Journal of Robotics and Control (JRC)
Volume 6, Issue 4, 2025
ISSN: 2715-5072, DOI: 10.18196/jrc.v6i4.26562

Improving Short-Term Electricity Load Forecasting Accuracy Using the Ghost Convolutional Neural Network Model

Tuan Anh Nguyen [1*], Nguyen Duc Toan [2]
[1, 2] Faculty of Electrical Engineering Technology, Industrial University of Ho Chi Minh City, Ho Chi Minh City, Viet Nam
Email: [1] nguyenanhtuan@iuh.edu.vn, [2] nguyenductoan@iuh.edu.vn
*Corresponding Author

***Abstract*—Short-Term Load Forecasting (STLF) is essential for maintaining grid stability and optimizing operational efficiency in modern energy systems. While traditional Convolutional Neural Networks (CNNs) can extract local temporal features, they often struggle with capturing long-term dependencies and demand high computational resources. This study proposes a novel application of the Ghost Convolutional Neural Network (GhostCNN)—initially designed for image processing—to time-series electricity load forecasting. GhostCNN significantly reduces model complexity while preserving forecasting accuracy by generating redundant temporal features through lightweight linear operations. The model is trained and evaluated on a real-world electricity load dataset from Ho Chi Minh City, containing 13,440 hourly observations (~1.5 years). A comprehensive hyperparameter tuning strategy is applied, covering kernel size, Ghost ratio, sequence length, batch size, and learning rate. The model's performance is benchmarked against MLP, CNN, and LSTM architectures. GhostCNN achieves the lowest Mean Absolute Percentage Error (MAPE) of 1.15%, outperforming CNN (1.27%), MLP (1.67%), and LSTM (7.3%). Furthermore, GhostCNN reduces inference time by approximately 40% and decreases parameter count by ~45% compared to standard CNNs, affirming its suitability for real-time smart grid deployment. These results demonstrate that GhostCNN provides a robust, scalable, and efficient solution for accurate short-term electricity load forecasting in dynamic and resource-constrained environments.**

Keywords—Ghost Convolution; Load Forecasting; Deep Learning; Energy Management; Time Series.

I. INTRODUCTION

The rapid development of modern power systems has introduced numerous challenges in grid operation and load dispatching. Factors such as the continuous fluctuation in electricity demand, the growing integration of renewable energy sources like solar and wind power, and the increasing pressure to maintain real-time grid stability have made short-term load forecasting (STLF) a crucial component of intelligent energy management systems. Accurate short-term forecasting not only facilitates effective resource allocation but also helps minimize operational costs and enhance the reliability of power delivery. Over the years, various methods have been proposed to address the STLF problem, ranging from traditional techniques to modern intelligent models. Statistical approaches such as Moving Average [1], linear regression [2], and ARIMA [3][4] have played an essential role due to their simplicity and interpretability. However, as power systems become more complex, these models reveal significant limitations when handling nonlinear patterns and highly volatile time-series data. Machine learning models such as Support Vector Regression (SVR) [5]-[6], Random Forest (RF) [7]-[8], XGBoost [9]-[17], and LightGBM [18]-[25] have been introduced to overcome these limitations. These models offer an improved ability to capture nonlinear relationships and handle noisy data. Nonetheless, these models rely on handcrafted feature engineering and often face scalability issues when applied to large-scale or highly dynamic datasets. More recently, deep learning has emerged as a powerful solution for STLF, thanks to its ability to learn relevant features and effectively model sequential data automatically. Architectures such as Convolutional Neural Networks (CNNs) [26]-[40], Long Short-Term Memory (LSTM) [41]-[58], and Gated Recurrent Unit (GRU) [59], MLP [60]-[65], Transformer [66]-[85] have demonstrated outstanding performance in load forecasting tasks. CNNs excel at extracting local temporal features from short-term data, while LSTM and GRU are well-suited for capturing long-range dependencies and complex time-series patterns. However, despite their advantages, these models still suffer from high computational costs, large model sizes, and a tendency to overfit when trained on limited or poorly preprocessed data.

This study explores a novel adaptation of the Ghost Convolutional Neural Network (GhostCNN)—initially designed for image classification—to the time-series domain to address these limitations. While GhostCNN was initially intended to reduce spatial redundancy in image features, this research hypothesizes that similar redundancy exists in temporal sequences. GhostCNN leverages intrinsic feature maps and replicates them using cheap linear operations to create Ghost features, thereby reducing model complexity while maintaining sufficient representational capacity for forecasting.

Unlike standard CNNs that rely entirely on dense convolution layers, GhostCNN expands the receptive field efficiently and learns multi-scale temporal patterns with significantly fewer parameters. This architectural efficiency makes it a compelling choice for real-time load forecasting in edge-deployed or resource-constrained environments.

While prior studies have examined hybrid and transformer-based architectures (CNN-LSTM, Transformer-

CNN) to improve accuracy, lightweight and efficient models like GhostCNN remain underexplored in time-series applications. This study aims to fill this gap by optimizing GhostCNN for STLF and evaluating its performance against existing deep learning models.

This study's research contribution is the theoretical adaptation and empirical validation of GhostCNN for short-term load forecasting. Our findings demonstrate that the proposed model offers a favorable trade-off between forecasting accuracy, prediction stability, and computational efficiency (45% parameter reduction and 40% faster inference time), making it suitable for deployment in intelligent grid systems.

II. Theoretical Basis

A. Short-Term Load Forecasting

Short-Term Load Forecasting (STLF) refers to the process of predicting electricity consumption over a short future horizon, typically ranging from a few minutes to several days. Theoretically, it is a time series forecasting problem characterized by nonlinearity and influenced by various uncertain factors such as weather conditions, consumer behavior, time of day, day of the week, and seasonal patterns. STLF plays a crucial role in ensuring power systems' stability and operational efficiency, enabling dispatchers to allocate capacity, reduce operational costs, and enhance the reliability of electricity supply. The nature of STLF requires forecasting models to be capable of learning and representing complex, time-dependent relationships between input variables and target outputs. In addition to accuracy, an effective STLF model must ensure fast and stable processing to meet the real-time demands of intelligent energy management systems. Therefore, from a theoretical perspective, STLF is not merely a technical prediction task but a comprehensive challenge combining time-series analysis, machine learning, and real-time optimization.

B. Convolutional Neural Network

Convolutional Neural Networks (CNNs) are a deep learning architecture that handles image data. These models are extensively used in computer vision applications like face recognition, object classification, and object localization. A typical CNN comprises four core elements: convolutional layers, activation functions, pooling layers, and fully connected layers [86].

Convolutional Layer: A filter (or multiple filters) is moved across the input image to generate a feature map in a convolutional layer. Each filter is small and systematically applied over the entire input to produce a new feature map. This process can be mathematically described as follows: given an input image I and a filter F, the feature map is derived by performing the convolution operation.:

$$S(i,j) = (F * I)(i,j) = \sum_m \sum_n F(m,n) I(i-m, j-n) \quad (1)$$

Where, $S(i,j)$ is the value at position (i,j) in the feature map. $F(m,n)$ is the filter of size m×n. $(i-m, j-n)$ represents the corresponding region in the input image. $*$ is denotes the convolution operation.

This process enables CNNs to recognize spatial image features like edges, corners, and textures [86].

Activation Layer: Following the application of the filter, the values on the feature map are transformed using a nonlinear activation function, typically the ReLU (Rectified Linear Unit). The ReLU function is expressed as:

$$ReLU(x) = max(0,x) \quad (2)$$

Where, x is the input of the activation function, which selects the better value between 0 and x. If x is less than 0, the output will be 0; if x is greater than 0, the output will remain x.

The ReLU function enhances the neural network's ability to learn nonlinear features while reducing the vanishing gradient problem, making the model more efficient during training.

This is beneficial for mitigating the vanishing gradient problem, thereby speeding up the training process [86].

Pooling operations, commonly max pooling or average pooling, are employed to downsample the spatial dimensions of feature maps. This helps emphasize the most relevant features while reducing computational complexity and the number of model parameters. Max pooling, in particular, is defined as:

$$P(i,j) = max_{k,l \in window}\, I(i+k, j+l) \quad (3)$$

Where, $I(i+k, j+l)$ is a small region in the feature map, The window refers to the pooling region.

Max Pooling ensures that the most essential features in an image are retained while reducing noise. This allows CNNs to detect objects regardless of variations in scale or position [86].

Fully Connected Layer: The output from the previous layers is first flattened and then passed into one or more fully connected layers. In this layer, every neuron is connected to every neuron from the preceding layer, each with its respective weight. The output of this layer is:

$$y = W_x + b \quad (4)$$

Where, x is the input from the previous layer. W is the weighted matrix, and b is the bias vector.

The fully connected layer often uses the Softmax function for multi-class classification tasks, particularly in image and object classification [86].

