Automation and Machine Learning (2026)
Clausius Scientific Press, Canada

DOI: 10.23977/autml.2026.070204
ISSN 2516-5003 Vol. 7 Num. 2

A Smart Control Method for Short-Term Load of Electricity Meters Based on TCN-BiGRU Attention Network and Model Predictive Control

Zhuoyue Xu^{1,a}, Wencan Huang^{1,b,*}

^{1}TUT Maritime College, Tianjin University of Technology, Tianjin, China
^{a}15932413633@163.com, ^{b}cangreg@126.com
**Corresponding author*

Keywords: Temporal Convolutional Network; Bidirectional Gated Recurrent Unit; Short-Term Load Forecasting; Model Predictive Control

Abstract: To address the problems of prediction lag, insufficient identification of sudden load changes, and untimely response of control strategies in short-term load control of electricity meters, this paper proposes an intelligent load control method based on Temporal Convolutional Network (TCN), Bidirectional Gated Recurrent Unit (BiGRU), Attention Mechanism (AM), and Model Predictive Control (MPC). This method takes the historical load sequence of the electricity meter as input. First, it uses the TCN to extract local fluctuation and periodic variation features. Then, it uses BiGRU to model the bidirectional temporal dependency of the load sequence and combines AM to enhance the expression of key time step information. Based on this, the predicted load is input into the model predictive control module for rolling optimization to generate real-time control commands. Experimental results show that the proposed method outperforms the comparative model in terms of prediction accuracy, error stability, and load control effect, effectively reducing peak load fluctuations and improving the real-time performance and reliability of electricity meter load regulation.

1. Introduction

With the continuous advancement of smart grid and new power system construction [1], dynamic sensing, prediction, and control of power load have gradually become important research directions in the field of energy internet [2-3]. As a crucial component of power system terminal sensing equipment, smart meters have been widely deployed in residential, industrial, and commercial electricity consumption scenarios [4-5]. Their high-frequency sampling capabilities can continuously generate a large amount of time-series load data, providing a data foundation for refined load management. Against the backdrop of deepening "dual-carbon" goals and demand-side response mechanisms, the power grid operating environment exhibits characteristics such as expanded distributed energy access, increased load fluctuations, and more complex electricity consumption behaviors. Traditional load control methods relying on static rules and empirical thresholds are no longer sufficient to meet the real-time, accuracy, and intelligence requirements of modern power grids. Therefore, how to fully exploit the time-series correlation characteristics in smart meter load data and construct intelligent load control methods with predictive and control synergy capabilities has become an important aspect of current power system intelligence research.

Currently, researchers have proposed numerous methods based on statistical analysis and machine learning to address the problem of short-term power load prediction. Traditional time series methods using the Autoregressive Integrated Moving Average (ARIMA) model are effective for handling stationary load sequences, but their adaptability to the nonlinear, non-stationary, and abrupt load scenarios prevalent in real-world power grids is weak [6-7]. Subsequently, shallow machine learning models such as Support Vector Machines (SVM) and Back Propagation Neural Networks (BPNN) have been increasingly applied to load forecasting tasks [8-9]. While these have improved nonlinear fitting capabilities to some extent, their performance is highly dependent on manual feature engineering, making it difficult to automatically extract deep features from complex time series data. However, existing research still has some shortcomings: on the one hand, some models lack sensitivity to load peak-valley abrupt changes and local fluctuations [10], easily resulting in prediction lag; on the other hand, most studies focus only on improving prediction accuracy [11], neglecting the coupling relationship between prediction results and real-time control strategies, leading to a significant disconnect between prediction and control, making it difficult to meet the actual needs of closed-loop dynamic regulation in smart grid scenarios [12].

To address these issues, the fusion of Temporal Convolutional Networks (TCNs) and gated recurrent networks is gradually becoming an important direction in load forecasting research. Compared to traditional Recurrent Neural Networks (RNNs), temporal convolutional networks, through causal convolution and dilated convolution structures, can achieve a larger temporal receptive field with lower computational complexity, exhibiting strong extraction capabilities for local fluctuation features and periodic patterns. Bidirectional Gated Recurrent Units (BiGRUs) can simultaneously learn forward and backward temporal information, thereby enhancing the model's ability to express long-term dependencies. Furthermore, the Attention Mechanism (AM) can dynamically allocate importance weights at different time steps, enabling the model to pay closer attention to critical periods such as load abrupt changes and peak-to-valley transitions, improving prediction accuracy in complex scenarios. At the control level, Model Predictive Control (MPC) possesses rolling optimization and dynamic feedback correction capabilities, generating control commands in real time based on future predictions, offering significant advantages in load shaving and valley filling [13], and ensuring safe equipment operation. Therefore, combining temporal convolutional networks, bidirectional gated recurrent unit networks (GRUs), and attention mechanisms, and further coupling them with a model predictive control (MMC) framework to construct an integrated closed-loop system of "prediction-optimization-control," is of significant research importance for improving the short-term load intelligent control performance of electricity meters.

Based on this, this paper proposes a short-term load intelligent control method for electricity meters that integrates temporal convolutional networks, bidirectional gated recurrent unit attention networks, and model predictive control. First, local temporal fluctuation features in the load sequence are extracted using a temporal convolutional network module, and a bidirectional long-term dependency relationship is established using a bidirectional gated recurrent unit network. Second, an attention mechanism is introduced to dynamically allocate weights to key time steps, thereby enhancing the model's ability to perceive peak-valley changes and sudden load changes. Finally, the prediction results are embedded into the MMC optimization framework, and real-time load control commands are generated through rolling optimization, realizing an integrated closed-loop decision-making process from load prediction to dynamic control. Experimental results show that the proposed method can effectively improve the accuracy of short-term load prediction and maintain good control stability and real-time response capabilities under complex load scenarios, providing a new solution for intelligent load regulation of electricity meters in a smart grid environment.

