2026 29th International Conference on Computer Supported Cooperative Work in Design (CSCWD) | 979-8-3315-8470-2/26/$31.00 ©2026 IEEE | DOI: 10.1109/CSCWD68734.2026.11581548

A Hybrid TCN-Mamba Model for Power Load Forecasting

Mingjie Xie, Xinyao Yan, Ye Cao, Guosheng Kang✉, Xiaoqing Chen, Guotai He
[1]Hunan Provincial Key Lab. for Services Computing and Novel Software Technology, HNUST, Xiangtan, Hunan, China
[2]School of Computer Science and Engineering, Hunan University of Science and Technology, Xiangtan, China
Email: {mingjiexie.trch, xinyaoyanhn, caoye6233, guoshengkang, xiaoqingchen696, guotaihe3}@gmail.com

***Abstract*—Accurate short-term power load forecasting is pivotal for efficient energy management, grid stability, and operational cost optimization in modern power systems. However, existing deep learning models face a fundamental trade-off between capturing local temporal patterns and modeling long-range dependencies, limiting their forecasting accuracy and computational efficiency. To address this challenge, we propose a novel hybrid TCN-Mamba architecture that synergistically combines Temporal Convolutional Networks (TCN) for parallelizable local feature extraction with Mamba's selective state space mechanism for efficient long-sequence modeling. Our framework achieves linear computational complexity while effectively capturing both short-term fluctuations and long-term temporal correlations in power load data. Comprehensive experiments on the UCI LD2011_2014 Dataset demonstrate that the proposed model consistently outperforms state-of-the-art baselines (BiLSTM, LSTM-Transformer, and CNN-LSTM-Attention), achieving the lowest MAE (0.0173) and NMAE (0.0204), with improvements of up to 14.36% and 17.41%, respectively. Ablation studies confirm that the TCN component enhances sensitivity to local temporal dynamics, while Mamba predominantly contributes to global dependency learning. These results validate that our hybrid architecture effectively balances accuracy, efficiency, and scalability, providing a robust foundation for intelligent energy management systems.**

***Index Terms*—Power Load Forecasting, Temporal Convolutional Network (TCN), Mamba, State Space Model (SSM), Hybrid Neural Network, Long-Sequence Modeling.**

I. Introduction

Short-term power load forecasting plays a crucial role in modern energy management systems, supporting tasks such as economic dispatch, grid stabilization, and intelligent control across buildings and district-level infrastructures. Even marginal improvements in forecasting accuracy (e.g., 0.01%) yield substantial economic benefits for utilities and contribute to more efficient HVAC operation, thus reducing energy waste caused by conservative control strategies [1]. Despite its importance, short-term load prediction remains challenging due to the intricate temporal characteristics of power consumption, including daily fluctuations associated with human routines, long-term seasonal variations, and stochastic disturbances induced by occupant behaviors and weather conditions such as temperature and solar radiation [2].

Traditional statistical models, such as ARIMA and SARIMA, have been widely used because of their transparency and ease of deployment [3]. However, these approaches struggle to capture nonlinear patterns and long-range dependencies inherent in electricity load profiles [4]. To address these limitations, machine learning (ML) techniques, including Support Vector Regression, Random Forest, and XGBoost, have been applied to enhance nonlinear mapping [5]. Nonetheless, they often depend heavily on handcrafted features and exhibit poor robustness under dynamic load conditions [6].

With the rapid advances in deep learning (DL), neural network–based approaches such as RNN, GRU, and LSTM have demonstrated strong abilities in modeling temporal dependencies in multivariate load sequences [7]. LSTM-based variants, including BiLSTM architectures [8], further improve prediction accuracy through bidirectional temporal modeling. However, sequential computation prevents parallelism, leading to training latency, and the vanishing gradient problem limits their performance on long sequences [9]. Hybrid frameworks such as CNN–LSTM-Attention [10], and VMD–GRU–TCN [11] combine local feature extraction with sequence modeling, yet they still struggle to capture both global and local dependencies.

