EAI Endorsed Transactions
on Energy Web
Research Article EAI.EU

A Medium and Long-term Forecasting Method for Electric Load Curves Integrating Dynamic Variable Selection and Sparse Attention Mechanism

Bingqian Chen, Jinlin Liao*, Shiyuan Ni and Sudan Lai

State Grid Fujian Economic and Technology Research Institute, Fuzhou, 350011, China

Abstract

Accurate forecasting of electric load curves is critical for stable power system operation, efficient dispatch, and scientific planning, with medium- and long-term forecasting providing key support for grid expansion, energy allocation, and electricity market decisions. Existing methods struggle with insufficient use of load-related multi-variable data and inefficient extraction of long-term temporal features from load measurements, limiting forecasting accuracy and efficiency. To address these issues, this paper proposes a medium- and long-term electric load curve forecasting model integrating dynamic variable selection and a sparse attention mechanism, with a focus on load measurement correlation. The model takes static variables and temporal variables related to load changes as inputs. Static variables include industry type and location, and temporal variables cover historical load and meteorological data. It uses a gated feedforward network to adaptively weight variables based on their correlation with actual load measurements and filters redundant information. Meanwhile, it adopts a dual-layer encoding structure with sparse attention to prioritize key features from long load measurement sequences, reducing computational complexity while enhancing capture of long-term dependencies between historical load measurements and future loads. Experiments on three datasets show the model outperforms baselines including Informer, Autoformer and TFT. The proposed model reduces average relative error by 14.3%–55.1% and improves computational efficiency, providing reliable technical support for power system medium- and long-term planning by closely linking load measurements to forecasting performance.

Keywords: electric load curve forecasting, medium and long-term prediction, dynamic variable selection, sparse attention mechanism, power system planning.

Received on 09 September 2025, accepted on 21 December 2025, published on 15 April 2026

doi: 10.4108/ew.12161

1. Introduction

In modern power systems, accurate forecasting of electric load curves is essential for ensuring grid stability, optimizing energy resource allocation, and supporting strategic decision-making in electricity markets [1]. Among various forecasting horizons, medium and long-term predictions (ranging from several months to decades) are particularly critical, as they underpin long-term grid expansion plans, energy investment strategies, and policy development [2], [3]. However, the complexity of power systems, characterized by multiple influencing factors and intricate temporal correlations, poses significant challenges to existing forecasting methodologies [4].

Traditional approaches, such as time series analysis and regression-based models, often rely on predefined static variables and struggle to adapt to the dynamic nature of load data [5]-[7]. They typically suffer from two major limitations: the ineffective utilization of multi-variable information and the insufficient extraction of long-term temporal features. As power systems become more interconnected and integrated with renewable energy sources, the volume and variety of

* Corresponding author. Email: 343292467@qq.com

available data have grown exponentially [8]. Static variable selection methods may fail to identify the most relevant factors, leading to information redundancy and degraded prediction accuracy [9]. Meanwhile, handling long sequences of load data requires high computational resources, making it difficult for conventional models to capture long-range dependencies efficiently [10].

Recent advancements in deep learning, especially attention mechanisms, have shown great potential in addressing these challenges [11]-[13]. An electricity load forecasting framework based on an attention mechanism and depthwise separable convolutional neural network is proposed in [14], which integrates seasonal decomposition and feature engineering to capture cyclical and stochastic features. A short-term multi-energy load forecasting method for integrated energy systems is proposed in [15], where a CNN-BiGRU model enhanced with attention modules and a multi-task loss function is developed to capture the coupling among electricity, cooling, and heating loads. However, existing attention-based models often employ full attention mechanisms, which can cause high computational complexity and memory consumption when processing long-term data. Moreover, they lack a systematic approach to dynamically select variables based on their relevance to load curves, resulting in suboptimal performance in practical scenarios.

To address the above challenges, this paper develops a novel forecasting method for electric load curves that integrates dynamic variable selection and a sparse attention mechanism. The key contributions are as follows:

1) A unified framework is established to jointly exploit static and temporal variables, where a gated feedforward network adaptively filters redundant information to improve the efficiency of feature utilization.

2) A dual-layer encoding structure with sparse attention is designed to emphasize key temporal dependencies, thereby enhancing long-term feature extraction while reducing computational costs.

3) Extensive experiments are conducted on multiple real-world datasets, including electricity transformer temperature (ETT), electricity consumption load (ECL), and domestic load data, demonstrating the superiority of the proposed method over existing approaches.

Overall, the proposed method provides a more accurate and computationally efficient solution for medium- and long-term power system load forecasting and planning.

2. Related Work on Transformer

The Transformer architecture, first proposed by Vaswani et al. [16], has revolutionized the field of sequence modeling by introducing a self-attention mechanism as its core component, replacing the recurrent structures (e.g., RNNs, LSTMs) that dominated earlier time-series analysis. Unlike recurrent models, which process data sequentially and suffer from limited parallelization, the Transformer leverages self-attention to capture dependencies between all time steps in parallel, enabling more efficient processing of long-sequence data—an attribute particularly critical for medium-and long-term power load forecasting, where historical data spans months to years. The standard Transformer consists of two main parts: an encoder that extracts hierarchical features from input sequences and a decoder that generates predictive outputs by attending to the encoder's features, making it inherently suitable for modeling complex temporal correlations in power systems.

