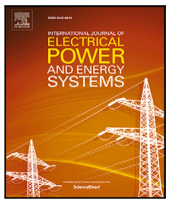
Electrical Power and Energy Systems 166 (2025) 110549

Contents lists available at ScienceDirect

International Journal of Electrical Power and Energy Systems

journal homepage: www.elsevier.com/locate/ijepes



TFTformer: A novel transformer based model for short-term load forecasting

Ahmad Ahmad [a], Xun Xiao [b,*], Huadong Mo [c,*], Daoyi Dong [d]

[a] School of Engineering and Technology, University of New South Wales, Canberra, Australia
[b] Department of Mathematics and Statistics, University of Otago, Dunedin, New Zealand
[c] School of Systems and Computing, University of New South Wales, Canberra, Australia
[d] Australian Artificial Intelligence Institute, Faculty of Engineering and Information Technology, University of Technology Sydney, Sydney, Australia

ARTICLE INFO

Keywords:
ANOVA
Feature embedding
Transposed embedding
Temporal convolutional networks

ABSTRACT

Electrical load forecasting is essential for the efficient operation and planning of power systems. Recent studies have employed Transformer models in forecasting due to their unique attention mechanisms and ability to extract correlations in data. However, these models face challenges in integrating varied data types and capturing long-term dependencies. To address these limitations, this study proposes a TFTformer, a transformer-based neural network designed to enhance the accuracy and generalisability of load forecasting models. The TFTformer incorporates transposed feature-specific embeddings for weather, time, and load data to more accurately capture their unique characteristics. A linear transformation layer post embedding improves feature representation, aligning and standardising features across sequences for improved pattern recognition. Additionally, a Temporal Convolutional Network is integrated within the Transformer's encoder, employing causal convolutions and dilation to adapt to the sequential nature of data with an expanded receptive field. The effectiveness of the TFTformer is demonstrated through a comparative study against several state-of-the-art methods using load datasets from Belgium, New Zealand, and five Australian states. The results demonstrate that the TFTformer achieves significant MSE improvements across different locations, with over 50% improvement over most models, 42% over CARD, and 16%–17% improvement compared to iFlowformer and iReformer. Furthermore, an Analysis of Variance is conducted to evaluate the impact of each component of the TFTformer. A SHAP-based interpretability analysis, using surrogate models, is conducted to elucidate the decision-making process of TFTformer, highlighting the critical role of time factors and weather features in its predictions.

1. Introduction

Energy Networks Australia predicts that by 2027, at least 40% of energy consumers will be using distributed energy resources, with this number projected to exceed 60% by 2050 [1]. The rapid adoption of distributed energy resources presents significant challenges for demand management and grid stability, creating a pressing need for effective demand response strategies [2]. The growing complexity of power systems emphasises the critical role of short-term load forecasting in the immediate dispatch of generating units [3]. Accurate short-term forecasting is essential to adjust for real-time demand fluctuations, ensuring operational efficiency and cost-effectiveness in grid management [4].

Statistical models have long been the foundation of load forecasting. Key techniques, such as the AutoRegressive Moving Average and AutoRegressive Integrated Moving Average (ARIMA), have shown competence in capturing basic temporal patterns by using autoregressive and differencing components [5,6]. Seasonal ARIMA extends these concepts to seasonal data, rendering it effective for certain cyclical load patterns [7,8]. Other statistical approaches, such as nonparametric regression [9], local linear regression [10], and exponential smoothing [11], have been used in identifying trends and seasonal patterns in load data. However, statistical methods often struggle with highly nonlinear relationships and non-stationary data in modern power systems. The assumptions about linear patterns and unchanging data characteristics may not hold given the non-stationary nature of grid load data, affecting their accuracy in forecasting [12].