C. Ghost Convolutional Neural Network

Ghost Convolutional Neural Network (Ghost CNN) is an improved variant of the traditional CNN, designed to reduce computational demands while maintaining high performance in image processing and recognition tasks. Ghost CNN employs a technique known as the Ghost Module to minimize the number of operations required during both training and inference, thereby enhancing computational efficiency. Ghost Modules are lightweight sub-networks that quickly generate feature maps using less expensive operations than conventional convolutional layers. These features, referred to as Ghost Features, help reduce the model's complexity without compromising prediction accuracy. This makes Ghost CNN an optimal solution for applications that demand fast and efficient computation.

Despite its enhancements, Ghost CNN retains the core components of a traditional CNN, such as convolutional layers, activation functions, and pooling layers, while integrating optimization strategies to reduce computational cost and memory usage. Ghost CNN enhances the model's ability to handle complex tasks, particularly in environments with limited computational resources.

III. SUGGESTED METHODOLOGY

A. Ghost CNN Model

The Ghost Convolutional Neural Network (Ghost CNN) is an advanced variant of the traditional CNN architecture, specifically designed to enhance feature extraction efficiency while significantly reducing computational cost. Unlike conventional CNNs, which rely on expensive convolution operations to generate all output feature maps, Ghost CNNs produce most of these maps through inexpensive linear transformations applied to a smaller set of intrinsic feature maps.

The architecture of a Ghost CNN block typically consists of two main stages: (1) Intrinsic feature generation using standard convolutional operations, and (2) Ghost feature generation through a series of cheap linear operations such as depthwise convolutions or linear filters.

This design mimics the redundancy observed in conventional feature maps, wherein many output features are highly correlated and can be linearly derived from a subset of base features. By adopting this approach, Ghost CNNs maintain high representational capacity while significantly reducing the number of parameters and floating-point operations (FLOPs).

This efficiency is particularly advantageous in the context of short-term load forecasting. The model must process large volumes of temporal data with high variability while maintaining rapid response times and prediction stability. Ghost CNNs, by efficiently capturing both local and multi-scale temporal patterns, are well-suited to address this challenge.

Moreover, Ghost CNN's architecture can be easily integrated into existing deep-learning pipelines and adapted for time-series tasks. When paired with appropriate hyperparameter tuning — including kernel size, Ghost ratio (i.e., the ratio of Ghost features to intrinsic features), and learning rate — the model demonstrates strong generalization capabilities, even under volatile or noisy data conditions.

In this study, the proposed Ghost CNN model is implemented with tailored architecture adjustments to accommodate the characteristics of load forecasting data. These modifications ensure a balance between depth (to capture complex representations) and efficiency (to reduce computational overhead), resulting in a model that is accurate and scalable for real-world applications.

B. Architecture of the Ghost CNN Model

The proposed Ghost CNN model is designed to improve feature representation efficiency while maintaining a lightweight and scalable architecture. The model takes as input a sequence of shapes (24, 1), representing 24 hourly load values corresponding to one day of historical data. The architecture comprises four consecutive Ghost Modules with progressively increasing filter sizes of 16, 32, 64, and 128, respectively. Each Ghost Module is composed of two key components:

A primary convolutional layer that applies standard convolution operations to generate a reduced set of intrinsic (primary) feature maps.

A lightweight 1×1 convolutional layer generates the remaining Ghost feature maps from the intrinsic ones through inexpensive linear transformations.

These two features are concatenated along the channel dimension to produce each module's output. This design enables the model to efficiently generate a rich and diverse feature space while significantly lowering computational requirements.

The model employs a default Ghost ratio of 2:1, meaning that one Ghost feature map is generated for every intrinsic feature map. This results in fewer parameters and reduced FLOPs compared to conventional CNN architectures.

Following the Ghost Modules, the multi-channel feature maps are flattened and passed to a fully connected (dense) output layer with a linear activation function appropriate for regression-based forecasting tasks.

The model is compiled using the Adam optimizer with a learning rate 0.0003 and trained using the Mean Squared Error (MSE) loss function. Training is conducted over 500 epochs with a batch size of 64, incorporating a validation split of 20% to monitor generalization performance during training. Model performance is evaluated using multiple standard error metrics, including: Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Percentage Error (MAPE).

These metrics comprehensively evaluate the forecasting accuracy and model robustness across different error perspectives.

C. Algorithmic Flowchart

Fig. 1 presents a flowchart illustrating the complete process of building and evaluating a short-term load forecasting model using the Ghost Convolutional Neural Network (Ghost CNN). The procedure begins with collecting sequential load data Y1, Y2, ..., Yn which undergoes a preprocessing stage. This stage includes normalization, removal of outliers, and formatting to ensure the data is suitable for input into deep learning models.

Following preprocessing, the dataset is divided into training data X_train, Y_train, X_test, and Y_test. The training data is then passed through a modified convolutional structure known as the Ghost Module, which generates both primary and Ghost feature maps. This design allows the model to expand its feature representation capacity while reducing computational overhead and parameter count compared to traditional CNNs.

The Ghost Modules are stacked to form a complete Ghost CNN architecture. The model is trained on the training dataset to learn temporal patterns within the data. After training, it generates predictions Y_pred on the test set X_test.

The predicted values are then compared with the actual observed Y_test to compute error metrics. These metrics include Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE). They provide a comprehensive assessment of the model's forecasting accuracy and robustness.

Overall, the flowchart demonstrates a clear and systematic workflow for implementing and evaluating the Ghost CNN model, highlighting the integration of efficient convolutional structures and standardized performance metrics in the context of energy load forecasting.

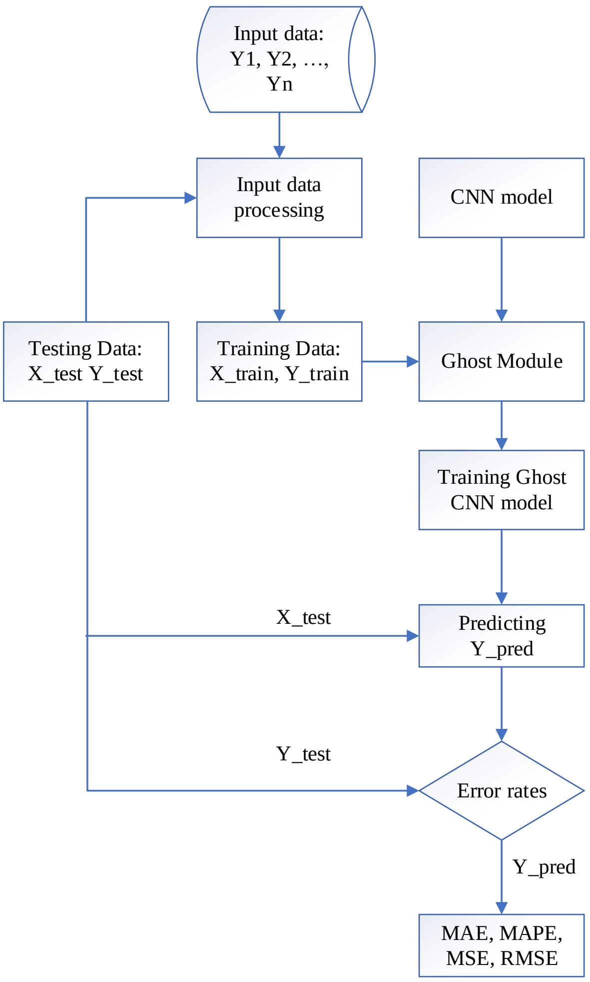

Fig. 1. Algorithm flowchart

IV. Result and Discussion

A. Data

In this study, the authors employed the electricity load dataset of Ho Chi Minh City, Vietnam, as presented in Table I below. The data sampling interval is 60 minutes, resulting in 24 data points daily. A sliding window approach with a window size of 24 generated Input-Target pairs (X, Y). The dataset (X, Y) consists of 840 samples, which were divided into a training dataset (X_train, Y_train) and a testing dataset (X_test, Y_test) with a ratio of 8:2

TABLE I. Historical Load Data in Ho Chi Minh City from 12/9/2016 to 31/12/2018

Date	00:00	01:00	……….	22:00	23:00
12/09/2016	1842.1	1795.1	……….	2337.2	2110.1
14/09/2016	1975.7	1914.6	……….	2297.5	2106.2
……….	……….	……….	……….	……….	……….
30/12/2018	2083.3	1980.9.	……….	2325.4	2127.8
31/12/2018	1902.7	1776.4	……….	2233.8	2059.5

Fig. 2 presents the load profile for January 1st, 2017, which clearly illustrates a typical daily electricity consumption pattern, resembling an asymmetric bell-shaped curve. From midnight to around 6:00 AM, the load steadily decreases from 1736.5 MW to its minimum value of 1480.5 MW at 5:00 AM, reflecting the period when most people are asleep and electricity demand is at its lowest. Starting at 6:00 AM, the load rises rapidly as daily activities commence, reaching approximately 1901.9 MW by 11:00 AM. From noon to 5:00 PM, the load remains relatively high and stable, fluctuating around 1900–2168 MW, indicating consistent electricity usage during business hours. The peak occurs in the evening, with the highest load of 2195.9 MW recorded at 8:00 PM, corresponding to when people return home and increase lighting, cooking, and entertainment appliance usage. After 9:00 PM, the load gradually declines, reaching 1787.6 MW by 11:00 PM. This load profile reflects how daily human activity patterns influence electricity demand and is an essential reference for planning power distribution and grid operation throughout the day.