In recent years, researchers have extended the Transformer's application to power load forecasting and optimized its performance for specific challenges in the field. A multi-horizon time-series forecasting method based on temporal attention learning is proposed in [17]. This method enhanced the Transformer's ability to capture both long-term dependencies and periodic patterns by weighting attention scores according to temporal relevance. For park-level integrated energy systems, Huang et al. [18] combined the Transformer with multi-task learning to predict both electric and thermal loads simultaneously. By sharing feature extraction layers across load types, their model effectively learned the coupled relationships between different energy demands, significantly improving prediction accuracy and robustness compared to single-task models.

To address the Transformer's limited "attention scope" when processing ultra-long sequences, Ti et al. [19] proposed a ConvGRU-Transformer hybrid model integrated with a recurrent dilation mechanism. The recurrent dilation expanded the model's receptive field to capture long-range temporal patterns, while the ConvGRU module extracted local spatial-temporal features. This hybrid design not only improved prediction accuracy but also reduced training time by avoiding redundant feature extraction. Li et al. [20] further integrated convolutional layers tightly with the Transformer, where convolutional operations first captured local time-series patterns, and the Transformer then modeled global dependencies. This local-to-global feature learning framework effectively balanced the model's ability to capture fine-grained and macro-scale temporal characteristics.

Another key direction of Transformer optimization is reducing computational complexity, as full self-attention in standard models incurs a quadratic time complexity relative to sequence length—an issue that becomes prohibitive for long-term load forecasting with thousands of time steps. Zhao et al. [21] proposed an Explicit Sparse Transformer that introduced an explicit selection mechanism to filter out trivial attention weights. By retaining only the top-k most relevant attention scores instead of computing attention for all pairs, the model reduced memory consumption and computational cost without sacrificing prediction performance, laying a foundation for sparse attention design in subsequent load forecasting models.

Zhou et al. [22] addressed long-sequence forecasting challenges by developing the Informer model, which incorporated a probabilistic sparse self-attention mechanism. This mechanism clustered similar attention scores to reduce redundant calculations and introduced a hierarchical encoder to compress long input sequences, enabling efficient processing of sequences with lengths up to 10,000 time steps. Wu et al. [23] proposed the Autoformer, which integrated a decomposition module with auto-correlation-based attention.

The decomposition module split load sequences into trend and seasonal components, and the auto-correlation mechanism focused on capturing periodic dependencies within each component—greatly improving the model's efficiency for long-term load forecasting. However, both the Informer and Autoformer primarily focus on optimizing temporal dependency capture and lack a dedicated module to handle multi-variable inputs, which are essential for accurate power load prediction.

Lim et al. [24] introduced the Temporal Fusion Transformer (TFT), a model designed explicitly for interpretable multi-horizon forecasting. The TFT incorporated gating mechanisms to weight feature importance and handle missing data, making it more adaptable to real-world power system scenarios where data quality varies. Despite its interpretability advantages, the TFT relies on full attention for temporal modeling, leading to higher computational costs when processing long sequences—limiting its efficiency for medium-and long-term load forecasting tasks.

Existing Transformer-based load forecasting methods have made significant progress in capturing temporal dependencies and reducing complexity, but two critical gaps remain: (1) Most models lack a dynamic variable selection mechanism to adaptively weight multi-source inputs, leading to information redundancy or underutilization of critical variables; (2) While sparse attention reduces complexity, few studies integrate it with variable selection to balance prediction accuracy and computational efficiency. To address these gaps, this paper proposes a medium-and long-term load forecasting model that combines dynamic variable selection with a sparse Transformer. Compared to Wu et al.'s autocorrelation mechanism — which focuses solely on temporal dependencies within single-type time-series—our model introduces a gated feedforward network (GFFN)-based variable selection component to weight multi-variable inputs. Additionally, the dual-layer encoder with sparse attention enhances the capture of long-term dependencies while minimizing computational overhead, achieving a better trade-off between accuracy and efficiency for medium-and long-term power load forecasting.

3. Model Structure

To address the limitations of existing medium-and long-term power load forecasting methods, such as inefficient multi-variable utilization and high computational complexity in long-sequence feature extraction, a forecasting model integrating dynamic variable selection and a sparse Transformer is proposed. The overall architecture of the model is illustrated in Fig. 1, which consists of four core layers: a multi-variable input layer, a variable weighting layer, a double-layer encoding layer, and a decoding output layer. Among these, the decoding output layer further includes an information processing sub-layer, a sparse attention sub-layer, and a gated feedforward sub-layer. The design of each layer is tailored to enhance the model's ability to capture critical features while reducing redundancy and computational cost, thereby achieving both high prediction accuracy and efficiency for medium-and long-term load curves.