2. Data Preprocessing and Feature Engineering

2.1. Dataset Description

The power load data used in this study comes from the real-time data acquisition system of the smart grid dispatching platform in a certain region. The data is uploaded to the power grid monitoring center by the smart meter terminals through the Advanced Metering Infrastructure (AMI)[14-15]. The data acquisition spans one year, with a sampling frequency of 1 hour, and includes load operation data for the entire year. The raw data is stored in time series format. Each record mainly includes a timestamp and the corresponding load power value. The time information is recorded using a unified standard time format, and the load value is in kW.

Because smart meters have continuous high-frequency sampling capabilities, the raw load series not only contains obvious daily periodic variations but also a large number of random fluctuations and local abrupt changes caused by user electricity consumption behavior, industrial equipment start-up and shutdown, and changes in the external environment. This type of load data has typical nonlinear, non-stationary, and strongly time-series correlated characteristics, posing a significant challenge to traditional prediction models. Furthermore, during actual data acquisition, factors such as communication anomalies, equipment calibration, and transient disturbances inevitably lead to missing values, outliers, and noise interference in the raw data. Therefore, before model training, systematic data cleaning and feature preprocessing of the raw data are necessary to improve the training stability and prediction accuracy of the subsequent deep learning model.

To ensure temporal continuity between the training and test sets, this paper adopts a chronological approach, dividing the dataset into the first 80% as the training set and the last 20% as the test set.

2.2. Data Cleaning

Since the raw electricity meter load data comes from the actual power grid operating environment, it is easily affected by factors such as equipment communication abnormalities, instantaneous pulse interference, and sampling errors during long-term continuous collection. Therefore, it is necessary to perform systematic data cleaning and anomaly processing on the raw data to improve data quality and enhance the stability of model training.

First, duplicate records and missing values in the raw dataset are detected. For a small number of missing samples, a linear interpolation method is adopted to ensure the continuity of the time series. Let the adjacent values before and after the missing point be represented as x_{t-1} and x_{t+1}, respectively. The interpolated value can be expressed as follows:

$$x_t = \frac{x_{t-1} + x_{t+1}}{2} \tag{1}$$

Subsequently, in order to reduce the influence of random noise on model training, a sliding window filtering method is employed to smooth the original load sequence. Assuming that the window length is k, the sliding average can be calculated as:

$$\hat{x}_t = \frac{1}{k}\sum_{i=0}^{k-1} x_{t-i} \tag{2}$$

In the anomaly detection stage, the Interquartile Range (IQR)-based method is adopted to identify outliers in the load sequence. First, the first quartile Q_1 and the third quartile Q_3 are calculated, and the interquartile range is defined as:

$$IQR = Q_3 - Q_1 \tag{3}$$

The abnormal value interval is further defined as:

$$x \notin [Q_1 - 1.5IQR, Q_3 + 1.5IQR] \tag{4}$$

For detected outliers, a neighborhood mean replacement strategy is used to repair the data, thereby reducing the impact of extreme values on parameter updates during model training.

2.3. Data Normalization

Because electricity load data fluctuates significantly across different time periods, directly inputting the raw data into a deep learning model may lead to unstable gradient updates and slow convergence.

This study employs a min-max normalization method to linearly map the raw load data to the interval [0,1]. The normalization formula is as follows:

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{5}$$

where x denotes the original load value, $x_{\min}$ and $x_{\max}$ represent the minimum and maximum values of the dataset, respectively, and x' is the normalized result. After normalization, the instability caused by large numerical scale differences can be effectively alleviated, while the model's capability of learning load variation patterns can also be enhanced.

2.4. Sliding Window Sample Construction

To transform continuous time-series load data into supervised learning samples suitable for model training, a sliding window strategy is adopted to establish the mapping relationship between the input sequence and prediction target. This method effectively preserves temporal dependency characteristics and improves the model's capability of learning short-term load variation patterns.

Assuming that the window length is T, the input sample is defined as a sequence consisting of T consecutive historical load values:

$$X_i = [x_i, x_{i+1}, \ldots, x_{i+T-1}] \tag{6}$$

The corresponding prediction target is defined as the load value at the next time step:

$$y_i = x_{i+T} \tag{7}$$

This study uses a fixed-length sliding window, moving forward one time step at a time to generate a large number of supervised learning samples.

3. The Proposed TCN-BiGRU-Attention-MPC Intelligent Load Control Framework

3.1. Overall Architecture

The overall structure of the proposed intelligent load control framework is shown in Figure 1. It mainly consists of a data preprocessing module, a TCN temporal feature extraction module, a BiGRU long-term dependency modeling module, an attention weight allocation module, a fully connected prediction layer, and an MPC optimization control module. The system first performs missing value repair, outlier cleaning, and normalization on the raw electricity meter load data. Then, a sliding time window is used to construct supervised learning samples, and the processed sequence is input into the deep prediction network.