More recently, Transformer-based architectures have emerged as strong contenders due to their globally attentive modeling capabilities. Models such as the Temporal Fusion Transformer (TFT) [12] and attention-augmented CNN–LSTM variants show promising performance in multivariate load forecasting. Nonetheless, the quadratic complexity of self-attention significantly restricts their applicability for long sequences and real-time deployment [9]. On the other hand, Temporal Convolutional Networks (TCNs) [9] offer efficient parallel computation and large receptive fields via dilated causal convolutions, but they may lose contextual continuity when dilation grows excessively [4].

These observations point to a fundamental methodological trade-off, where models optimized for *local* pattern extraction (e.g., CNNs, TCNs) often lack the ability to capture *global* long-range dependencies, while global modeling architectures (e.g., LSTM, Transformer) typically incur high computational costs and exhibit scalability issues.

To address these challenges, we propose a hybrid forecasting framework unifying the strengths of both modeling paradigms. The TCN–Mamba architecture leverages dilated causal convolutions for efficient extraction of short-term load patterns and integrates the Mamba selective state space model (SSM) [4] for scalable long-range dependency modeling. The selective mechanism in Mamba dynamically filters contextual information, enabling it to emphasize meaningful temporal

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

979-8-3315-8470-2/26/$31.00 ©2026 IEEE Authorized licensed use limited to: Mingjie Xie. Downloaded on August 27,2026 at 14:20:19 UTC from IEEE Xplore. Restrictions apply.

interactions and suppress noise and irrelevant fluctuations. This design makes it suitable for complex power load forecasting. The contributions of this paper are summarized as follows.

- A hybrid modeling approach is proposed, where a TCN–Mamba architecture leverages dilated causal convolutions and state-space modeling. This enables efficient extraction of short-term temporal dynamics while preserving long-range dependencies with linear complexity.
- A coordinated architectural interaction mechanism is developed, with TCN acting as a multi-scale temporal encoder and Mamba as a selective global dependency learner. This complementary architecture is tailored to the characteristics of power load sequences.
- A forecasting framework is constructed, incorporating data preprocessing, feature integration, multi-step prediction optimization, and a multi-scale loss formulation. These components form a robust pipeline for short-term power load forecasting.

The remainder of this paper is structured as follows. Section II reviews the related work on TCN and Mamba, which form the foundation of our framework. Section III details the proposed hybrid architecture. Section IV presents experiments and analysis. Section V concludes the work.

II. Framework of TCN–Mamba Based Power Load Forecasting

The TCN-Mamba framework comprises three core modules, namely *Sample Processing*, *Hybrid TCN–Mamba Load Forecasting*, and *Load Forecasting Optimization*, as shown in Fig. 1. First, *Sample Processing* formalizes the forecasting task, predicting future load values from a historical input window. This module ensures data quality and feature richness via preprocessing steps like Min-Max normalization, missing value handling, and integrating external features to capture deterministic temporal influences. Second, the core prediction engine is the *Hybrid TCN–Mamba Model*. It combines the parallelizable TCN component (utilizing dilated causal convolutions and residual blocks) to efficiently capture high-frequency local patterns, with the Mamba Selective State Space Model (SSM). The Mamba component is critical for scalable long-range dependency modeling, leveraging its input-conditioned state transition to selectively filter contextual information, a significant advantage over quadratic-complexity Transformer models. Finally, the *Load Forecasting Optimization* module ensures the model's robust performance, particularly for point-wise load prediction. This is achieved through a stable training strategy and a loss function that encourages high accuracy for immediate load fluctuations. This integrated system provides a comprehensive, efficient, and stable foundation for intelligent energy management.

III. TCN–Mamba Hybrid Load Forecasting Model

This section presents the proposed TCN–Mamba forecasting model, which consists of three main components. 1) **Sample Processing** details the procedures for transforming raw data into structured sequences, including normalization and feature integration; 2) **Hybrid TCN–Mamba Load Forecasting** describes the integrated model design, where TCN captures local dynamics and Mamba models long-range dependencies; 3) **Load Forecasting Optimization** outlines the techniques used to improve forecasting accuracy and model stability.

A. Sample Processing

The raw power load data is denoted as a time series $L = \{l_1, l_2, \ldots, l_T\}$, where l_t represents the load value at time step t. To ensure the data quality and suitability for the TCN-Mamba model, we implement a rigorous processing pipeline comprising data cleaning, feature integration, normalization, and sequence construction.