In response to the limitations inherent in traditional statistical models, research has been increasingly focused on various machine learning methodologies. Support Vector Machine (SVM) has demonstrated strong performance as part of hybrid forecasting approaches that combine different prediction techniques [13]. Further research with SVMs has shown promise in load forecasting when combined with phase space reconstruction [14]. Random forest models have proven

* Corresponding authors.
E-mail addresses: xun.xiao@otago.ac.nz (X. Xiao), huadong.mo@unsw.edu.au (H. Mo).

https://doi.org/10.1016/j.ijepes.2025.110549
Received 5 November 2024; Received in revised form 4 February 2025; Accepted 17 February 2025
Available online 4 March 2025

effective when integrated with probability maps and risk assessment indices [15]. While traditional machine learning techniques have certain advantages, they often struggle to capture long-range temporal dependencies. This limitation arises from their reliance on feature engineering or shallow architectures, which may fail to adequately represent the complex temporal correlations present in multivariate load data [16]. Recent research has also demonstrated the adaptability of Artificial Neural Networks (ANN) across different data scales in load forecasting applications [17]. With refined architectures and training methods, ANN applications in electric load forecasting have advanced further [18]. The integration of meta-learning techniques with ANN has pushed the boundaries of forecasting accuracy by enabling better model adaptation to varying load patterns [19].

More recently, deep learning methods, including Long Short-Term Memory (LSTM) networks, Graph Neural Network (GNN), and Temporal Convolutional Networks (TCN), have significantly broadened the scope of load forecasting models. Particularly, LSTM can capture short to medium range dependencies in sequential data by using memory cells and gates to retain information over time [20]. However, they may still face difficulties handling very long sequences and can suffer from vanishing or exploding gradients for extremely large input windows [21]. GNNs operate on graph-structured data, propagating information between nodes to capture both local and global dependencies within a graph [22], but they often require careful graph construction and might underperform when the load data does not exhibit clear spatial network characteristics [23]. TCN demonstrates exceptional ability to capture temporal dependencies through leveraging convolutions with causal padding to process sequences in parallel, allowing faster training [24,25].

Recently, the Transformer model [26] represents a significant breakthrough in deep learning, particularly enhancing Natural Language Processing (NLP) through its innovative attention mechanism [27]. Research has expanded its application to time series forecasting. Temporal Fusion Transformer [28], Informer [29], Reformer [30], Flashformer [31], Flowformer [32], Transformer neural network with multi-scale acceleration feature fusion [33], and Autoformer [34], have significantly outperformed conventional models. The Transformer variants incorporate advancements such as self-attention, reversible layers, hardware-accelerated attention mechanisms, dilated overlap patch embedding, and [35] integrated a hierarchical encoder–decoder pair with a temporal-channel attention block in the Transformer to extract multi-scale features and capture connections between net load and relevant factors. The work in [36] proposed a multi-decoder Transformer for multi-task learning, enabling the simultaneous prediction of loads of different types in an integrated energy system through one encoder and multiple decoders. The work in [37] introduced a short-term load forecasting model combining the Transformer with an improved empirical mode decomposition algorithm to decompose load sequences into multiple frequency subsequences, reducing noise impact. Moreover, [38] employed a spatial and temporal attention-enabled Transformer model to extract dynamic spatial and nonlinear temporal correlations between residential units, enabling joint predictions of multivariate residential loads.

However, a study by Zeng et al. [39] questioned the effectiveness of using complicated Transformer models for time series forecasting due to their permutation-invariant self-attention mechanism, which disregards the order of elements. While semantic meaning can be kept under certain reordering operations in NLP tasks, time series forecasting requires strict preservation of the temporal order in data points. The study suggested that simpler methods, such as a simple linear layer with preserving the sequential ordering of data, might offer a more effective solution. By maintaining the temporal structure, these linear models could better capture the underlying patterns in time series data, leading to potentially superior accuracy and computational efficiency compared to Transformer-based approaches. Based on this insight, Liu et al. [40] and Nie et al. [41] highlighted potential limitations in the existing design of Transformer models for forecasting multivariate time series. They observed that merging data points within the same time step into a single token, which represents a fundamental unit of input data that the model processes, can pose challenges. Single token representation overlooks correlations between different data types and inadequately captures time-sensitive information. The parallel processing architecture of Transformers creates challenges in capturing long-term dependencies. The need to address these temporal dependency limitations motivated the development of i-Transformer and PatchTST. Both models embed the entire time series of each variable independently into separate variate tokens, enhancing the capture of unique sequential data dynamics. Fig. 1 illustrates the difference between one-time step and separate variate tokens.

