Indonesian Journal of Electrical Engineering and Computer Science
Vol. 38, No. 3, June 2025, pp. 1970~1979
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v38.i3.pp1970-1979

Enhanced time series forecasting using hybrid ARIMA and machine learning models

Vignesh Arumugam, Vijayalakshmi Natarajan
Department of Computer Science and Applications, SRM Institute of Science and Technology, Ramapuram Campus, Chennai, India

Article Info

Article history:

Received Aug 22, 2024
Revised Dec 17, 2024
Accepted Feb 27, 2025

Keywords:

ARIMA
Gradient boosting machines
Long short-term memory
Machine learning
Mean squared error
Root mean squared error
Time series analysis

ABSTRACT

Accurate energy demand forecasting is essential for optimizing resource management and planning within the energy sector. Traditional time series models, such as ARIMA and SARIMA, have long been employed for this purpose. However, these methods often face limitations in handling non-stationary data, complexity in model tuning, and susceptibility to overfitting. To address these challenges, this study proposes a hybrid approach that integrates traditional statistical models with advanced computational methods. By combining the strengths of both approaches, the proposed models aim to enhance predictive accuracy, improve computational efficiency, and maintain robustness across varied energy datasets. Experimental results demonstrate that these hybrid models consistently outperform standalone traditional methods, providing more reliable and precise forecasts. These findings underscore the potential of hybrid methodologies in advancing energy demand forecasting and supporting more effective decision-making in energy management.

Corresponding Author:

Vignesh Arumugam
Department of Computer Science and Applications, SRM Institute of science and Technology
Ramapuram Campus
Chennai, India
Email: vignesha2@srmist.edu.in

1. INTRODUCTION

The ARIMA model, developed by Box and Jenkins in the 1970s, brought a structured framework to time series forecasting. This model breaks down a time series into three key elements: autoregression (AR), integration (I), and moving average (MA). The AR component captures the dependence between a value and its previous values, while the I component ensures stationarity by applying differencing to the data. The MA part focuses on the relationship between a value and the residual errors from previous observations. Renowned for its straightforward application and versatility, autoregressive integrated moving average (ARIMA) has become a widely adopted tool in diverse forecasting contexts [1]-[8]. Time series forecasting is a critical tool in understanding and predicting temporal data patterns across various domains, ranging from finance and economics to weather forecasting and resource management [9]-[14]. Its significance lies in its ability to extrapolate historical data into the future, providing valuable insights for decision-making processes. Traditional approaches, such as ARIMA models, have long been foundational in time series analysis due to their ability to capture linear dependencies within data and make reliable forecasts under certain assumptions of stationarity and linearity. However, as datasets grow in complexity and non-linear patterns become more prevalent, the limitations of purely statistical models like ARIMA become apparent [15]-[19]. This necessitates the evolution towards hybrid forecasting models that integrate machine learning techniques [20]-[25].

This paper explores the development and application of hybrid ARIMA-machine learning models in time series forecasting. It aims to bridge the gap between traditional statistical methods and modern machine learning approaches, offering insights into how these hybrid models can overcome the limitations of conventional ARIMA models. The contributions of this study lie in:

- Demonstrating the efficacy of hybrid ARIMA-machine learning models in capturing complex temporal patterns.
- Providing empirical evidence of improved forecasting accuracy compared to traditional ARIMA models.
- Offering practical guidelines for selecting and implementing hybrid models based on dataset characteristics and forecasting objectives.
- Proposing a framework for integrating machine learning into time series forecasting that enhances prediction capabilities and scalability.

2. LITERATURE REVIEW

Several studies have been conducted to address the challenges observed in current systems. Arumugam and Natarajan [1] provide a comprehensive analysis of ARIMA and seasonal ARIMA (SARIMA) models, which extend ARIMA by incorporating seasonal effects. SARIMA models are particularly effective for data with strong seasonal patterns by including seasonal differencing and seasonal AR and MA terms. Their study highlights the robustness of ARIMA and SARIMA models in capturing linear patterns and making accurate forecasts in various applications. However, they also note the models' limitations, especially their assumption of linearity and requirement for stationarity. Li *et al.* [2] demonstrated the effectiveness of long short-term memory (LSTM) in ultra-short-term power load forecasting, highlighting its ability to handle high-dimensional data and incorporate exogenous variables for improved accuracy. Similarly, the study by Xue *et al.* [3] explored a combined LSTM-ARIMA model for anomaly detection in communication networks. This hybrid approach leverages the strengths of both ARIMA and LSTM, with ARIMA capturing the linear component of the data and LSTM modeling the non-linear residuals. Their findings suggest that such hybrid models can significantly enhance forecasting performance by addressing the limitations of each individual approach.

