

Seven models were evaluated in this study to forecast monthly coffee prices. Each model identifies a unique structural aspect of the data trend, seasonality, and/or volatility. The mathematical formulation and theoretical foundation for each model are described below.

ARIMA

The ARIMA model, developed by Box and Jenkins [6], is defined as

$$y_t = c + \sum_{i=1}^{p} \phi_i y_{t-i} + \sum_{j=1}^{q} \theta_j \varepsilon_{t-j} + \varepsilon_t$$

where:

y_t represents the value of the time series at time t after applying d levels of differencing to remove trends or seasonality.

c is a constant term, which accounts for the average value of the series if it does not fluctuate around zero.

ϕ_i are the coefficients of the autoregressive component. Each coefficient quantifies the influence of a previous observation (y_{t-i}) on the current value.

θ_j are the coefficients of the moving average component. These describe the impact of past error terms (ϵ_{t-j}) on the current observation.

ϵ_t denotes the random error at time t, also known as white noise. It is assumed to have a mean of zero and constant variance and to be uncorrelated over time.

p indicates the number of lagged values used in the autoregressive part of the model.

q specifies how many past forecast errors are included in the moving average part of the model.

ARIMA models short-term linear dependencies and is appropriate for stationary series [6].

SARIMA-GARCH

The SARIMA-GARCH hybrid model combines SARIMA for modeling trend and seasonality with GARCH for modeling volatility in residuals.

Step 1: Fit SARIMA Residuals e_t are obtained from the SARIMA model.

Step 2: Fit GARCH on residuals.

$$e_t = \sigma_t z_t, \quad z_t \sim \mathcal{N}(0,1), \quad \sigma_t^2 = \omega + \sum_{i=1}^{p} \alpha_i e_{t-i}^2 + \sum_{j=1}^{q} \beta_j \sigma_{t-j}^2$$

where:

e_t is the residual from the SARIMA model.

$\sigma^2{}_t$ is the conditional variance of e_t, which changes over time.

z_t is a standard normal random variable.

α_i and β_j are GARCH parameters capturing the effect of past squared residuals and past variances, respectively.

ω is a constant that ensures positive variance.

p and q are the orders of the GARCH and ARCH terms, respectively.

This hybrid approach effectively captures both seasonality and volatility, making it highly suitable for commodities like coffee [7,8].

SARIMA

The SARIMA model extends ARIMA by incorporating seasonal patterns. It is denoted as:

$$\mathrm{SARIMA}(p, d, q)(P, D, Q)_s$$

where:

(p, d, q) are the non-seasonal orders: autoregressive terms, differencing, and moving average terms, respectively.

(P, D, Q) are the seasonal counterparts, applied to past values and errors at lag s.

s is the seasonal period.

SARIMA is especially suited for agricultural commodities with cyclical trends, such as coffee [8].

LSTM

LSTM is a deep learning architecture from the family of Recurrent Neural Networks, capable of learning long-term dependencies. LSTMs use a memory cell controlled by input, output, and forget gates to maintain information over time. While there is no single defining equation, the core operations include

$$\begin{aligned}
f_t &= \sigma(W_f[h_{t-1}, x_t] + b_f) \\
i_t &= \sigma(W_i[h_{t-1}, x_t] + b_i) \\
\tilde{C}_t &= \tanh(W_C[h_{t-1}, x_t] + b_C) \\
C_t &= f_t * C_{t-1} + i_t * \tilde{C}_t \\
O_t &= \sigma(W_O[h_{t-1}, x_t] + b_O) \\
h_t &= O_t * \tanh(C_t)
\end{aligned}$$

where:

x_t is the input vector at time t, such as a price or feature at that time step.

h_t is the hidden state at time t, representing the output passed to the next LSTM unit.

C_t is the cell state, which carries long-term memory across the sequence.

f_t is the forget gate, controlling how much of the previous cell state (C_{t-1}) is retained.

i_t is the input gate, determining how much of the new information ($\tilde{C}_t$) is added to the cell state.