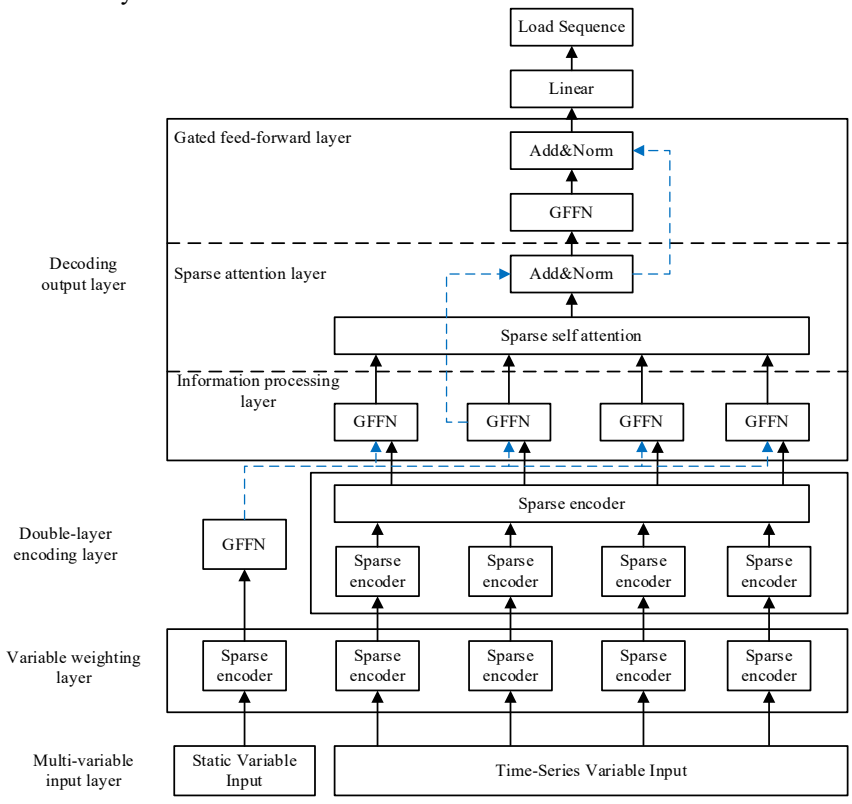

Figure 1. Structure of forecasting model

3.1 Multi-variable Input Layer

Power load is influenced by both time-invariant and time-varying factors. To fully leverage the complementary information from these factors, the model introduces static variables and temporal variables as joint inputs, addressing the issue of insufficient feature diversity in traditional single-variable or fixed multi-variable models.

Static variables refer to factors that do not change with time but have a persistent impact on load patterns. These primarily include industry-related attributes and location-related attributes. Such variables provide global contextual information—for instance, industrial areas typically exhibit stable high-load patterns during working hours, while residential areas show load peaks in the morning and evening.

Temporal variables are time-dependent factors that directly drive short-term to long-term load fluctuations. These include historical load data, meteorological data, and time-related indicators. Historical load data capture inherent temporal trends, meteorological data reflect environmental impacts, and time indicators account for periodic patterns.

To integrate these two types of variables effectively, the initial input of the model is constructed using feature concatenation, as defined in (1).

$$\mathbf{X}^{t} = \mathrm{concat}\left(\mathbf{S}, \mathbf{D}^{t}\right) = \left[\mathbf{S}, \mathbf{D}^{t}\right] \tag{1}$$

where $\mathrm{concat}(\cdot)$ denotes the feature concatenation operation; $\mathbf{S} = \left[S_1, S_2, \cdots S_n\right]$ represents the static variable vector, with S_i being the value of the i-th static variable; and $\mathbf{D} = \left[d_1^t, d_2^t, \cdots, d_n^t\right]$ denotes the temporal variable vector, with d_j^t being the value of the j-th temporal variable at time t.

Notably, static variables are not only fed into the subsequent variable weighting layer but also distributed to the double-layer encoding layer through a multi-path flow. In the first path, static variables are combined with temporal variables to calculate the correlation between each variable and the load, guiding the selection of critical features. In the second path, static variables provide contextual constraints for the encoder, preventing the model from overfitting to local temporal patterns and ensuring consistency with global load characteristics.

3.2 Variable Weighting Layer

Multi-variable input introduces rich information but also brings redundant or irrelevant features. Traditional manual feature selection or fixed-weight methods lack adaptability—they either retain redundant features (increasing computational cost) or discard critical ones (degrading accuracy). To solve this, the model designs a variable weighting component based on a GFFN, which adaptively assigns weights to variables according to their contribution to load prediction and filters out irrelevant information.

The GFFN is an optimized version of the standard Transformer feedforward layer. It replaces the ReLU activation function, which has limited ability to handle non-linear dependencies, with a Gated Recurrent Unit (GRU). The GRU's gating mechanism including reset gate and update gate enables dynamic control over the flow of information, allowing the network to retain useful feature information and suppress noise from irrelevant variables. The mathematical formulation of the GFFN is given in (2) and (3).

$$\mathrm{GFFN}_{\omega}(x) = \mathrm{LayerNorm}\left(x + \mathrm{GRU}_{\omega}(\boldsymbol{\psi})\right) \tag{2}$$

$$\mathrm{GRU}_{\omega}(\boldsymbol{\psi}) = \tanh\left(W_{\omega}''' \boldsymbol{\psi} + c\right) \otimes \sigma\left(W_{\omega}''' \boldsymbol{\psi} + d\right) \tag{3}$$

where x and $\boldsymbol{\psi}$ are the original input vectors, $W_{\omega}''' \in \mathbb{R}^{d_{out} \times d_{in}}$ is the weight matrix, $c, d \in \mathbb{R}^{d_{out}}$ are bias vectors, $\mathrm{LayerNorm}(\cdot)$ denotes layer normalization to stabilize training by reducing internal covariate shift, ω represents the shared weights of the model, $\tanh(\cdot)$ is the hyperbolic tangent activation function for generating candidate feature states, $\sigma(\cdot)$ is the sigmoid function for generating gate values between 0 and 1 and $\otimes$ denotes element-wise multiplication.

