

Transformer-based correction scheme for short-term bus load prediction in holidays

Tang Ningkai^{1, 2} Lu Jixiang^{1, 2} Chen Tianyu² Shu Jiao^{1, 2} Chang Li² Chen Tao³

(¹ State Key Laboratory of Technology and Equipment for Defense Against Power System Operational Risks, Nanjing 211106, China)

(² NARI Group Corporation (State Grid Electric Power Research Institute), Nanjing 211106, China)

(³ School of Electrical Engineering, Southeast University, Nanjing 211189, China)

Abstract: To tackle the problem of inaccurate short-term bus load prediction, especially during holidays, a Transformer-based scheme with tailored architectural enhancements is proposed. First, the input data are clustered to reduce complexity and capture inherent characteristics more effectively. Gated residual connections are then employed to selectively propagate salient features across layers, while an attention mechanism focuses on identifying prominent patterns in multivariate time-series data. Ultimately, a pre-trained structure is incorporated to reduce computational complexity. Experimental results based on extensive data show that the proposed scheme achieves improved prediction accuracy over comparative algorithms by at least 32.00% consistently across all buses evaluated, and the fitting effect of holiday load curves is outstanding. Meanwhile, the pre-trained structure drastically reduces the training time of the proposed algorithm by more than 65.75%. The proposed scheme can efficiently predict bus load results while enhancing robustness for holiday predictions, making it better adapted to real-world prediction scenarios.

Key words: short-term bus load prediction; Transformer network; holiday load; pre-training model; load clustering

DOI: 10.3969/j.issn.1003-7985.2024.03.010

Power scheduling poses a significant challenge in power system automation^[1]. Accurate prediction is essential for ensuring grid safety and economic scheduling. The power grid forecasting problem can be divided into two main categories: power generation prediction^[2] and load prediction^[3]. Specifically, precise bus load prediction provides critical input for generator commitment and economic dispatch, ensuring the security and economic efficiency of regional power grid operations^[4]. During holidays, electricity consumption behaviors undergo significant changes, resulting in fluctuations in bus

loads that deviate substantially from conventional trends^[5]. These complex variations present unique challenges for load prediction models. Factors such as weather conditions, special public events, and population shifts^[6] interact to produce bus load profiles with high uncertainty^[7]. Conventional statistical methods often fail to capture these complex holiday load characteristics. Therefore, it is imperative to develop advanced bus load prediction techniques for holidays to adapt to the intrinsic complexity and improve prediction accuracy.

Deep learning has recently shown promising performance in bus load prediction problems. Recurrent neural networks (RNNs), like long short-term memory (LSTM) networks, outperform traditional statistical models by extracting temporal features and modeling sequential dependencies in time-series data^[8]. However, RNNs have limitations in capturing long-term dependencies owing to the vanishing gradient problem.

More recently, the Transformer architecture has demonstrated state-of-the-art results across various sequential modeling tasks by effectively addressing these issues^[9]. The Transformer's self-attention mechanism establishes direct connections between all positions and learns contextual representations through weighted aggregation. This allows for the implicit modeling of both local and global dependencies regardless of distance. Transformers leverage residual connections and layer normalization to overcome the vanishing gradient problem and enhance model performance. Furthermore, by leveraging pre-training on massive datasets followed by fine-tuning on specific tasks, Transformer models achieve superior performance and generalization capabilities compared to traditional training methods.

Originally designed for natural language processing tasks^[10], the Transformer architecture has also noted great success in time-series forecasting^[11]. Compared to RNNs and convolutional neural networks (CNNs), Transformers can better capture long-term dependencies and facilitate parallel computations. Several well-known Transformer variations have been developed for time-series problem modeling. For instance, the Informer^[12] introduced ProbSparse self-attention to streamline attention computation. Another example is Autoformer^[13], a Transformer-based

Received 2024-05-08, **Revised** 2024-06-11.

Biographies: Tang Ningkai (1987—), male, doctor, tangningkai@sgepri.sgcc.com.cn; Chen Tao (corresponding author), male, doctor, associate professor, taoc@seu.edu.cn.

Foundation item: State Grid Science & Technology Program (No. 1400-202140341A-0-0-00).

Citation: Tang Ningkai, Lu Jixiang, Chen Tianyu, et al. Transformer-based correction scheme for short-term bus load prediction in holidays [J]. Journal of Southeast University (English Edition), 2024, 40(3): 304 – 312. DOI: 10.3969/j.issn.1003-7985.2024.03.010.

network that employs fast Fourier transform for temporal decomposition and correlation. To further reduce training time, Zhou et al.^[14] applied low-rank approximation in frequency, achieving state-of-the-art performance with $O(N)$ complexity.

The Transformer has also been applied to forecasting problems in the electrical sector. For instance, researchers used the Temporal Fusion Transformer^[15] (TFT), a variation of the Transformer developed by Google, to provide day-ahead photovoltaic power forecasting^[16]. Based on spatiotemporal information, researchers came up with a multi-step wind power forecasting algorithm^[17]. Within the scope of load prediction, several papers^[18-19] have focused on system load forecasting. However, owing to computational costs and the constant mutation of load trends, the application of Transformer on bus load prediction remains underexplored, presenting an opportunity to improve power system reliability^[20].