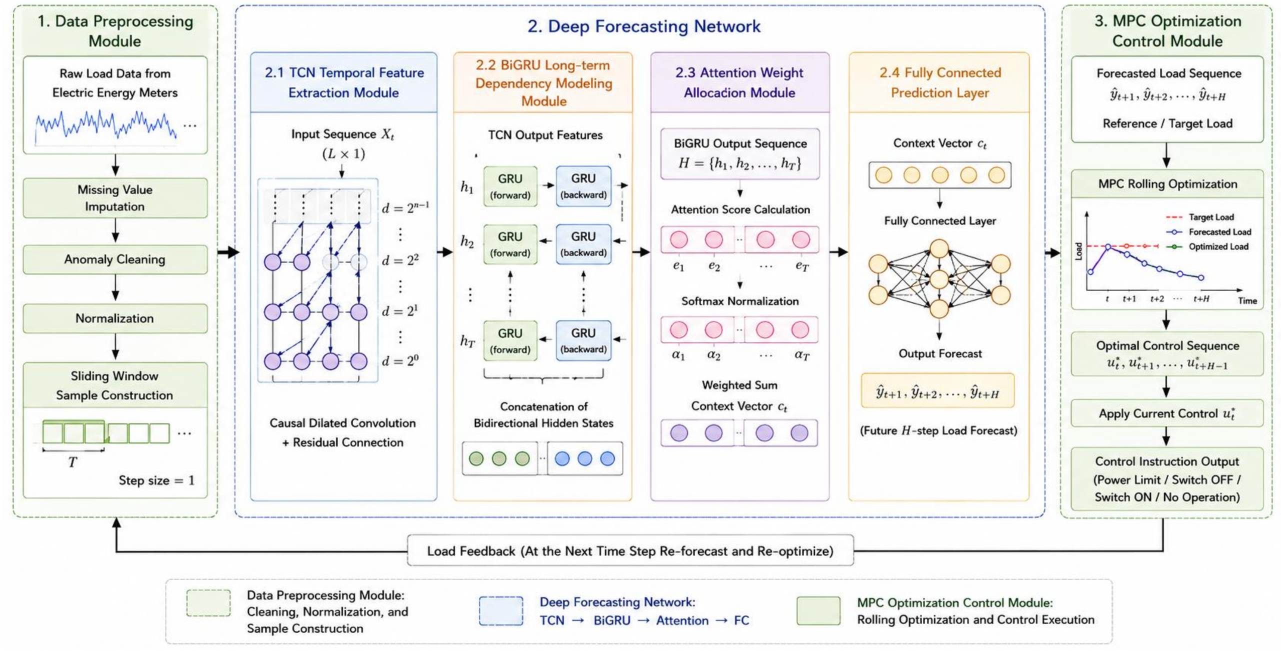

Figure 1: Overall framework of intelligent load control.

In the feature extraction stage, the temporal convolutional network uses causal convolution and dilated convolution structures to model the local fluctuations and periodic change patterns in the load sequence. Subsequently, a bidirectional gated recurrent unit network further extracts long-term dependencies and bidirectional temporal information from the time series, thereby enhancing the model's ability to express complex temporal dynamic changes. Based on this, an attention mechanism is introduced to dynamically allocate the importance of different time steps, enabling the model to focus on load peak-valley changes and abrupt change regions, thereby improving the stability and accuracy of the prediction results. Finally, the predicted load values for future time periods are output through the fully connected layer, and the prediction results are input into the model predictive controller. Real-time control commands are generated through rolling optimization to achieve dynamic adjustment of the electricity meter load.

The entire model structure can be represented as:

$$\hat{y}_{t+1} = f_{MPC}(f_{FC}(f_{Att}(f_{BiGRU}(f_{TCN}(X_t))))) \tag{8}$$

where X_t denotes the input load sequence, while f_{TCN}, f_{BiGRU}, f_{Att}, f_{FC}, and f_{MPC} represent the TCN module, BiGRU module, attention module, fully connected prediction layer, and MPC module, respectively.

3.2. TCN Temporal Feature Extraction Module

The TCN module is the primary core feature extraction structure of the proposed model. Its main function is to extract local temporal fluctuation features and periodic change patterns from the original load sequence. It adopts a causal convolutional structure, which effectively avoids the leakage of future information, and uses dilated convolution to expand the temporal receptive field without significantly increasing the network parameters, thereby enhancing the model's ability to model long sequences.

Assuming that the input time series is represented as $X=[x_1,x_2......x_T]$, the causal convolution process can be formulated as:

$$F(s)=\sum_{i=0}^{k-1} f(i)x_{s-d\cdot i} \tag{9}$$

where k denotes the convolution kernel size, d represents the dilation factor, $f(i)$ denotes the convolution kernel parameter, and $x_{s-d\cdot i}$ represents the input at the corresponding time step.

To further improve the training stability of deep networks, a residual connection structure is introduced into the TCN module, whose output can be expressed as:

$$H(x)=F(x)+x \tag{10}$$

where $F(x)$ denotes the convolution mapping result and x denotes the original input.

By combining dilated convolutions and residual structures, the TCN module effectively enhances the model's ability to extract local fluctuation features and periodic patterns, while mitigating the gradient vanishing problem during deep network training.

Although TCN can effectively capture local temporal features, convolutional structures alone are still insufficient for modeling long-term temporal dependencies.

3.3. BiGRU Long-Term Dependency Modeling Module

To further enhance the model's ability to learn long-term dynamic patterns in time-series data, BiGRU is introduced after the TCN module. The GRU structure is more lightweight, has fewer parameters, and is more efficient in training. Simultaneously, its bidirectional structure allows the model to learn both historical and future temporal information, thereby improving its ability to represent complex temporal dependencies.

In the GRU structure, the update gate and reset gate control the degree of preservation of historical information and the degree of integration of current input information, respectively. Their expressions are as follows.

Update gate:

$$z_t=\sigma(W_z[h_{t-1},x_t]) \tag{11}$$

Reset gate:

$$r_t=\sigma(W_r[h_{t-1},x_t]) \tag{12}$$

Hidden state updating process:

$$h_t=(1-z_t)\odot h_{t-1}+z_t\odot\tilde{h}_t \tag{13}$$

where $\sigma(\cdot)$ denotes the Sigmoid activation function and $\odot$ denotes the Hadamard product operation.