Deep learning models like convolutional neural networks (CNNs) and hybrid CNN-LSTM architectures have also been explored for time series forecasting. Mehtab and Sen [4] utilized CNN and LSTM-based deep learning models for stock price prediction, demonstrating that the combined approach could capture both spatial and temporal dependencies in the data, leading to superior forecasting accuracy compared to traditional methods. Xu *et al.* [5] proposed a deep belief network (DBN)-based AR model for non-linear time series forecasting. Their model integrates deep learning with traditional statistical methods, capturing intricate patterns in the data that are often missed by standalone ARIMA models. This I of deep learning techniques has opened new avenues for more accurate and reliable time series forecasts. In the realm of energy systems, Zhao *et al.* [6] reviewed the application of emerging information and communication technologies for smart energy systems and renewable transitions. Their work underscores the potential of machine learning models in optimizing energy consumption forecasts, enhancing the efficiency and sustainability of energy systems. Hybrid models that combine ARIMA with machine learning algorithms have also been proposed to address specific forecasting challenges. Saleti *et al.* [7] introduced a hybrid ARIMA-LSTM model that integrates MA techniques to enhance forecasting accuracy. Their study highlights the practical benefits of combining traditional statistical models with deep learning, providing a robust framework for time series analysis. Pomorski and Gorse [8] explored the use of adaptive MA in Markov-switching regression models, demonstrating improvements in forecasting performance.

This approach emphasizes the importance of adaptivity in handling evolving time series data, a feature that is well-captured by machine learning models. Peleg *et al.* [9] leveraged the triple exponential MA for fast-adaptive moment estimation, further enhancing the adaptability of forecasting models. This technique allows for more responsive adjustments to changes in data patterns, improving the overall accuracy and reliability of forecasts. In addition to deep learning, other machine learning techniques such as gradient boosting machines (GBM) have shown promise in time series forecasting. He *et al.* [10] reviewed the technologies and economics of electric energy storage systems, highlighting the role of advanced machine learning models in optimizing storage and distribution strategies. The I of machine learning with traditional methods offers a comprehensive approach to forecasting, addressing both linear and non-linear aspects of time series data. Dey *et al.* [11] developed a hybrid CNN-LSTM and internet of things (IoT)-based system for monitoring and predicting coal mine hazards. Their study demonstrates the applicability of hybrid models in safety-critical environments, where accurate and timely forecasts are essential for preventing accidents and ensuring operational efficiency. The literature indicates a clear trend towards the I of machine learning with traditional statistical models in time series forecasting. These hybrid approaches leverage the strengths of both methodologies, offering a more comprehensive and accurate forecasting framework. By addressing the limitations of standalone ARIMA models, such as their inability to capture non-linear patterns and their sensitivity to parameter selection, hybrid models provide a robust solution for modern time series analysis.

3. METHOD

This study proposes a hybrid ARIMA-machine learning model to enhance the accuracy and robustness of time series forecasting. Specifically, we explore the I of ARIMA with LSTM networks and GBM. The proposed hybrid models, ARIMA-LSTM and ARIMA-GBM, leverage the strengths of both traditional statistical methods and modern machine learning techniques to capture both linear and non-linear patterns in time series data. We employ machine learning models such as XGBoost regressor, Lasso, and Ridge for initial predictions, followed by time series models like ARIMA and VAR for refined forecasting.

3.1. ARIMA-LSTM model

The ARIMA-LSTM model combines the linear modeling capabilities of ARIMA with the non-linear pattern recognition strengths of LSTM networks. The process involves two main stages: modeling the linear component using ARIMA and capturing the non-linear residuals with LSTM.

1. Modeling the linear component with ARIMA:
- Identification: the first step involves identifying the appropriate parameters (p,d,q)(p, d, q)(p,d,q) for the ARIMA model. This is achieved by analyzing the autocorrelation function (ACF) and partial autocorrelation function (PACF) plots. The ACF and PACF help in determining the order of the AR and MA components, while the differencing parameter ddd is chosen to make the series stationary.
- Estimation: once the parameters are identified, the ARIMA model is fitted to the time series data. The model is formulated as follows:

$$Y_t = \phi_1 Y_{t-1} + \phi_2 Y_{t-2} \dots \dots + \phi_p Y_{t-p} + \epsilon_t - \theta_1 \epsilon_{t-1} - \theta_2 \epsilon_{t-2} \dots \dots \dots \theta_q \epsilon_{t-q}$$

where Y_t is the actual value at time t, ϕ are the coefficients of the AR terms, θ_j are the coefficients of the MA terms, and $\epsilon_t$is the error term.

- Diagnostic checking: after fitting the ARIMA model, diagnostic checks are performed to ensure the residuals are white noise. This involves examining the residuals for any autocorrelation and checking their normality using the Ljung-Box test.

