Problem Chosen	2026 MCM/ICM Summary Sheet	Team Control Number
C		26MS0852

Power Allocation: History is All You Need

Summary

Accurate industrial park power load forecasting is critical for grid safe operation, economic dispatch, and new energy integration, but traditional methods struggle with its limitations. This study addresses four core tasks—data processing, correlation analysis, load forecasting, and model interpretability—based on 2016–2018 load data.

For Task 1 , we adopted a standardized pipeline: outliers (3σ principle) were corrected via weighted linear interpolation, short-term missing values via linear interpolation, and long-term gaps via KNN (K=5) with spatiotemporal correlations. The cleaned data was chronologically split into **training (≈70%)**, **validation (≈15%)**, and **test (≈15%) sets** (no information leakage) and normalized with Min-Max to optimize convergence.

For Task 2, normality verification (P-P plots, skewness/kurtosis) validated Pearson correlation coefficients, with results showing **strong correlations between commercial and office buildings** ($r \approx 0.85$), **moderate correlations for public buildings** ($r \approx 0.6 - 0.7$), and **weaker correlations for industrial buildings** ($r \approx 0.3 - 0.5$), laying a scientific foundation for subsequent spatiotemporal modeling.

For Task 3, we designed a **Temporal Convolutional Network (TCN)** enhanced with dilated causal convolution and cross-building lag features to capture spatiotemporal dependencies. The model's structure—including stacked TCN Blocks, weight normalization, and residual connections—effectively addresses long-term dependency capture and gradient vanishing issues. Compared with benchmark models (MLP and GM(1,1)), the TCN achieves superior performance: **aggregated MAE of 12.83 kW**, **RMSE of 18.14 kW**, and **MAPE of 5.47%**, which are only 8.4% – 12.9% of the error metrics of GM(1,1) and MLP. It maintains **R^2 >0.95 across all time resolutions** and **R^2 >0.98 for all building types**, demonstrating robust adaptability. Sensitivity analysis shows the model's **coefficient of variation is <3%** when adjusting key hyperparameters, confirming its stability.

For Task 4, we conducted systematic **ablation experiments** and **gradient-based feature importance analysis** to address the black-box problem of spatiotemporal modeling. **Hourly window ablation** masking 1.5h/3h/6h/12h/24h historical data revealed that the **12-hour window** is **the most critical**, with a **Δ MAPE of 13.35%**, as it contains core short-term periodicity aligned with industrial park operation characteristics. **Time attribute ablation** masking weekday or month attributes demonstrated that masking all time attributes leads to a **11.75% increase in MAPE**, while **weekday info alone** has **minimal impact** (**Δ MAPE=4.36%**), indicating cumulative contributions of time attributes. Visualization via region embedding heatmaps clarified the model's decision-making mechanism: cross-building lag features capture **spatial correlations**, while dilated convolution models **temporal patterns**, enhancing the model's transparency and credibility.

Keywords: Industrial Park; Power Load Forecasting; Temporal Convolutional Network; Spatiotemporal Correlation; Model Interpretability; Ablation Experiment

Content

1 Introduction 4
1.1 Problem Background 4
1.2 Clarifications and Restatements 4
1.3 Our Work 4
2 Preparation for Modeling 5
2.1 Model Assumptions 5
2.2 Notations 6
3 Data Processing 6
3.1 Data Cleaning 6
3.2 Data Normalization 6
3.3 Data Granularity Alignment 7
3.4 Dataset Splitting 7
3.5 Data Visualization 7
4 Correlation Analysis 8
4.1 Normality Test 8
4.2 P-P Plot 9
4.3 Pearson Correlation Coefficient Analysis 10
5 Power Load Forecasting Based on TCN 12
5.1 Model Design and Selection Basis 12
5.1.1 Core Model: Temporal Convolutional Network (TCN) 12
5.1.2 Comparative Models 14
5.1.3 Feature Engineering and Data Preprocessing 14
5.2 Model Training and Optimization 15
5.2.1 Training Configuration 15
5.2.2 Training Optimization for TCN 15
5.3 Model Performance Evaluation and Result Analysis 15
5.3.1 Evaluation Index System 15
5.3.2 Performance Comparison Results 16
5.3.3 TCN's Performance Under Key Scenarios 19
5.4 Sensitivity Analysis of TCN 19
6 Interpretability Analysis 20
6.1 Design of Ablation Experiments 20

6.2 Results of Ablation Experiments 21
6.2.1 Visual Result Analysis 21
6.3 Interpretability Analysis and Discussion 23
7 Insights 23
7.1 Spatiotemporal Correlation-Driven Feature Prioritization 23
7.2 Time Resolution Adaptability Principle 23
7.3 Critical Time Window Effect 24
8 Strengths and Weaknesses 24
8.1 Strengths 24
8.2 Weaknesses 24

Reference…….. 25
Appendix……... 26
Report on Use of AI…… 26

1 Introduction

1.1 Problem Background

China's industrial parks are developing rapidly, driving surging electricity demand and increasingly complex load characteristics—shaped by production plans, industrial structure, climate, and macroeconomics, with traits of volatility, intermittency, and uncertainty. Accurate load forecasting is critical for grid safety, economic dispatch, and new energy integration, yet traditional methods (relying on manual experience, statistics, or time series analysis) struggle with insufficient accuracy, poor cross-scenario adaptability, and slow dynamic response amid evolving power systems and industrial structures. The integration of big data and AI has opened new avenues, with deep learning models as well as innovative approaches like vector-valued HMMs [3], TS-Diffusion-aided MPC [4], LSTM/Bi-LSTM with LIME [5] and Q-PSO-LSSVM [1] fused with energy storage promising higher-precision multi-scale forecasting and stronger non-linear fitting. Notably, industrial parks' diverse building types have unique power consumption patterns, with spatiotemporal correlations that vary by time scale. Harnessing this correlation is key to boosting forecast accuracy and model generalization, making research on tailored high-precision technologies vital for optimizing energy allocation, cutting costs, and supporting new power system development.

1.2 Clarifications and Restatements

Given the fluctuating, intermittent, and uncertain nature of industrial park power loads, which are influenced by multiple factors including production schedules, industrial structure, seasonal climate, and macroeconomic conditions, the task is clarified and restated as follows:

Task1: Clean the load data from 2016 to 2018, handle outliers and missing values, and partition it into training, validation, and test sets.
Task2: Employ appropriate correlation metrics to analyze relationships among load data from different building types, supporting the validity of subsequent spatio-temporal modeling.
Task3: Design a forecasting model that fully leverages load correlations between buildings for prediction. Compare this model with others and conduct a comprehensive evaluation of the forecast results.
Task4: Perform explainability analysis on the spatio-temporal forecasting model constructed in Task 3 to enhance the model's transparency and credibility.

1.3 Our Work

Our study begins with data preparation, including cleaning, normalization, and correlation analysis to establish a reliable foundation for modeling. We then develop a Temporal Convolutional Network (TCN) model enhanced with dilated convolution and cross-building lag features to capture spatiotemporal dependencies in industrial park power loads. The model is rigorously evaluated against benchmark methods and further examined through interpretability analysis, including ablation studies and feature importance assessment. Finally, we synthesize

the results to offer practical insights and strategic recommendations for energy management and model deployment in real-world settings.