In this paper, we propose a Transformer-based scheme to accurately predict short-term electricity bus loads, especially during holidays. To capture the unique characteristics of holiday loads, the presented scheme incorporates several innovations. First, it categorizes the input data by type, which reduces training complexity and leverages inherent periodicities. Different holidays are distinguished to isolate their distinct effects from normal days. As part of the Transformer-based algorithm, gated residual connections are employed to selectively pass on meaningful features (such as holiday types) across layers, focusing the model on holiday-specific patterns. Attention mechanisms enable concentrating on salient patterns within lengthy multivariate time series, while the self-attention

layer establishes global connections to learn contextual representations. Ultimately, pre-training the model and freezing some layers significantly cut down on training time, making the approach practical for real-world use.

1 Data Analysis and Problem Statement

1.1 Data description

The real-world bus load dataset used for this research is collected from 1 399 buses across multiple substations in Jiangsu Province, China. The data span from June 2021 to May 2023, with bus loads measured every 15 min. Additionally, the dataset includes weather forecast data such as temperature, humidity, wind speed, and solar irradiance, respectively. Seasonal impacts and trends are clearly visible in the data of certain buses, showing heavy loads in summer owing to air conditioning and light loads during holidays such as Chinese New Year (see Fig. 1). Furthermore, most buses exhibit intraday fluctuations driven by consumer demand and power generation patterns.

Accurate forecasting of bus load plays a crucial role in power system operation and planning, enabling efficient resource allocation, load balancing, and grid stability. However, bus load prediction is a challenging task owing to the complex interplay of various factors, including weather conditions, human activities, and stochastic demand patterns. This real-world dataset forms an ideal test-bed for developing and evaluating advanced forecasting models. Successfully modeling bus load can lead to significant economic and environmental benefits through optimized energy management and reduced carbon emissions, underscoring the practical importance of this research.

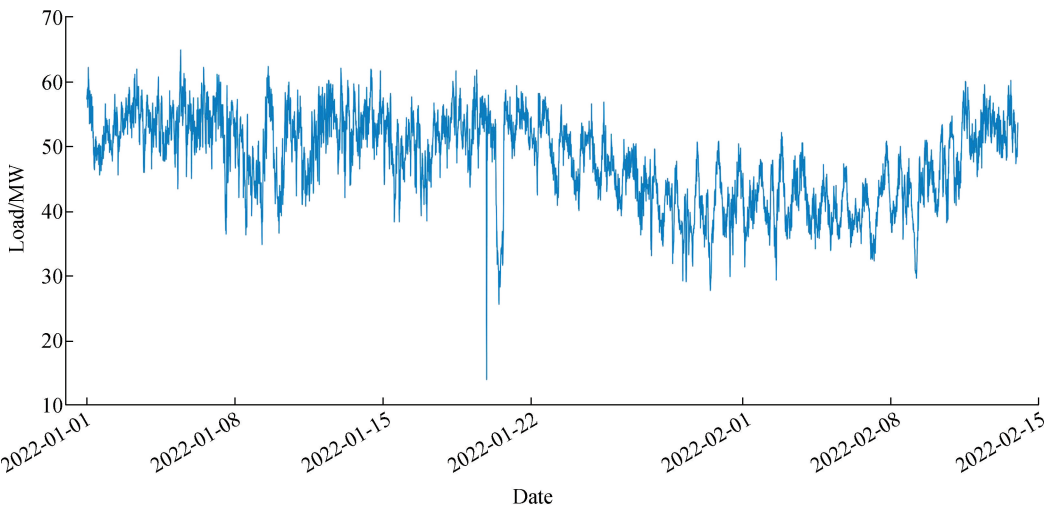


Fig. 1 Bus load during the Chinese New Year period (2022-01-31—2022-02-06)

1.2 Problem formulation

The problem of short-term bus load prediction can be formulated as follows. Given a set of historical bus load data $D = (x_t, y_t), t = 1, 2, \dots, T$, where x_t represents the

input features at time t , and y_t is the corresponding bus load, the goal is to learn a function $f: X \rightarrow Y$ that maps the input features $x_t \in X$ to the predicted bus load $\hat{y}_t \in Y$, minimizing the error between the predicted and actual bus load values.

In the study, the overall precision, which includes all 1 399 buses, is considered. The input features x_t comprise both historical bus load data and exogenous variables related to weather conditions. Specifically, the input features can be represented as: $x_t = \{L_{t-96}, L_{t-97}, \dots, L_{t-95-p}, W_t, O_t\}$, where L_{t-i} denotes the historical bus load data at time $t-i$, with $i \in \{1, 2, \dots, p\}$ and p is the number of considered lagged bus loads. Since day-ahead prediction is performed in this paper and the resolution is 15 min, x_t starts with L_{t-96} . W_t represents the weather forecast data at time t , such as temperature, solar radiation, humidity, and wind speed. O_t stands for other possible features to be considered, such as month, day of year, and prediction time.

The target variable y_t corresponds to the bus load at time t which is aimed to be predicted accurately. The goal is to learn the mapping function f which minimizes a given loss function l between the predicted bus load $\hat{y}_t = f(x_t)$ and the actual bus load y_t :

$$\min_f \sum_{t=1}^T l(y_t, f(x_t)) \quad (1)$$

According to previous studies^[21-23], the possible evaluation metrics for load forecasting results generally include root mean squared error (RMSE), mean absolute percentage error, and forecast accuracy. The definition of RMSE is shown in the following equation:

$$r_{\text{RMSE}} = \sqrt{\frac{\sum_{j=1}^N (y_{\text{act}}(j) - y_{\text{pred}}(j))^2}{N}} \quad (2)$$

where N represents the total number of predicted points; $y_{\text{act}}(j)$ denotes the actual load value at the j -th point; $y_{\text{pred}}(j)$ represents the predicted load value at the j -th point.