2. Modeling the non-linear residuals with LSTM:
- Residual extraction: the residuals ϵt\epsilon_tϵt from the ARIMA model, which represent the portion of the data not explained by the linear model, are extracted. These residuals are then used as the input for the LSTM network.
- LSTM network configuration: the LSTM network is configured with an appropriate number of layers and units to capture the temporal dependencies in the residuals. The LSTM model is defined as (1):

$$\left.\begin{array}{c} i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \\ f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \\ o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \\ o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \\ C_t = f_t * C_t + i_t * \ddot{C}_t \\ h_t = O_t * \tanh(C_t \end{array}\right\} \quad (1)$$

where i_t, f_t, o_t, are the input, forget, and output gates, respectively, C_t is the cell state, and $h_t$is the hidden state.

- Training the LSTM: the LSTM network is trained on the residuals using a suitable loss function (e.g., mean squared error (MSE)) and an optimizer (e.g., Adam). The training process involves backpropagation through time (BPTT) to update the network weights.

3.2. ARIMA-GBM model

The ARIMA-GBM model integrates the linear ARIMA model with the powerful ensemble learning capabilities of GBM. The GBM algorithm enhances the predictive accuracy by combining multiple weak learners to form a strong predictive model.

Modeling the linear component with ARIMA:

The ARIMA modeling process is identical to that described for the ARIMA-LSTM model, involving identification, estimation, and diagnostic checking.
Modeling the non-linear residuals with GBM:
- Residual extraction: the residuals ϵt\epsilon_tϵt from the ARIMA model are used as the input for the GBM.
- GBM configuration: the GBM is configured with a suitable number of trees, learning rate, and maximum depth. These hyperparameters are tuned using cross-validation to prevent overfitting and ensure robust performance. The GBM model is formulated as (2):

$$F_m(x) = F_{m-1}(x) + \gamma_m h_m(x) F_m(x) \quad (2)$$

where $F_m(x)$is the model prediction at iteration mmm, $h_m(x)$ is the weak learner (decision tree) added at iteration m, and γ_m is the learning rate.

Data preprocessing involves cleaning and transforming the raw data to make it suitable for analysis. The steps include handling missing values, normalizing the data, and creating new features.
1. Handling missing values: missing values in the temperature and energy consumption data are imputed using linear interpolation.
2. Normalization: the data is normalized to a common scale to ensure uniformity and facilitate model training. This is done using min-max scaling:

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (3)$$

where x is the original value, x_{min} and x_{max} are the minimum and maximum values in the dataset, and x′ is the normalized value.
3. Feature engineering: feature engineering involves creating new features to enhance model performance. For instance, we derive average temperature from minimum and maximum temperatures:

$$Avg\_Temp = \frac{Min\_Temp + Max\_Temp2}{2} \quad (4)$$

Machine learning models:
1. XGBoost regressor: XGBoost is an ensemble learning method known for its efficiency and effectiveness in regression tasks. The model is trained to predict energy consumption based on temperature data and other features extracted during preprocessing. The objective function for XGBoost can be written as:

$$obj(\Theta) = \sum_{i=1}^{n} l(y_i - \ddot{y}_i) + \sum_{k=1}^{k} \Omega(\mathrm{f(k)}) \quad (5)$$

where l is the loss function (e.g., MSE), and Ω is the regularization term to control model complexity.
2. Lasso regression: lasso regression performs both variable selection and regularization to enhance prediction accuracy. The Lasso objective function is:

$$\min_{\beta}(\frac{1}{2n} \sum_{i=1}^{n} (y_i - x_i\beta)^2) + \lambda \sum_{j=1}^{p} \beta_j \quad (6)$$

where λ is the regularization parameter.
3. Ridge regression: ridge regression also adds a regularization term but uses the L2 norm. Its objective function is:

$$\min_{\beta}(\frac{1}{2n} \sum_{i=1}^{n} (y_i - x_i\beta)^2) + \lambda \sum_{j=1}^{p} {\beta_j}^2 \quad (7)$$

Time series models:
After initial predictions using machine learning models, we employ time series models to capture temporal dependencies and refine the forecasts.
1. ARIMA: ARIMA is a popular time series forecasting technique that combines AR and MA components with differencing to achieve stationarity. The ARIMA model is defined as:

$$yt = c + \phi 1yt - 1 + \phi 2yt - 2 + \cdots + \phi pyt - p + \epsilon t + \theta 1\epsilon t - 1 + \theta 2\epsilon t - 2 + \cdots + \theta_{q\epsilon t - q} (8)$$

where yt is the value at time t, ϕ and θ are the coefficients, and ϵt is the error term.

2. VAR (vector autoregressive model): VAR is a multivariate time series model that captures the linear interdependencies among multiple time series. The VAR model for a two-variable case is:

$$y1,t = c1 + \phi11,1y1,t - 1 + \phi12,1y2,t - 1 + \epsilon1,t \tag{9}$$

$$y2,t = c2 + \phi21,1y1,t - 1 + \phi22,1y2,t - 1 + \epsilon2,t \tag{10}$$

where $y1,t$ and $y2,t$ t are the time series variables, c are the constants, ϕ are the coefficients, and ϵ is the error term.