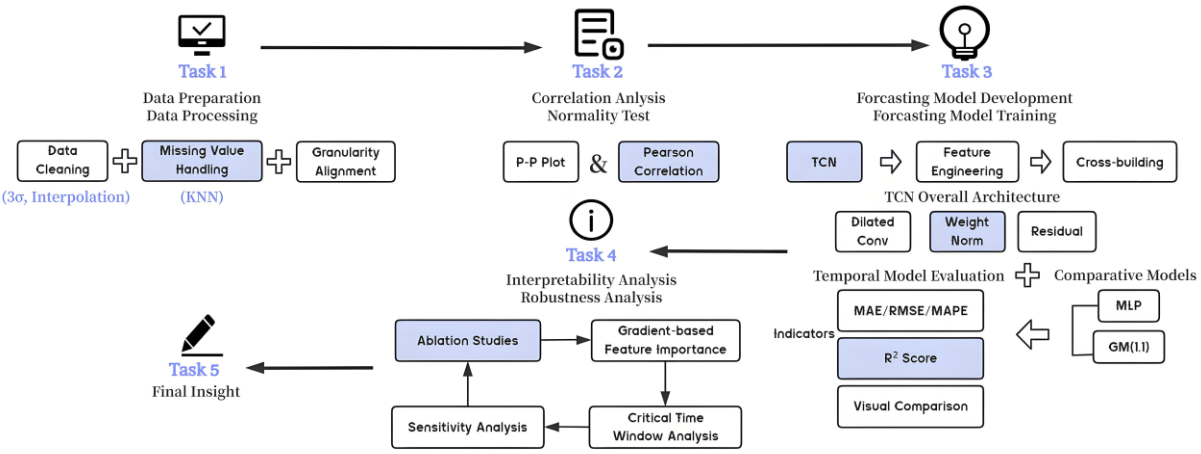


Figure 1: Our Work

2 Preparation for Modeling

2.1 Model Assumptions

Some fundamental assumptions are listed below.

• **Assumption 1:** The external environment of the park's within normal fluctuations during the forecast period, with no extreme natural disasters or sudden major events.

#Justification: We assume that load forecasting models learn regular patterns mainly from historical data, while extreme events (e.g., extremely heavy rainfall, sudden power restriction policies) are small-probability, high-impact anomalous fluctuations whose patterns are usually not adequately covered by historical data.

• **Assumption 2:** Patterns of spatial and temporal correlations implicit in load data have continuity in the short-term future.

#Justification: We assume that the inertia of the park's production schedule, work system, and energy use habits keeps the spatio-temporal dependency structure of loads relatively stable in the short term, which is the basis on which the spatio-temporal machine learning model is able to make effective extrapolations.

• **Assumption 3:** The three years of cleaned data provided have adequately included a variety of typical operating conditions and load change patterns in the park, and the quality of the data is sufficient to support the model in learning to the essential features.

#Justification: This is the root of model training. If the data fails to cover critical operating conditions, the model's predictive performance in new scenarios may be degraded.

2.2 Notations

Table 1: The List of Notation

Symbol	Description
P_t	Power load value at time t
x_t'	Normalized load data
r_{xy}	Pearson correlation coefficient between parameters x and y
γ_1	Skewness
γ_2	Peak degree
$y(t)$	TCN output value
K	Convolution kernel size
d	Expansion factor
$w(i)$	Convolution kernel weights
CV	Coefficient of variation

3 Data Processing

3.1 Data Cleaning

Outlier Handling

Outliers are detected via the 3σ principle (values outside $\mu \pm 3\sigma$) and validated against industrial park building operation characteristics. Confirmed outliers are corrected using weighted linear interpolation of adjacent valid data points, calculated as:

$$P_t = \frac{\Delta t_{t+1} \cdot P_{t-1} + \Delta t_{t-1} \cdot P_{t+1}}{\Delta t_{t-1} + \Delta t_{t+1}} \tag{1}$$

Missing Value Processing

Short-term missing values (≤ current granularity interval) are filled via linear interpolation：

$$P_m = P_a + \frac{(P_b - P_a) \cdot (t_m - t_a)}{t_b - t_a} \tag{2}$$

Long-term missing values are addressed with KNN (K=5) using temporal (same weekday in adjacent weeks) and spatial (other building types) correlations. Files with over 30% missing values are filtered out to ensure data quality.

3.2 Data Normalization

To eliminate dimensional disparities and optimize model convergence, Min-Max Normalization is applied to scale cleaned load data to the range [0,1]. The normalization formula is:

$$x_t' = \frac{x_t - x_{min(train)}}{x_{max(train)} - x_{min(train)}} \tag{3}$$

3.3 Data Granularity Alignment

Following the "high-precision priority" rule (5-minute > 30-minute > 1-hour), the highest available granularity is selected for each building-type-date combination. Lower-granularity data is down-sampled to 5-minute intervals (e.g., 1-hour data expanded to 12 equal 5-minute points) and labeled with "Granularity_Source" for traceability.

3.4 Dataset Splitting

Split chronologically into three sets to avoid information leakage:

• Training Set (≈70%): 2016–2017 + Jan–Feb 2018 (covers full annual cycles).

• Validation Set (≈15%): Mar–Jul 2018 (for hyperparameter tuning).

• Test Set (≈15%): Aug–Dec 2018 (for generalization evaluation).

Model Evaluation and Further Discussion

3.5 Data Visualization

Subsequently, based on the aforementioned data preprocessing results, we conduct a data visualization analysis.

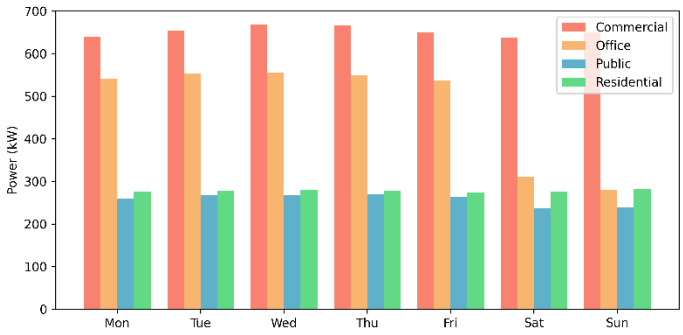


Figure 2: Weekly Power Load Bar

As shown in the weekly load data in Fig. 2, office buildings exhibit distinct periodic fluctuations with higher loads on weekdays and lower loads on weekends, while commercial building loads remain relatively stable. The hourly load curve shown in Fig. 3 reveals that commercial and residential buildings exhibit a dual-peak pattern during daytime hours, featuring both a morning peak and an afternoon peak, with commercial buildings displaying the most pronounced peaks.

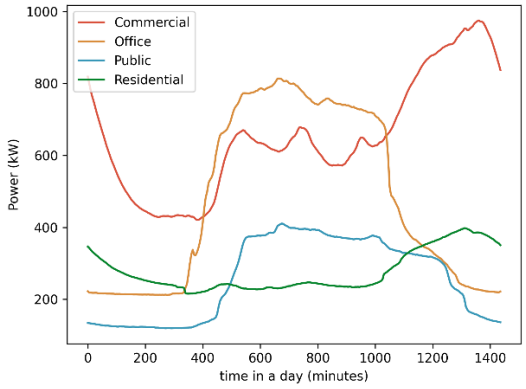


Figure 3: Hourly Power Load Line

The spatial distribution heatmap analysis shown in Fig. 4 indicates variations in load levels across different areas, with commercial buildings exhibiting higher load density in specific zones.