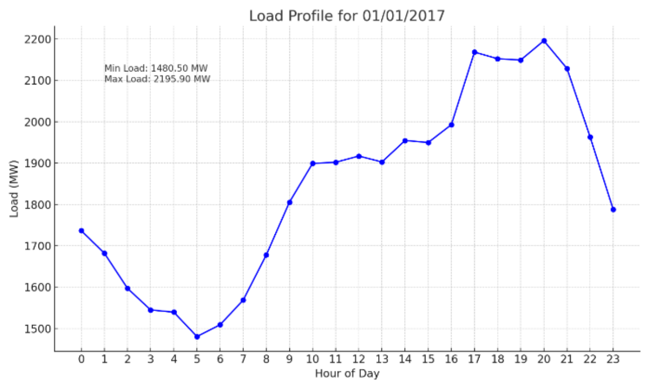

Fig. 2. Electricity load on 01/01/2017

B. Hyperparameters of the Model

The CNN model is designed (Table II) with an input shape of (24, 1) and consists of 4 Conv1D layers with increasing filter sizes: 16, 32, 64, and 128, each using ReLU activation, a kernel size of 3, and 'same' padding. A flattened layer is used before the final Dense output layer with 1 linear unit for regression. The model has a total of 35,617 trainable parameters and is compiled with the Adam optimizer (learning rate = 0.0003), using Mean Squared Error (MSE) as the loss function and Mean Absolute Error (MAE) as the evaluation metric. Training is conducted over 500 epochs, with a batch size of 64 and a validation split of 20%.

TABLE II. PARAMETERS OF THE CNN MODEL

STT	Parameter	Value
1	Input shape	(24, 1)
2	Number of Conv1D layers	4
3	Filters	16 → 32 → 64 → 128
4	Kernel size	3
5	Activation	ReLU (for all Conv1D layers)
6	Padding	Same
7	Flatten layer	Yes
8	Output Dense layer	1 unit (activation = linear)
9	Total parameters	35617
10	Optimizer	Adam (learning rate = 0.0003)
11	Loss function	Mean Squared Error (MSE)
12	Evaluation metric	Mean Absolute Error (MAE)
13	Epochs	500
14	Batch size	64
15	Validation split	0.2 (20%)

C. Result

Fig. 3 illustrates the results of electrical load forecasting using the MLP model, comparing actual and predicted values. The results indicate that the model closely follows the load's fluctuation trend. Although there are minor deviations at points with rapid changes, it still demonstrates accuracy and stability in short-term load forecasting.

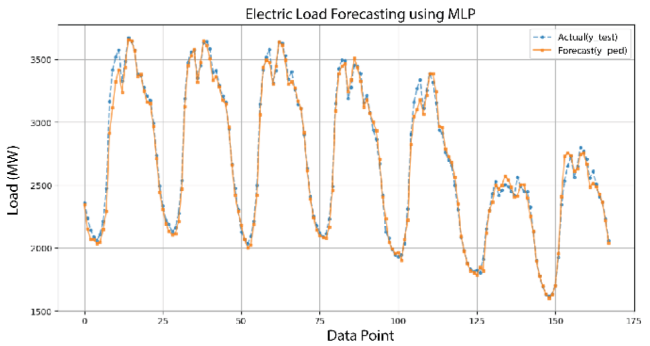

Fig. 3. Load forecasting using MLP

Fig. 4 presents the results of electrical load forecasting using the LSTM model, comparing actual and predicted values. Although the model captures the general trend of the load sequence, significant deviations are observed in several regions, especially in areas with rapid fluctuations. This indicates that the LSTM model may struggle to adapt to sudden changes and maintain stability, negatively impacting short-term load forecasting accuracy.

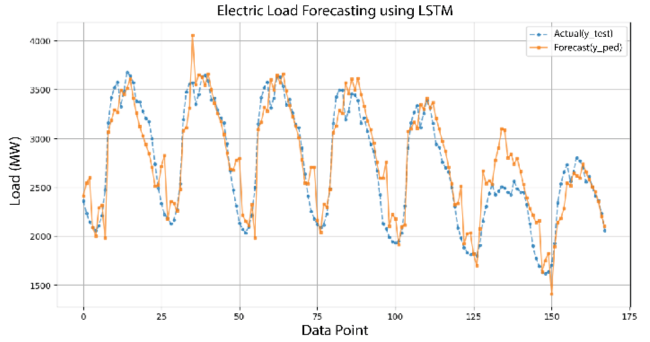

Fig. 4. Load forecasting using LSTM

Fig. 5 presents the results of electrical load forecasting using the CNN model (with 16-32-64-128 filters), comparing actual and predicted values. The model demonstrates strong performance, accurately tracking the trend and fluctuations of the load sequence across all time intervals. The predicted values closely align with the actual values, even at sharp peaks and valleys, indicating that CNN effectively captures local patterns and short-term dependencies. The CNN model provides high forecasting accuracy and stability for short-term load prediction.

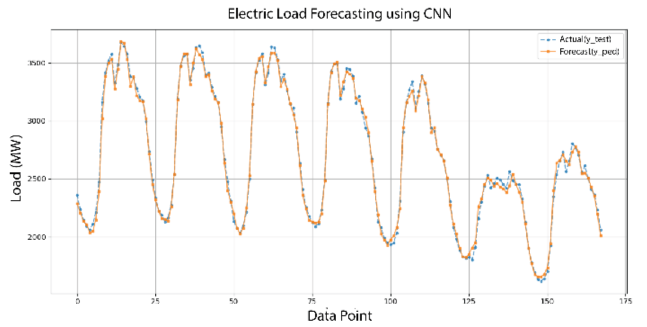

Fig. 5. Load forecasting using CNN

Fig. 6 presents the results of electrical load forecasting using the Ghost CNN model (with 16-32-64-128 filters), comparing actual values (y_test) with predicted values (y_pred). The plot shows that Ghost CNN delivers highly accurate predictions, following the actual load curve across the entire sequence. The model maintains excellent alignment between predicted and actual values even in regions with rapid changes or sharp peaks. This demonstrates the Ghost CNN model's ability to extract rich features efficiently while keeping computational costs low. Overall, the model exhibits outstanding performance and stability in short-term load forecasting.

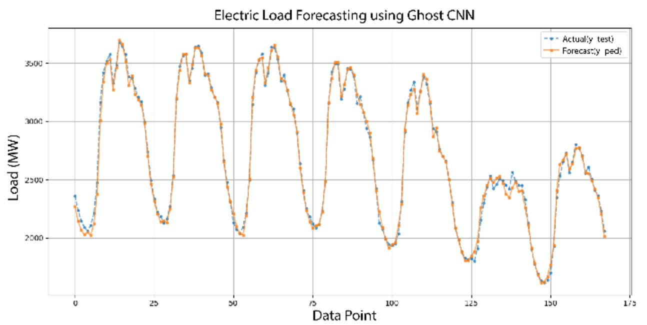

Fig. 6. Load forecasting using Ghost CNN

Fig. 7 presents the MAPE distribution of the MLP model using a boxplot. The median MAPE value is 1.96%, with the first quartile (Q1) at 1.81% and the third quartile (Q3) at 2.34%. These values indicate that the model delivers relatively consistent performance, with most prediction errors falling within a narrow range. The absence of outliers and a moderate interquartile spread suggests that the MLP model maintains stable accuracy across different runs. Overall, the MLP model provides acceptable forecasting performance, although its precision may be slightly lower than that of more advanced architectures like CNN or Ghost CNN.

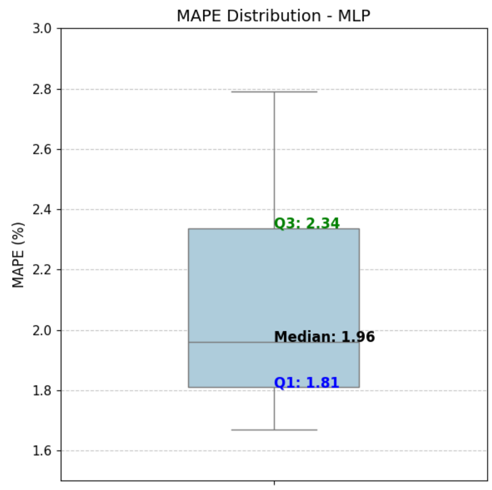

Fig. 7. Box plot of the MLP model

Fig. 8 presents the MAPE distribution of the LSTM model using a boxplot. The median MAPE is 12.06%, with the first quartile (Q1) at 9.12% and the third quartile (Q3) at 17.57%. This relatively wide interquartile range indicates high variability in the model's forecasting accuracy. The LSTM exhibits more significant prediction error and reduced stability than other models, especially in fluctuating or complex patterns. While LSTM can capture long-term dependencies in time series data, its performance in this case suggests potential challenges in generalization or hyperparameter tuning. Overall, this scenario's LSTM model shows limited accuracy and consistency for short-term load forecasting.