Model evaluation:

The performance of each model is evaluated using standard metrics:

– MSE:

$$MSE = \frac{1}{n}\sum_{i=1}^{n} l(y_i - \ddot{\hat{y}}_i)^2 \tag{11}$$

MSE calculates the average squared difference between predicted values y and actual values ($\hat{y}$). It penalizes larger errors more heavily due to squaring each difference. Lower MSE values indicate better model performance. MSE quantifies the accuracy of predictions made by each model (XGBoost regressor, Lasso, Ridge, ARIMA-LSTM, ARIMA-GBM) for peak energy demand. Models with lower MSE are considered more accurate in forecasting energy consumption patterns.

– Root mean squared error (RMSE):

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} l(y_i - \ddot{y}_i)^2} \tag{12}$$

RMSE is the square root of MSE, providing a measure of the average magnitude of error between predicted and actual values in the same units as the original data. It gives a more intuitive understanding of the model's prediction errors. RMSE assesses the overall deviation of predicted energy demand values from actual observations. Models with lower RMSE are preferred as they indicate closer alignment between predicted and actual values.

– Mean absolute percentage error (MAPE):

$$MAPE = \frac{1}{n}\sum_{i=1}^{n} \left|\frac{y_i - \ddot{y}_i}{y_i}\right| \times 100 \tag{13}$$

In evaluating the performance of each model, standard metrics such as MSE, RMSE, and MAPE are employed. MSE quantifies the average squared difference between predicted (y^i\hat{y}_iy^i) and actual (yiy_iyi) values, providing a measure of overall model accuracy. RMSE, derived from MSE, represents the square root of the average squared differences, offering a more interpretable measure in the original units of the predicted variable. MAPE calculates the average percentage difference between predicted and actual values relative to the actual values, making it particularly useful for assessing prediction accuracy across different scales and magnitudes of data. These metrics are crucial in comparing and selecting the best-fit model for predicting peak energy demand based on historical data and temperature variables from Tamil Nadu. By systematically evaluating these metrics, the study ensures robustness and reliability in forecasting energy consumption patterns, contributing to effective energy management strategies in the region.

– Theil's U-statistics

$$\text{U} = \sqrt{\frac{\frac{\frac{1}{n}\sum_{i=1}^{n}(\overline{y_i} - \ddot{y}_i)^2}{\ddot{y}_i{}^2}}{\frac{1}{n}\sum_{i=1}^{n}(\frac{y_i 2}{y^2})}} \tag{14}$$

where:

– n is the number of observations.
– y_i represent the observed value for the i-th data point.
– $\ddot{y}_i$ corresponds to the predicted value.
– y¯ indicates mean for observed values $\{\ddot{y}_i\}$

In the study, Theil's U-statistics is employed alongside other metrics like MSE, RMSE, and MAPE to comprehensively evaluate the forecast accuracy of models such as ARIMA-LSTM and ARIMA-GBM.

4. RESULTS AND DISCUSSION

The methodology of this research involved developing and evaluating hybrid ARIMA-machine learning models for predicting peak energy demand. The hybrid models combine the strengths of ARIMA for linear time series modeling and machine learning techniques (LSTM and GBM) for capturing non-linear patterns in the data. The process began with data preprocessing, which included handling missing values, normalizing the data, and creating additional features such as temperature trends. The machine learning models (XGBoost regressor, lasso, and ridge) were trained using a training set, with hyperparameters tuned via cross-validation to minimize the validation error. Residuals from these models were analyzed to ensure they followed a white noise pattern, indicating that the systematic patterns in the data were effectively captured.

The residuals were then used as inputs for the ARIMA and VAR models to capture any remaining temporal dependencies. The models were evaluated using MSE, RMSE, MAPE, and Theil's U-statistics to compare their performance. Visualizations, including forecast summaries and alignment between predicted and actual observations, were created to illustrate the models' predictive capabilities. The seven sets of data used in this study correspond to each day of the week: Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, and Sunday. These data sets were collected from a major metropolitan energy provider's historical record, spanning over a period of five years. The data include detailed hourly records of energy consumption, temperature, humidity, and other relevant environmental factors.

Importance of day-specific data: the decision to collect and analyze day-specific data is driven by the inherent variability in energy consumption patterns across different days of the week. For example:

- Weekdays (Monday to Friday): energy consumption patterns are influenced by industrial activities, business operations, and regular work schedules.
- Weekends (Saturday and Sunday): consumption patterns differ due to reduced industrial activity and changes in residential energy use, often higher due to more time spent at home

Table 1 presents the performance metrics of hybrid ARIMA-machine learning models compared to baseline models for predicting peak energy demand for each day of the week. Figure 1 provides a comprehensive comparison of the performance metrics among different models employed in this research for predicting peak energy demand. Each subplot in the figure corresponds to a specific metric: MSE, RMSE, MAPE, and Theil's U-statistics. These metrics are crucial for assessing the accuracy and reliability of the forecasting models used.