$\tilde{C}_t$ is the candidate value for updating the cell state, computed using the current input and previous hidden state.

O_t is the output gate, deciding how much of the cell state should be exposed to the output.

W_f, W_i, W_C, W_O are weight matrices learned during training.

b_f, b_i, b_C, b_O are the corresponding bias terms.

σ denotes the sigmoid activation function, which outputs values between 0 and 1.

tan$h$ is the hyperbolic tangent function, outputting values between -1 and 1.

* denotes element-wise multiplication.

LSTM models are especially useful when data exhibit complex or nonlinear dependencies [9-11].

In this study, an LSTM model was implemented with a stacked configuration consisting of two LSTM layers, with 100 neurons in the first layer and 50 neurons in the second. The larger first layer allows the network to extract a broad range of temporal features, while the second layer, with fewer neurons, condenses these features to enhance generalization and avoid overfitting. This architecture was chosen after initial experimentation to strike an optimal balance between predictive accuracy, model complexity, and computational efficiency, particularly given the dataset size of approximately 400 monthly observations.

To further reduce the risk of overfitting, each LSTM layer was followed by a dropout layer with a dropout rate of 0.2. The network concludes with a dense output layer containing a single neuron, which generates the forecast for the next month's coffee price. A sliding time window of 12 months was used to create the input sequences, and the data was scaled to the range (0, 1) using the MinMaxScaler.

The model was trained with the Adam optimizer (learning rate = 0.001) and mean squared error (MSE) as the loss function. Training was performed for up to 200 epochs, with early stopping (patience = 10) to halt the process once the loss stabilized and no further improvement was observed. A summary of the model architecture and hyperparameters is presented in Table *1*.

Hyperparameter	Value
Layers	2 LSTM + 2 Dropout + 1 Dense
Neurons per layer	100, 50
Dropout rate	0.2 (after each LSTM layer)
Activation	tanh
Optimizer	Adam (lr = 0.001)
Loss function	Mean Squared Error
Batch size	32
Epochs	200
Input window size	12 months
Data scaling	MinMaxScaler (0, 1)

TABLE 1: LSTM model architecture and training parameters.

LSTM, Long Short-Term Memory

Prophet

Prophet, developed by Facebook, models a time series as a sum of trend, seasonality, and holiday effects:

$$y(t) = g(t) + s(t) + h(t) + \varepsilon_t$$

where:

$y(t)$ represents the actual observed value of the time series at a specific time t. It is the real data point recorded or measured at that moment.

$g(t)$ is the trend component, which can be piecewise linear or logistic, used to capture non-periodic changes over time.

$s(t)$ models recurring seasonal patterns, typically using Fourier series.

$h(t)$ accounts for effects of holidays or irregular events defined by the user.

ε_t is the error term representing unexplained noise, assumed to be normally distributed.

It is designed for business and social science time series with strong, regular seasonalities [12].

ETS

The ETS model represents a family of exponential smoothing models, which decompose a time series into its components. The additive error-trend-seasonality form is given by

$$y_t = l_{t-1} + b_{t-1} + s_{t-m} + \varepsilon_t$$

where:

y_t is the observed value of the time series at time t.

l_{t-1} is the level component at time t-1, representing the baseline value of the series.

b_{t-1} is the trend component at time t-1, indicating the slope or rate of change in the level over time.

s_{t-m} is the seasonal component from the same season in the previous cycle, where m is the length of the seasonality period.

m is the seasonal cycle length.

ε_t is the error term at time t, capturing the random variation not explained by the other components. It is typically assumed to have mean zero and constant variance.

This model is based on the state space framework of exponential smoothing [13].