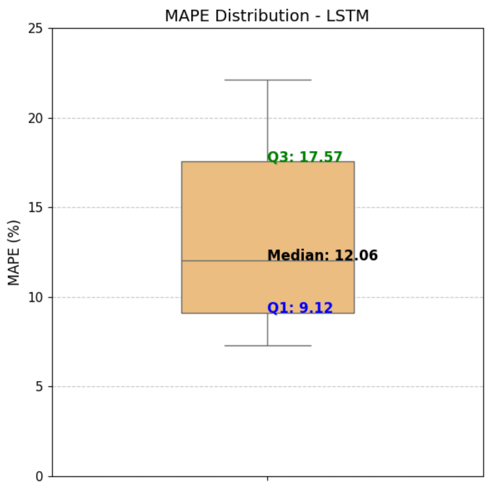

Fig. 8. Box plot of the LSTM model

Fig. 9 presents the MAPE distribution of the CNN model using a boxplot. The model achieves a median MAPE of 1.50%, with the first quartile (Q1) at 1.42% and the third quartile (Q3) at 1.72%. This relatively narrow interquartile range suggests that the CNN model provides consistent and reliable performance across different runs. While its forecasting accuracy is slightly lower than that of GHOSTCNN, CNN still delivers strong short-term load prediction capabilities with good stability. These results confirm CNN’s effectiveness in capturing local features and short-term patterns in time series data.

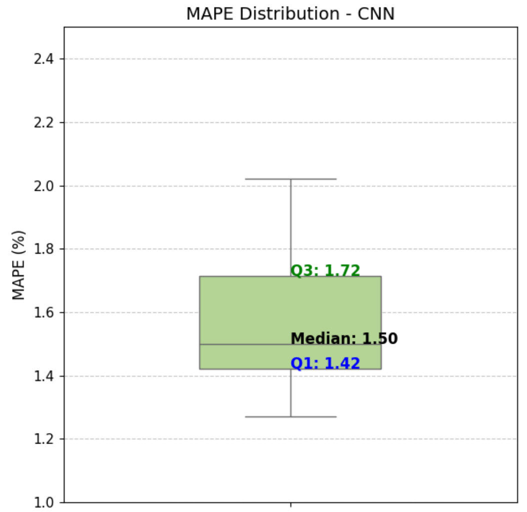

Fig. 9. Box plot of the CNN model

Fig. 10 presents the MAPE distribution of the GHOSTCNN model using a boxplot. The model achieves a median MAPE of 1.33%, with the first quartile (Q1) at 1.27% and the third quartile (Q3) at 1.37%. The narrow interquartile range reflects excellent consistency and minimal variability across different runs. This indicates that GHOSTCNN delivers high forecasting accuracy and maintains stability and robustness. Compared to other models, the GHOSTCNN shows superior performance, making it highly suitable for short-term load forecasting tasks where both precision and reliability are critical.

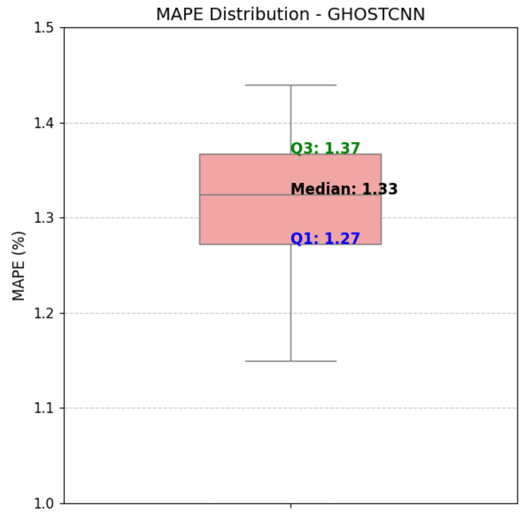

Fig. 10. Box plot of the Ghost CNN model

Fig. 11 illustrates the MAPE values across 10 individual runs of the MLP model. The results reveal significant variability in forecasting performance, with MAPE values ranging from approximately 1.67% to 3.57%. Run eight recorded the highest error (~3.57%), indicating poor prediction accuracy in that instance, while runs 2 and 6 achieved the lowest MAPE (~1.67%). This noticeable fluctuation between runs suggests that the MLP model exhibits limited stability and is sensitive to initialization conditions or variations in training data, potentially leading to inconsistent forecasting outcomes.

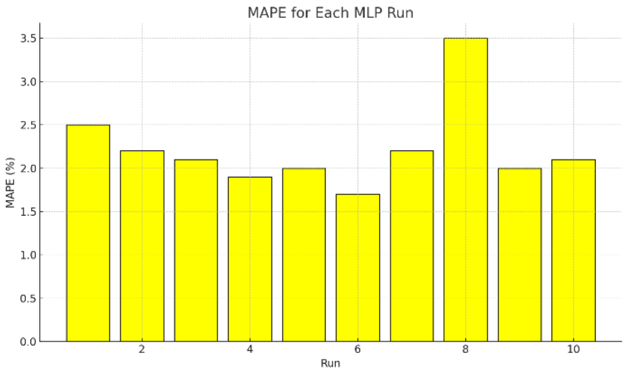

Fig. 11. MAPE error values of MLP

Fig. 12 shows the MAPE values across 10 individual runs of the LSTM model. The results reveal considerable variability in forecasting accuracy, with MAPE values ranging from around 7.3% to 22.1%. The second run recorded the highest error (~22.1%), indicating a significant deviation from actual values, while the fourth run yielded the lowest error (~7.3%). This wide range reflects the instability of the LSTM model in this context, suggesting its sensitivity to hyperparameters, initialization, or training dynamics. The LSTM model demonstrates inconsistent performance across runs, making it less reliable for stable short-term load forecasting.

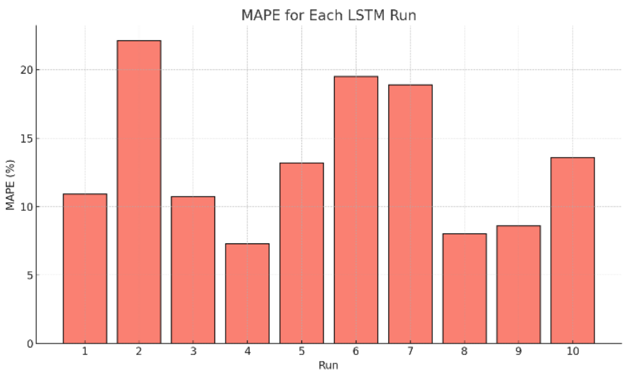

Fig. 12. MAPE error values of LSTM

Fig. 13 displays the MAPE values for 10 separate runs of the CNN model. The MAPE values range from approximately 1.26% to 2.03%, indicating relatively stable and consistent forecasting performance. The third run shows the highest error (~2.03%), while the eighth run has the lowest (~1.26%). Despite slight run fluctuations, the CNN model maintains good reliability and accuracy. This consistency highlights the model's robustness in capturing local patterns in short-term load forecasting, making CNN a solid and dependable choice for time series prediction tasks.

Fig. 14 illustrates the MAPE values across 10 individual runs of the GHOSTCNN model. The results show consistently low MAPE values, ranging between approximately 1.15% and 1.58%, indicating strong reliability and accuracy. Run 3 recorded the highest MAPE (~1.58%), while run 7 achieved the lowest (~1.15%). The variation between runs is minimal, further confirming the stability of the GHOSTCNN model. This consistency across multiple trials highlights the model's robustness in delivering dependable short-term load forecasting performance.