Data Cleaning and Feature Integration. Real-world load data often contains noise and irregularities. We first address missing values using linear interpolation and correct statistical outliers based on the interquartile range (IQR) criterion.

Normalization. To eliminate dimensional disparities between load values and external features, and to accelerate model convergence, we apply Min-Max normalization to scale the data into the range $[0, 1]$. The normalization is defined as shown in Formula (1), where x is the original value, and $x_{\min}$ and $x_{\max}$ are the minimum and maximum.

$$\tilde{x} = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{1}$$

Sequence Construction. The normalized time series is transformed into supervised learning samples using a sliding window mechanism. We construct fixed-length input sequences $X_t = \{x_{t-W+1}, \ldots, x_t\}$ with a historical window size of $W = 28$ (corresponding to 7 hours of 15-minute resolution data) to predict the load y_{t+1} at the next time step.

B. Hybrid TCN–Mamba Load Forecasting

The proposed hybrid model synergistically integrates *TCN* and the *Mamba* to handle the complex temporal characteristics of power loads. As illustrated in Fig. 1, the architecture follows a hierarchical design: the TCN module first extracts high-frequency local variations and short-term trends from the input sequence X, generating a dense feature representation. Subsequently, the Mamba module models the long-range dependencies and global correlations based on these features.

1) TCN

TCN is composed of three fundamental components—causal convolution, dilated convolution, and residual blocks. The dilated causal convolution used in TCN is obtained by introducing dilation into the causal convolution, allowing the model to enlarge its receptive field while strictly preserving temporal causality.

Causal Convolution. TCN enforces temporal causality by ensuring that the output at time t depends on current and past inputs. Given convolution kernel size n, causal convolution is defined as shown in Formula (2), where $f(i)$ is the i-th kernel weight and x_{t-i} denotes the input at lag i. This

Authorized licensed use limited to: mingjie Xie. Downloaded on August 27,2026 at 14:20:19 UTC from IEEE Xplore. Restrictions apply.