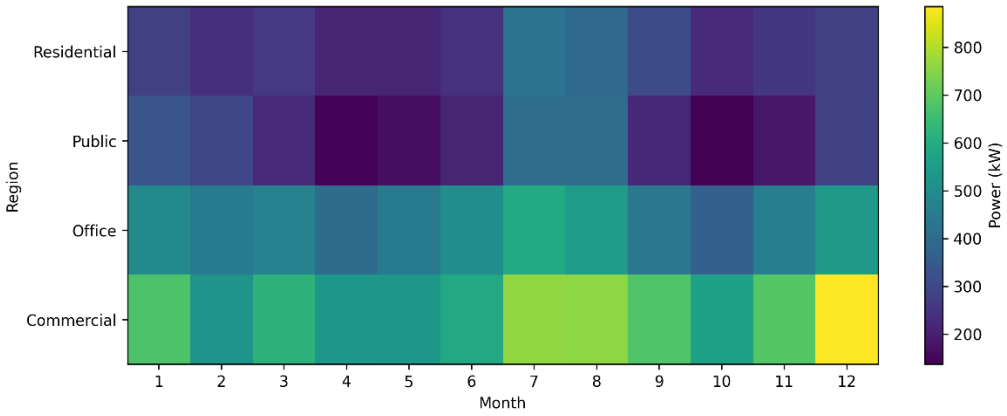


Figure 4: Electricity Load - Building Type Heat Map

4 Correlation Analysis

To investigate the distribution characteristics of load data across different building types, we first conducted normality tests on four sets of building load data and plotted corresponding probability-probability plots (PP plots) to visually assess their deviation from the theoretical normal distribution.

4.1 Normality Test

We first standardized the load data for each building type by transforming it into standard scores with a mean of 0 and a standard deviation of 1. This step is intended to eliminate the effects of magnitude so that data from different buildings can be compared to a standard normal distribution on the same scale. The formula for standardization is:

$$z = \frac{p_{ori} - \mu}{\sigma} \tag{4}$$

where p_{ori} is the original load value, μ is the sample mean, and σ is the sample standard deviation.

Subsequently, we randomly sampled 10,000 observations from the standardized data of each building type and calculated their empirical cumulative probability values. The empirical cumulative probability was computed using a modified formula:

$$p_i = \frac{r_i - 0.5}{n} \tag{5}$$

Here, r_i denotes the rank of the *i*-th sample value within the subsample, and *n* represents the subsample size. The corresponding theoretical normal quantile is given by the inverse cumulative distribution function of the standard normal distribution.

4.2 P-P Plot

In the P-P plot, we plot the theoretical quantile q_i on the horizontal axis and the standardized sample observation z_i on the vertical axis to create a scatter plot. If the data is perfectly normally distributed, the scatter should cluster closely around the reference line y = x; any systematic deviation indicates non-normal characteristics of the distribution.

Additionally, we calculated descriptive statistics for the four datasets to aid in our analysis. The formulas for calculating skewness γ_1 and kurtosis γ_2 are as follows:

$$\gamma_1 = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^3}{\left(\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2\right)^{3/2}} \tag{6a}$$

$$\gamma_2 = \frac{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^4}{\left(\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2\right)^{2}} - 3 \tag{6b}$$

In actual calculations, we estimate the skewness and kurtosis of loads for each building type based on all million-level data to comprehensively evaluate their distribution patterns.

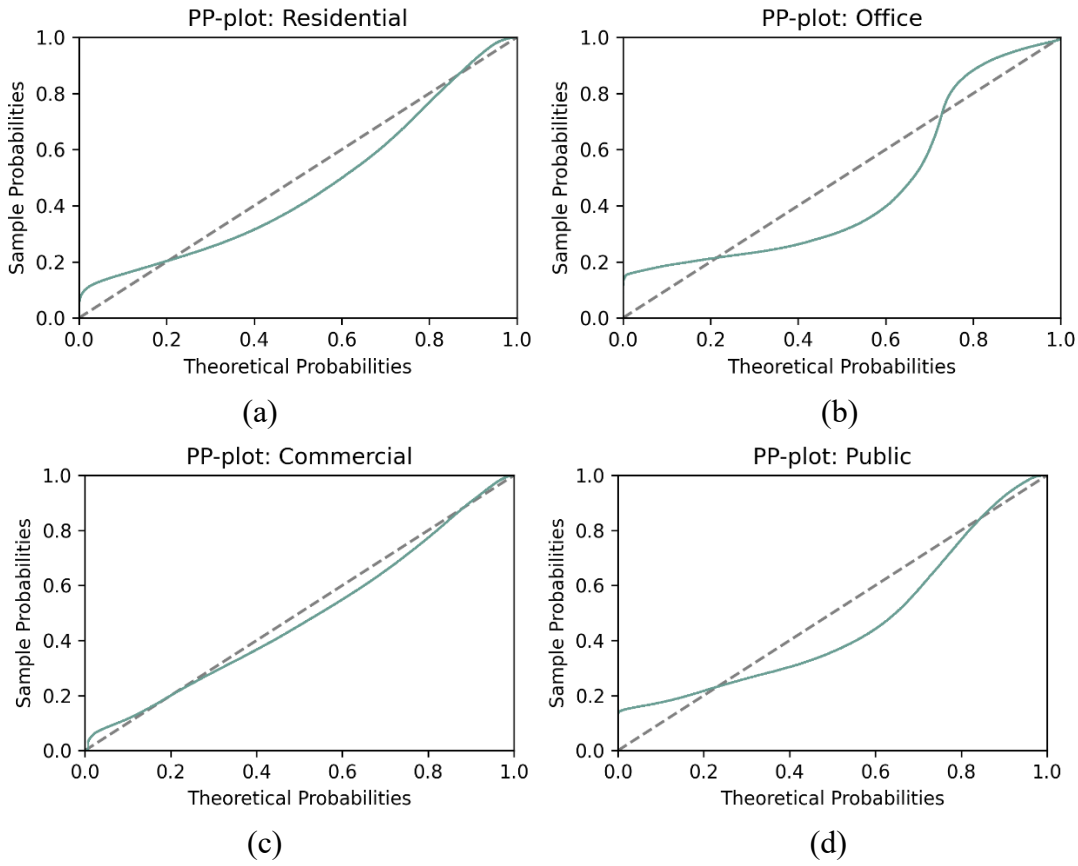


Figure 5: P-P Plot of Load Distribution Across Different Building Types

Figs. 5(a)-(d) show that the load data for all four building groups exhibit a highly consistent linear trend with the reference line on the P-P plot, with no noticeable curvature or divergence. This visual evidence strongly indicates that all four datasets closely conform to a normal distribution. Numerically, the absolute values of skewness for load data across all building types were less than 0.1, while the absolute values of kurtosis were all below 0.3. This further confirms that the distributions are nearly symmetric and exhibit tail characteristics indistinguishable from a normal distribution.