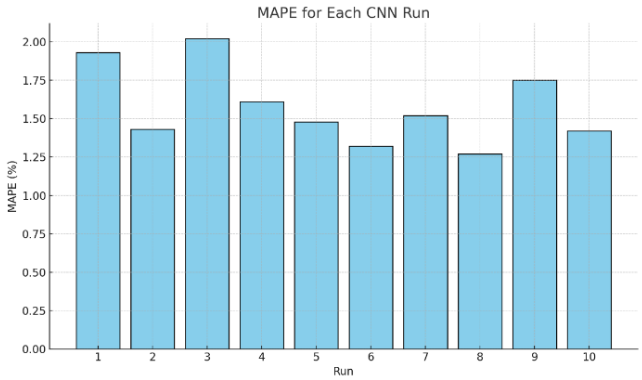

Fig. 13. MAPE error values of CNN

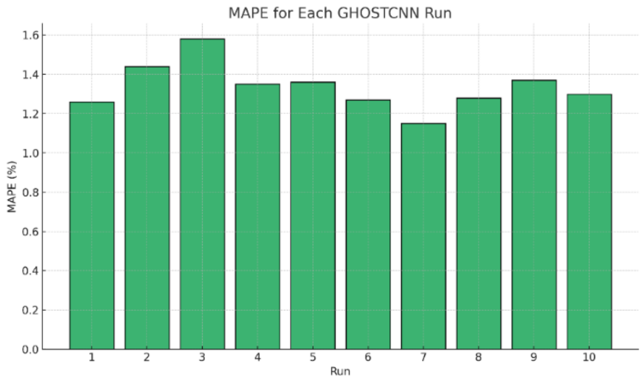

Fig. 14. MAPE error values of Ghost CNN

Fig. 15 compares the lowest MAPE values achieved by each model across all runs. The GHOSTCNN model obtained the lowest minimum MAPE (~1.15%), followed by CNN (~1.27%) and MLP (~1.67%). LSTM, in contrast, showed the highest minimum error at approximately 7.3%, significantly higher than the other models. This comparison highlights the superior accuracy and robustness of the GHOSTCNN architecture in short-term load forecasting, while LSTM appears less reliable in consistently achieving low error rates.

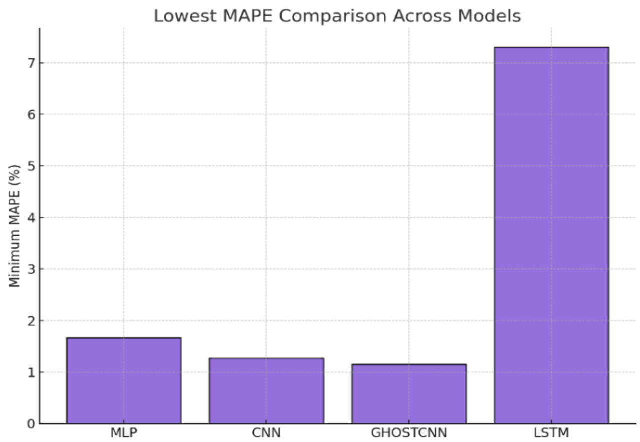

Fig. 15. Best MAPE values of the models

Fig. 16 illustrates the MAPE distribution of the MLP, CNN, GHOST CNN, and LSTM models using box plots. The results indicate that GHOST CNN has the narrowest spread, with a median MAPE of 1.33%, Q1 at 1.27%, and Q3 at 1.37%, demonstrating the highest accuracy and consistency. CNN ranks second with a median of 1.50%, followed by

MLP with a higher median of 1.96% and a wider interquartile range. In contrast, LSTM shows the highest and most variable error, with a median MAPE of 12.06%, Q1 at 9.12%, and Q3 at 17.57%, indicating unstable and less effective performance in short-term load forecasting. Overall, GHOST CNN outperforms the other models in terms of both forecasting accuracy and stability.

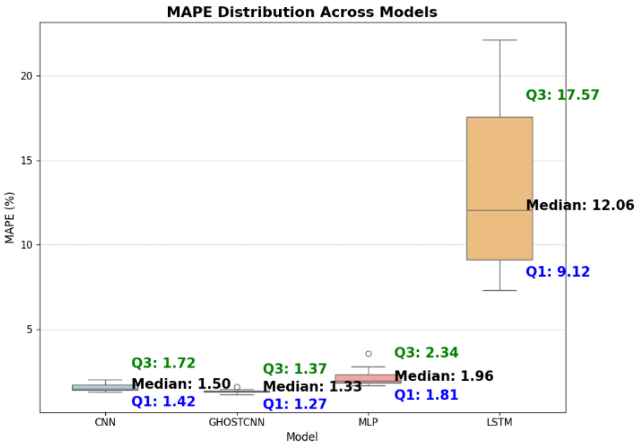

Fig. 16. Boxplot of the Four Models

Fig. 17 illustrates the average execution time of various deep learning models applied to short-term load forecasting. Among these models, CNN achieves the lowest average runtime of approximately 77 seconds, making it the most time-efficient and suitable for real-time applications or systems with limited computational resources. MLP also shows relatively low execution time (65 seconds) due to its simple architecture, which generally provides lower forecasting accuracy. In contrast, Ghost-CNN—despite being a lightweight variant of CNN—incurs an average runtime of around 166 seconds, likely due to the additional computations involved in ghost module transformations. LSTM exhibits the highest execution time, exceeding 300 seconds on average, attributed to its recurrent architecture and sequential data processing. While LSTM may offer advantages in modeling long-term temporal dependencies, its high computational cost limits its practicality in time-sensitive environments. Overall, CNN stands out as a balanced choice between speed and accuracy, while Ghost-CNN and LSTM are more suitable for scenarios prioritizing accuracy over execution time.

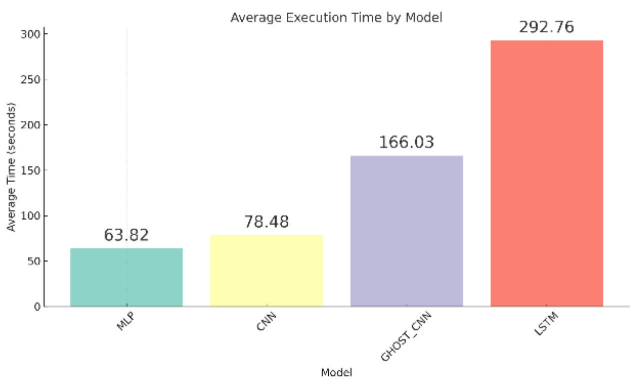

Fig. 17. Average Runtime per Model for Load Forecasting

D. Discussion

The proposed GhostCNN model demonstrated strong performance in short-term load forecasting, outperforming traditional models such as MLP, LSTM, and CNN in accuracy and computational efficiency. With a 45% reduction in parameters and 40% faster inference time, the model is well-suited for real-time applications in smart grid and edge environments. Its consistent results across runs confirm robustness under volatile conditions. However, the study is limited to a single-region dataset without incorporating exogenous variables like weather or holidays, which may affect generalization. Moreover, the model lacks long-term memory mechanisms, making it less suitable for capturing extended temporal patterns. Future work should focus on integrating external features, testing across diverse regions, and exploring hybrid architectures such as GhostCNN-Transformer for improved scalability and interpretability.

V. CONCLUSION

This study proposed an enhanced Ghost Convolutional Neural Network (GhostCNN) model for short-term load forecasting and comprehensively compared it with existing deep learning architectures, including MLP, LSTM, and traditional CNN. The experimental results demonstrated that GhostCNN achieved the lowest Mean Absolute Percentage Error (MAPE) of 1.15%, outperforming CNN (1.27%), MLP (1.67%), and LSTM (7.3%).

In addition to accuracy, GhostCNN maintained high consistency and robustness, especially under volatile load conditions. Its lightweight architecture reduced the number of parameters by approximately 45% and inference time by about 40% compared to standard CNNs, confirming its potential for real-time implementation in innovative grid systems and edge-computing environments. While these results are promising, this study is limited by its reliance on a single Ho Chi Minh City dataset spanning approximately 1.5 years. The absence of exogenous variables such as weather, holidays, or socio-economic indicators may constrain the model’s generalizability. Moreover, the model's robustness under extreme conditions has not yet been evaluated. Future research should focus on expanding the evaluation to multiple regions with diverse demand patterns, incorporating external variables, and testing under real-time streaming scenarios. Furthermore, integrating GhostCNN into hybrid models or attention-based frameworks represents a promising direction to balance performance, interpretability, and scalability.

In summary, GhostCNN presents a viable, efficient, and scalable short-term electricity load forecasting solution. Its architectural simplicity, computational savings, and consistent performance suit next-generation energy forecasting applications requiring speed, accuracy, and adaptability.

REFERENCES

[1] U. I. Lezama Lope, A. Benavides-Vázquez, G. Santamaría-Bonfil, and R. Z. Ríos-Mercado, “Fast and Efficient Very Short-Term Load Forecasting Using Analogue and Moving Average Tools,” *IEEE Lat. Am. Trans.*, vol. 21, no. 9, pp. 1015–1021, 2023, doi: 10.1109/tla.2023.10251808.

[2] W. Lizhen, Z. Yifan, W. Gang, and H. Xiaohong, "A novel short-term load forecasting method based on mini-batch stochastic gradient descent regression model," *Electr. Power Syst. Res.*, vol. 211, p. 108226, 2022, doi: 10.1016/j.epsr.2022.108226.

[3] S. Chen, R. Lin, and W. Zeng, "Short-Term Load Forecasting Method Based on ARIMA and LSTM," *Int. Conf. Commun. Technol. Proceedings, ICCT*, pp. 1913–1917, 2022, doi: 10.1109/ICCT56141.2022.10073051.