Owing to significant variations in load among different buses and the possibility of load actual values approaching or equaling zero on certain buses, the evaluation or the loss function defined in Eq. (1) relies on RMSE as the main criterion for assessment.

2 Proposed Scheme

2.1 Description of the proposed Transformer-based algorithm

Differently from the vanilla Transformer model, the proposed architecture of the Transformer-based model is meticulously designed to integrate information from feature metadata, such as holiday information in this research. Fig. 2 displays the overall design of the proposed Transformer-based network.

From Fig. 2, it is evident that the proposed Transformer-based algorithm deviates significantly from the vanilla Transformer architecture in its structural design. The proposed algorithm adopts the gated residual network

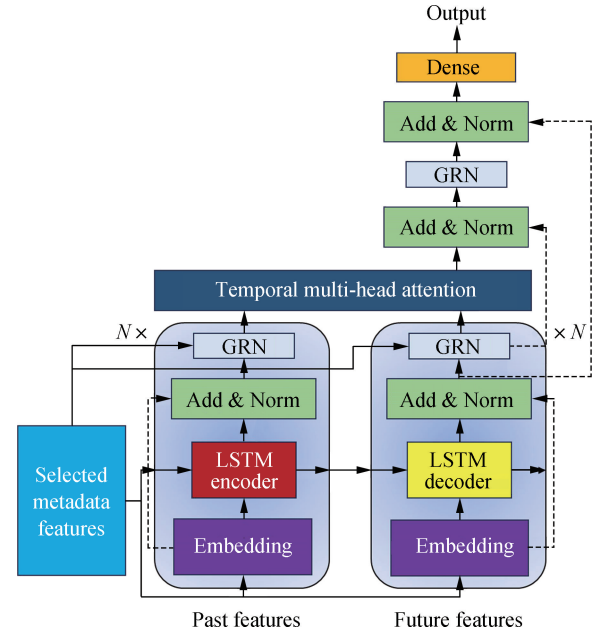


Fig. 2 Proposed Transformer-based network

(GRN)^[13] to selectively produce distinct context vectors. Additionally, it employs an improved LSTM-based positional encoding strategy that more effectively encodes the positional information of time steps, strengthening the algorithm's ability to model time-series data. The algorithm further introduces a novel temporal multi-head attention mechanism that effectively captures both short-term and long-term dependencies. This mechanism enhances trend learning and cyclical patterns in time-series data by capturing both local and global patterns, thereby increasing the model's expressive power.

2.1.1 Gated residual network

GRN modules are adopted to produce distinct context vectors. These context vectors are then wired into various locations in the decoders, where certain vectors play an important role in processing. The designed GRN is shown in Fig. 3. The exponential linear unit (ELU) is defined as follows:

$$\text{ELU}(x) = \begin{cases} x & x > 0 \\ \alpha(e^x - 1) & x \leq 0 \end{cases} \quad (3)$$

where α is a hyperparameter that controls the slope of the negative region. The ELU function is an activation func-

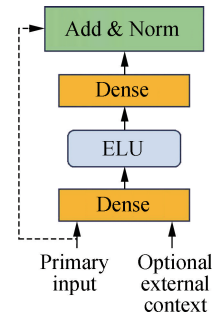


Fig. 3 Gated residual network

tion that maps negative inputs to the negative region using exponential and linear transformations while preserving positive inputs.

2.1.2 Temporal multi-head attention

In the proposed model, a self-attention mechanism is employed to capture long-range temporal dependencies across different time steps of bus load and weather data. This mechanism is a modification of the multi-head attention used in Transformer-based architectures, specifically designed to enhance temporal awareness. Attention mechanisms, in general, scale values $V \in \mathbf{R}^{N \times d_v}$ based on the relationships between keys $K \in \mathbf{R}^{N \times d_m}$ and queries $Q \in \mathbf{R}^{N \times d_m}$ as follows:

$$\text{Attention}(Q, K, V) = A(Q, K)V \quad (4)$$

where $A(\cdot)$ is a normalization function. A commonly used normalization is the scaled dot-product attention:

$$A(Q, K) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d_{\text{attn}}}}\right) \quad (5)$$

The multi-head attention mechanism is introduced to enhance the learning capacity of the standard attention mechanism. The former employs multiple independent attention heads to capture different temporal representation subspaces:

$$\text{MultiHead}(Q, K, V) = [H_1 \ H_2 \ \dots \ H_{m_h}]W_H \quad (6)$$

$$H_h = \text{Attention}(QW_h^Q, KW_h^K, VW_h^V) \quad (7)$$

where $W_h^Q \in \mathbf{R}^{d_{\text{model}} \times d_m}$, $W_h^K \in \mathbf{R}^{d_{\text{model}} \times d_m}$, $W_h^V \in \mathbf{R}^{d_{\text{model}} \times d_v}$ are head-specific weight matrices for queries, keys, and values, respectively; and $W_H \in \mathbf{R}^{(m_h d_v) \times d_{\text{model}}}$ linearly combines the concatenated temporal outputs from all heads H_h .

However, using different value vectors in each head means that the attention weights alone cannot fully indicate the importance of specific features. To address this issue, the proposed model modifies the multi-head attention mechanism to share value vectors across all heads and employs additive aggregation of the attention outputs.

$$\text{TemporalMultiHead}(Q, K, V) = \tilde{H}W_H \quad (8)$$

$$\tilde{H} = \tilde{A}(Q, K)VW_V \quad (9)$$

$$\tilde{A}(Q, K) = \frac{1}{H} \sum_{h=1}^{m_h} \text{Attention}(QW_h^Q, KW_h^K, VW_h^V) \quad (10)$$

where $W_V \in \mathbf{R}^{d_{\text{model}} \times d_v}$ are shared value weights across all heads; and $W_H \in \mathbf{R}^{d_{\text{attn}} \times d_{\text{model}}}$ is used for the final linear mapping.