TBATS

TBATS is ideal for capturing multiple or complex seasonal patterns. The model is generally expressed as:

$$y_t = l_t + \sum_{i=1}^{k} \left[\alpha_i \sin(\lambda_i t) + \beta_i \cos(\lambda_i t)\right] + \varepsilon_t$$

where:

y_t is the observed value of the time series at time t.

l_t is the level component, representing the baseline value of the series.

α_i and β_i are coefficients for the sine and cosine terms, respectively, which capture seasonal fluctuations.

λ_i are the seasonal frequencies used to model multiple seasonal cycles.

k is the number of seasonal harmonics included.

t represents time, typically measured in consistent intervals such as days, months, or years.

$\sin(\lambda_i t)$ and $\cos(\lambda_i t)$ are trigonometric expressions that represent seasonal variation by modeling repeating patterns in time series data.

ε_t is the error term or noise component at time t, assumed to be independently distributed, representing random variations not captured by the model.

This model was proposed by De Livera et al. [14].

Performance assessment

To evaluate how well time series models perform, statistical indicators are used to measure the gap between predicted outcomes and actual observations. Among the most widely used indicators are the Mean Absolute Error (MAE) and the Root Mean Squared Error (RMSE) [6,13]. The MAE measures the average absolute difference between the actual values and the model's predictions. It provides a straightforward explanation of forecast accuracy by treating all errors equally, regardless of their direction. The MAE is mathematically expressed in the following equation:

$$MAE = \frac{1}{N}\sum_{n=1}^{N}|d_n - y_n|$$

where:

N is the total number of observations.

d_n is the actual (observed) value.

y_n is the predicted value for the n-th observation.

The RMSE takes a slightly different method by calculating the root of the mean squared gap between what was predicted and what actually occurred. Since the errors are squared before averaging, RMSE gives more weight to larger errors, making it especially sensitive to significant mistakes. Its formula is presented in the following equation:

$$RMSE = \sqrt{\frac{1}{N}\sum_{n=1}^{N}(d_n - y_n)^2}$$

In general, lower values of both MAE and RMSE point to better forecasting results. Both indicators were used to evaluate how effectively the models forecasted coffee price trends, providing a straightforward basis for comparing their accuracy [6,13].

Results And Discussion

The dataset used in this study contained 12,540 daily observations of international coffee prices, spanning the years 1973 to 2023. To reduce high-frequency fluctuations and highlight long-term dynamics, the data was resampled to a monthly frequency using arithmetic means. This transformation produced a univariate time series comprising approximately 400 monthly data points.

To evaluate the models, the last 24 months of data were isolated as the test set, while the remaining portion served as the training set. This split was selected to ensure that evaluation was based on two full seasonal cycles. Seven forecasting models were implemented using identical data ARIMA, SARIMA, SARIMA-GARCH, ETS, TBATS, Prophet, and LSTM neural network. All models were trained and tested using the same train-test configuration. Out-of-sample evaluation was performed using the last 24 months of data, ensuring that all models were tested over two complete seasonal cycles. For ARIMA and SARIMA, hyperparameters (p, d, q) were optimized using AIC and BIC selection criteria. For LSTM, a 12-month rolling input window and 50 training epochs were determined after initial experiments to balance model stability and performance. Early stopping was applied to prevent overfitting. The LSTM model was trained for a maximum of 200 epochs with early stopping based on validation loss. Training concluded at epoch 55, where the model's loss stabilized at approximately 0.0032, indicating convergence without overfitting. The model's performance was assessed using two commonly used accuracy measures MAE and RMSE, calculated across the 24-month test period.

LSTM achieved the lowest forecasting errors, and its deep learning architecture is less interpretable compared to traditional models like ARIMA and SARIMA, which provide transparent parameter-based insights into trends and seasonality. This trade-off between accuracy and interpretability is an important consideration for decision-makers and stakeholders.