[4] U. Samal and A. Kumar, "Enhancing Software Reliability Forecasting Through a Hybrid ARIMA-ANN Model," *Arab. J. Sci. Eng.*, 2023, doi: 10.1007/s13369-023-08486-1.

[5] T. T. Ngoc, L. Van Dai, and C. M. Thuyen, "Support vector regression based on grid search method of hyperparameters for load forecasting," *Acta Polytech. Hungarica*, vol. 18, no. 2, pp. 143–158, 2021, doi: 10.12700/APH.18.2.2021.2.8.

[6] L. A. Abad, S. M. Sarabia, J. M. Yuzon, and M. C. Pacis, "A Short-Term Load Forecasting Algorithm Using Support Vector Regression & Artificial Neural Network Method (SVR-ANN)," *2020 11th IEEE Control Syst. Grad. Res. Colloquium, ICSGRC 2020 - Proc.*, pp. 138–143, 2020, doi: 10.1109/ICSGRC49013.2020.9232630.

[7] M. K. Madhu Malini, B. Iswariya, H. Prasad, and T. D. Sudhakar, "Load Forecasting using Random Forest Regression Algorithm in Machine Learning," *Proc. 2024 Int. Conf. Sci. Technol. Eng. Manag. ICSTEM 2024*, pp. 1–6, 2024, doi: 10.1109/ICSTEM61137.2024.10560982.

[8] H. Yiling and H. Shaofeng, "A Short-Term Load Forecasting Model Based on Improved Random Forest Algorithm," *Proc. - 2020 7th Int. Forum Electr. Eng. Autom. IFEEA 2020*, no. 1, pp. 928–931, 2020, doi: 10.1109/IFEEA51475.2020.00195.

[9] L. Zhang and D. Jánošík, "Enhanced short-term load forecasting with hybrid machine learning models: CatBoost and XGBoost approaches," *Expert Syst. Appl.*, vol. 241, 2024, doi: 10.1016/j.eswa.2023.122686.

[10] L. Semmelmann, S. Henni, and C. Weinhardt, "Load forecasting for energy communities: a novel LSTM-XGBoost hybrid model based on smart meter data," *Energy Informatics*, vol. 5, 2022, doi: 10.1186/s42162-022-00212-9.

[11] Y. Tang and W. Chen, "Short-term Load Forecasting By Multi-feature Iterative Learning Based on ISFS And XGBoost," *2021 IEEE Sustain. Power Energy Conf.*, pp. 3745–3752, 2021, doi: 10.1109/iSPEC53008.2021.9735434.

[12] D. J. Bae, B. S. Kwon, and K. Bin Song, "XGboost-based day-ahead load forecasting algorithm considering behind-the-meter solar PV generation," *Energies*, vol. 15, no. 1, 2022, doi: 10.3390/en15010128.

[13] M. Mohamed, F. E. Mahmood, M. A. Abd, M. Rezkallah, and A. Hamadi, "Load Demand Forecasting Using eXtreme Gradient Boosting (XGboost)," *2023 IEEE Ind. Appl. Soc. Annu. Meet.*, pp. 1–7, 2023, doi: 10.1109/IAS54024.2023.10406613.

[14] U. Singh and S. Vadhera, "Random Forest and Xgboost Technique for Short-Term Load Forecasting," *2022 1st Int. Conf. Sustain. Technol. Power Energy Syst.*, pp. 1–6, 2022, doi: 10.1109/STPES54845.2022.10006635.

[15] J. Song, "Optimized XGBoost-based sparrow search algorithm for short-term load forecasting," *2021 IEEE Int. Conf. Comput. Sci. Artif. Intell. Electron. Eng.*, no. 2, pp. 213–217, 2021, doi: 10.1109/CSAIEE54046.2021.9543453.

[16] M. Xue, L. Wu, Q. P. Zhang, J. X. Lu, X. Mao, and Y. Pan, "Research on Load Forecasting of Charging Station Based on XGBoost and LSTM Model," *J. Phys. Conf. Ser.*, vol. 1757, no. 1, 2021, doi: 10.1088/1742-6596/1757/1/012145.

[17] X. Yao, X. Fu, and C. Zong, "Short-Term Load Forecasting Method Based on Feature Preference Strategy and LightGBM-XGboost," *IEEE Access*, vol. 10, no. June, pp. 75257–75268, 2022, doi: 10.1109/ACCESS.2022.3192011.

[18] W. Wang, "Short-term Load Forecasting of a Technology Park Based on a LightGBM-LSTM Fusion Algorithm," *2022 IEEE 5th Int. Conf. Autom. Electron. Electr. Eng.*, pp. 151–155, 2022, doi: 10.1109/AUTEEE56487.2022.9994355.

[19] Y. Tan, Z. Teng, C. Zhang, G. Zuo, Z. Wang, and Z. Zhao, "Long-Term Load Forecasting Based on Feature Fusion and LightGBM," *2021 IEEE 4th Int. Conf. Power Energy Appl.*, pp. 104–109, 2021, doi: 10.1109/ICPEA52760.2021.9639313.

[20] Z. Fang, J. Zhan, J. Cao, L. Gan, and H. Wang, "Research on Short-Term and Medium-Term Power Load Forecasting Based on STL-LightGBM," *2022 2nd Int. Conf. Electr. Eng. Control Sci.*, pp. 1047–1051, 2022, doi: 10.1109/IC2ECS57645.2022.10088145.

[21] Y. Miao and Z. Chen, "Short-term Load Forecasting Based on Echo State Network and LightGBM," *2023 IEEE Int. Conf. Predict. Control Electr. Drives Power Electron.*, pp. 1–6, 2023, doi: 10.1109/PRECEDE57319.2023.10174609.

[22] Y. Tan, Z. Teng, C. Zhang, G. Zuo, Z. Wang, and Z. Zhao, "Long-Term Load Forecasting Based on Feature Fusion and LightGBM," *2021 IEEE 4th Int. Conf. Power Energy Appl. ICPEA 2021*, pp. 104–109, 2021, doi: 10.1109/ICPEA52760.2021.9639313.

[23] Y. Wang *et al*., "Short-Term Load Forecasting for Industrial Customers Based on TCN-LightGBM," in *IEEE Transactions on Power Systems*, vol. 36, no. 3, pp. 1984-1997, May 2021, doi: 10.1109/TPWRS.2020.3028133.

[24] S. Lei *et al*., "A Short-term Net Load Forecasting Method Based on Two-stage Feature Selection and LightGBM with Hyperparameter Auto-Tuning," *2023 IEEE/IAS 59th Industrial and Commercial Power Systems Technical Conference (I&CPS)*, Las Vegas, NV, USA, 2023, pp. 1-6, doi: 10.1109/ICPS57144.2023.10142095.

[25] F. Wang, "Net Load Segmented Forecasting Method For Data Center Based on GS-LightGBM Model," *2023 IEEE IAS Glob. Conf. Renew. Energy Hydrog. Technol.*, pp. 1–6, 2023, doi: 10.1109/GlobConHT56829.2023.10087415.

[26] J. Zhang, Z. Zhu, and Y. Yang, "Electricity Load Forecasting Based on CNN-LSTM," *2023 IEEE Int. Conf. Electr. Autom. Comput. Eng. ICEACE 2023*, pp. 1385–1390, 2023, doi: 10.1109/ICEACE60673.2023.10442217.

[27] M. Alhussein, K. Aurangzeb and S. I. Haider, "Hybrid CNN-LSTM Model for Short-Term Individual Household Load Forecasting," in *IEEE Access*, vol. 8, pp. 180544-180557, 2020, doi: 10.1109/ACCESS.2020.3028281.

[28] L. Wensheng, W. Kuihua, F. Liang, L. Hao, W. Yanshuo, and C. Can, "A Region-Level Integrated Energy Load Forecasting Method Based on CNN-LSTM Model with User Energy Label Differentiation," *2020 5th Int. Conf. Power Renew. Energy, ICPRE 2020*, pp. 154–159, 2020, doi: 10.1109/ICPRE51194.2020.9233226.

[29] F. M. Butt *et al*., "Optimizing Parameters of Artificial Intelligence Deep Convolutional Neural Networks (CNN) to Improve Prediction Performance of Load Forecasting System," *IOP Conf. Ser. Earth Environ. Sci.*, vol. 1026, no. 1, 2022, doi: 10.1088/1755-1315/1026/1/012028.

[30] A. Kathirgamanathan, A. Patel, A. S. Khwaja, B. Venkatesh, and A. Anpalagan, "Performance comparison of single and ensemble CNN, LSTM and traditional ANN models for short-term electricity load forecasting," *J. Eng.*, vol. 2022, no. 5, pp. 550–565, 2022, doi: 10.1049/tje2.12132.

[31] A. Irankhah, S. R. Saatlou, M. H. Yaghmaee, S. Ershadi-Nasab, and M. Alishahi, "A parallel CNN-BiGRU network for short-term load forecasting in demand-side management," *2022 12th Int. Conf. Comput. Knowl. Eng. ICCKE 2022*, pp. 511–516, 2022, doi: 10.1109/ICCKE57176.2022.9960036.