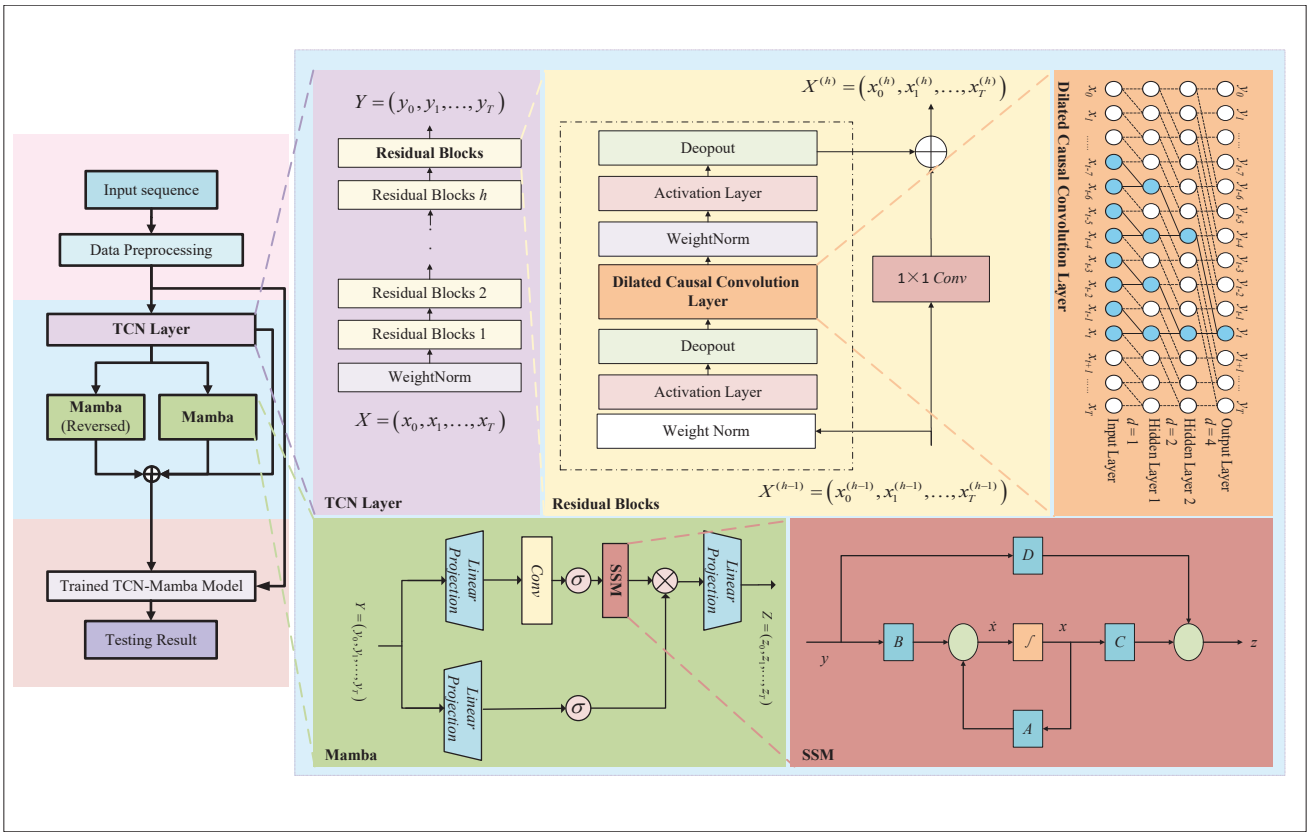

Fig. 1. Overall framework of TCN-Mamba, the left side of the figure presents the overall architecture of our model. The right side of the figure details the components of the TCN-Mamba Block.

formulation preserves the input length through padding and eliminates future-leakage, facilitating reliable extraction.

$$y_t = \sum_{i=0}^{n-1} f(i)\, x_{t-i} \tag{2}$$

Dilated Convolution. To expand the receptive field without increasing network depth, TCN introduces dilation, allowing kernels to skip intermediate inputs. The dilated convolution is defined as shown in Formula (3), where d is the dilation factor. This operation enables each layer to aggregate over progressively larger spans. The effective receptive field for L layers is expressed in Formula (4), capturing multi-scale patterns as dilation grows exponentially across layers, which is beneficial for modeling daily or weekly behaviors.

$$y_t = \sum_{i=0}^{n-1} f(i)\, x_{t-d\cdot i} \tag{3}$$

$$R = 1 + (n-1)\sum_{l=0}^{L-1} 2^l \tag{4}$$

A Dilated Causal Convolution is obtained by augmenting the causal convolution with a dilation factor, thereby expanding the receptive field without violating temporal causality.

Residual Block. To stabilize deep architectures, each TCN block incorporates a residual connection that projects the input and adds it to the transformed representation, as shown in Formula (5), where $F(\cdot)$ refers to the dilated convolutional processing sequence with normalization and dropout, and $P(\cdot)$ is a 1×1 projection used to synchronize channel dimensions. This mechanism facilitates effective gradient transmission and ensures that short-term temporal details remain available to deeper layers for hierarchical feature extraction.

$$X^{(h)} = \mathrm{ReLU}\Big(F(X^{(h-1)}) + P(X^{(h-1)})\Big) \tag{5}$$

2) Mamba

While TCN effectively extracts short-term dependencies, long-range temporal structures require a more expressive and efficient mechanism. The Mamba module addresses this through an input-conditioned State Space Model (SSM) that couples continuous-time dynamics with a selective mechanism enabling content-aware state transitions.

Continuous-Time SSM Formulation. The backbone of Mamba is a continuous-time SSM defined in Formula (6) and Formula (7), where $x(t)$ is the input, $h(t)$ is the hidden state, and A, B, and C are system matrices. Formula 6 describes how the hidden state evolves based on current dynamics, while Formula 7 decodes the hidden state into the output.

$$h'(t) = Ah(t) + Bx(t) \tag{6}$$

$$y(t) = Ch(t) \tag{7}$$

Zero-Order Hold Discretization. For discrete-time forecasting, the continuous system is discretized using zero-order

Authorized licensed use limited to: mingjie Xie. Downloaded on August 27,2026 at 14:20:19 UTC from IEEE Xplore. Restrictions apply.

hold, giving the discrete-time SSM shown in Formula (8) and Formula (9). The discrete-time matrices $\bar{A}$ and $\bar{B}$ are computed as shown in Formula (10) and Formula (11). With sampling interval Δ, these equations ensure stable state propagation, enabling Mamba to effectively model slow variations and long-term seasonal structures.