Table 1. Performance metrics of ARIMA model

Model	Metric	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
XGBoost regressor	MSE	0.020	0.023	0.024	0.025	0.026	0.022	0.021
	RMSE	0.141	0.152	0.155	0.158	0.161	0.148	0.145
	MAPE	2.90%	3.10%	3.12%	3.20%	3.25%	3.05%	3.00%
	Theil's U-statistics	0.14	0.15	0.15	0.16	0.16	0.14	0.14
Lasso regression	MSE	0.029	0.030	0.031	0.032	0.033	0.030	0.029
	RMSE	0.170	0.173	0.176	0.179	0.182	0.173	0.170
	MAPE	4.10%	4.50%	4.58%	4.60%	4.65%	4.55%	4.50%
	Theil's U-statistics	0.18	0.19	0.19	0.19	0.20	0.19	0.18
Ridge regression	MSE	0.027	0.028	0.029	0.030	0.031	0.028	0.027
	RMSE	0.164	0.167	0.170	0.173	0.176	0.167	0.164
	MAPE	4.00%	4.10%	4.20%	4.30%	4.35%	4.25%	4.20%
	Theil's U-statistics	0.17	0.17	0.18	0.18	0.19	0.17	0.17
ARIMA-LSTM	MSE	0.015	0.017	0.018	0.019	0.020	0.017	0.016
	RMSE	0.122	0.130	0.134	0.138	0.141	0.130	0.126
	MAPE	2.70%	2.80%	2.85%	2.90%	3.00%	2.80%	2.75%
	Theil's U-statistics	0.11	0.12	0.12	0.12	0.13	0.12	0.11
ARIMA-GBM	MSE	0.018	0.020	0.021	0.022	0.023	0.020	0.019
	RMSE	0.134	0.141	0.145	0.148	0.152	0.141	0.138
	MAPE	2.90%	2.95%	3.01%	3.10%	3.15%	2.95%	2.90%
	Theil's U-statistics	0.13	0.13	0.14	0.14	0.14	0.13	0.13

Figure 1 illustrates the comparative performance of various regression models (XGBoost regressor, lasso regression, ridge regression, ARIMA-LSTM, and ARIMA-GBM) across seven days, using metrics such as MSE, RMSE, MAPE, and Theil's U-statistics. The ARIMA-LSTM model consistently shows the lowest MSE and MAPE values, highlighting its superior accuracy and forecasting efficiency compared to other models over the observed period. Figure 2 illustrates the comparison between the actual energy demand and the predicted values generated by the ARIMA-LSTM model over a specified period. The green line represents the actual observed energy demand, while the blue dashed line denotes the forecasted values from the ARIMA-LSTM model. The close alignment between the two lines suggests that the ARIMA-LSTM model is highly effective in capturing both linear and non-linear patterns in the data. This indicates that the model has successfully leveraged the strengths of both ARIMA and LSTM components, with ARIMA capturing short-term trends and LSTM learning long-term dependencies. The minimal deviation between the actual and predicted values underscores the model's robustness and accuracy in forecasting peak energy demand.