Due to the varying contributions of multivariate data to prediction performance, it is necessary to select different variables. Traditional methods of analyzing contribution and manually assigning weights are inefficient and inflexible. This article constructs a variable selection component based on GFFN, which selects correlated variable data while excluding variables that have a negative impact on model performance. Its main structure is shown in Fig. 2.

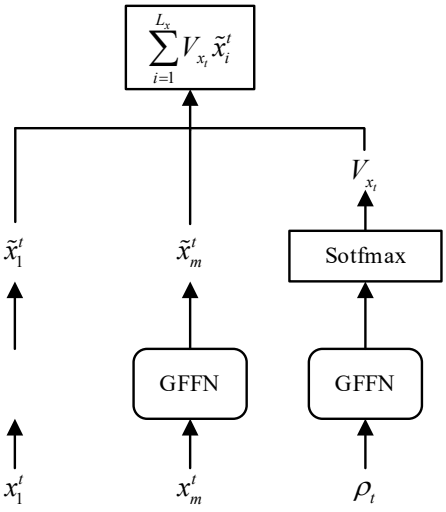

Figure 2. Variable selection component

The variable weighting component takes the GFFN as its core and incorporates a Softmax layer to convert feature importance scores into probabilistic weights. The workflow is as follows:

(1) For each input variable x_i^t at time t, the GFFN performs non-linear transformation to extract its high-dimensional feature representation, as shown in (4).

$$\tilde{x}_i^t = \mathrm{GFFN}\left(x_i^t\right) \tag{4}$$

(2) All transformed variables at time t are aggregated into a feature matrix ρ_t, defined in (5).

$$\rho_t = \left[x_1^{t^T}, x_2^{t^T}, \cdots, x_{L_x}^{t^T}\right]^T \tag{5}$$

where L_x is the total number of input variables.

(3) The feature matrix ρ_t is fed into another GFFN to calculate the importance score of each variable, and the Softmax layer normalizes these scores into weights V_{X_i} as in (6).

$$V_{X_i} = \mathrm{Softmax}\left(\mathrm{GFFN}\left(\rho_t\right)\right) \tag{6}$$

(4) The weighted variable vector $\tilde{x}_t$ at time t is obtained by element-wise multiplication of each variable's feature representation and its corresponding weight, followed by summation.

$$\tilde{x}_t = \sum_{i=1}^{L_x} V_{X_i} \cdot \tilde{x}_i^t \tag{7}$$

This adaptive weighting mechanism ensures that variables with strong correlations to load are assigned high weights, while irrelevant variables are assigned near-zero weights, effectively reducing feature redundancy and improving the signal-to-noise ratio of the input.

3.3 Double-layer Encoding Layer

The double-layer encoding layer is designed to address the challenge of inefficient long-sequence feature extraction in traditional Transformer models. It consists of two cascaded encoder sub-layers, each integrating sparse multi-head attention and GFFN, to sequentially capture cross-variable correlations and cross-temporal dependencies. The structure of the layer is illustrated in Fig. 3.

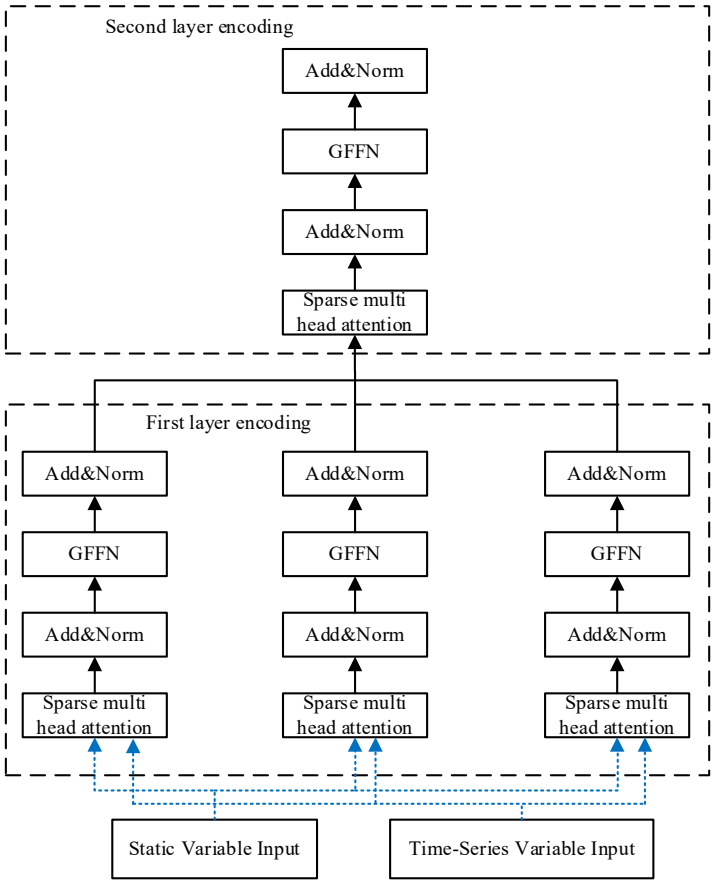

Figure 3. Double layer coding layer structure

The first sub-layer focuses on learning the correlation between static and temporal variables, as well as among different temporal variables. It takes the weighted variable vector $\tilde{x}_t$ from the variable weighting layer and static variables $\mathbf{S}$ as inputs, and uses sparse multi-head attention to model pairwise relationships between variables.