To simultaneously learn forward and backward temporal information, the bidirectional structure concatenates hidden states as follows:

$$h_t=[\overrightarrow{h_t};\overleftarrow{h_t}] \tag{14}$$

where $\overrightarrow{h_t}$ and $\overleftarrow{h_t}$ represent the forward and backward hidden states, respectively.

Through the Bidirectional Gated Recurrent Unit network, the model can further learn long-term trend variations and complex temporal correlations in load sequences, thereby enhancing the characterization capability for peak load fluctuations and periodic variation patterns

3.4. Attention Dynamic Weight Allocation Module

To improve the model's perception capability for critical temporal information, an attention mechanism is introduced after the BiGRU module. By dynamically assigning importance weights to different time steps, the model can focus on load mutation, peak-valley transitions, and abnormal fluctuation regions, thereby further improving forecasting accuracy and robustness.

Assuming that the hidden state sequence output by the BiGRU module is:

$$H=\{h_1,h_2,\cdots,h_T\} \tag{15}$$

First, the attention score corresponding to each time step is calculated as:

$$e_t=h_t^T W_a s_{t-1} \tag{16}$$

Then, the Softmax function is employed for normalization to obtain the attention weights:

$$\alpha_t=\frac{\exp(e_t)}{\sum_{k=1}^{T}\exp(e_k)} \tag{17}$$

Finally, the context vector is generated as:

$$c_t=\sum_{t=1}^{T}\alpha_t h_t \tag{18}$$

where α_t denotes the importance weight of the t-th time step and c_t denotes the context vector integrating critical temporal features.

3.5. MPC Optimization Control Module

The Model Predictive Control module is an essential component for achieving closed-loop "prediction-control" decision-making in this study. Its core idea is to dynamically solve the optimal control sequence based on future forecasting results within each control cycle and achieve real-time load regulation through rolling optimization.

Assuming that the system prediction output is $y(t+k\,|\,t)$, the target load is $r(t+k)$, and the control input is $u(t+k\,|\,t)$, the MPC objective function can be formulated as:

$$J=\sum_{k=0}^{N}(y_{t+k}-r_{t+k})^T Q(y_{t+k}-r_{t+k})+\sum_{k=0}^{N}(\Delta u_{t+k})^T R(\Delta u_{t+k}) \tag{19}$$

where Q and R denote the output error weighting matrix and control smoothness weighting matrix, respectively, and Δu_{t+k} denotes the variation of the control input.

At each sampling instant, the MPC controller performs rolling optimization over the future control horizon based on the current system state and future load forecasting results. Only the optimal control instruction at the current time step is output, while forecasting and optimization are recalculated at the next sampling instant, thereby realizing dynamic feedback regulation.

Through the joint design of the MPC module and deep forecasting network, the proposed method constructs a complete short-term intelligent load control closed-loop framework, which not only improves load forecasting accuracy but also significantly enhances control stability and real-time responsiveness under complex operating environments.

4. Experimental Results and Analysis

To verify the effectiveness of the proposed TCN-BiGRU-Attention-MPC intelligent load control method in short-term load forecasting and dynamic control tasks, this paper constructs an experimental environment based on real electricity meter load data and conducts experimental analysis from multiple perspectives, including prediction accuracy, model stability, and control effect.

4.1. Experimental Environment

This experiment was developed using a Python deep learning environment, primarily employing the PyTorch framework for model training and testing. The experimental hardware platform included an Intel Core i7 series processor and an NVIDIA RTX series graphics processor to meet the computational resource requirements during deep temporal network training.

The experimental environment and main parameter configurations are shown in Table 1.

Table 1: Experimental environment and model parameter settings.

Category	Parameter	Configuration
CPU	Processor	Intel Core i7
GPU	Graphics Processor	NVIDIA RTX 4060
Development Environment	Python Version	Python 3.10
Deep Learning Framework	Framework	PyTorch 2.0
Optimizer	Optimizer	Adam
Learning Rate	Learning Rate	0.001
Batch Size	Batch Size	64
Epoch	Training Epochs	100
TCN	Kernel Size	3
TCN	Dilation Factors	1, 2, 4
BiGRU	Hidden Units	64
Attention	Attention Heads	1
Loss Function	Loss Function	MSE

4.2. Evaluation Metrics

To comprehensively evaluate the performance of the proposed model in short-term load forecasting tasks, Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and Coefficient of Determination (R^2) are adopted as the primary evaluation metrics

$$MAE = \frac{1}{N}\sum_{i=1}^{N} |y_i - \hat{y}_i| \tag{20}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N} (y_i - \hat{y}_i)^2} \tag{21}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{N} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{N} (y_i - \bar{y})^2} \tag{22}$$

where $\bar{y}$ denotes the average value of the actual load sequence. y_i represents the actual load value, $\hat{y}_i$ represents the predicted load value, and N denotes the total number of testing samples.

4.3. Comparative Experiment Analysis

To verify the effectiveness of the proposed model in short-term load forecasting tasks, this paper selects traditional statistical models and several mainstream deep learning models as comparison methods. All models use the same dataset and training/testing partitioning to ensure the fairness of the experimental results.

Table 2 shows a comparison of the prediction results of each model on the test set.

As shown in Table 2, the traditional ARIMA model exhibits relatively poor prediction performance under complex nonlinear load scenarios. Its MAE and RMSE values are significantly higher than those of deep learning models, indicating that traditional statistical methods are insufficient to effectively characterize complex load temporal variation patterns. In contrast, recurrent neural network models such as LSTM and GRU can effectively learn long-term dependencies in time series data, thus significantly improving prediction accuracy. Among them, GRU, due to its lighter structure, demonstrates better prediction stability than LSTM.

Table 2: Prediction performance comparison results of different models.