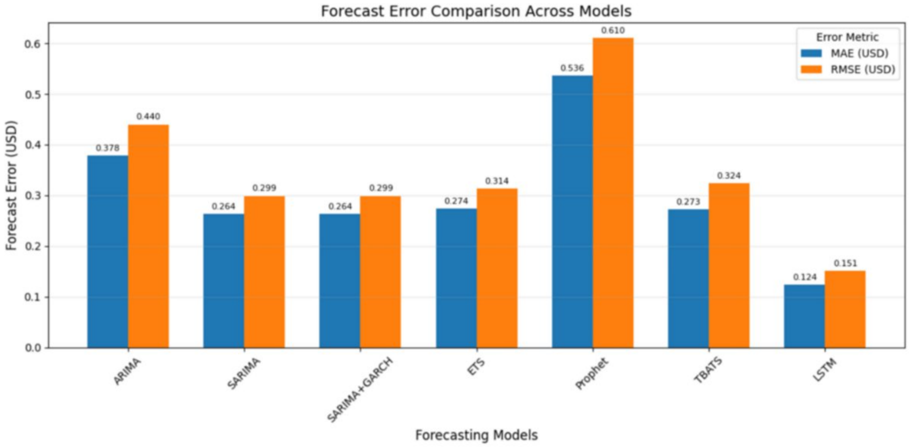

FIGURE 4: Model performance comparison based on forecast errors.

ARIMA, AutoRegressive Integrated Moving Average; SARIMA, Seasonal AutoRegressive Integrated Moving Average; ETS, Error, Trend, Seasonality; TBATS, Trigonometric, Box-Cox, ARMA errors, Trend, and Seasonal components; LSTM, Long Short-Term Memory; GARCH, Generalized AutoRegressive Conditional Heteroskedasticity; MAE, Mean Absolute Error; RMSE, Root Mean Squared Error

The comparative evaluation of model performance is presented in Figure 4, which summarizes forecasting errors across all models using MAE and RMSE. Among the seven models tested, the LSTM network achieved the lowest error values, followed by the SARIMA-GARCH and SARIMA models. The remaining models ETS, TBATS, ARIMA, and Prophet showed progressively higher error levels, with Prophet exhibiting the least accurate results overall.

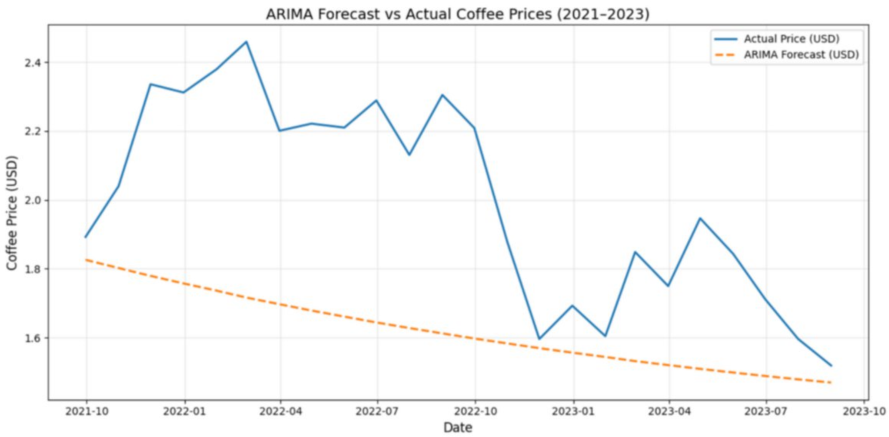

FIGURE 5: Forecasting performance of ARIMA.

ARIMA, AutoRegressive Integrated Moving Average

Figure 5 illustrates the ARIMA forecast compared with actual coffee prices. The model follows a general downward trend but significantly underestimates the observed fluctuations. This suggests that ARIMA struggles with capturing non-linear movements and seasonality in volatile agricultural markets.