Multi-head attention splits the input into multiple parallel attention heads, allowing the model to capture different types of correlations simultaneously . However, standard multi-head attention computes attention scores for all variable pairs, leading to quadratic computational complexity. To mitigate this, the model introduces sparse attention: for each query variable, only the top-k key variables with the highest attention scores are retained, and the remaining scores are set to $-\infty$. This reduces the complexity from $O\left(L_x^2\right)$ to $O\left(L_x \cdot k\right)$ where $k \ll L_x$, significantly lowering computational cost.

The attention score calculation follows the scaled dot-product mechanism, as shown in (8).

$$P_t\left(Q,K\right)=\frac{QK^T}{\sqrt{d_k}} \tag{8}$$

where Q and K are matrices derived from the input variables query and key, d_k is the dimension of Q and K , and the scaling factor $\sqrt{d_k}$ prevents the attention scores from becoming too large avoiding saturation of the softmax function.

After sparse processing, the output of the first encoder sub-layer is normalized and combined with the residual connection to alleviate vanishing gradients, then fed into the GFFN for further non-linear feature enhancement.

The second sub-layer builds on the features from the first sub-layer and focuses on capturing long-term temporal dependencies (e.g., weekly, monthly load trends). It treats the output of the first sub-layer at each time step as a temporal sequence and applies sparse multi-head attention again. The query and key are derived from temporal positions rather than variables.

For example, when predicting the load at time $t+720$ (30 days later), the model emphasizes attention to time steps $t-720$ (30 days prior) and $t-1440$ (60 days prior) capturing monthly periodicity, while downplaying attention to irrelevant time steps. This targeted attention mechanism enables efficient modeling of long-sequence dependencies without excessive computation.

The output of the second encoder sub-layer Enc is obtained through residual connection, layer normalization, and GFFN processing, as defined in (9).

$$\text{Enc}=\text{LayerNorm}\left(\text{GFFN}_\omega\left(\phi(t)\right)+\phi(t)\right) \tag{9}$$

where $\phi(t)$ is the sparse attention output at time t.

Take the attention score as the evaluation standard, assume that the score set of query$_i$ and key$_i$ is P, sort the scores in the set P, retain the first k scores, and set all the remaining scores to infinity, and finally get the attention layer output $\phi(t)$.

$$\phi(t)=\begin{cases}P_{t(ij)}, & 0<P_{t(ij)}<k\\ -\infty, & P_{t(ij)}\le 0 \text{ or } P_{t(ij)} \gg k\end{cases} \tag{10}$$

3.4 Output Layer

The decoding output layer converts the high-dimensional features from the double-layer encoding layer into concrete load prediction values. It consists of three sequential sub-layers, each addressing a specific task: information fusion, temporal refinement, and output generation.

This sub-layer integrates static variables $\mathbf{S}$ with the encoder output Enc to supplement global contextual information. Static variables are reintroduced here to ensure that the decoded features remain consistent with the inherent load characteristics of the target region. The feature fusion is implemented via GFFN, as shown in (11).

$$\psi=\text{GFFN}_s\left(\{\mathbf{S},\text{Enc}\}\right) \tag{11}$$

where ψ is the fused feature vector, and the subscript s indicates that the GFFN parameters are optimized for static-temporal fusion.

This sub-layer refines the fused features by focusing on critical temporal patterns. It applies the same sparse attention mechanism as the encoding layer to ψ , further suppressing noise and enhancing the model's focus on key dependencies. The output of this sub-layer is computed as

$$\eta=\text{LayerNorm}\left(\psi+\text{Attention}_\psi\right) \tag{12}$$

where Attention_ψ denotes the sparse attention output for the fused feature vector ψ , and layer normalization ensures training stability.

The gated feedforward sub-layer uses GFFN to perform the final non-linear transformation on η , refining the feature representation to match the distribution of load values. The output of this sub-layer Dec is given by (13).

$$\text{Dec}=\text{LayerNorm}\left(\eta+\text{GFFN}_\omega\left(\eta\right)\right) \tag{13}$$

Finally, a linear layer maps the high-dimensional feature vector Dec to the dimension of the target load sequence, and a Softmax function used here for probability distribution normalization in multi-step prediction generates the final predicted load sequence Output, as shown in (14).

$$\text{Output}=\text{Softmax}\left(\text{Linear}\left(\text{Dec}\right)\right) \tag{14}$$

This multi-stage decoding process ensures that the model converts abstract features into accurate, interpretable load predictions, laying the foundation for reliable medium-and long-term power system planning.