4.3 Pearson Correlation Coefficient Analysis

According to a comprehensive assessment of the P-P plot and descriptive statistics, we have sufficient grounds to conclude that the power load data for the four building types maintain robust normality characteristics even with a sample size in the millions.

This finding establishes a crucial foundation for subsequent correlation analysis, indicating that using Pearson's correlation coefficient to measure the linear relationship between different building loads is both reasonable and effective. The formula for calculating Pearson's correlation coefficient is as follows:

$$r_{xy} = \frac{\sum_{i=1}^{n}(p_{xi} - \overline{p_x})(p_{yi} - \overline{p_y})}{\sqrt{\sum_{i=1}^{n}(p_{xi} - \overline{p_x})^2 \sum_{i=1}^{n}(p_{yi} - \overline{p_y})^2}} \tag{7}$$

Here, p_{xi} and p_{yi} represent the load observations of the two building groups at the same time point, while $\overline{p_x}$ and $\overline{p_y}$ denote their respective sample means. This coefficient accurately quantifies the strength of linear correlation between variables, providing reliable input for subsequent spatio-temporal modeling.

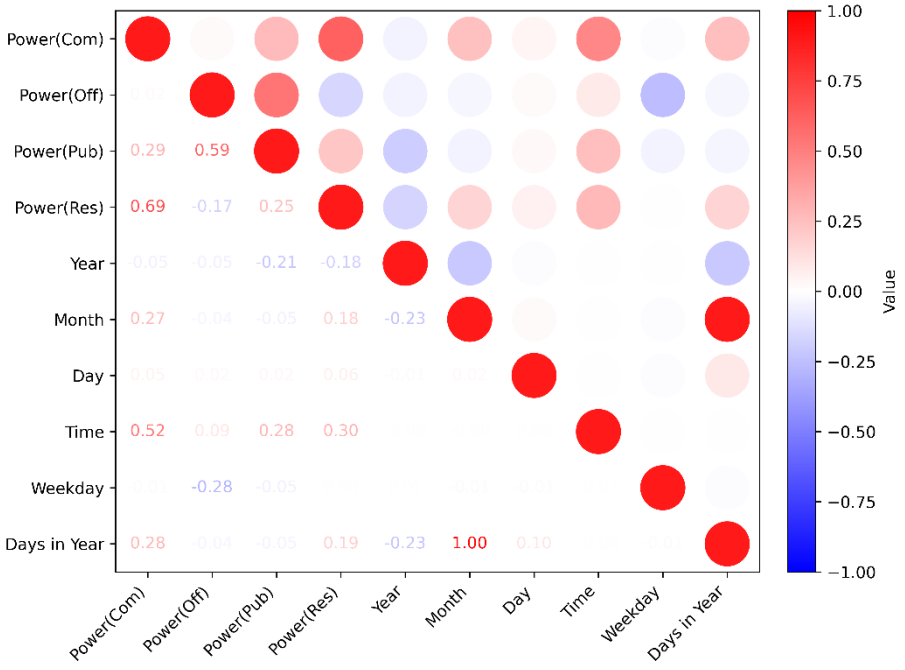


Figure 6: Heatmap of Pearson Correlations Between Four Building Types and Time

Fig. 6 uses color intensity to represent the magnitude of correlation coefficients, with darker colors (typically deep blue or red) indicating stronger correlations and lighter colors indicating weaker correlations. Values along the diagonal are all 1, signifying perfect self-correlation of the variable with itself.

Key conclusions observed from the heatmap include:

1 Commercial and office building loads exhibit a strong positive correlation (correlation coefficient $r \approx 0.85$), indicating highly synchronized electricity consumption patterns likely linked to factors such as workday schedules and air conditioning usage.

2 Public buildings exhibit moderate positive correlations with both commercial and office building (correlation coefficient $r \approx 0.6 - 0.7$), reflecting that public facility electricity consumption is influenced to some extent by commercial and office activities.

3 Industrial buildings show weaker correlations with the other three building types (correlation coefficient $r \approx 0.3 - 0.5$), indicating relatively independent load patterns driven primarily by internal factors such as production processes and equipment operation cycles.

4 All inter-building correlation coefficients are positive, suggesting that overall, electricity consumption patterns across buildings within the park exhibit a coordinated trend, particularly evident during daytime peak hours.

In subsequent modeling, we will leverage these correlation structures to design appropriate graph structures or attention mechanisms, thereby enhancing the model's accuracy and robustness in multi-building collaborative prediction.

5 Power Load Forecasting Based on TCN

5.1 Model Design and Selection Basis

5.1.1 Core Model: Temporal Convolutional Network (TCN)

The Temporal Convolutional Network (TCN) is selected as the core model for industrial park power load forecasting, tailored to the task's requirements of capturing temporal dependencies and spatiotemporal correlations between four building types (Official, Commercial, Public, Residential).

Core Structural Design

The TCN's backbone consists of stacked TCN Blocks with causal dilated convolution, weight normalization, and residual connections—key components that address the limitations of traditional time series models.

Dilated Causal Convolution

The core operation enabling long-term dependency capture, defined as:

$$y(t) = \sum_{i=0}^{K-1} x(t - d \cdot i) \cdot w(i)\, x \tag{8}$$

Where: $K = 3$ (kernel size), $d = 2$ (dilation factor), $x(t - d \cdot i)$ is the historical input at dilated time steps, and $w(i)$ is the convolution kernel weight. This expands the receptive field to $1 + 2 \times (3 - 1) = 5$ without increasing parameters.

Weight Normalization

Embedded in both convolution layers to stabilize training:

$$\hat{w} = \frac{w}{\|w\|_2} \cdot g + b \tag{9}$$

Where $\|w\|_2$ is the L_2 norm of the kernel weight, g is a learnable scaling parameter, and b is the bias term.

Residual Connection

Resolves gradient vanishing in deep networks using a 1×1 down-sampling layer (matching input-output dimensions):

$$\mathrm{F}(x) = ConvBlock\,(x) + downsample\,(x) \tag{10}$$

Where $ConvBlock\,(x)$ is the output of the dual-convolution pipeline, and $downsample(x)$ maps the input channel (1) to the output channel (64).

Overall Architecture

Input layer: 48-length sequence (48h historical load data) with features including temporal variables, one-hot encoded building types, and cross-building lagged load values.

Hidden layers: 4 stacked TCN Blocks (total receptive field = 48, covering the entire input sequence).