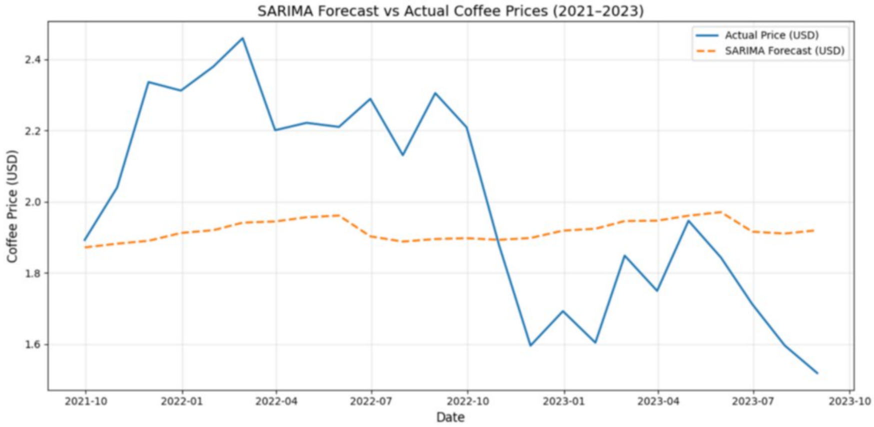

FIGURE 6: Forecasting performance of SARIMA.

SARIMA, Seasonal AutoRegressive Integrated Moving Average

Figure 6 presents the SARIMA model output. Compared to ARIMA, SARIMA incorporates seasonality and performs slightly better, capturing a more stable trend. However, its forecast line still remains too flat, failing to reflect sharp changes in prices, particularly those seen post-2022.

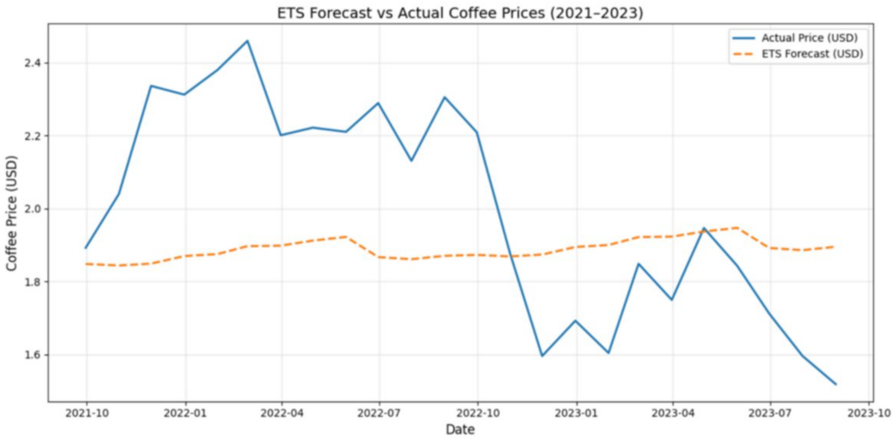

FIGURE 7: Forecasting performance of ETS.

ETS, Error, Trend, Seasonality

Figure 7 shows the ETS model, which also applies exponential smoothing and considers trend and seasonality. The ETS forecast follows a similar pattern to SARIMA but is marginally more responsive to mid-term price changes. Nonetheless, it underperforms during rapid fluctuations, such as the price drop around early 2023.