2.1.3 Information extended sequence-to-sequence layer

The proposed approach aims to leverage local context and handle cases with observed inputs by employing a sequence-to-sequence model. In bus load data, significant

points often relate to their surrounding values, such as anomalies, change points, or cyclical patterns. By incorporating features in addition to point-wise values, attention-based architectures can potentially achieve performance improvements by leveraging local contexts like contiguous load and weather data.

However, when observed inputs are present, the number of past and future inputs may differ, rendering the use of local context features unsuitable. To address this challenge, the proposed solution uses a sequence-to-sequence model, which can naturally handle these differences. Specifically, the model takes data from time $t - k$ to t , denoted by $\tilde{\xi}(t - k : t)$ as input to the encoder and all future input $\tilde{\xi}(t + 1 : t + \tau_{\text{max}})$ to the decoder. This process generates a set of uniform temporal features, denoted by $\varphi(t, n) \in \{\varphi(t, -k), \varphi(t, -k + 1), \dots, \varphi(t, \tau_{\text{max}})\}$, where n is a position index. These temporal features serve as inputs to the decoder itself.

Unlike traditional Transformer-based architectures, this architecture uses an LSTM-based architecture to serve as an alternative to standard positional encoding, providing an appropriate inductive bias for the time ordering of the load and weather data. Moreover, selected metadata is allowed to influence local processing. In addition, a skip connection over this layer is employed to mitigate potential gradient issues.

2.1.4 GRN-based feed-forward layer

Different from the vanilla Transformer, the feed-forward layer of the proposed model is designed based on the GRN. By applying additional non-linear processing to the outputs of the self-attention layer, the GRN weights are shared across the entire layer. This design allows for more complex temporal relationship extraction from the attention outputs.

2.2 Description of the proposed methodologies

2.2.1 Load data clustering

In the proposed scheme, load data is first clustered using the K-means before the training stage. The objective is to identify patterns and group similar load profiles together, allowing each incoming pre-trained model to train on statistically similar data for better efficiency. The features used for clustering include the mean, maximum, minimum, and standard deviation of load data from different buses.

2.2.2 Frozen neuron mechanism

To mitigate overfitting and reduce computational complexity when training models on different buses, a frozen neuron mechanism is implemented in the pre-trained Transformer models. This technique leverages transfer learning by preserving the knowledge gained during pre-training on a large corpus while allowing the final layer to adapt to particular target tasks and domains.

Formally, θ represents the full set of trainable parameters in the original Transformer model. The hyperparam-

eter set θ is partitioned into frozen parameters θ_f and trainable parameters θ_t , such that $\theta = \theta_f \cup \theta_t$. During fine-tuning, the gradients are computed only with respect to θ_t , effectively freezing the weights of θ_f .

By freezing a portion of the model parameters, the frozen neuron mechanism reduces the computation graph during the backward propagation step, leading to lower memory requirements and more efficient training iterations. Furthermore, it mitigates the risk of overfitting by limiting the model capacity, thereby acting as an implicit regularizer.

2.2.3 Other methodologies

During the data preprocessing stage, the Chebyshev method is utilized to remove outliers, ensuring data quality. Furthermore, for categorical features such as month and holiday type, one-hot encoding technique is used to transform them into numerical vectors. These trivial but useful processing techniques will be discussed below to demonstrate their capability to enhance the model generalization capability and prediction accuracy.

3 Validation

3.1 Experimental setup

The experiment was conducted on a Lenovo SR650 server equipped with two Intel Xeon 4216 @ 2.1 GHz processors, 256 GB DDR4 memory, and an NVIDIA Tesla V100 GPU with 32 GB VRAM. The machine

learning environment was set up using Python 3.7, with implementation carried out on PyTorch 1.8.

The dataset was split into training, validation, and test sets. The training set, consisting of 80% of each bus time-series data, was used to fit the model parameters. The validation set, comprising 10% of the data, was used for hyperparameter tuning. The test set, containing the remaining 10%, was used for the final model evaluation. This multi-year dataset, with a 15-min resolution from 1 399 real-world buses, provides a challenging and representative validation for the bus load forecasting model. The diversity in locations, seasonal trends, and fluctuations rigorously demonstrate the capability of the proposed model.

3.2 Data preprocessing

Bus load data often includes extreme values that are undesirable, necessitating additional processing steps. The diverse patterns in the distribution of bus load curves (see Fig. 4) make outlier detection challenging for methods assuming a normal distribution, such as the commonly used z-score method. While both the z-score method and the Chebyshev method utilize mean and standard deviation to identify anomalies, the z-score method’s reliance on normal distribution can lead to inaccurate results, given the data characteristics.