4. Experiments and Result Analysis

To verify the accuracy, efficiency, and practical applicability of the proposed medium-and long-term power load forecasting model integrating dynamic variable selection and sparse attention mechanism, a series of experiments were conducted. This section details the experimental setup including dataset selection, data preprocessing, and

evaluation metrics, parameter optimization process, and comprehensive analysis of prediction results, with a focus on comparing the proposed model against state-of-the-art baseline models.

4.1 Experimental Setup

Three representative datasets were used to cover different application scenarios and verify the model's generalization ability. The detailed specifications of each dataset are as follows.

ETT Load Dataset: This dataset contains transformer load data from a regional power grid in China, spanning July 2022 to July 2024. The sampling interval is 15 minutes, resulting in 96 samples per day (24 hours × 4 samples/hour). In addition to load values, the dataset includes transformer temperature readings—a critical factor for understanding load variations related to equipment operation status. This dataset is widely used in power system research to validate models for medium-term load forecasting.

ECL Dataset: This dataset aggregates electricity consumption load data from multiple European countries, collected from January 2021 to December 2024. It adopts the same sampling interval (15 minutes) and daily sample count (96 samples) as the ETT dataset, enabling direct cross-dataset comparison. The ECL dataset captures load patterns across diverse geographical and climatic conditions, making it suitable for testing the model's adaptability to cross-region load characteristics.

Domestic Regional Transformer Historical Load Dataset: To further validate the model's performance in practical Chinese power system scenarios, a custom dataset was constructed using historical load data from a domestic regional power grid. The dataset covers the entire year of 2019, with a sampling interval of 1 hour (24 samples per day) and includes both dynamic variables and static variables. This dataset addresses the limitation of ETT and ECL, which only include power system internal data, by incorporating static variables, allowing comprehensive evaluation of the proposed model's dynamic variable selection mechanism.

Data preprocessing is critical for eliminating noise and standardizing feature formats, ensuring stable model training and reliable prediction results. The key steps are as follows.

(1) Dataset Splitting. For the ETT and ECL datasets with continuous long-term data, a 6:2:2 ratio was used to split the data into training set for model parameter learning, validation set for hyperparameter tuning, and test set for final performance evaluation. To avoid data leakage, the split was performed in chronological order rather than random shuffling. For the domestic regional dataset, a quarterly splitting strategy was adopted: the first 2 months of each quarter were used as training data, and the last month was split equally into validation and test sets. This approach aligns with practical medium-and long-term forecasting scenarios, where models are typically trained on historical seasonal data.

(2) Feature Processing: Input features were categorized into static and temporal variables, with tailored processing methods based on their attributes (Table 1). For the static variables, industry type was processed using one-hot encoding to convert categorical labels into numerical vectors. Location data was normalized to the range (-1, 1) using min-max scaling to eliminate the impact of different value magnitudes. For the temporal variables, continuous variables were normalized to (-1, 1) to ensure consistent feature scales. Discrete time indicators were one-hot encoded: for example, "8:00" was represented as a 24-dimensional vector with 1 at the 8th position and 0 elsewhere; "Friday" as a 7-dimensional vector with 1 at the 5th position; and "Labor Day" as a binary vector.

Table 1. Feature attributes and processing methods

Category	Attribute	Processing Method
Static Data	Industry Type	Discrete (One-hot Encoding)
	Location	Continuous (Min-Max Normalization)
Temporal Data	Electricity Consumption	Continuous (Min-Max Normalization)
	Load	Continuous (Min-Max Normalization)
	Temperature	Continuous (Min-Max Normalization)
	Hour of Day	Discrete (One-hot Encoding)
	Day of Week	Discrete (One-hot Encoding)
	Holiday Flag	Discrete (One-hot Encoding)

Two widely used metrics in time-series forecasting were selected to evaluate model performance: Mean Absolute Percentage Error (MAPE) and Root Mean Square Error (RMSE). MAPE quantifies the relative error in percentage to reflect prediction accuracy from a proportional perspective, while RMSE emphasizes the impact of large errors to evaluate the model's robustness. The formulas are as follows.

$$\mathrm{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100\% \tag{15}$$

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

where y_i is the actual load value at time i, $\hat{y}_i$ is the predicted load value, and n is the number of test samples. For fairness, all models were evaluated using the same test set, and metrics were calculated only after model training converged.

To validate the superiority of the proposed model, three state-of-the-art Transformer-based time-series forecasting models were selected as baselines:

Informer: A long-sequence forecasting model that uses probabilistic sparse self-attention and hierarchical encoding to reduce computational complexity, widely recognized for its performance in long-term load forecasting [22].

Autoformer: A decomposition-based model that splits time series into trend and seasonal components, using auto-correlation attention to capture periodic dependencies [23].

TFT (Temporal Fusion Transformer): An interpretable model with gating mechanisms for feature weighting and missing data handling, designed for multi-horizon forecasting [24].

All baseline models were implemented with their official hyperparameters adjusted slightly to match the input feature dimensions of the datasets to ensure a fair comparison.

4.2 Hyperparameter Optimization

The performance of deep learning models is highly sensitive to hyperparameters. To determine the optimal hyperparameter combination for the proposed model, a grid search was conducted on the validation set, with MAPE as the key evaluation metric. The hyperparameters and their optimization ranges are listed below.