$$h_t = \bar{A}h_{t-1} + \bar{B}x_t \tag{8}$$

$$y_t = Ch_t \tag{9}$$

$$\bar{A} = \exp(\Delta A) \tag{10}$$

$$\bar{B} = A^{-1}\left(\exp(\Delta A) - I\right)\Delta B \tag{11}$$

Selective Mechanism. A distinctive feature of Mamba is its input-conditioned modulation mechanism, formulated in Formula (12), where W_Δ and b_Δ are learnable parameters and $\mathrm{Broadcast}_D(\cdot)$ expands the scalar to match the hidden dimension D. This mechanism allows the model to selectively enhance or suppress temporal transitions, enabling content-aware long-range dependency modeling that is especially useful for extreme or anomalous load patterns.

$$s_\Delta(x) = \mathrm{Broadcast}_D(W_\Delta x + b_\Delta) \tag{12}$$

C. Load Forecasting Optimization

Hyperparameter Tuning. A grid search is performed to determine the optimal hyperparameters based on the jointly best MAE, NMAE, PMAE, and RMSE. The optimal input window length is obtained at $W = 28$, while the Temporal Convolutional Network (TCN) achieves its best performance with $K = 2$ residual blocks and a kernel size of $n = 7$. For the Mamba component, the optimal state dimension is found to be $D_h = 4$. A dropout rate of 0.2 is applied throughout the model to mitigate overfitting.

Training Strategy. Training uses the Adam optimizer with a learning rate of 1×10^{-3} and a batch size of 64. The model follows a standard supervised training paradigm for one-step-ahead forecasting. For each training sample, the model maps the input sequence $X_t = \{x_{t-W+1}, \ldots, x_t\}$ to the ground-truth value y_{t+1} at the next time step. This strategy eliminates the risk of error accumulation associated with recursive multi-step forecasting and focuses on maximizing the precision of immediate load fluctuations.

Loss Function. To ensure accurate point-wise prediction and robust convergence, we utilize the Mean Absolute Error (MAE) as our primary loss function. As shown in Formula (13), MAE measures the average absolute difference between the predicted value $\hat{y}_i$ and the actual ground truth y_i. This loss is particularly effective for short-term power load forecasting as it provides stable gradients and is less sensitive to stochastic noise compared to quadratic loss functions.

$$\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i| \tag{13}$$

IV. Experiments

A. Experimental Setup

Dataset Description. We evaluate TCN-Mamba on the publicly available UCI LD2011_2014 Dataset[1], which provides 15-minute resolution electricity consumption of 370 residential users from 2011 to 2015. Following standard practice for short-term load forecasting, the data is chronologically split into training (2012), validation (January 2013), and testing (February 2013) sets to prevent information leakage. Each sample includes historical load (kW) and calendar-based features such as hour-of-day, day-of-week, and holiday indicators.

Evaluation Metrics. Five evaluation metrics are used to comprehensively assess model performance: MAE, NMAE, PMAE, and RMSE. Their mathematical definitions are provided below, with each metric referenced individually.

a) *Mean Absolute Error (MAE)*. MAE measures the average magnitude of prediction errors and is defined in Formula (14).

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i| \tag{14}$$

b) *Normalized Mean Absolute Error (NMAE)*. NMAE normalizes prediction error by the true load and is shown in Formula (15).

$$\mathrm{NMAE} = \frac{\sum_{i=1}^{N}|y_i - \hat{y}_i|}{\sum_{i=1}^{N} y_i} \tag{15}$$

c) *Root Mean Square Error (RMSE)*. RMSE emphasizes larger errors by applying a quadratic penalty, as shown in Formula (16).

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2} \tag{16}$$

d) *Normalized Root Mean Square Error (NRMSE)*. NRMSE is a normalized version of RMSE that facilitates comparison across datasets with scales. It is defined in Formula (17).

$$\mathrm{NRMSE} = \frac{\sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}}{y_{\max} - y_{\min}} \tag{17}$$

e) *Peak Mean Absolute Error (PMAE)*. PMAE evaluates performance on high-load periods, focusing on peak accuracy. Its definition is given in Formula (18), where $\mathcal{P}$ denotes the set of samples corresponding to the top 10% true load values.

$$\mathrm{PMAE} = \frac{1}{|\mathcal{P}|}\sum_{i\in\mathcal{P}}|y_i - \hat{y}_i| \tag{18}$$

These metrics collectively evaluate absolute deviation, percentage deviation, peak-period performance, and scale-invariant error characteristics.