Method	MAE	RMSE	R^2
ARIMA	0.8124	0.9651	0.8612
LSTM	0.6035	0.7428	0.9214
GRU	0.5712	0.7036	0.9347
TCN	0.5386	0.6715	0.9472
BiGRU	0.5148	0.6427	0.9563
TCN-BiGRU	0.4725	0.5981	0.9716
Proposed Model	0.4381	0.5637	0.9812

Further observation of the TCN and BiGRU models reveals that the TCN model has a strong ability to extract local temporal fluctuation features, while the BiGRU model is more suitable for modeling bidirectional long-term temporal dependencies. Therefore, both models outperform traditional recurrent neural network models in prediction performance. When TCN and BiGRU are jointly integrated, the model can simultaneously capture local temporal features and long-term dynamic change patterns, further reducing prediction errors. This demonstrates the strong complementary advantages of convolutional and recurrent structures.

Building on this, the proposed model further integrates an attention mechanism and MPC framework, enabling it to dynamically focus on key temporal features while achieving integrated optimization of prediction and control.

4.4. Ablation Experiment Analysis

To further verify the contribution of each component module in the proposed model to short-term load forecasting and intelligent control performance, this paper designs an ablation experiment. The ablation experiment mainly focuses on two aspects: the prediction network structure and the control module. Model variants were constructed with and without attention mechanisms, temporal convolutional network modules, bidirectional gated recurrent unit network modules, and a model without a prediction control module, and compared with the complete model. All ablation models used the same dataset partitioning, training epochs, optimizer, and evaluation metrics to ensure the comparability of the experimental results.

Table 3 presents the ablation experiment results under different model structures. As can be seen from the table, removing the attention mechanism increases both the mean absolute error and root mean square error of the model, indicating that the attention mechanism effectively enhances the model's ability to focus on peak-valley changes and abrupt changes.

Table 3: Ablation test results.

Model	MAE	RMSE	R ²	Controlling tracking error /kW
w/o Attention	0.4716	0.6014	0.9693	3.684
w/o TCN	0.4968	0.6275	0.9621	3.927
w/o BiGRU	0.5123	0.6482	0.9574	4.216
w/o MPC	0.4381	0.5637	0.9812	5.384
Proposed Model	0.4381	0.5637	0.9812	2.951

Furthermore, when the model predicts the control module is not used, although the model can still complete the load prediction task, it cannot further generate dynamic control commands based on the prediction results, resulting in a significant increase in the tracking error between the controlled load and the target load. In contrast, the complete model achieves the best results in terms of mean absolute error, root mean square error, and coefficient of determination, while also exhibiting the lowest control tracking error, demonstrating the good structural synergy of the TCN-BiGRU-Attention-MPC framework designed in this paper.

4.5. Visual Analysis of Prediction Results

To more intuitively compare the fitting capabilities of different models in short-term load forecasting, this paper selects a continuous load sample from the test set and plots the actual load curve and the prediction results of each comparative model in the same graph, as shown in Figure 2.

As shown in Figure 2, the ARIMA model has a certain ability to follow the periodic change trend of the load sequence, but there are significant deviations in the peak and trough regions, especially during periods of rapid load increase or decrease, where lag is likely to occur. LSTM and GRU have better time series modeling capabilities than traditional statistical models, and their prediction curves are generally closer to the actual load, but there are still some smoothing issues in local abrupt change regions. TCN can capture local fluctuation characteristics well, while BiGRU performs more stably in terms of overall trend fitting. The combined TCN-BiGRU model further improves the prediction effect in peak and trough regions.

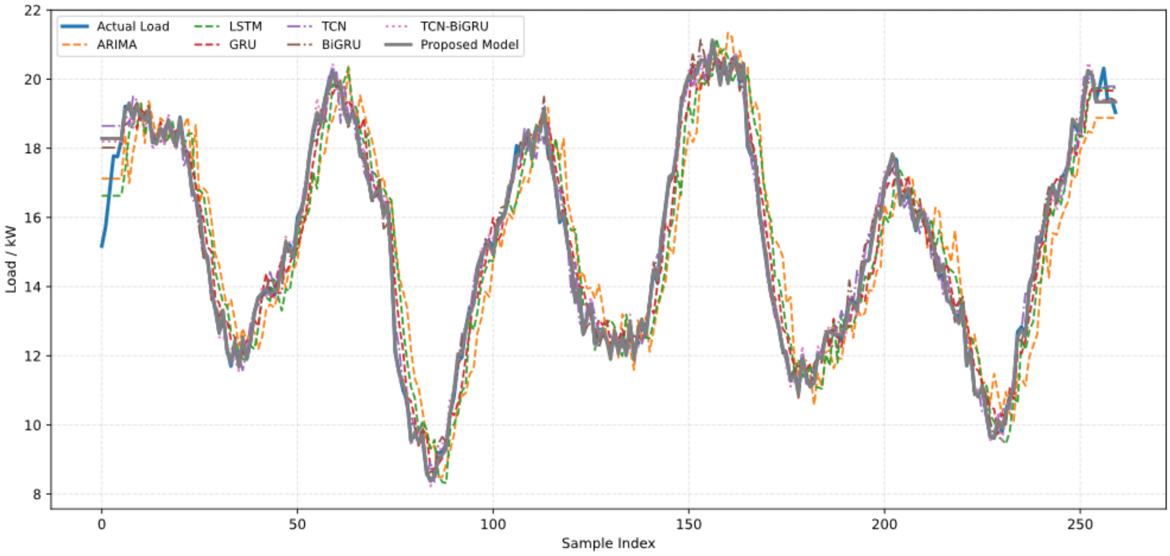

Figure 2: Visual comparison of prediction results from different models.