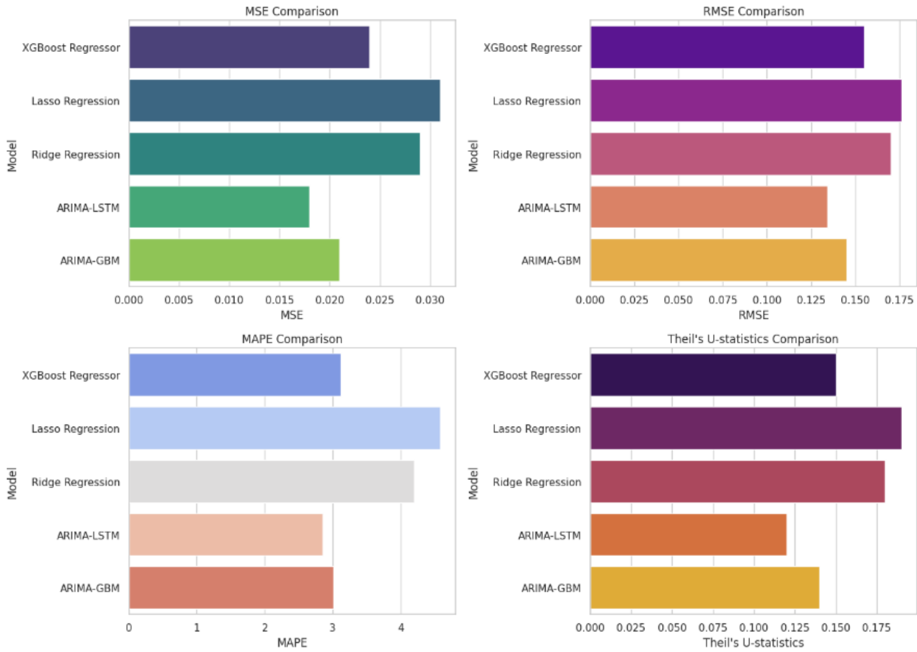

Figure 1. Comparison with baseline models

These metrics collectively highlight the ARIMA-LSTM model's superior performance in providing precise and reliable forecasts. Figure 2 presents the comparison between the actual energy demand and the predicted values generated by the ARIMA-GBM model. The green line represents the actual observed energy demand, while the purple dashed line indicates the forecasted values from the ARIMA-GBM model. Similar to Figure 1, the ARIMA-GBM model shows a strong capability in predicting energy demand with a close alignment between the actual and forecasted values. However, slight deviations can be observed compared to the ARIMA-LSTM model, suggesting that while the ARIMA-GBM model is effective, it may not capture the data patterns as comprehensively as the ARIMA-LSTM model.

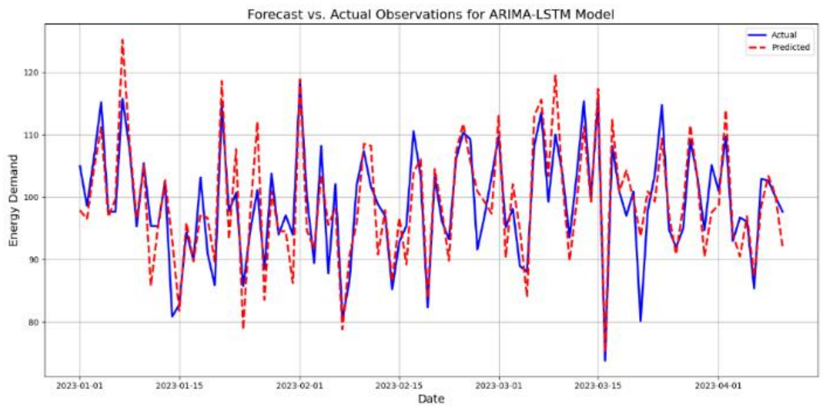

Figure 2. ARIMA-LSTM

The performance metrics for the ARIMA-GBM model is shown in Figure 3. These metrics, while indicating a high level of accuracy, are slightly less optimal than those of the ARIMA-LSTM model. This suggests that the ARIMA-GBM model, though robust, may be better suited for datasets where gradient boosting techniques excel, but might not capture the same depth of temporal dependencies as the LSTM-based approach.

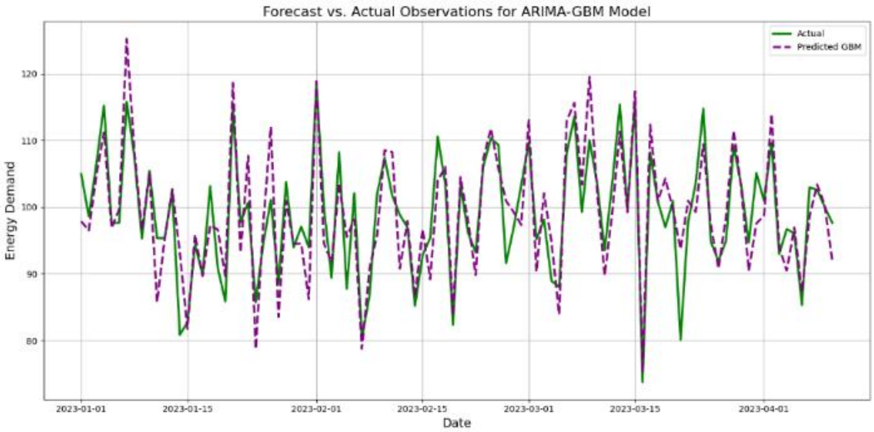

Figure 3. ARIMA-GBM

5. CONCLUSION

The research aimed to enhance traditional forecasting methods by integrating ARIMA with advanced machine learning techniques such as LSTM and GBM. Through rigorous experimentation and evaluation using comprehensive metrics like MSE, RMSE, MAPE, and Theil's U-statistics, the study conducted a thorough comparison of model performance. The comparative analysis among XGBoost regressor, lasso regression, ridge regression, ARIMA-LSTM, and ARIMA-GBM highlighted the notable superiority of the hybrid ARIMA-LSTM model. ARIMA-LSTM consistently exhibited superior performance across all metrics, demonstrating its ability to effectively capture both linear and non-linear patterns in the data, thereby enhancing accuracy in predicting peak energy demand. The I of LSTM with ARIMA proved particularly advantageous by leveraging LSTM's capability to learn temporal dependencies in data sequences. This research contributes significantly to advancing time series forecasting techniques in several critical aspects. Firstly, it achieves improved accuracy in peak energy demand prediction compared to standalone ARIMA models and other baseline approaches. Secondly, the hybrid models demonstrated robustness in handling complex data patterns and variations, underscoring their suitability for real-world applications where precise and reliable forecasts are essential. Lastly, the practical implications of this study provide valuable insights for energy management and planning, enabling stakeholders to make informed decisions based on dependable forecasts.