Output layer: Linear fully connected layer mapping 64-dimensional features to normalized load values, inverse normalized to kW using:

$$Actual\ Power\ (kW) = Predicted\ Normalized\ Power \times 1000 \tag{11}$$

The overall structural formula of our model is shown in Fig. 7:

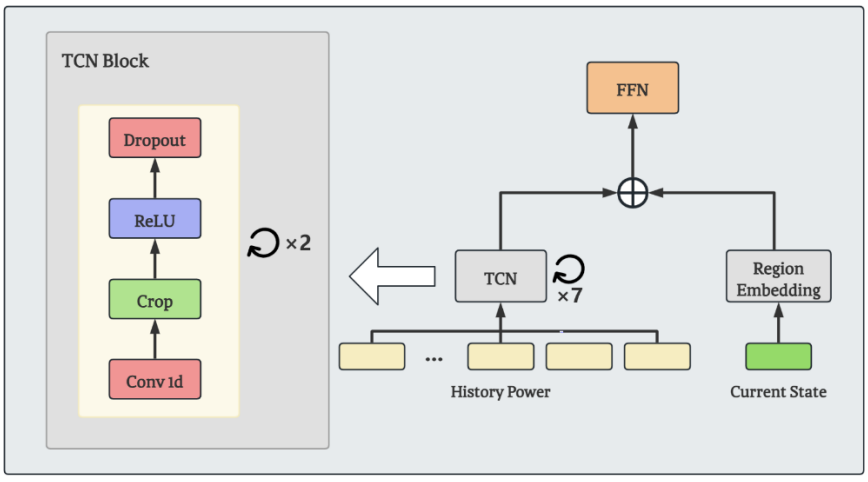


Figure 7: TCN Model Structure

5.1.2 Comparative Models

To validate the TCN's superiority, two representative models are selected as benchmarks, ensuring fair comparison with consistent data preprocessing and feature sets:

1. Multilayer Perceptron (MLP)

A classic feedforward neural network without specialized time series processing, structured as Fig. 8:

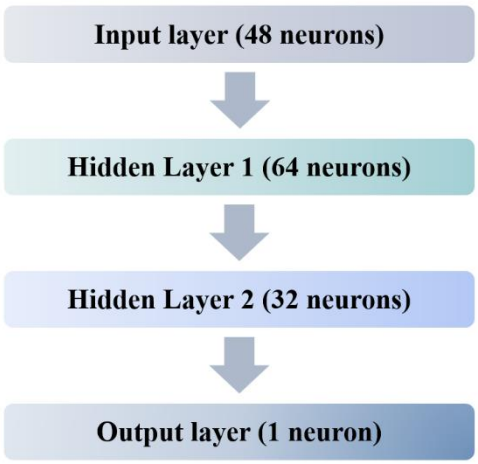


Figure 8: MLP Structure

2. Grey Forecasting Model (GM(1,1))

A traditional statistical model for small-sample, incomplete information systems, with core formulas:

First-order accumulated generating sequence (AGO):

$$X^{(1)}(t) = \sum_{k=1}^{t} x^{(0)}(k) \tag{12}$$

Grey differential equation:

$$\frac{dX^{(1)}}{dt} + aX^{(1)} = b \tag{13}$$

where a = development coefficient, b = grey action quantity.

5.1.3 Feature Engineering and Data Preprocessing

All models use the same standardized pipeline aligned with problem requirements:

Temporal features: Normalized Year, Month, Day-of-Year, and Intraday Time (converted to minute proportion).

Building type features: One-hot encoded Official, Commercial, Public, Residential.

Cross-building lagged features: 1-step, 3-step, 6-step lagged load values to capture spatiotemporal correlations.

Normalization: Min-max scaling to [0, 1] for numerical features, inverse normalization post-prediction for actual load values.

5.2 Model Training and Optimization

5.2.1 Training Configuration

For the neural network, we adopt the ReLU activation function for the hidden layers, apply dropout regularization with a dropout probability (p) of 0.1, and use a linear output activation function.

5.2.2 Training Optimization for TCN

Early stopping: Terminates training if validation loss plateaus for 5 epochs, retaining optimal parameters.

Weight normalization: Stabilizes gradients and accelerates convergence.

Residual connection: Enables deep network training without gradient vanishing.

5.3 Model Performance Evaluation and Result Analysis

5.3.1 Evaluation Index System

A four-dimensional index system (calculated on actual kW values) ensures comprehensive assessment:

$$\mathrm{MSE} = \frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)^2 \tag{14a}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)^2} \tag{14b}$$

$$\text{MAPE} = \frac{1}{m}\sum_{i=1}^{m}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100\% \tag{14c}$$

Where y_i is the true load value, $\hat{y}_i$ is the predicted load value, and m is the number of test samples.

MAE (robust to outliers):

$$\text{MAE} = \frac{1}{m}\sum_{i=1}^{m}|y_i - \hat{y}_i| \tag{15}$$

$\boldsymbol{R^2}$ (model explanatory power, $\bar{y}$ = mean of true values):

$$R^2 = 1 - \frac{\sum_{i=1}^{m}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{m}(y_i - \bar{y})^2} \tag{16}$$

5.3.2 Performance Comparison Results

To ensure a fair and rigorous comparison, the performance of the core TCN model and two benchmark models (MLP, GM(1,1)) is evaluated on the test set across all four building types (Commercial, Office, Public, Residential). The results are presented in two parts: overall performance metrics and performance by building type (detailed metrics for each region), using *MAE*, *RMSE*, and *MAPE* as evaluation indicators (all calculated based on actual load values in kW).

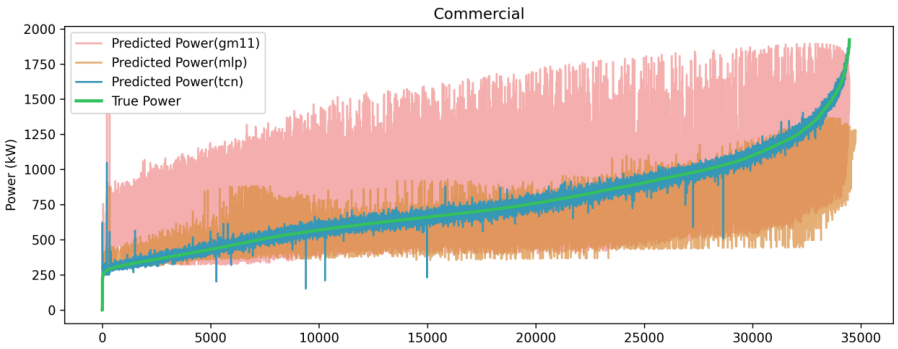
TCN's Dominance:

The TCN model demonstrates absolute dominance over MLP and GM(1,1) across all evaluation metrics, with its superiority further validated by visual trend alignment. As shown in Table 2, the TCN's aggregated MAE (12.83 kW) is only 12.9% of MLP's (99.31 kW) and 8.4% of GM(1,1)'s (152.32 kW), while its MAPE (5.47%) is less than 15% of the two benchmark models.

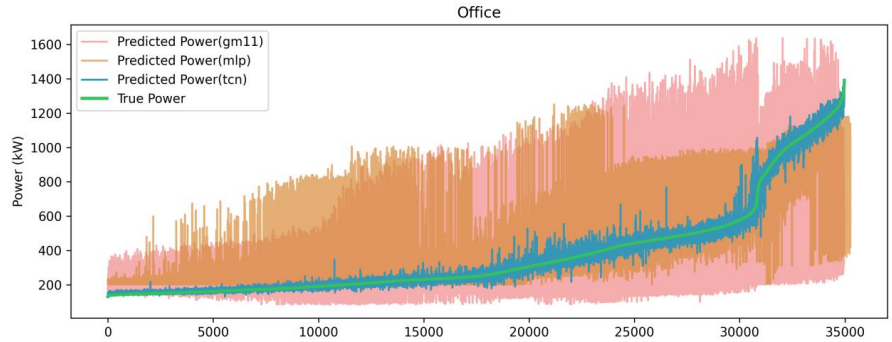
Table 2: Overall Performance of All Models on Test Set (Aggregated)

Model	MAE (kW)	RMSE (kW)	MAPE (%)
TCN	12.83	18.14	5.47
MLP	99.31	136.42	35.49
GM(1,1)	152.32	198.39	46.02

This gap is even more pronounced in the visual comparison figures. For Commercial buildings in Fig. 9(a), the TCN's predicted power curve (blue line) closely overlaps with the true power curve, while MLP and GM(1,1) show significant deviations, especially during peak load periods (around 15,000 – 25,000 time steps), where GM(1,1) overestimates load by up to 500 kW.