Time Span: The length of historical data used for prediction (6, 12, 24, 48, 72, 96, 120, 144, 168, 192, 216, 240 hours). This parameter directly affects the model's ability to capture long-term dependencies.

Learning Rate: Controls the step size of parameter updates (0.001, 0.005, 0.01, 0.05, 0.1). A too-small learning rate leads to slow convergence, while a too-large one causes training instability.

Number of Attention Heads: Determines the parallel attention channels in multi-head attention (1, 2, 4, 8, 10, 12). More heads can capture diverse dependencies but increase computational cost.

Training Epochs: The number of full passes over the training set (10, 20, 50, 100, 120, 140). Too few epochs result in underfitting, while too many lead to overfitting.

Figure 4 shows the impact of each hyperparameter on the model's MAPE. The optimal combination was determined as follows:

Time Span: 192 hours (8 days of historical data), which balances the capture of long-term trends and avoidance of redundant historical information.

Learning Rate: 0.01, which ensures fast convergence without training oscillations.

Number of Attention Heads: 8, which achieves sufficient parallel feature learning with moderate computational cost.

Training Epochs: 100, after which the validation MAPE stops improving (indicating convergence).

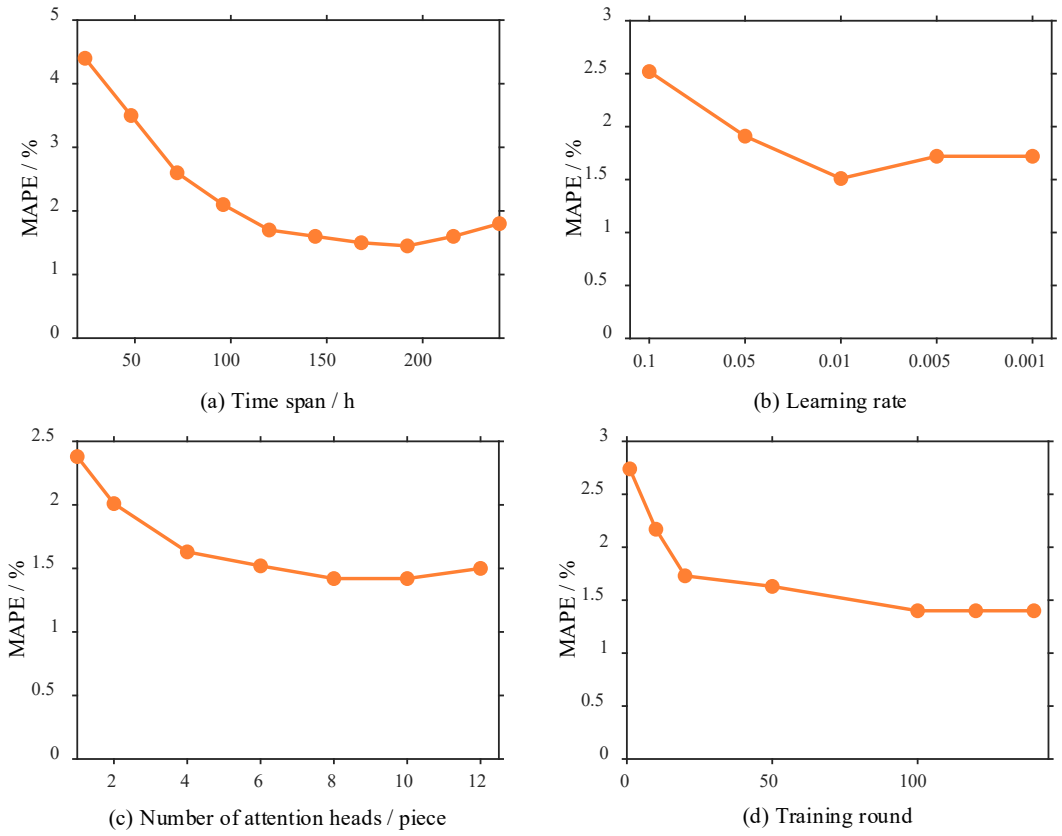

Figure 4. Effect of different parameters on model MAPE in validation set

Additionally, the number of GRU layers in the GFFN was optimized using a control variable method (Table 2). The results show that 3 GRU layers yield the lowest MAPE, as increasing layers beyond 3 leads to overfitting due to excessive model complexity, while fewer than 3 layers result in underfitting insufficient non-linear feature learning.

Table 2. Determination of GRU network layers

Number of GRU Layers	Time Span (h)	Learning Rate	Numer of Attention Heads	Training Epochs	MAPE (%)
2	192	0.01	8	100	1.45
3	192	0.01	8	100	1.42
4	192	0.01	8	100	1.43
5	192	0.01	8	100	1.43

5. Conclusions

In this paper, a novel forecasting method for electric load curves has been proposed, which integrates dynamic variable selection with a sparse attention mechanism. By jointly considering static and temporal variables in a unified framework and employing a gated feedforward network, the model effectively filters redundant information and enhances the efficiency of feature utilization. Furthermore, the dual-layer encoding structure with sparse attention enables the model to capture long-term temporal dependencies while reducing computational costs.