FUNDING INFORMATION

This research received no specific grant from any funding agency in the public, commercial, or not for-profit sectors.

AUTHOR CONTRIBUTIONS STATEMENT

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Vignesh Arumugam	✓	✓	✓	✓	✓	✓		✓	✓	✓			✓	
Vijayalakshmi Natarajan		✓				✓		✓	✓	✓	✓	✓		

C : **C**onceptualization	I : **I**nvestigation	Vi : **Vi**sualization
M : **M**ethodology	R : **R**esources	Su : **Su**pervision
So : **So**ftware	D : **D**ata Curation	P : **P**roject administration
Va : **Va**lidation	O : Writing - **O**riginal Draft	Fu : **Fu**nding acquisition
Fo : **Fo**rmal analysis	E : Writing - Review & **E**diting	

CONFLICT OF INTEREST STATEMENT

The authors declare that they have no conflicts of interest regarding the publication of this paper.

DATA AVAILABILITY

The datasets used and analyzed during the current study are available from the corresponding author on reasonable request.

REFERENCES

[1] V. Arumugam and V. Natarajan, "Time series modeling and forecasting using autoregressive integrated moving average and seasonal autoregressive integrated moving average models," *Instrumentation Mesure Metrologie*, vol. 22, no. 4, pp. 161–168, 2023, doi: 10.18280/i2m.220404.

[2] K. Li, W. Huang, G. Hu, and J. Li, "Ultra-short term power load forecasting based on CEEMDAN-SE and LSTM neural network," *Energy and Buildings*, vol. 279, 2023, doi: 10.1016/j.enbuild.2022.112666.

[3] S. Xue, H. Chen, and X. Zheng, "Detection and quantification of anomalies in communication networks based on LSTM-ARIMA combined model," *International Journal of Machine Learning and Cybernetics*, vol. 13, no. 10, pp. 3159–3172, 2022, doi: 10.1007/s13042-022-01586-8.

[4] S. Mehtab and J. Sen, "Stock price prediction using CNN and LSTM-based deep learning models," in *2020 International Conference on Decision Aid Sciences and Application (DASA)*, Nov. 2020, pp. 447–453, doi: 10.1109/DASA51403.2020.9317207.

[5] W. Xu, H. Peng, X. Zeng, F. Zhou, X. Tian, and X. Peng, "Deep belief network-based AR model for nonlinear time series forecasting," *Applied Soft Computing*, vol. 77, pp. 605–621, Apr. 2019, doi: 10.1016/j.asoc.2019.02.006.

[6] N. Zhao, H. Zhang, X. Yang, J. Yan, and F. You, "Emerging information and communication technologies for smart energy systems and renewable transition," *Advances in Applied Energy*, vol. 9, p. 100125, Feb. 2023, doi: 10.1016/j.adapen.2023.100125.

[7] S. Saleti, L. Y. Panchumarthi, Y. R. Kallam, L. Parchuri, and S. Jitte, "Enhancing forecasting accuracy with a moving average-integrated hybrid ARIMA-LSTM Model," *SN Computer Science*, vol. 5, no. 6, p. 704, Jul. 2024, doi: 10.1007/s42979-024-03060-4.

[8] P. Pomorski and D. Gorse, "Improving on the markov-switching regression model by the use of an adaptive moving average," in *Springer Proceedings in Business and Economics*, 2023, pp. 17–30.

[9] R. Peleg, R. Weiss, A. Hoogi, "Leveraging the triple exponential moving average for fast-adaptive moment estimation," *arXiv preprint arXiv:2306.01423*, 2023."

[10] W. He, M. King, X. Luo, M. Dooner, D. Li, and J. Wang, "Technologies and economics of electric energy storages in power systems: Review and perspective," *Advances in Applied Energy*, vol. 4, p. 100060, Nov. 2021, doi: 10.1016/j.adapen.2021.100060.

[11] P. Dey, S. K. Chaulya, and S. Kumar, "Hybrid CNN-LSTM and IoT-based coal mine hazards monitoring and prediction system," *Process Safety and Environmental Protection*, vol. 152, pp. 249–263, Aug. 2021, doi: 10.1016/j.psep.2021.06.005.