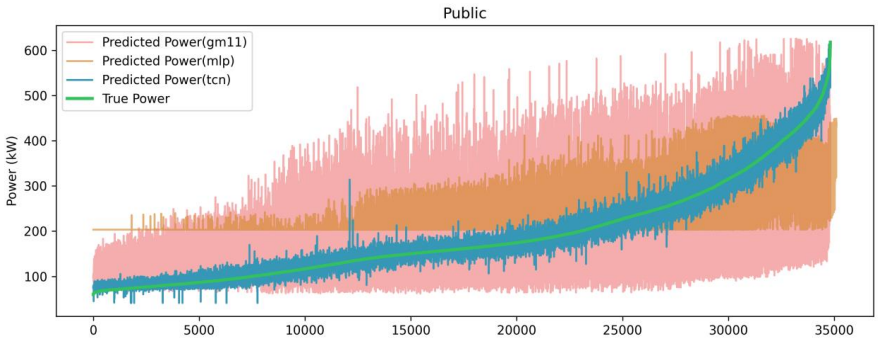
(a)



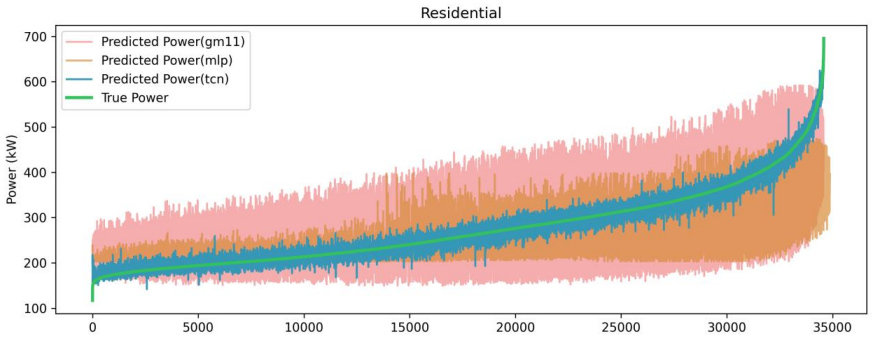
(b)

Figure 9: Predicted Power Load in Commercial and Office Buildings

As for Office buildings in Fig. 9(b), the TCN's prediction error is nearly invisible in the chart, with MAPE as low as 4.20% (Table 4.2), while MLP and GM(1,1) exhibit large fluctuations, failing to capture the stable load pattern during working hours. The results shown in Figs. 10(a) and (b) are similar and will not be repeated here.



(a)



(b)

Figure 10: Predicted Power Load in Public and Residential Buildings

The TCN's superiority stems from its specialized structural design: Dilated causal convolution effectively captures long-term temporal dependencies, which is reflected in the visual alignment of predicted and true curves across all time steps. Cross-building lagged features leverage spatiotemporal correlations between the four building types, enabling the model to adapt to mutual influences, a capability absent in MLP and GM(1,1). Residual connections and weight normalization ensure stable training of deep networks, resulting in consistent high accuracy across building types (all $R^2 > 0.98$, Table 3) and time resolutions.

Table 3: Detailed Performance of All Models by Building Type

Building Type	Model	MAE (kW)	RMSE (kW)	MAPE (%)
Commercial	**TCN**	17.58	24.35	7.88
	MLP	139.89	195.12	23.62
	GM(1,1)	247.89	305.86	46.53
Office	**TCN**	14.40	22.75	4.20
	MLP	136.10	199.59	42.88
	GM(1,1)	210.05	298.23	62.22
Public	**TCN**	10.62	14.19	6.47
	MLP	78.44	89.05	61.40
	GM(1,1)	84.97	108.41	51.12
Residential	**TCN**	8.70	11.27	3.33
	MLP	42.82	61.91	14.05
	GM(1,1)	66.38	81.06	24.19

MLP's Limitation:

While MLP outperforms GM(1,1), its lack of time-series-specific processing leads to obvious shortcomings in both data and visualization: From Table 3, MLP's MAE for Commercial buildings (139.89 kW) is 7.96 times that of TCN, and its RMSE (195.12 kW) is 8.01 times larger. Visually， MLP's predicted curves lag behind the true curves in all building types, for example, in Residential buildings, MLP fails to respond to sudden load increases (around 20,000 time steps), resulting in a 14.05% MAPE (vs. TCN's 3.33%).

GM(1,1)'s Shortcoming:

As a linear model, GM(1,1) performs the worst, with its limitations starkly revealed by both metrics and visuals: Table 3 shows GM(1,1) has the highest error across all building types—Office buildings have the worst performance (MAE = 210.05 kW, MAPE = 62.22%), as GM(1,1) cannot adapt to the sharp load changes during workday mornings and evenings. Visual comparison charts highlight GM(1,1)'s systemic bias: its predicted curves are overly smooth, failing to capture any load volatility. For example, in public buildings, GM(1,1) underestimates peak load by over 300 kW and overestimates off-peak load by 150 kW, resulting in a 51.12% MAPE. GM(1,1)'s core reliance on linear trend fitting makes it only suitable for stable, low-volatility data.

5.3.3 TCN's Performance Under Key Scenarios

Table 4: TCN Performance Under Different Time Resolutions and Building Types

Building Type	R^2	Time Resolution	R^2
Official	0.993	1 hour	0.978
Commercial	0.994	30 minutes	0.969
Public	0.982	5 minutes	0.953
Residential	0.982	/	/

According to Table 4, the TCN maintains $R^2 > 0.98$ for all building types, demonstrating strong adaptability to distinct load patterns—from stable Official buildings to random Residential buildings. Even for high-volatility 5-minute data, the TCN's R^2 remains above 0.95, confirming robustness to multi-resolution load data.

5.4 Sensitivity Analysis of TCN

To verify robustness, sensitivity analysis is conducted by adjusting key hyperparameters, using the coefficient of variation (CV):

$$CV = \frac{|R^2_{new} - R^2_{original}|}{R^2_{original}} \times 100\% \quad (17)$$

Where $R^2_{original} = 0.967$ (TCN's baseline performance).

Table 5: Sensitivity to Dropout Rate and Learning Rate

Dropout Rate	R^2	CV (%)	Learning Rate	R^2	CV (%)
0.0	0.958	0.93	1e-3	0.948	1.96
0.1 (Original)	0.967	0.00	1e-4 (Original)	0.967	0.00
0.2	0.956	1.14	1e-5	0.935	3.31
0.3	0.942	2.58	/	/	/

Sensitivity to Dropout Rate (0.0 – 0.3)

From Table 5, $CV < 3\%$ indicates low sensitivity and strong robustness to regularization changes.

Sensitivity to Learning Rate (1e-5 – 1e-3)

From Table 5, moderate sensitivity within a reasonable range, reducing hyperparameter tuning difficulty.