Compared to the above models, the prediction curve of the method presented in this paper has the highest degree of fit with the actual load curve, especially maintaining good tracking ability in load peaks, troughs, and rapidly changing regions. The results further demonstrate that the model in this paper can not only learn the periodic variation patterns of the load sequence, but also enhance its ability to perceive local mutations and critical periods, thereby effectively improving the accuracy of short-term load forecasting.

4.6. Error Distribution Analysis

To further analyze the distribution characteristics of the model's prediction errors, this paper calculates the error between the predicted and actual values on the test set and plots an error distribution histogram. Figure 3 shows the distribution of the prediction errors of the proposed model. As can be seen from the figure, the vast majority of prediction errors are concentrated near zero, exhibiting an approximately symmetrical distribution, indicating that the model does not exhibit significant systematic overestimation or underestimation.

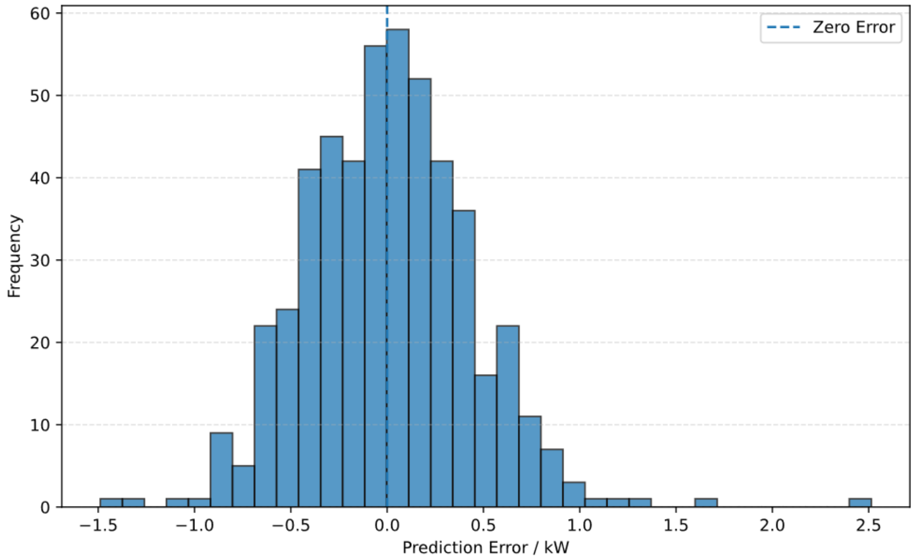

Figure 3: Prediction error distribution.

From the perspective of error range, most errors are distributed within a relatively small interval, indicating that the model has high prediction stability. A small number of larger errors mainly occur during periods of rapid load changes or severe local fluctuations, which is consistent with the characteristics of actual electricity meter load data being affected by user behavior and equipment start-up and shutdown. Overall, the error distribution results further verify the robustness of the proposed model in complex load prediction scenarios.

4.7. Attention Weight Heatmap Analysis

To verify whether the attention mechanism can effectively identify key time steps, this paper visualizes the attention weights in the test samples and plots an attention weight heatmap. Figure 4 shows the distribution of attention across different time steps. As can be seen from the figure, the model assigns higher weights to some key time areas, especially during periods of rapid load changes or significant peak-to-valley transitions, where the attention weights are significantly enhanced.

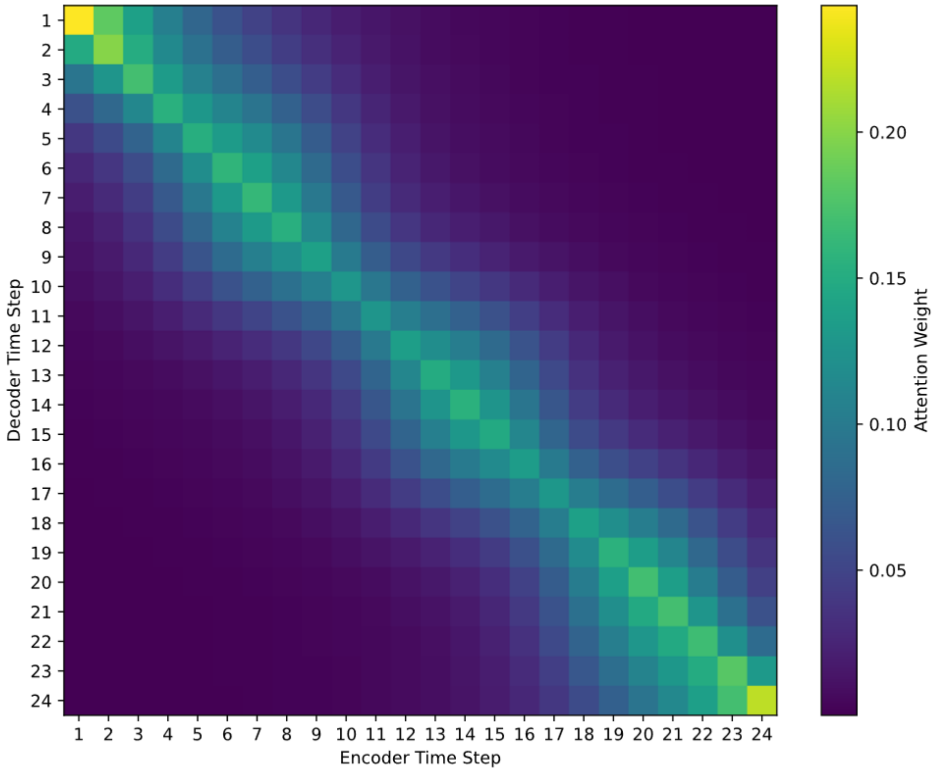

Figure 4: Attention weight heatmap.