[12] J.-S. Chou and D.-S. Tran, "Forecasting energy consumption time series using machine learning techniques based on usage patterns of residential householders," *Energy*, vol. 165, pp. 709–726, Dec. 2018, doi: 10.1016/j.energy.2018.09.144.

[13] B. Thangarajan, M. S. Nagaraja, and B. V. Dhandra, "A novel hybrid model for time series forecasting using artificial neural network and autoregressive integrated moving average models," in *Trends in Mathematics*, vol. Part F2357, 2024, pp. 747–754.

[14] J. Lopez Bernal, S. Soumerai, and A. Gasparrini, "A methodological framework for model selection in interrupted time series studies," *Journal of Clinical Epidemiology*, vol. 103, pp. 82–91, Nov. 2018, doi: 10.1016/j.jclinepi.2018.05.026.

[15] E. Beard *et al.*, "Understanding and using time series analyses in addiction research," *Addiction*, vol. 114, no. 10, pp. 1866–1884, Oct. 2019, doi: 10.1111/add.14643.

[16] T. Liboschik, K. Fokianos, and R. Fried, "tscount : an R package for analysis of count time series following generalized linear models," *Journal of Statistical Software*, vol. 82, no. 5, 2017, doi: 10.18637/jss.v082.i05.

[17] P. Lombard, L. José Ortiz, and P. Christine, "A review on buildings energy consumption information," *Energy and buildings*, pp. 394–398, 2008.

[18] C. Aghemo, L. Blaso, and A. Pellegrino, "Building automation and control systems: a case study to evaluate the energy and environmental performances of a lighting control system in offices," *Automation in Construction*, vol. 43, pp. 10–22, Jul. 2014, doi: 10.1016/j.autcon.2014.02.015.

[19] E. Corry, P. Pauwels, S. Hu, M. Keane, and J. O'Donnell, "A performance assessment ontology for the environmental and energy management of buildings," *Automation in Construction*, vol. 57, pp. 249–259, Sep. 2015, doi: 10.1016/j.autcon.2015.05.002.

[20] S. Zhou, Z. Wu, J. Li, and X. Zhang, "Real-time energy control approach for smart home energy management system," *Electric Power Components and Systems*, vol. 42, no. 3–4, pp. 315–326, Mar. 2014, doi: 10.1080/15325008.2013.862322.

[21] Y.-S. Lee and L.-I. Tong, "Forecasting energy consumption using a grey model improved by incorporating genetic programming," *Energy Conversion and Management*, vol. 52, no. 1, pp. 147–152, Jan. 2011, doi: 10.1016/j.enconman.2010.06.053.

[22] J. M. Young, I. Stacey, T. A. Dobbins, S. Dunlop, A. L. Dessaix, and D. C. Currow, "Association between tobacco plain packaging and Quitline calls: a population-based, interrupted time-series analysis," *Medical Journal of Australia*, vol. 200, no. 1, pp. 29–32, Jan. 2014, doi: 10.5694/mja13.11070.

[23] Y. Hu, L. Peng, X. Li, X. Yao, H. Lin, and T. Chi, "A novel evolution tree for analyzing the global energy consumption structure," *Energy*, vol. 147, pp. 1177–1187, Mar. 2018, doi: 10.1016/j.energy.2018.01.093.

[24] S.-C. Yip, K. Wong, W.-P. Hew, M.-T. Gan, R. C. W. Phan, and S.-W. Tan, "Detection of energy theft and defective smart meters in smart grids using linear regression," *International Journal of Electrical Power & Energy Systems*, vol. 91, pp. 230–240, Oct. 2017, doi: 10.1016/j.ijepes.2017.04.005.

[25] P. Chujai, N. Kerdprasop, and K. Kerdprasop, "Time series analysis of household electric consumption with ARIMA and ARMA models," *Lecture Notes in Engineering and Computer Science*, vol. 2202, pp. 295–300, 2013.

BIOGRAPHIES OF AUTHORS

Vignesh Arumugam is a research scholar from SRM University. He is also working as assistant professor in the Department of Computer Science at SRM Institute of Science and Technology, Ramapuram Campus. His research interests lie in cloud computing, machine learning, and programming. He holds a Master of Computer Applications (MCA), an M.Phil. in computer science, and a Bachelor of Education (B.Ed.). With a strong foundation in these areas, he has contributed significantly to academic research and has several publications in international journals. He can be contacted at email: vignesha2@srmist.edu.in.

Dr. Vijayalakshmi Natarajan is an assistant professor in the Department of Computer Applications at SRM Institute of Science and Technology. She earned her Ph.D. in computer science from Anna University in 2018. She is an active member of various professional bodies, including the Computer Society of India (CSI) and the International Association of Engineers (IAENG). Additionally, she serves on the Board of Studies at Pondicherry University as a nominee. She has numerous publications in international journals, reflecting her dedication to research and academia. She can be contacted at email: vijayaln@srmist.edu.in.