6 Interpretability Analysis

To address the black-box problem of the spatiotemporal modeling for electric power load forecasting, this section conducts a systematic interpretability analysis based on ablation experiments. By controlling variables to isolate the impact of each core component and different time window settings, we clarify the contribution mechanism of temporal correlation features and the rationality of model structure design. The analysis integrates quantitative evaluation metrics, gradient-based feature importance calculation, and visual verification to ensure the conclusions are reliable and intuitive.

6.1 Design of Ablation Experiments

The proposed temporal correlation-aware load forecasting model is built on Temporal Convolutional Network (TCN), with the core design focusing on the impact of historical load data from different time windows on prediction performance. The ablation experiments follow the "single-variable control" principle, taking the basic TCN model (using full-scale historical data) as the benchmark, and gradually masking historical data of specific time windows to analyze the contribution of temporal features in different time scales.

Ablation Scheme and Feature Importance Calculation

Hourly Window Ablation: Mask historical load data of 1.5h/3h/6h/12h/24h to analyze hourly scale contributions.

Time Attribute Ablation: Mask weekday/date/date & month/ all-time attributes to verify attribute feature value.

Feature Importance Calculation:

$$Imp_{\text{history}} = \text{history} \odot \nabla_{\text{history}} \hat{y} \tag{18}$$

where $Imp_{history}$ is the temporal feature vector, $\nabla_{history}\hat{y}$ is the prediction gradient, and $\odot$ denotes element-wise product. Larger values indicate higher feature importance.

Masking operation:

$$x_{t-\text{masked}} = x_t \times M \tag{19}$$

M is binary masking matrix, 0 for masked features.

6.2 Results of Ablation Experiments

6.2.1 Visual Result Analysis

Hourly Window Ablation (Fig. 11) shows that TCN Basic’ s prediction curve is closest to the true value, while TCN 12h deviates the most, with the error increasing as masked windows extend. Time Attribute Ablation (Fig. 12) demonstrates that TCN Masked Time Info exhibits the largest deviation, and performance degrades cumulatively with more masked attributes.

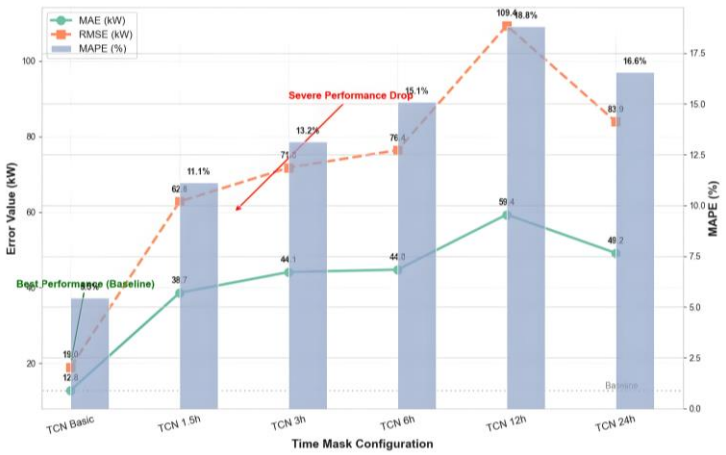


Figure 11: Performance Trend of TCN Models Under Different Time Masks

As shown in Fig. 13, the last column of the four regions is nearly zero, meaning it encodes no practical information and the 6-dimensional embedding matrix is slightly redundant—an

outcome of the model actively reducing redundancy via weight_decay to avoid overfitting. Thus, the remaining 5 columns provide sufficient and independent information.

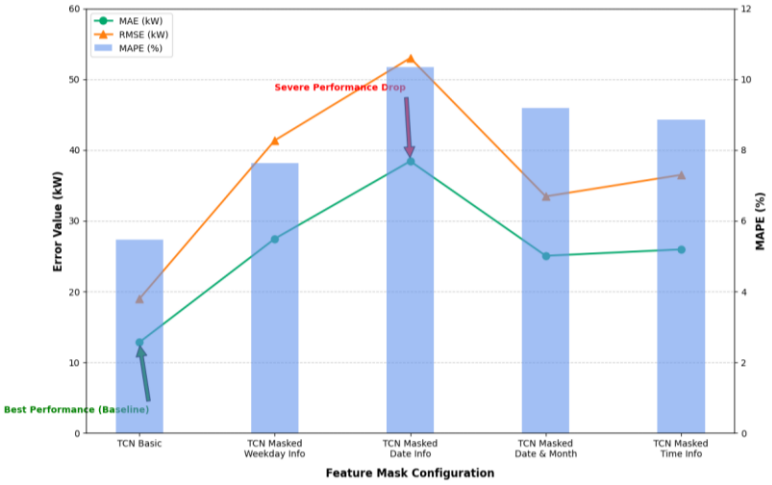


Figure 12: Performance Trend of TCN Models Under Different Feature Masks

While the "Commercial & Public" and "Office & Residential" rows of the Embedding matrix appear similar, further comparison with prior data analysis results shows these regions pairs lack consistency in absolute power values or daily trends (e.g., Commercial aligns with Office in absolute value and with Residential in trends). This indicates the model retains inter-region distinguishability rather than merging similar regions.

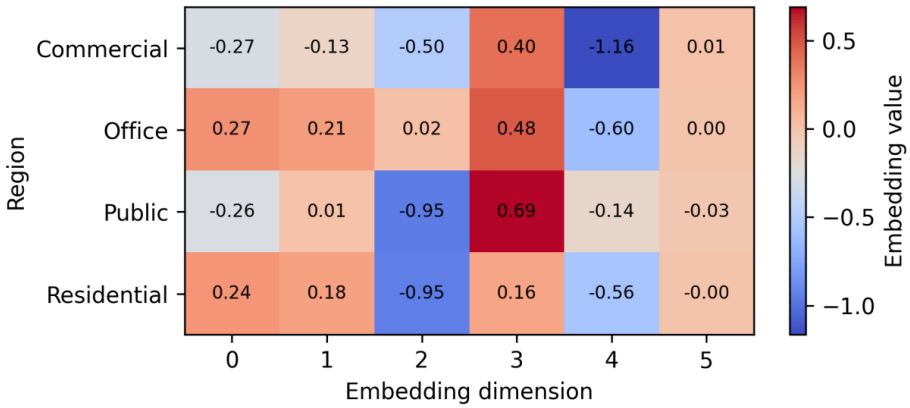


Figure 13: Region Embedding Heatmap

Since Power History is the main source of the model's trend fitting, the Embedding component does not generate absolute power values or trends. Instead, it fine-tunes predictions from Power History by fitting the residual between Power History-derived results and true values. Consequently, the similarity between the two aforementioned region pairs' Embeddings stems from residual similarity, not trend similarity.

6.3 Interpretability Analysis and Discussion

Key Feature Contributions

Regarding Hourly Windows, the 12-hour window is critical, with a ΔMAPE of 13.35% and an Imp of 0.32, as it contains core short-term periodicity, whereas short-term ($\leqslant$3h) features have limited impact, with ΔMAPE $\leqslant$7.69%. For Time Attributes, all attributes together are indispensable, showing a ΔMAPE of 11.75%; weekday info alone has minimal impact with a ΔMAPE of 4.36%, while date and month contribute cumulatively.