This indicates that the attention mechanism can automatically adjust the time step weights based on the changing characteristics of the load sequence, making the model pay more attention to historical moments that significantly contribute to future load forecasting. Compared to models without an attention mechanism, the proposed method can better capture the short-term inertial characteristics before and after load abrupt changes, thereby reducing prediction errors in peak-to-valley regions. Simultaneously, the attention weight heatmap also provides some interpretability to the model's prediction process, helping to analyze the model's decision-making basis at different time periods.

4.8. Load Control Effect Analysis

To further verify the effectiveness of the model predictive control module in dynamic load regulation, this paper visualizes the changes before and after load control and plots the relationship curves between the actual load, target load, post-control load, and safety threshold, as shown in Figure 5. In the figure, the target load represents the reference operating load that the system expects to maintain, the safety threshold represents the maximum safe operating boundary allowed by the electricity meter load, and the post-control load represents the dynamic regulation result obtained after the model predictive control module performs rolling optimization.

Without control, the actual load curve exceeds the safety threshold in multiple time periods, especially in peak load areas, where significant over-limit phenomena are observed, easily leading to overload operation of electricity meters or increased pressure on local equipment. After introducing the model predictive control module, the system can dynamically adjust the control strategy based on future load forecasts, ensuring that the overall load remains below the safety threshold and gradually approaches the target load.

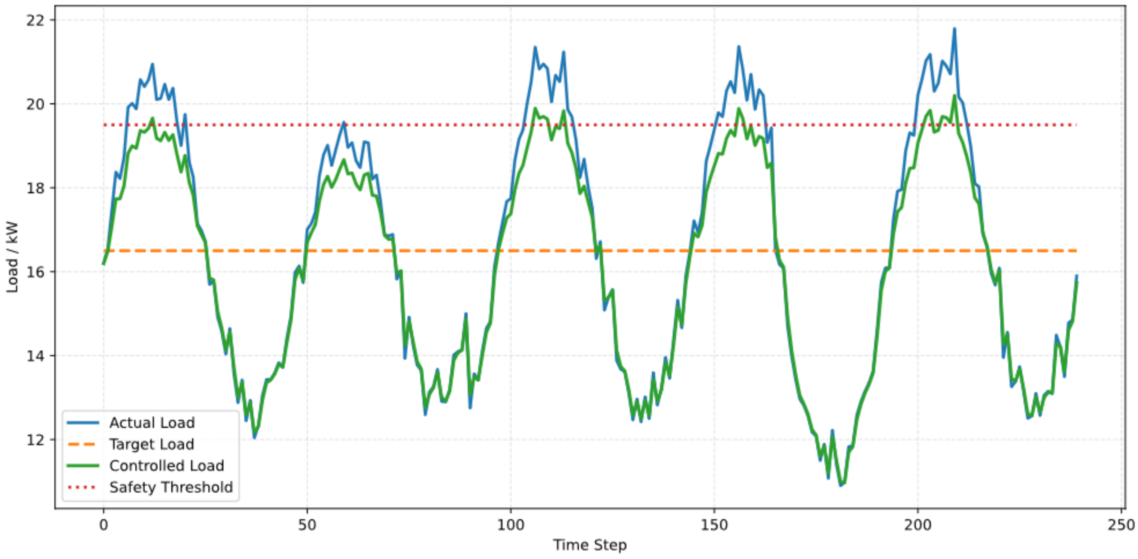

Figure 5: Load control effect analysis.

Further observation of the peak region reveals that the model predictive controller can adjust in advance before the load rises rapidly, effectively reducing peak fluctuations. This is mainly due to the model predictive control rolling optimization mechanism's ability to plan the control sequence in advance based on future load forecast information, thus exhibiting better foresight and dynamic response capabilities compared to traditional static threshold control methods.

4.9. Visual Analysis of Control Commands

To further analyze the control behavior of the model predictive control module under different load scenarios, this paper performs statistical and visual analysis on the control commands output by the controller and plots the distribution of control actions at different time steps, as shown in Figure 6. The control commands mainly include four types: power limit, switch OFF, switch ON, and no operation.

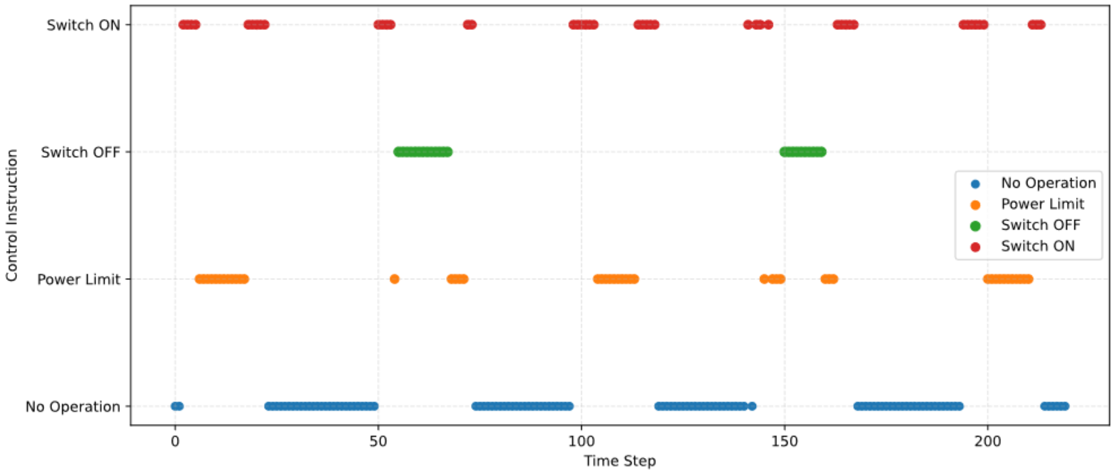

Figure 6: Visual analysis of control commands.