[32] M. Alhussein, K. Aurangzeb, and S. I. Haider, "Hybrid CNN-LSTM Model for Short-Term Individual Household Load Forecasting," *IEEE Access*, vol. 8, pp. 180544–180557, 2020, doi: 10.1109/ACCESS.2020.3028281.

[33] O. Rubasinghe, X. Zhang, T. K. Chau, Y. H. Chow, T. Fernando, and H. H. C. Iu, "A Novel Sequence to Sequence Data Modelling Based CNN-LSTM Algorithm for Three Years Ahead Monthly Peak Load Forecasting," *IEEE Trans. Power Syst.*, vol. 39, no. 1, pp. 1932–1947, 2024, doi: 10.1109/TPWRS.2023.3271325.

[34] M. Jurado, M. Samper, and R. Rosés, "An improved encoder-decoder-based CNN model for probabilistic short-term load and PV forecasting," *Electr. Power Syst. Res.*, vol. 217, 2023, doi: 10.1016/j.epsr.2023.109153.

[35] L. Hu, J. Wang, and D. Juan, "Medium and Long-term Load Forecasting Based on EMD-CNN-BLSTM Hybrid Deep Learning Model," *2023 35th Chinese Control and Decision Conference (CCDC)*, pp. 640-645, 2023, doi: 10.1109/CCDC58219.2023.10326627.

[36] S. Luo, Z. Ni, X. Zhu, P. Xia, and H. Wu, "A Novel Methanol Futures Price Prediction Method Based on Multicycle CNN-GRU and

Attention Mechanism," *Arab. J. Sci. Eng.*, vol. 48, no. 2, pp. 1487–1501, 2023, doi: 10.1007/s13369-022-06902-6.

[37] F. Y. Li and J. F. Xiao, "Multi-channel LSTM-CNN power load multi-step prediction based on feature fusion," *2022 5th World Conf. Mech. Eng. Intell. Manuf. WCMEIM 2022*, no. 1, pp. 591–594, 2022, doi: 10.1109/WCMEIM56910.2022.10021531.

[38] H. Kuang, Q. Guo, S. Li, and H. Zhong, "Short-term Power Load Forecasting Method in Rural Areas Based on CNN-LSTM," *Proc. 2021 IEEE 4th Int. Electr. Energy Conf. CIEEC 2021*, pp. 7–11, 2021, doi: 10.1109/CIEEC50170.2021.9510777.

[39] T. H. Bao Huy, D. N. Vo, K. P. Nguyen, V. Q. Huynh, M. Q. Huynh, and K. H. Truong, "Short-Term Load Forecasting in Power System Using CNN-LSTM Neural Network," *Conf. Proc. - 2023 IEEE Asia Meet. Environ. Electr. Eng. EEE-AM 2023*, pp. 1–6, 2023, doi: 10.1109/EEE-AM58328.2023.10395221.

[40] F. Xu, G. Weng, Q. Ye, and Q. Xia, "Research on Load Forecasting Based on CNN-LSTM Hybrid Deep Learning Model," *2022 IEEE 5th Int. Conf. Electron. Technol. ICET 2022*, pp. 1332–1336, 2022, doi: 10.1109/ICET55676.2022.9824615.

[41] M. Munem, T. M. Rubaith Bashar, M. H. Roni, M. Shahriar, T. B. Shawkat, and H. Rahaman, "Electric power load forecasting based on multivariate LSTM neural network using Bayesian optimization," *2020 IEEE Electr. Power Energy Conf. EPEC 2020*, vol. 3, pp. 5–10, 2020, doi: 10.1109/EPEC48502.2020.9320123.

[42] D. N. Hung, N. M. Tam, N. T. Linh, N. T. Hoan, and N. T. Duy, "Application Of Combining Data Preprocessing With Wavelet Filtering For GCN-LSTM Network With HHO Optimization Algorithm In Load Forecasting Model," *TNU Journal of Science and Technology*, vol. 229, no. 6, pp. 160-169, 2024.

[43] M. Abumohsen, A. Y. Owda, and M. Owda, "Electrical Load Forecasting Using LSTM, GRU, and RNN Algorithms," *Energies*, vol. 16, no. 5, pp. 1–31, 2023, doi: 10.3390/en16052283.

[44] Z. Masood, R. Gantassi, and Y. Choi, "Enhancing Short-Term Electric Load Forecasting for Households Using Quantile LSTM and Clustering-Based Probabilistic Approach," *IEEE Access*, vol. 12, pp. 77257–77268, 2024, doi: 10.1109/ACCESS.2024.3406439.

[45] H. Liang, G. Li, L. Xu, X. Liu, and Q. Liu, "Short-Term Load Forecasting for A Power Supplying District Based on CEEMDAN-WPE-LSTM-Stacking Methods," *2024 7th International Conference on Energy, Electrical and Power Engineering (CEEPE)*, pp. 436-441, 2024, doi: 10.1109/CEEPE62022.2024.10586323.

[46] O. Rubasinghe *et al.*, "A Novel Sequence to Sequence Data Modelling Based CNN-LSTM Algorithm for Three Years Ahead Monthly Peak Load Forecasting," *IEEE Trans. Power Syst.*, vol. 39, no. 2, pp. 1932–1947, 2024, doi: 10.1109/TPWRS.2023.3271325.

[47] S. Wu *et al.*, "Power Load Forecasting Method Based on Random Matrix Theory and CNN-LSTM Model," *2022 IEEE 2nd Int. Conf. Digit. Twins Parallel Intell. DTPI 2022*, pp. 1–6, 2022, doi: 10.1109/DTPI55838.2022.9998910.

[48] K. Zhu, Y. Li, W. Mao, F. Li, and J. Yan, "LSTM enhanced by dual-attention-based encoder-decoder for daily peak load forecasting," *Electr. Power Syst. Res.*, vol. 208, p. 107860, 2022, doi: 10.1016/j.epsr.2022.107860.

[49] A. Agga, A. Abbou, M. Labbadi, and Y. El Houm, "Short-Term Load Forecasting: Based on Hybrid CNN-LSTM Neural Network," *2021 6th Int. Conf. Power Renew. Energy, ICPRE 2021*, pp. 886–891, 2021, doi: 10.1109/ICPRE52634.2021.9635488.

[50] A. Altan and S. Karasu, "Crude oil time series prediction model based on LSTM network with chaotic Henry gas solubility optimization," *Energy*, vol. 242, p. 122964, 2022, doi: 10.1016/j.energy.2021.122964.

[51] W. Xiong, "Bus load forecasting based on maximum information coefficient and CNN-LSTM model," *2023 IEEE Int. Conf. Image Process. Comput. Appl.*, pp. 659–663, 2023, doi: 10.1109/ICIPCA59209.2023.10257944.

[52] O. Rubasinghe, X. Zhang, T. K. Chau, T. Fernando, and H. H. C. Iu, "A Novel Sequence to Sequence based CNN-LSTM Model for Long Term Load Forecasting," *Proc. - 2022 IEEE Sustain. Power Energy Conf. iSPEC 2022*, pp. 1–5, 2022, doi: 10.1109/iSPEC54162.2022.10033062.

[53] J. Yedukondalu, D. Sharma, and L. Dev, "Subject-Wise Cognitive Load Detection Using Time-Frequency EEG and Bi-LSTM," *Arab. J. Sci. Eng.*, 2023, doi: 10.1007/s13369-023-08494-1.

[54] C. Li, R. Hu, C. Y. Hsu, and Y. Han, "Short-term Power Load Forecasting based on Feature Fusion of Parallel LSTM-CNN," *2022 IEEE 4th Int. Conf. Power, Intell. Comput. Syst. ICPICS 2022*, no. 4, pp. 448–452, 2022, doi: 10.1109/ICPICS55264.2022.9873566.

[55] C. Ren, L. Jia, and Z. Wang, "A CNN-LSTM Hybrid Model Based Short-term Power Load Forecasting," *2021 Power Syst. Green Energy Conf.*, pp. 182–186, 2021, doi: 10.1109/PSGEC51302.2021.9542404.

[56] W. Qian, C. Gu, C. Zhu, Z. Jiang, B. Han, and M. Yu, "Short-term Load Forecasting Based on Multi-model Fusion of CNN-LSTM-LGBM," *Proc. - 2021 Int. Conf. Power Syst. Technol. Carbon Neutrality New Type Power Syst. POWERCON 2021*, pp. 934–939, 2021, doi: 10.1109/POWERCON53785.2021.9697619.

[57] X. Chen, M. Yang, Y. Zhang, J. Liu, and S. Yin, "Load Prediction Model of Integrated Energy System Based on CNN-LSTM," *2023 3rd Int. Conf. Energy Eng. Power Syst. EEPS 2023*, pp. 248–251, 2023, doi: 10.1109/EEPS58791.2023.10257124.