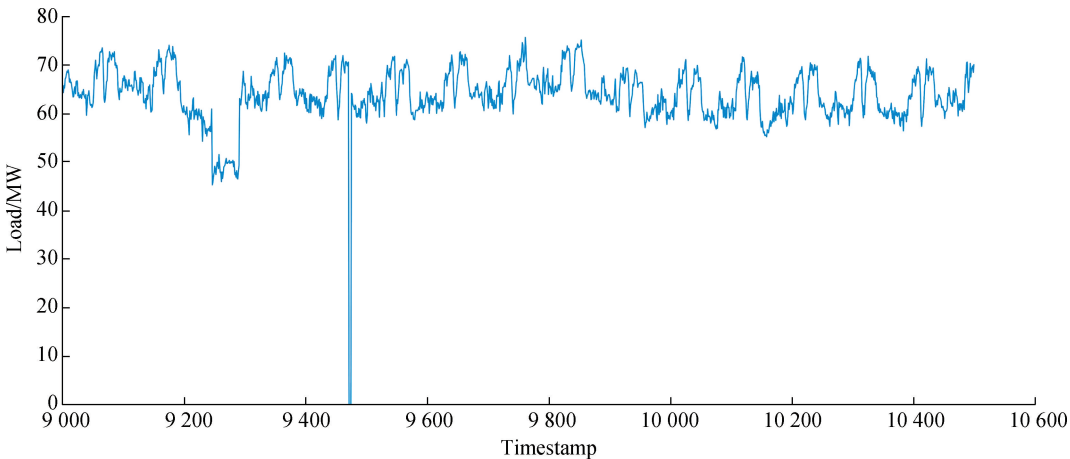


Fig. 4 Bus load with outliers

The load distribution of each bus in the dataset displays various non-normal shapes, including heavy tails (see Fig. 5(a)), skewness (see Fig. 5(b)), and multimodality (see Fig. 5(c)). Factors such as factory demand, integrated distributed renewable energy resources, and repair schedules contribute to these patterns. As a result, assuming a normal distribution can lead to the misidentification of genuine outliers or failure to capture significant deviations in the bus load data.

To address this issue, the Chebyshev method, which does not assume any specific distribution, is utilized. It

identifies anomaly points based on the mean and standard deviation of the data, aiming to identify outliers that deviate significantly from the mean.

After removing outliers, load clustering is performed. The clustering analysis results are summarized in Table 1. Cluster 0 exhibits an average bus load with a moderate variation, having a mean value of 37.543 673. Cluster 1 has the lowest mean of 7.063 455, indicating lighter bus loads with minimal variation. Cluster 2 is characterized by a relatively higher average of 64.617 019 but shows considerable variation. Cluster 3 has the highest mean

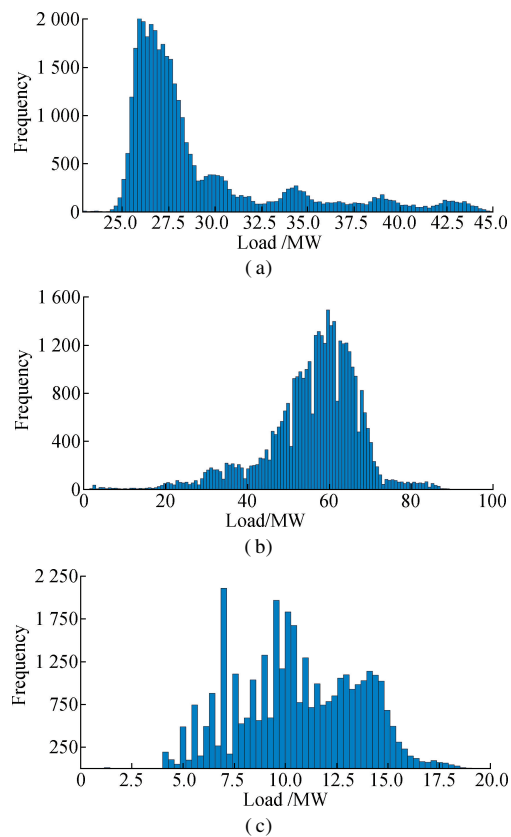


Fig. 5 Non-normal bus load distribution. (a) Heavy tails; (b) Skewness; (c) Multimodality

Table 1 Cluster statistics of the bus load

Cluster	Feature name	Mean	Min	Max	Std
0	Mean	37.543 673	−16.758 643	62.044 625	10.834 189
	Max	70.574 910	43.687 900	111.327 400	12.617 533
	Min	5.025 613	−140.843 600	38.197 105	16.143 793
	Std	14.487 689	2.738 775	61.762 897	5.709 163
1	Mean	7.063 455	−15.944 471	30.439 850	8.798 253
	Max	21.972 385	0.000 000	58.222 900	15.287 870
	Min	−6.380 830	−82.146 500	22.874 500	14.865 714
	Std	5.853 832	0.000 000	32.099 781	5.499 260
2	Mean	64.617 019	20.786 855	87.796 465	10.524 663
	Max	110.225 025	86.103 200	141.494 600	11.949 742
	Min	17.522 436	−62.070 100	48.944 500	13.379 479
	Std	19.733 701	4.462 211	48.188 159	6.106 540
3	Mean	95.565 477	46.219 579	170.637 302	17.141 038
	Max	154.038 688	129.508 500	246.395 800	20.364 005
	Min	31.682 036	−80.797 200	123.979 500	22.618 865
	Std	25.373 632	6.096 517	77.428 246	9.849 551

and represents large bus loads, with relatively unobtrusive variation considering the high mean. These 4 different clusters will be used as the base data source for the pre-trained model.