Baseline Methods. We compare the proposed TCN–Mamba model with several representative baselines, including BiLSTM [8], LSTM-Transformer [13], CNN-LSTM-Attention [10], LSTM-RNN [14], XGBoost [5], and CNN1D [15].

[1]https://archive.ics.uci.edu/dataset/321/electricityloaddiagrams20112014

Authorized licensed use limited to: mingjie Xie. Downloaded on August 27,2026 at 14:20:19 UTC from IEEE Xplore. Restrictions apply.

B. Performance Comparison

The purpose of this experiment is to evaluate the proposed TCN–Mamba model's capability to achieve accurate and stable load forecasting, particularly in comparison with representative recurrent, convolutional, attention-based, and machine-learning baselines. As shown in Table I, TCN–Mamba achieves the best overall performance across all evaluation metrics. First, the model obtains the lowest MAE (0.0173), outperforming the best baseline by 14.36%, which demonstrates its superior ability to capture fine-grained load fluctuations; this gain can be attributed to the TCN module, whose dilated causal convolutions effectively extract multi-scale temporal features. Second, TCN–Mamba achieves the lowest NMAE (0.0204), indicating strong robustness and stable prediction quality across different load magnitudes; this improvement is supported by the synergistic hybrid architecture that maintains stable mapping of complex temporal correlations within historical sequences. Third, the model yields the lowest PMAE (0.2040), verifying its capability to forecast peak-load periods that are typically challenging for conventional architectures; this advantage arises from the complementary interaction between TCN and Mamba, which enables rapid response while maintaining long-range contextual awareness. Finally, TCN–Mamba attains the lowest RMSE (0.0390), highlighting its effectiveness in learning stable global representations of nonlinear load dynamics; this improvement mainly benefits from the selective state-space modeling of the Mamba component, which captures long-range dependencies with high efficiency. Collectively, these results confirm that TCN–Mamba integrates local pattern extraction and long-range temporal dependency modeling more effectively than recurrent, convolutional, and attention-based baselines. The consistent performance gains across all metrics underscore the architectural advantages of the proposed model and establish TCN–Mamba as an accurate robust, and scalable solution for real-world power load forecasting applications.

TABLE I. Performance comparison with baseline models

Model	MAE	NMAE	PMAE	RMSE
BiLSTM [8]	0.0211	0.0256	0.2358	0.0452
LSTM-Transformer [13]	0.0215	0.0261	0.2294	0.0416
CNN-LSTM-Attention [10]	0.0237	0.0288	0.2540	0.0438
LSTM-RNN [14]	0.0428	0.0417	0.2445	0.0594
XGBoost [5]	0.0254	0.0280	0.2290	0.0452
CNN1D [15]	0.0299	0.0254	0.2139	0.0452
TCN-only [9]	0.0248	0.0304	0.2666	0.0462
Mamba-only [4]	0.0202	0.0247	0.2169	0.0423
TCN-Mamba	**0.0173**	**0.0204**	**0.2040**	**0.0390**
Improv. (%)	**14.36%**	**17.41%**	**4.63%**	**6.25%**

C. Ablation Analysis

The ablation results shown in Fig. 2 verify the model's ability to capture both fine-grained local temporal patterns and long-range dependencies, thereby confirming the effectiveness of the hybrid TCN-Mamba architecture. For MAE, the full TCN-Mamba model achieves the lowest error (0.0173); removing Mamba increases it by 43.6%, and removing TCN raises it to 0.0202, indicating that Mamba is crucial for long-range modeling while TCN ensures accurate short-term fluctuations. For NMAE, the hybrid model scores 0.0204, outperforming TCN-only (0.0304) and Mamba-only (0.0247), suggesting stable performance across varying loads due to TCN's dense local encoding enhancing Mamba's global context. For PMAE, the model reaches 0.2040, with TCN or Mamba removal significantly degrading performance, highlighting the need for both rapid local response and long-horizon filtering via dilated causal convolutions and the selective state-space mechanism. For RMSE, the full model achieves 0.0390, demonstrating reduced large deviations and stable long-range predictions, attributed to Mamba's input-conditioned state transition. Overall, the ablation confirms that TCN–Mamba integrates local and global temporal modeling in a mutually reinforcing manner, providing accuracy, robustness, and generalization for intelligent energy management and long-sequence forecasting.