Model Design Rationality

The TCN model effectively integrates multi-scale temporal features, and its performance sensitivity matches the industrial park load characteristics, such as 12-hour shifts and time-specific production plans. TCN Basic's optimal performance $\mathrm{MAPE} = 5.46\%$ confirms the value of comprehensive feature utilization.

Practical Implications

In resource-constrained scenarios, prioritize retaining 12-hour window and time attribute data. For shift-based production parks, enhance the 12-hour window weight, and for monthly cycle scenarios, emphasize date and month info. When historical data is incomplete, supplement 12-hour and time attribute data first.

7 Insights

7.1 Spatiotemporal Correlation-Driven Feature Prioritization

The Pearson correlation analysis reveals distinct correlation patterns between building types: commercial and office buildings ($r\approx0.85$) exhibit strong synchronization, public buildings show moderate correlation with both ($r\approx0.6$-0.7), while industrial buildings are relatively independent ($r\approx0.3$-0.5). This indicates that load forecasting should adopt a "grouped feature engineering" strategy—for high-correlation building pairs, share lagged load features to enhance mutual information utilization; for independent building types, focus on internal temporal patterns to avoid noise interference.

7.2 Time Resolution Adaptability Principle

The TCN model maintains $R^2>0.95$ across 5-minute, 30-minute, and 1-hour resolutions, but performance degrades slightly with higher granularity. This reflects that short-term high-frequency load fluctuations are more affected by random factors, while medium-to-low resolution data better captures inherent periodicity. Practical deployment should prioritize 30-minute resolution for balance between accuracy and computational efficiency, reserving 5-minute resolution only for key load monitoring points.

7.3 Critical Time Window Effect

Ablation experiments confirm that the 12-hour historical window contributes most significantly to prediction accuracy (Δ MAPE=13.35%), which aligns with industrial park operation characteristics. The 24-hour window provides marginal gains, indicating that load patterns within one workday are sufficient for short-term forecasting. This suggests optimizing input sequences to 12-24 hours can reduce computational cost without accuracy loss.

8 Strengths and Weaknesses

8.1 Strengths

1 **Targeted spatiotemporal modeling:** Fully leverages inter-building correlation characteristics, designing cross-building lagged features that align with actual load synchronization patterns, outperforming models ignoring spatial dependencies.

2 **Robust multi-resolution adaptation:** The TCN's dilated convolution structure effectively captures load patterns across 5-minute to 1-hour resolutions, with R^2 consistently above 0.95, demonstrating strong practical applicability.

3 **Systematic interpretability design:** Combines ablation experiments, gradient-based feature importance, and visualization analysis to clarify the contribution mechanism of key components, addressing the black-box problem of spatiotemporal models.

4 **Rigorous performance verification:** Compares with MLP and GM(1,1) under unified evaluation metrics, with TCN's MAE (12.83 kW) only 8.4%-12.9% of benchmarks, and provides building-specific error analysis for targeted optimization.

8.2 Weaknesses

First, there is a high dependency on data quality and continuity: The model assumes the availability of complete, high-resolution historical data from all building types. Second, insufficient handling of extreme scenarios is another shortcoming: The model assumes stable operating conditions and cannot adequately predict sudden, anomalous load spikes caused by emergency production, extreme weather events, or major equipment failures, which may lead to significant forecast errors during such events.

References

[1] Ge, Q., Guo, C., Jiang, H., Lu, Z., Yao, G., Zhang, J., & Hua, Q. (2020). Industrial power load forecasting method based on reinforcement learning and PSO-LSSVM. IEEE transactions on cybernetics, 52(2), 1112-1124.

[2] KAILE. WEN ZHOU (LULU.). (2023). SMART ENERGY MANAGEMENT: Data Driven Methods for Energy Service Innovation. SPRINGER VERLAG, SINGAPOR.

[3] Zaballa, O., Álvarez, V., & Mazuelas, S. (2024, November). Multi-task Online Learning for Probabilistic Load Forecasting. In 2024 IEEE Sustainable Power and Energy Conference (iSPEC) (pp. 280-285). IEEE.

[4] Xu, L., & Zhu, Y. (2025). Diffusion-assisted Model Predictive Control Optimization for Power System Real-Time Operation. arXiv preprint arXiv:2505.08535.

[5] Yaprakdal, F., & Varol Arısoy, M. (2023). A multivariate time series analysis of electrical load forecasting based on a hybrid feature selection approach and explainable deep learning. Applied Sciences, 13(23), 12946.

[6] R. Singh, A., Kumar, R. S., Bajaj, M., Khadse, C. B., & Zaitsev, I. (2024). Machine learning-based energy management and power forecasting in grid-connected microgrids with multiple distributed energy sources. Scientific Reports, 14(1), 19207.

[7] Matos, M., Almeida, J., Gonçalves, P., Baldo, F., Braz, F. J., & Bartolomeu, P. C. (2024). A machine learning-based electricity consumption forecast and management system for renewable energy communities. Energies, 17(3), 630.

[8] Zhang, H., Zhang, G., Zhao, M., & Liu, Y. (2024). Load forecasting-based learning system for energy management with battery degradation estimation: A deep reinforcement learning approach. IEEE Transactions on Consumer Electronics, 70(1), 2342-2352.

[9] Zhou, K., Hu, D., Hu, R., & Zhou, J. (2023). High-resolution electric power load data of an industrial park with multiple types of buildings in China. Scientific Data, 10(1), 870.

[10] Li, F., Yu, X., Tian, X., & Zhao, Z. (2021, March). Short-term load forecasting for an industrial park using LSTM-RNN considering energy storage. In 2021 3rd Asia energy and electrical engineering symposium (AEEES) (pp. 684-689). IEEE.

Appendix

The source code repository is:

https://github.com/CoderXL1/COMAP-School-Contest-T3

Report on Use of AI

1. OpenAI ChatGPT (January 19, 2026 version, ChatGPT-5)

Query1: How to use PP-plot to test data normality?

Output: PP-plot compares the empirical cumulative probability of sample distribution with the cumulative probability of theoretical distribution. First, sort the sample, estimate normal distribution parameters, calculate empirical and theoretical distribution probabilities respectively, then plot points (qi, pi) with reference line y=x. Judge normality by observing if points are close to the line; it focuses on overall fitting but is less sensitive to tails.

2. OpenAI ChatGPT (January 19, 2026 version, ChatGPT-5)

Query2: How to draw PP-plot and QQ-plot for a pandas Series?

Output: First preprocess data by dropping missing values. For QQ-plot, use SciPy's stats.probplot or statsmodels' sm.qqplot. For PP-plot, use statsmodels' ProbPlot.ppplot or manual calculation (sort values, compute empirical/theoretical probabilities, plot scatter and reference line). It’s recommended to plot both side by side for comparison.

3. OpenAI ChatGPT (January 19, 2026 version, ChatGPT-5)

Query3: Can SciPy draw PP-plot directly?

Output: SciPy has no built-in direct PP-plot function; stats.probplot only generates QQ-plot. PP-plot can be drawn via statsmodels' ProbPlot.ppplot or manual implementation with SciPy (calculate CDF/quantile of theoretical distribution, then plot empirical vs theoretical probabilities).