[58] X. Yang and Z. Chen, "A Hybrid Short-Term Load Forecasting Model Based on CatBoost and LSTM," *2021 IEEE 6th Int. Conf. Intell. Comput. Signal Process. ICSP 2021*, pp. 328–332, 2021, doi: 10.1109/ICSP51882.2021.9408768.

[59] H. Hua, M. Liu, Y. Li, S. Deng, and Q. Wang, "An ensemble framework for short-term load forecasting based on parallel CNN and GRU with improved ResNet," *Electr. Power Syst. Res.*, vol. 216, p. 109057, 2023, doi: 10.1016/j.epsr.2022.109057.

[60] A. P. Wibawa, A. B. P. Utama, H. Elmunsyah, U. Pujianto, F. A. Dwiyanto, and L. Hernandez, "Time-series analysis with smoothed Convolutional Neural Network," *J. Big Data*, vol. 9, no. 1, 2022, doi: 10.1186/s40537-022-00599-y.

[61] J. H. Kim, B. S. Lee, and C. H. Kim, "A Study on the development of long-term hybrid electrical load forecasting model based on MLP and statistics using massive actual data considering field applications," *Electron. Power Syst. Res.*, vol. 221, p. 109415, 2023, doi: 10.1016/j.epsr.2023.109415.

[62] S. S. Subbiah and J. Chinnappan, "Deep learning based short term load forecasting with hybrid feature selection," *Electr. Power Syst. Res.*, vol. 210, p. 108065, 2022, doi: 10.1016/j.epsr.2022.108065.

[63] I. C. Figueiró, A. R. Abaide, N. K. Neto, L. N. F. Silva, and L. L. C. Santos, "Bottom-Up Short-Term Load Forecasting Considering Macro-Region and Weighting by Meteorological Region," *Energies*, vol. 16, no. 19, pp. 1191–1198, 2023, doi: 10.3390/en16196857.

[64] M. Abdurohman and A. G. Putrada, "Forecasting Model for Lighting Electricity Load with a Limited Dataset using XGBoost," *Kinet. Game Technol. Inf. Syst. Comput. Network, Comput. Electron. Control*, vol. 8, no. 2, pp. 571–580, 2023, doi: 10.22219/kinetik.v8i2.1687.

[65] S. Ryu and Y. Yu, "Quantile-Mixer: A Novel Deep Learning Approach for Probabilistic Short-term Load Forecasting," *IEEE Trans. Smart Grid*, vol. 15, no. 2, pp. 2237–2250, 2023, doi: 10.1109/TSG.2023.3290180.

[66] I. Diahovchenko, A. Chuprun, and Z. Čonka, "Assessment and mitigation of the influence of rising charging demand of electric vehicles on the aging of distribution transformers," *Electr. Power Syst. Res.*, vol. 221, no. May, 2023, doi: 10.1016/j.epsr.2023.109455.

[67] M. Zhong *et al.*, "Dissolved gas in transformer oil forecasting for transformer fault evaluation based on HATT-RLSTM," *Electr. Power Syst. Res.*, vol. 221, 2023, doi: 10.1016/j.epsr.2023.109431.

[68] P. Ran, K. Dong, X. Liu, and J. Wang, "Short-term load forecasting based on CEEMDAN and Transformer," *Electr. Power Syst. Res.*, vol. 214, p. 108885, 2023, doi: 10.1016/j.epsr.2022.108885.

[69] Z. Tian, W. Liu, W. Jiang, and C. Wu, "CNNs-Transformer based day-ahead probabilistic load forecasting for weekends with limited data availability," *Energy*, vol. 293, p. 130666, 2024, doi: 10.1016/j.energy.2024.130666.

[70] Z. Tian, W. Liu, W. Jiang, and C. Wu, "CNNs-Transformer based day-ahead probabilistic load forecasting for weekends with limited data availability," *Energy*, vol. 293, p. 130666, 2024, doi: 10.1016/j.energy.2024.130666.

[71] Y. Shang *et al.*, "Loss of life estimation of distribution transformers considering corrupted AMI data recovery and field verification," *IEEE Trans. Power Deliv.*, vol. 36, no. 1, pp. 180–190, 2021, doi: 10.1109/TPWRD.2020.2978809.

[72] H. Zhao, Y. Wu, L. Ma, and S. Pan, "Spatial and Temporal Attention-Enabled Transformer Network for Multivariate Short-Term Residential

Load Forecasting," *IEEE Trans. Instrum. Meas.*, vol. 72, pp. 1–11, 2023, doi: 10.1109/TIM.2023.3305655.

[73] W. Zeng *et al.*, "Hybrid CEEMDAN-DBN-ELM for online DGA serials and transformer status forecasting," *Electr. Power Syst. Res.*, vol. 217, 2023, doi: 10.1016/j.epsr.2023.109176.

[74] C. Xu and G. Chen, "Interpretable transformer-based model for probabilistic short-term forecasting of residential net load," *Int. J. Electr. Power Energy Syst.*, vol. 155, p. 109515, 2024, doi: 10.1016/j.ijepes.2023.109515.

[75] K. Qu, G. Si, Z. Shan, Q. Wang, X. Liu, and C. Yang, "Forwardformer : Efficient Transformer With Multi-Scale Forward Self-Attention for Day-Ahead Load Forecasting," *IEEE Trans. Power Syst.*, vol. 39, no. 2, pp. 1421–1433, 2024, doi: 10.1109/TPWRS.2023.3266369.

[76] Z. Tian, W. Liu, W. Jiang, and C. Wu, "CNNs-Transformer based day-ahead probabilistic load forecasting for weekends with limited data availability ☆," *Energy*, vol. 293, p. 130666, 2024, doi: 10.1016/j.energy.2024.130666.

[77] H. Ma, P. Yang, F. Wang, X. Wang, D. Yang, and B. Feng, "Short-Term Heavy Overload Forecasting of Public Transformers Based on Combined LSTM-XGBoost Model," *Energies*, vol. 16, no. 3, 2023, doi: 10.3390/en16031507.

[78] I. U. Khalil, A. Ul Haq, and N. ul Islam, "A deep learning-based transformer model for photovoltaic fault forecasting and classification," *Electr. Power Syst. Res.*, vol. 228, p. 110063, 2024, doi: 10.1016/j.epsr.2023.110063.

[79] S. Luo, Y. Rao, J. Chen, H. Wang, and Z. Wang, "Short-Term Load Forecasting Model of Distribution Transformer Based on CNN and LSTM," *7th IEEE Int. Conf. High Volt. Eng. Appl. ICHVE 2020 - Proc.*, pp. 1–4, 2020, doi: 10.1109/ICHVE49031.2020.9279813.

[80] C. Wang, Y. Wang, Z. Ding, and K. Zhang, "Probabilistic Multi-energy Load Forecasting for Integrated Energy System Based on Bayesian Transformer Network," *IEEE Trans. Smart Grid*, vol. 15, no. 2, pp. 1495–1508, 2023, doi: 10.1109/TSG.2023.3296647.

[81] S. Cen and C. G. Lim, "Multi-Task Learning of the PatchTCN-TST Model for Short-Term Multi-Load Energy Forecasting Considering Indoor Environments in a Smart Building," *IEEE Access*, vol. 12, pp. 19553–19568, 2024, doi: 10.1109/ACCESS.2024.3355448.

[82] M. Grabner, Y. Wang, Q. Wen, B. Blazic, and V. Struc, "A Global Modeling Framework for Load Forecasting in Distribution Networks," *IEEE Trans. Smart Grid*, vol. 14, no. 6, pp. 4927–4941, 2023, doi: 10.1109/TSG.2023.3264525.

[83] Q. Liu, J. Cao, J. Zhang, Y. Zhong, T. Ba, and Y. Zhang, "Short-Term Power Load Forecasting in FGSM-Bi-LSTM Networks Based on Empirical Wavelet Transform," *IEEE Access*, vol. 11, pp. 105057–105068, 2023, doi: 10.1109/ACCESS.2023.3316516.

[84] I. Diahovchenko, R. G. Olsen, V. Zalotov, A. Chuprun, B. Dolník, and O. Boiko, "Development of effective shielding against electricity meters tampering with strong magnetic fields," *Electr. Power Syst. Res.*, vol. 213, p. 108722, 2022, doi: 10.1016/j.epsr.2022.108722.

[85] W. Li and K. L. E. Law, "Deep Learning Models for Time Series Forecasting: A Review," *IEEE Access*, vol. 12, pp. 92306–92327, 2024, doi: 10.1109/ACCESS.2024.3422528.

[86] T. A. Nguyen and T. N. Tran, "Improving Short-Term Electrical Load Forecasting with Dilated Convolutional Neural Networks: A Comparative Analysis," *J. Robot. Control*, vol. 6, no. 2, pp. 560–569, 2025, doi: 10.18196/jrc.v6i2.24967.