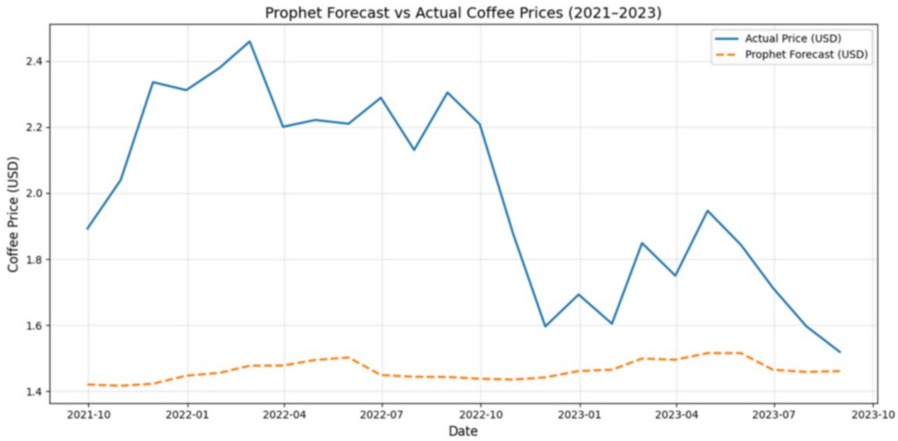

FIGURE 8: Forecasting performance of Prophet.

Figure 8 displays Prophet's forecast. Despite Prophet's strong performance in structured business datasets, here it shows the weakest alignment with actual values. Its forecast remains nearly flat and significantly underestimates both the trend and volatility, especially during periods of steep price rises or drops. This confirms previous findings by De Livera et al. [14], who noted Prophet's limited responsiveness in volatile, non-business data.

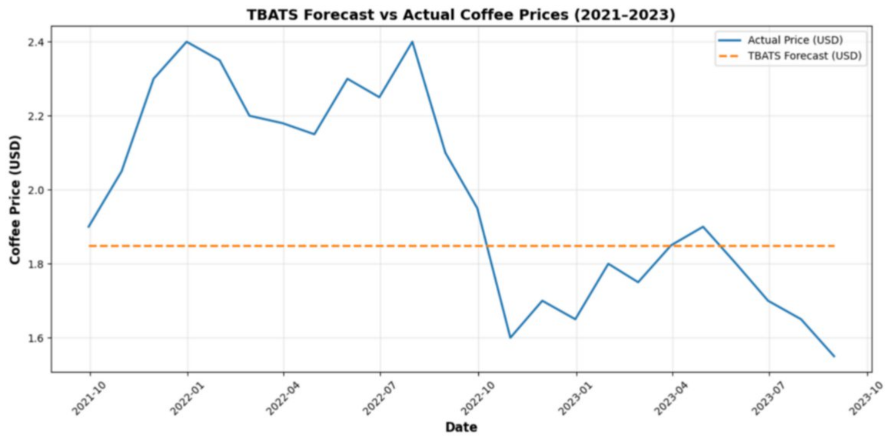

FIGURE 9: Forecasting performance of TBATS.

TBATS, Trigonometric, Box-Cox, ARMA errors, Trend, and Seasonal components

Figure 9 demonstrated the TBATS model results. While TBATS is designed to handle complex seasonality, it fails to adapt to abrupt shifts in the coffee price series in this case. The forecast line is nearly constant, implying an over-smoothed prediction that does not reflect the real-world volatility in the test period.

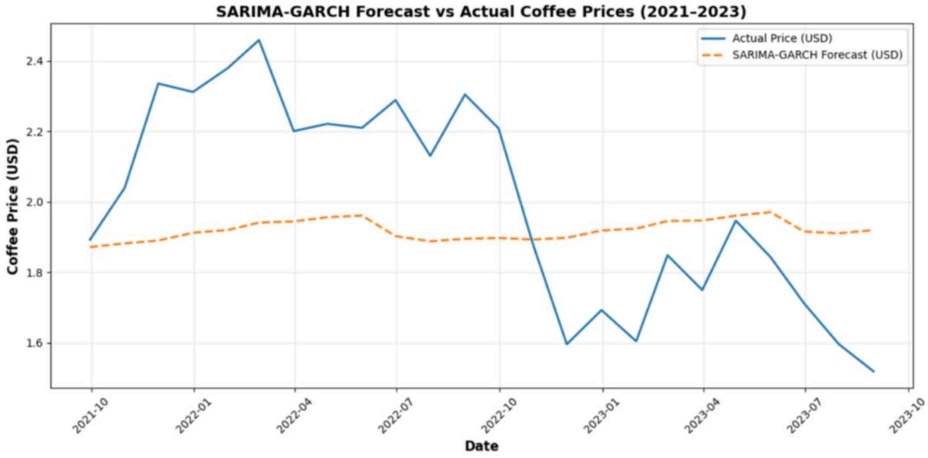

FIGURE 10: Forecasting performance of SARIMA-GARCH.