3.3 Feature selection and one-hot coding

In the proposed approach, relevant features were carefully selected to capture patterns and dependencies in the bus load dataset. Feature selection was guided by domain knowledge and exploratory data analysis. We first per-

formed correlation analysis and feature importance ranking to identify the most influential features. From this analysis, bus load from the previous week and weather features such as temperature and humidity were selected. Domain knowledge further motivated the inclusion of holiday information and temporal features such as month, day, hour, and minute to capture periodic patterns in the bus load data. Highly correlated features, such as different types of solar irradiance data, were removed to mitigate multicollinearity, while low-importance features, such as sunrise and sunset times, were discarded based on feature importance ranking to reduce noise and improve model parsimony.

Moreover, one-hot encoding was applied to the month, day, and holiday type features. This technique helps capture non-linear relationships and avoids assuming ordinal relationships, recognizing that their impact on bus load patterns may vary significantly. However, the hour and minute features were retained in their numerical form to effectively learn and incorporate temporal dependencies throughout the day.

Using all these detailed features, the 4 identified clusters were utilized to train the pre-trained models with all algorithms. Subsequently, the single-bus training process was carried out with these pre-trained models. To ensure an unbiased comparison across models, Optuna^[24], an automatic hyperparameter optimization tool, was deployed during the entire training procedure. Furthermore, early stopping was set to prevent overfitting.

3.4 Results during common days

In this subsection, we evaluate the performance of the proposed scheme alongside three representative baseline algorithms, focusing on the results for a specific bus. As illustrated in Fig. 6, the proposed scheme consistently outperforms all baselines, accurately capturing intricate patterns and trends in bus load data during regular days. The predicted values closely align with the ground truth, showcasing minimal deviations and demonstrating its capability to effectively model the underlying dynamics and dependencies.

By contrast, the vanilla Transformer baseline exhibited the weakest performance, struggling to accurately capture the underlying trends and patterns. The Autoformer baseline demonstrated moderate performance but still had noticeable deviations from the ground truth, resulting in higher errors compared to the proposed scheme. The Informer baseline, while capturing general trends, exhibited larger deviations and higher error values compared to the proposed scheme.

To provide a comprehensive evaluation, additional common statistical metrics such as mean absolute error (MAE) and R-squared (R^2) are included to compare overall performance across all 1 399 buses, as presented

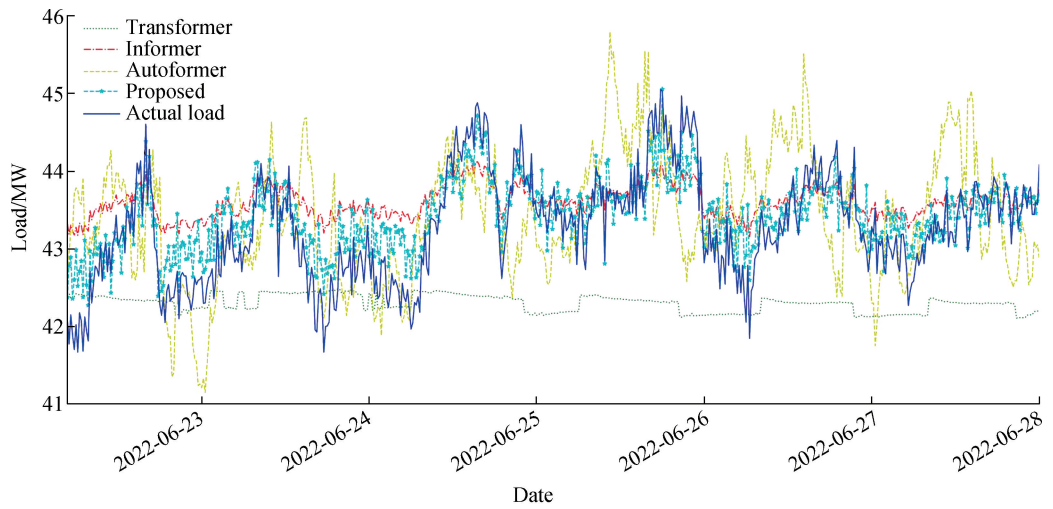


Fig. 6 Prediction results on regular days

in Table 2. The proposed scheme demonstrates superior accuracy, evidenced by significant improvements across all metrisss, most notably a minimum 32.00% reduction in RMSE.

Table 2 Performance comparisons

Statistical metric	RMSE/ MW	RMSE reduction of proposed/%	MAE/ MW	MAE reduction of proposed/%	R ²
Proposed	4.723 1		4.241 2		0.874 2
Transformer	13.618 2	65.31	8.178 8	48.14	0.217 9
Autoformer	6.946 0	32.00	5.399 5	21.45	0.769 3
Informer	9.449 3	50.02	7.926 0	46.49	0.300 4

3.5 Results during the Chinese New Year

To further evaluate the performance of the proposed scheme alongside the baseline algorithms, experiments

were conducted focusing on the Chinese New Year period, which is the most significant holiday in the region where data is collected. The results are presented in Fig. 7.

The proposed scheme remarkably captured the unique trends associated with this holiday, achieving the lowest RMSE. The Transformer baseline exhibited no discernible trend, failing to capture holiday dynamics. The Autoreformer baseline demonstrated better performance than the Transformer but experienced a noticeable degeneration compared to its performance on regular days. The Informer baseline preserved some trends but displayed bias and produced higher RMSE than that of the Transformer during the Chinese New Year period.