Extensive simulation experiments conducted on multiple real-world datasets, including ETT, ECL, and domestic load data, have demonstrated that the proposed method consistently outperforms state-of-the-art baselines. Specifically, the model achieves significant improvements in forecasting accuracy and robustness across different load patterns, validating its effectiveness for practical applications.

Overall, the proposed approach provides a more accurate and computationally efficient solution for medium- and long-term load forecasting. These findings highlight its potential to support reliable power system operation and long-term planning, while also offering a promising direction for future research on interpretable and scalable load forecasting models.

Conflict of interest

The authors declare that they have no conflict of interest.

References

[1] Nti I K, Teimeh M, Nyarko-Boateng O, et al. Electricity load forecasting: a systematic review[J]. Journal of Electrical Systems and Information Technology, 2020, 7(1): 13.

[2] Khuntia S R, Rueda J L, van Der Meijden M A M M. Forecasting the load of electrical power systems in mid-and long-term horizons: a review[J]. IET Generation, Transmission & Distribution, 2016, 10(16): 3971-3977.

[3] Aditya T, Jaipuria S, Kumar Dadabada P. A Review of Methods for Long-Term Electric Load Forecasting[J]. Journal of Forecasting, 2025, 44(4): 1403-1423.

[4] Wang H, Zhang N, Du E, et al. A comprehensive review for wind, solar, and electrical load forecasting methods[J]. Global Energy Interconnection, 2022, 5(1): 9-30.

[5] Liu D, Wang H. Time series analysis model for forecasting unsteady electric load in buildings[J]. Energy and Built Environment, 2024, 5(6): 900-910.

[6] Gasparin A, Lukovic S, Alippi C. Deep learning for time series forecasting: The electric load case[J]. CAAI Transactions on Intelligence Technology, 2022, 7(1): 1-25.

[7] Madhukumar M, Sebastian A, Liang X, et al. Regression model-based short-term load forecasting for university campus load[J]. IEEE Access, 2022, 10: 8891-8905.

[8] Mladenov V, Chobanov V, Georgiev A. Impact of renewable energy sources on power system flexibility requirements[J]. Energies, 2021, 14(10): 2813.

[9] Lin J, Ma J, Zhu J, et al. Short-term load forecasting based on LSTM networks considering attention mechanism[J]. International Journal of Electrical Power & Energy Systems, 2022, 137: 107818.

[10] Mounir N, Ouadi H, Jrhilifa I. Short-term electric load forecasting using an EMD-BI-LSTM approach for smart grid energy management system[J]. Energy and Buildings, 2023, 288: 113022.

[11] Cordeiro-Costas M, Villanueva D, Eguía-Oller P, et al. Load forecasting with machine learning and deep learning methods[J]. Applied Sciences, 2023, 13(13): 7933.

[12] Aslam S, Herodotou H, Mohsin S M, et al. A survey on deep learning methods for power load and renewable energy forecasting in smart microgrids[J]. Renewable and Sustainable Energy Reviews, 2021, 144: 110992.

[13] Wan A, Chang Q, Khalil A L B, et al. Short-term power load forecasting for combined heat and power using CNN-LSTM

enhanced by attention mechanism[J]. Energy, 2023, 282: 128274.
[14] Xu H, Hu F, Liang X, et al. A framework for electricity load forecasting based on attention mechanism time series depthwise separable convolutional neural network[J]. Energy, 2024, 299: 131258.
[15] Niu D, Yu M, Sun L, et al. Short-term multi-energy load forecasting for integrated energy systems based on CNN-BiGRU optimized by attention mechanism[J]. Applied Energy, 2022, 313: 118801.
[16] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[J]. Advances in neural information processing systems, 2017, 30.
[17] Fan C, Zhang Y, Pan Y, et al. Multi-horizon time series forecasting with temporal attention learning[C]//Proceedings of the 25th ACM SIGKDD International conference on knowledge discovery & data mining. 2019: 2527-2535.
[18] Xurui H, Fengyuan Y, Bo Y. Short-term Electric-thermal Load Forecasting Method of Campus Integrated Energy System Based on Transformer Network and multi-task Learning [J][J]. China Southern Power Grid Technology, 2023, 17(01): 152-160.
[19] Baozhong T I, Gengyin L I, Zhaoyuan W U. A short-term load forecasting method based on recurrent and dilated mechanism of ConvGRU-transformer[J]. Journal of North China Electric Power University, 2022, 49(03): 34-43.
[20] Shen L, Wang Y. TCCT: Tightly-coupled convolutional transformer on time series forecasting[J]. Neurocomputing, 2022, 480: 131-145.
[21] Zhao G, Lin J, Zhang Z, et al. Explicit sparse transformer: Concentrated attention through explicit selection[J]. arXiv preprint arXiv:1912.11637, 2019.
[22] Zhou H, Zhang S, Peng J, et al. Informer: Beyond efficient transformer for long sequence time-series forecasting[C]//Proceedings of the AAAI conference on artificial intelligence. 2021, 35(12): 11106-11115.
[23] Wu H, Xu J, Wang J, et al. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting[J]. Advances in neural information processing systems, 2021, 34: 22419-22430.
[24] Lim B, Arık 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.