SARIMA, Seasonal AutoRegressive Integrated Moving Average; GARCH, AutoRegressive Conditional Heteroskedasticity

Figure 10 presents the forecast from the SARIMA-GARCH model. This hybrid approach, by integrating seasonality with volatility adjustments, offers better performance compared to the basic SARIMA model. The addition of the GARCH structure improves the model's ability to account for changes in price variability over time. Despite this improvement, the model still struggles to fully capture sudden spikes or rapid market shifts, often producing a smoother trajectory than the actual observed data.

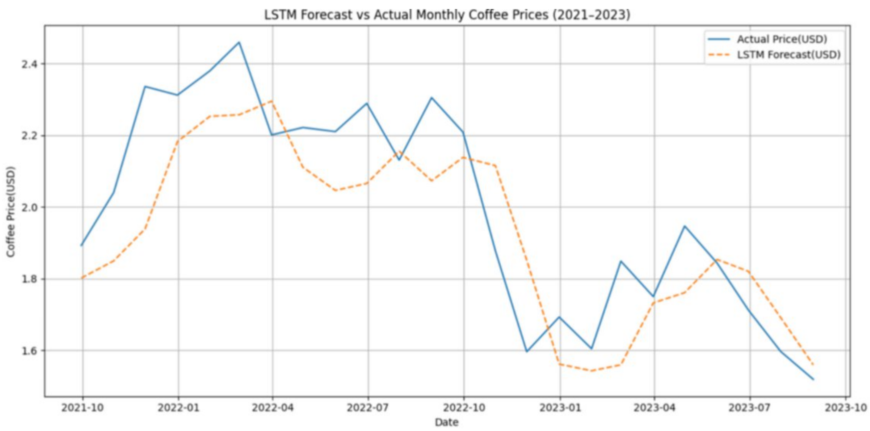

FIGURE 11: Forecasting performance of LSTM.

LSTM, Long Short-Term Memory

Figure 11 displays the forecast generated by the LSTM model. Compared to all other methods, LSTM gains the closest match to the real coffee price movements. It accurately reflects both the sharp rises and steep declines during the forecast period, maintaining strong agreement with the actual data across the 24 months. This result shows clearly that LSTM's advantage in capturing complex patterns and non-linear relationships, a strength noted previously as it relates to financial and commodity temporal data forecasting.

The analysis of Figure 5 through Figure 11 highlights the relative strengths and weaknesses of each model. Classical statistical methods such as ARIMA and SARIMA are useful for identifying general trends but fail to respond well to irregular or volatile shifts. Hybrid models like SARIMA-GARCH provide moderate improvements through volatility modeling, while deep learning approaches particularly LSTM offer a robust framework for forecasting complex, non-linear time series such as international coffee prices.