When the predicted load approaches or exceeds the safety threshold, the model predictive controller prioritizes the power limit strategy, dynamically reducing some load to mitigate system operational risks. When the load in a local area continues to exceed limits, a relay OFF operation is triggered to prevent equipment overload. After the load returns to normal, the system re-executes the relay ON operation to restore normal power supply to the equipment. During periods of stable operation where the target load is not exceeded, the controller remains in a no-operation state to avoid system oscillations caused by frequent control.

During most normal operating periods, the controller remains in a no-operation state, indicating that the overall system operation is relatively stable. When the load enters the peak region, the number of power limit and relay actions increases significantly, indicating that the model predictive control module can dynamically adjust the control strategy according to future load change trends. Furthermore, the relay operation is mainly concentrated in a small number of extremely high load areas, indicating that the method in this paper can reduce the frequency of forced disconnection operations by controlling power in advance, thereby reducing the operating losses caused by frequent start-stop of the equipment.

5. Conclusion

This paper addresses the shortcomings of traditional short-term load forecasting methods, such as insufficient ability to model complex time series, delayed response to peak-valley abrupt changes, and decoupling between forecasting and control. It proposes a smart short-term load control method for electricity meters that integrates TCN, BiGRU, AM, and MPC. First, TCN extracts local fluctuation features from the load sequence and BiGRU establishes long-term time series dependencies. Then, AM is introduced to dynamically assign weights to key time steps, thereby enhancing the model's ability to perceive peak-valley change regions. Finally, the forecast results are combined with the MPC framework to achieve integrated closed-loop optimization from load forecasting to dynamic control.

Experimental results show that the proposed TCN-BiGRU-Attention-MPC model outperforms traditional statistical models and mainstream deep learning models in terms of MAE, RMSE, and R^2. Specifically, TCN effectively enhances the model's ability to extract local short-term fluctuation features, BiGRU improves long-term trend modeling capabilities, and AM further strengthens the model's focus on key time regions. Meanwhile, the MPC module can dynamically generate control commands based on future load forecasts, enabling advance adjustment of peak loads, thereby effectively reducing the risk of system over-limit operation and improving system stability.

Overall, the method presented in this paper can achieve high-precision short-term load forecasting and dynamic control in complex load scenarios, demonstrating significant engineering application value. However, this research still has certain limitations. For example, the current experiments are mainly based on single-region electricity meter load data for verification, and the issues of multi-region collaborative control and multi-source heterogeneous data fusion have not been fully considered.

References

[1] Al-Shetwi A Q. Latest advancements in smart grid technologies and their transformative role in shaping the power systems of tomorrow: An overview[J]. Progress in Energy, 2025, 7(1): 012004.

[2] Majhi A A K, Mohanty S. A comprehensive review on internet of things applications in power systems[J]. IEEE Internet of Things Journal, 2024, 11(21): 34896-34923.

[3] Miraftabzadeh S M. Deep learning in power systems: A bibliometric analysis and future trends[J]. IEEE Access, 2024, 12: 163172-163196.

[4] Tabassum S, Vijay Babu A R, Dheer D K. A comprehensive exploration of IoT-enabled smart grid systems: power quality issues, solutions, and challenges[J]. Science and technology for energy transition, 2024, 79: 62.

[5] Rajendran G, Raute R, Caruana C. The brain behind the grid: a comprehensive review on advanced control strategies for smart energy management systems[J]. Energies, 2025, 18(15): 3963.

[6] Soni A, Yadav A. Seasonal autoregressive integrated moving average model for forecasting electrical load of

Chhattisgarh state[C]//2025 Fourth International Conference on Power, Control and Computing Technologies (ICPC2T). IEEE, 2025: 46-51.

[7] Thota V R. Comparative Study of Time Series Forecasting for Trucking Shipments: Evaluating Arima, LSTM, and Hybrid Arima-LSTM Models[D]. State University of New York at Binghamton, 2025.

[8] Waheed W, Xu Q. Data-driven short term load forecasting with deep neural networks: Unlocking insights for sustainable energy management[J]. Electric Power Systems Research, 2024, 232: 110376.

[9] Borunda M. Advances in Similar Day Methods for Short-Term Load Forecasting for Power Systems[J]. Forecasting, 2026, 8(2): 32.

[10] Liu K. Generation-side peak–valley time-of-use tariff optimization considering fluctuations in distributed photovoltaic grid-connected output[J]. Sustainable Energy Research, 2025, 12(1): 53.

[11] Kim J. A comprehensive survey of deep learning for time series forecasting: architectural diversity and open challenges[J]. Artificial Intelligence Review, 2025, 58(7): 216.

[12] Daryanavard S. Real-time predictive artificial intelligence: deep reinforcement learning for closed-loop control systems and open-loop signal processing[D]. University of Glasgow, 2024.

[13] Ao D, Wong P K, Huang W. Model predictive control allocation based on adaptive sliding mode control strategy for enhancing the lateral stability of four-wheel-drive electric vehicles[J]. Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering, 2024, 238(6): 1514-1534.

[14] Amuthan N, Sathya M, Rani N C. Empowering Consumers and Utilities for a Smarter Future: The Pivotal Role of Advanced Metering Infrastructure (AMI) in Smart Meter Technology[J]. Cloud Computing in Smart Energy Meter Management, 2025: 31-65.

[15] Zhao Y, Shen X, Chen C. Research on IP-based network architecture for advanced metering infrastructure of power grid network[J]. Journal of Computational Methods in Sciences and Engineering, 2025, 25(4): 2943-2963.