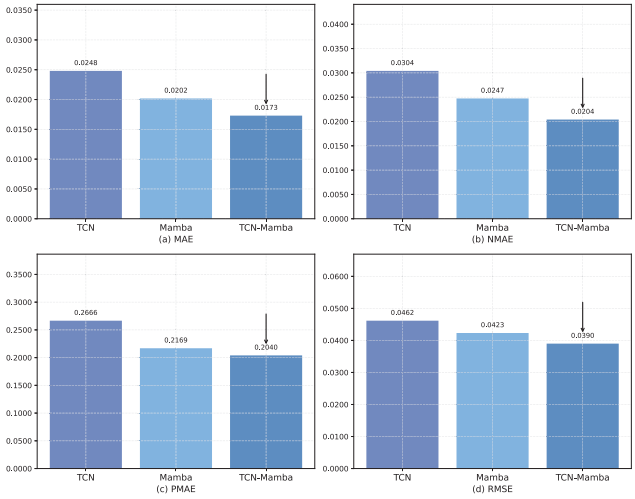

Fig. 2. Ablation study

D. Impact of Hyperparameter

This analysis systematically evaluates the influence of critical TCN and Mamba hyperparameters to verify the model's ability to efficiently balance local and long-range dependency modeling. As shown in Fig. 3, the optimal configuration is consistently aligned with computational efficiency, which is a crucial factor for real-time energy management systems. Firstly, the lowest MAE within the state dimension sweep (0.0218) is achieved at a remarkably low Mamba State Space Dimension ($D_h = 4$). This low D_h requirement significantly reduces the computational overhead and memory footprint during both training and inference, highlighting the high representational efficiency attributed to the Mamba's Selective State Space Model (SSM) design, which intelligently filters irrelevant input. Secondly, the TCN performs optimally at a shallow depth of K=2 and a moderate kernel size of n=7; this configuration effectively balances the trade-off between capturing short-term dynamics and maintaining model complexity. This result demonstrates robust local feature extraction

Authorized licensed use limited to: mingjie Xie. Downloaded on August 27,2026 at 14:20:19 UTC from IEEE Xplore. Restrictions apply.

attributed to TCN's effective dilated convolutions. Finally, the minimum MAE (0.0200) is found at an input Sliding Window Length ($W = 28$). This suggests that capturing 7 hours of historical data is crucial for accurate forecasting, successfully integrating local temporal dynamics. This excellent modeling capability is attributed to Mamba's $O(L)$ linear-time complexity, enabling the effective integration of input sequences without prohibitive cost. In summary, TCN-Mamba synergy confirms its potential as an efficient, accurate, and computationally viable foundation model for time series forecasting in critical domains like energy management.

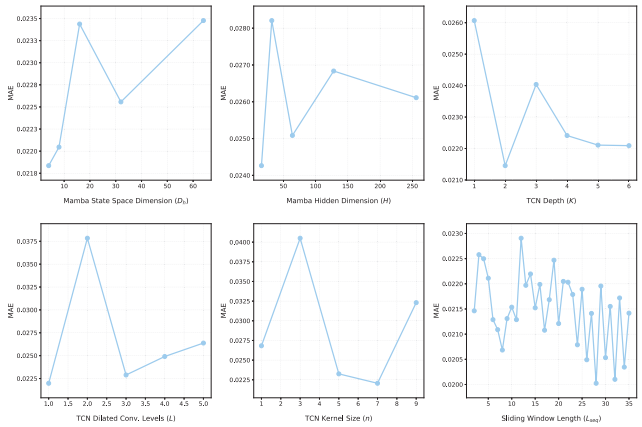

Fig. 3. The impact of hyperparameters

V. Conclusion

In this paper, we introduced the TCN-Mamba architecture for short-term power load forecasting, unifying the TCN for local pattern extraction and Mamba's selective state-space mechanism for efficient long-range dependency modeling. This hybrid design resolves the fundamental trade-off by achieving linear $O(L)$ complexity and superior performance. Experiments on the UCI LD2011_2014 Dataset demonstrated it outperforms state-of-the-art baselines, achieving the lowest MAE (0.0173). Hyperparameter analysis confirmed its operational efficiency, showing optimal performance at a low Mamba state dimension ($D_h = 4$). TCN-Mamba provides an accurate and efficient solution for intelligent energy management. Future work includes integrating external variables and exploring dynamic multi-regional forecasting environments.