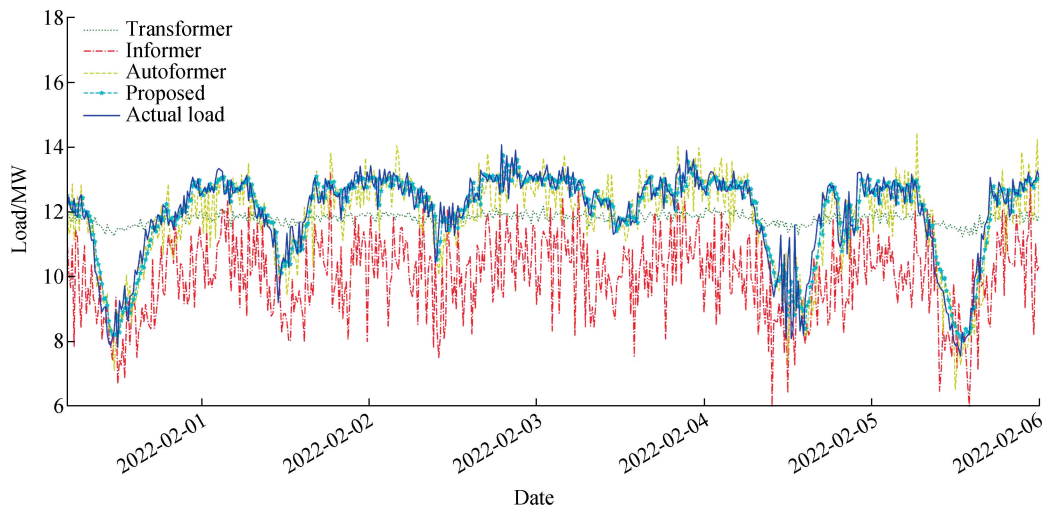


Fig. 7 Prediction results during the Chinese New Year

3.6 Computational burden comparison

During the single-bus model training stage, computation costs were significantly reduced by freezing all layers except for the final dense and output layers of the pre-

trained transformer. For the proposed network, the number of trainable parameters was reduced from 5.81×10^6 to 513. This substantial reduction led to a remarkable decrease in average training time, dropping from 288.94 s to 98.96 s, a 65.75% reduction. This strategy was also

applied to other algorithms for comparison, as listed in Table 3.

Table 3 Computation statistics before/after using the frozen neuron strategy

Statistical metric		Proposed	Transformer	Autoformer	Informer
Trainable parameter count	Before	5.81×10^6	13.61×10^6	14.91×10^6	14.39×10^6
	After	513	1 025	1 025	1 025
Average training time/s	Before	288.94	502.52	766.74	737.27
	After	98.96	180.51	251.48	306.16

By utilizing pre-trained knowledge and selectively fine-tuning a minimal set of parameters, the proposed approach strikes an optimal balance between model complexity and predictive accuracy. This contributes to the development of efficient and sustainable machine learning solutions, addressing the computational burden associated with large-scale Transformer models for bus load prediction.

4 Conclusions

- 1) A novel tailored Transformer-based scheme for bus load forecasting is proposed, exhibiting remarkable accuracy and considerably lower computational complexity compared to existing representative algorithms.
- 2) The proposed scheme demonstrates significantly enhanced predictive performance, particularly excelling in capturing atypical load patterns during holiday periods, a scenario that poses significant challenges to conventional forecasting techniques.
- 3) The incorporation of a frozen neuron strategy enables a substantial reduction in computational requirements, facilitating daily model updates and rendering the proposed solution highly competitive for bus load forecasting.

References

[1] Facchinetti T, Della Vedova M L. Real-time modeling for direct load control in cyber-physical power systems [J]. *IEEE Transactions on Industrial Informatics*, 2011, 7(4): 689 – 698. DOI: 10.1109/TII.2011.2166787.

[2] Tang N K, Mao S W, Wang Y, et al. Solar power generation forecasting with a LASSO-based approach [J]. *IEEE Internet of Things Journal*, 2018, 5(2): 1090 – 1099. DOI: 10.1109/JIOT.2018.2812155.

[3] Zhang R, Liu P F, Wang Q. Estimation model of EPC based on long time series of nighttime light data [J]. *Journal of Southeast University (Natural Science Edition)*, 2021, 51(6): 1094 – 1102. DOI: 10.3969/j.issn.1001-0505.2021.06.023. (in Chinese)

[4] Cao Y, Zheng L, Chen Y F, et al. Identification method and control strategy for superheated steam temperature of thermal power unit based on PFNN[J]. *Journal of South-*

east University (Natural Science Edition), 2022, 53(3): 417 – 424. DOI: 10.3969/j.issn.1001-0505.2022.03.001. (in Chinese)

[5] Luo J Z, Su C. Optimization of charging pricing strategy based on user behavior and time-of-use tariffs[J]. *Journal of Southeast University (Natural Science Edition)*, 2021, 51(6): 1109 – 1116. DOI: 10.3969/j.issn.1001-0505.2021.06.025. (in Chinese)

[6] Lu R Y, Guo X C, Li J C, et al. Tourist travel behavior in rural areas considering bus route preferences[J]. *Journal of Southeast University (English Edition)*, 2023, 39(1): 49 – 61. DOI: 10.3969/j.issn.1003-7985.2023.01.006.

[7] Bao Q, Tan X, Qu Q K, et al. Prediction of electric vehicle charging demand based on user space-time activities and fuzzy decision-making[J]. *Journal of Southeast University (Natural Science Edition)*, 2022, 52(6): 1209 – 1218. DOI: 10.3969/j.issn.1001-0505.2022.06.022. (in Chinese)

[8] Rubasinghe O, Zhang X N, Chau T K, et al. A novel sequence to sequence data modelling based CNN-LSTM algorithm for three years ahead monthly peak load forecasting[J]. *IEEE Transactions on Power Systems*, 2024, 39(1): 1932 – 1947. DOI: 10.1109/TPWRS.2023.3271325.