Conclusions

This work examined the forecasting of international coffee prices by comparing traditional time series models, hybrid approaches, and modern deep learning techniques. Models such as ARIMA, SARIMA, ETS, TBATS, Prophet, SARIMA-GARCH, and LSTM networks were applied to monthly coffee price data spanning several decades. Forecasting accuracy was evaluated using MAE and RMSE over a two-year test horizon. The results showed that while classical models like ARIMA and SARIMA performed reasonably well in capturing general trends and seasonal patterns, they were less effective in modeling the sharp price swings observed in coffee markets. The SARIMA-GARCH hybrid improved volatility handling to some extent but still struggled with abrupt shifts. Deep learning approaches, particularly the LSTM model, consistently outperformed traditional methods, achieving the lowest forecasting errors and better tracking the nonlinear and irregular behavior of coffee prices. Models like Prophet and TBATS, although efficient in stable, seasonally driven contexts, proved less adaptable to the unpredictable nature of commodity prices. The current forecasts are based solely on historical price data and do not incorporate external drivers such as weather anomalies, geopolitical events, or trade policies, all of which significantly impact coffee price volatility. Future studies will consider these factors to enhance predictive reliability. Additionally, statistical significance tests such as the Diebold-Mariano test will be incorporated in future work to verify whether performance improvements, especially by LSTM, are statistically significant. Future work may also focus on expanding model inputs to include external economic and climate variables, developing ensemble models that combine statistical and machine learning approaches, and applying more advanced deep learning architectures to further improve forecasting performance in highly volatile markets.

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: Christina EA, Hemalatha N, Sathish Kumar Narayanan

Acquisition, analysis, or interpretation of data: Christina EA

Drafting of the manuscript: Christina EA

Critical review of the manuscript for important intellectual content: Christina EA, Hemalatha N, Sathish Kumar Narayanan

Supervision: Hemalatha N, Sathish Kumar Narayanan

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.

References

1. Dicum G, Luttinger N: The Coffee Book: Anatomy of an Industry from Crop to the Last Drop. The New Press, New York, NY; 2006.
2. Barrett CB: Measuring food insecurity. Science. 2010, 327:825-828. 10.1126/science.1182768
3. Laderach P, Lundy M, Jarvis A, Ramirez J, Perez Portilla E, Schepp K, Eitzinger A: Predicted impact of climate change on coffee supply chains. The Economic, Social and Political Elements of Climate Change. Leal Filho W (ed): Springer, Berlin, Heidelberg; 2011. 703-723. 10.1007/978-3-642-14776-0_42
4. Coffee market report. (2023). Accessed: August 21, 2025: https://www.ico.org.
5. Global Grain and Coffee Price History (1973-2023). Kaggle. https://www.kaggle.com/datasets/mabdullahsajid/global-grain-and-coffee-price-history-1973-2023.
6. Box GEP, Jenkins GM, Reinsel GC, Ljung GM: Time Series Analysis: Forecasting and Control. Wiley, Hoboken, NJ; 2015.
7. Bollerslev T: Generalized autoregressive conditional heteroskedasticity. Journal of Econometrics. 1986, 31:307-327. 10.1016/0304-4076(86)90063-1
8. Ghiyal P, Kumar J: Use of hybrid SARIMA-GARCH model for predicting the prices of agricultural product in Haryana. International Journal of Statistics and Applied Mathematics. 2024, 9:101-107.
9. Hochreiter S, Schmidhuber J: Long short-term memory. Neural Computation. 1997, 9:1735-1780. 10.1162/neco.1997.9.8.1735
10. Fischer T, Krauss C: Deep learning with long short-term memory networks for financial market predictions. European Journal of Operational Research. 2018, 270:654-669. 10.1016/j.ejor.2017.11.054

11. Siami-Namini S, Tavakoli N, Namin AS: A comparison of ARIMA and LSTM in forecasting time series. 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA), Orlando, FL, USA. 2018, 1394-1401. 10.1109/ICMLA.2018.00227
12. Taylor SJ, Letham B: Forecasting at scale. The American Statistician. 2018, 72:37-45. 10.1080/00031305.2017.1380080
13. Hyndman R, Koehler A, Ord K, Snyder R: Forecasting with Exponential Smoothing: The State Space Approach. Springer, Berlin, Heidelberg; 2008. 10.1007/978-3-540-71918-2
14. De Livera AM, Hyndman RJ, Snyder RD: Forecasting time series with complex seasonal patterns using exponential smoothing. Journal of the American Statistical Association. 2011, 106:1513-1527. 10.1198/jasa.2011.tm09771