Acknowledgment

This work is partially supported by the Hunan Provincial Natural Science Foundation of China under Grant No: 2026JJ50519. *Guosheng Kang* is the corresponding author.

References

[1] Z. Liao, H. Pan, X. Huang, R. Mo, X. Fan, H. Chen, L. Liu, and Y. Li, "Short-term load forecasting with dense average network," *Expert Systems with Applications*, vol. 186, p. 115748, 2021.

[2] G. Kang, G. He, Y. Tang, J. Xu, X. You, and X. Xiao, "Ring matrix encoding-based global optimization for fault recovery of distribution network with distributed generation," *Expert Systems with Applications*, p. 130998, 2025.

[3] N. Ayub, M. Irfan, M. Awais, U. Ali, T. Ali, M. Hamdi, A. Alghamdi, and F. Muhammad, "Big data analytics for short and medium-term electricity load forecasting using an ai techniques ensembler," *Energies*, vol. 13, no. 19, p. 5193, 2020.

[4] A. Gu and T. Dao, "Mamba: Linear-time sequence modeling with selective state spaces," in *First conference on language modeling*, 2024.

[5] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 2016, pp. 785–794.

[6] E. Aguilar Madrid and N. Antonio, "Short-term electricity load forecasting with machine learning," *Information*, vol. 12, no. 2, p. 50, 2021.

[7] A. Azeem, I. Ismail, S. M. Jameel, and V. R. Harindran, "Electrical load forecasting models for different generation modalities: a review," *IEEe Access*, vol. 9, pp. 142 239–142 263, 2021.

[8] N. Mounir, H. Ouadi, and I. Jrhilifa, "Short-term electric load forecasting using an emd-bi-lstm approach for smart grid energy management system," *Energy and Buildings*, vol. 288, p. 113022, 2023.

[9] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," *arXiv preprint arXiv:1803.01271*, 2018.

[10] A. Wan, Q. Chang, A.-B. Khalil, and J. He, "Short-term power load forecasting for combined heat and power using cnn-lstm enhanced by attention mechanism," *Energy*, vol. 282, p. 128274, 2023.

[11] C. Cai, Y. Li, Z. Su, T. Zhu, and Y. He, "Short-term electrical load forecasting based on vmd and gru-tcn hybrid network," *Applied Sciences*, vol. 12, no. 13, p. 6647, 2022.

[12] P. C. Huy, N. Q. Minh, N. D. Tien, and T. T. Q. Anh, "Short-term electricity load forecasting based on temporal fusion transformer model," *Ieee Access*, vol. 10, pp. 106 296–106 304, 2022.

[13] B. Lim, S. Ö. Arik, N. Loeff, and T. Pfister, "Temporal Fusion Transformers for interpretable multi-horizon time series forecasting," *International Journal of Forecasting*, vol. 37, no. 4, pp. 1748–1764, 2021.

[14] B. Gunay, W. Shen, and G. Newsham, "Inverse blackbox modeling of the heating and cooling load in office buildings," *Energy and Buildings*, vol. 142, pp. 200–210, 2017.

[15] X. Dong, L. Qian, and L. Huang, "Short-term load forecasting in smart grid: A combined cnn and k-means clustering approach," in *2017 IEEE international conference on big data and smart computing (BigComp)*.

Authorized licensed use limited to: mingjie Xie. Downloaded on August 27,2026 at 14:20:19 UTC from IEEE Xplore. Restrictions apply.

IEEE, 2017, pp. 119–125.

Authorized licensed use limited to: mingjie Xie. Downloaded on August 27,2026 at 14:20:19 UTC from IEEE Xplore. Restrictions apply.