[9] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[C]// *31st Annual Conference on Neural Information Processing Systems (NIPS)*. Long Beach, CA, USA, 2017, 30: 6000 – 6010.

[10] He Z R, Shen Q F, Wu J X, et al. Transformer encoder-based multilevel representations with fusion feature input for speech emotion recognition [J]. *Journal of Southeast University (English Edition)*, 2023, 39(1): 68 – 73. DOI: 10.3969/j.issn.1003-7985.2023.01.008.

[11] Wen Q S, Zhou T, Zhang C L, et al. Transformers in time series: A survey[EB/OL]. (2022-02-15) [2024-05-08]. <http://arxiv.org/abs/2202.07125>.

[12] Zhou H Y, Zhang S H, Peng J Q, et al. Informer: Beyond efficient transformer for long sequence time-series forecasting[EB/OL]. (2020-12-14) [2024-05-08]. <http://arxiv.org/abs/2012.07436>.

[13] Wu H X, Xu J H, Wang J M, et al. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting[EB/OL]. (2021-06-24) [2024-05-08]. <http://arxiv.org/abs/2106.13008>.

[14] Zhou T, Ma Z Q, Wen Q S, et al. FEDformer: Frequency enhanced decomposed transformer for long-term series forecasting [EB/OL]. (2022-01-30) [2024-05-08]. <https://arxiv.org/abs/2201.12740>.

[15] Lim B, Arik S Ö, Loeff N, et al. Temporal Fusion Transformers for interpretable multi-horizon time series forecasting [J]. *International Journal of Forecasting*, 2021, 37(4): 1748 – 1764. DOI: 10.1016/j.ijforecast.

2021.03.012.

[16] López Santos M, García-Santiago X, Echevarría Camare-ro F, et al. Application of Temporal Fusion Transformer for day-ahead PV power forecasting[J]. *Energies*, 2022, **15**(14): 5232. DOI: 10.3390/en15145232.

[17] Sun S L, Liu Y K, Li Q, et al. Short-term multi-step wind power forecasting based on spatio-temporal correla-tions and transformer neural networks[J]. *Energy Con- version and Management*, 2023, **283**: 116916. DOI: 10.1016/j.enconman.2023.116916.

[18] L'Heureux A, Grolinger K, Capretz M A M. Transform-er-based model for electrical load forecasting[J]. *Ener-gies*, 2022, **15**(14): 4993. DOI: 10.3390/en15144993.

[19] Zhao Z Z, Xia C Q, Chi L, et al. Short-term load fore-casting based on the transformer model[J]. *Information*, 2021, **12**(12): 516. DOI: 10.3390/info12120516.

[20] Fu M Z, Qin M, Guo X J, et al. Magnetic field and coupling effect analysis of a novel dual-rotor dual-stator permanent magnet synchronous generator[J]. *Journal of Southeast University (English Edition)*, 2024, **40** (1): 89 – 96. DOI: 10.3969/j. issn. 1003-7985. 2024.01.010.

[21] Chen K J, Chen K L, Wang Q, et al. Short-term load forecasting with deep residual networks[J]. *IEEE Trans- actions on Smart Grid*, 2019, **10** (4): 3943 – 3952. DOI: 10.1109/TSG.2018.2844307.

[22] Li Z H, Liu J M, Lin Y Z, et al. Grid-constrained data cleansing method for enhanced bus load forecasting[J]. *IEEE Transactions on Instrumentation and Measurement*, 2021, **70**: 9002810. DOI: 10.1109/TIM.2021.3075538.

[23] Rafiei M, Niknam T, Aghaei J, et al. Probabilistic load forecasting using an improved wavelet neural network trained by generalized extreme learning machine [J]. *IEEE Transactions on Smart Grid*, 2018, **9**(6): 6961 – 6971. DOI: 10.1109/TSG.2018.2807845.

[24] Akiba T, Sano S, Yanase T, et al. Optuna: A next-gen-eration hyperparameter optimization framework [C]// *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. Anchorage, AK, USA, 2019: 2623 – 2631. DOI: 10.1145/3292500.3330701.

基于 Transformer 的节假日短期母线负荷预测修正机制

唐宁恺^{1, 2} 陆继翔^{1, 2} 陈天宇² 束 蛟^{1, 2} 昌 力² 陈 涛³

(¹电网运行风险防御技术与装备全国重点实验室,南京 211106)

(²南瑞集团有限公司(国网电力科学研究院有限公司),南京 211106)

(³东南大学电气工程学院,南京 211189)

摘要:为了解决短期母线负荷预测不够精准,且该现象在节假日期间尤为显著的问题,提出一种基于 Trans-former 的定制架构增强机制. 首先对输入数据进行聚类,以降低簇复杂性并捕获固有特性;然后利用门控残差连接有选择性地在各层之间传播显著特征,采用注意力机制专注识别多元时间序列数据中的显著模式;最后使用带有预训练架构减少训练计算资源需求. 基于大量数据的实验结果表明,所提机制在全母线评估上将预测准确度相对对比算法提高至少 32.00%,对节假日负荷曲线拟合效果突出,同时预训练方法将所提算法训练时间减少 65.75% 以上. 所提机制能在高效预测母线负荷结果的同时提升节假日预测鲁棒性,因而能更有效适应实际预测场景.

关键词:短期母线负荷预测; Transformer 网络; 节假日负荷; 预训练模型; 负荷聚类

中图分类号:TP274.2