Furthermore, the current research on Transformer models in load forecasting mainly focuses on numerical studies conducted using a limited collection of datasets. Many of these studies are based on the UCI Electricity Load Diagrams dataset [42], which provides load data from clients in Portugal. Additionally, datasets from European countries such as Austria, Belgium and France [35], and regions within the United States, including Arizona [36], Los Angeles [38], and New York [37,38] have been used. However, the limited geographical scope of studies can significantly restrict the understanding of model performance. Different regions present unique challenges to their power grids, influenced by environmental, social, and regulatory factors, variations in weather patterns, and consumer behaviours to different stages of grid modernisation and market regulations. The geographical limitation can significantly impact the validation of model performance across diverse power grid architectures and load patterns.

In examining model effectiveness, existing ablation studies typically focus on removing or modifying individual components of Transformer architectures to assess their impact on forecasting accuracy [[29]–[34]]. While this approach provides insights into component-level contributions to prediction accuracy, it overlooks broader performance aspects crucial for real-world applications. Critical factors such as model robustness under different operational conditions, and adaptability to evolving grid conditions remain largely unexplored. Furthermore, recent studies rarely employ statistical analysis techniques to quantify the significance of their findings or examine potential interactions between different model components.

From Table 1, it is evident that recent Transformer models have achieved notable success in time series forecasting, significant challenges remain in their ability to effectively handle multivariate data and capture both short-term and long-term dependencies. Despite advancements in the attention mechanisms and encoder–decoder structures, models such as Flowformer and Informer often struggle with modelling complex interactions between variables and preserving temporal patterns [43]. The limitations of Transformer become particularly problematic when dealing with load datasets where the intricate relationships between variables, such as weather, time, and load, are critical for accurate forecasting. Consequently, further improvements are needed in embedding techniques and the ability of Transformers to account for both local and global dependencies in multivariate time series data.

In response, the study proposes the TFTformer, which stands for Transposed Feature-specific embedding and Temporal convolution in Transformer, for load forecasting. The model is inspired by the iReformer, known for its efficiency in managing long data sequences with minimal memory usage. The primary focus is on optimising data preprocessing for different data types, followed by the transposed feature-specific embedding for each type of data, specifically weather, time, and load. Then, a linear transformation is added to enhance the representation of each feature. To further strengthen the model's ability to capture temporal dependencies, a TCN layer is incorporated into the encoder stage. The strategic implementation steps include:

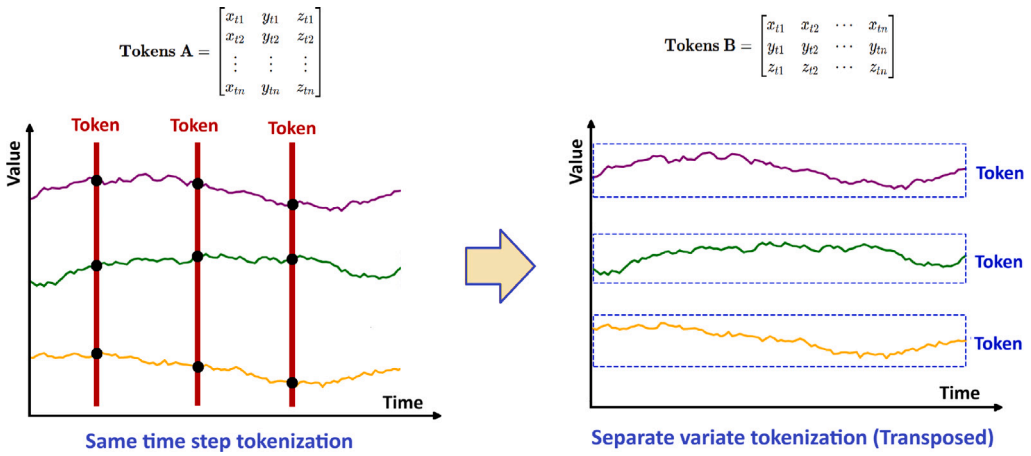


Fig. 1. The difference between same time and transposed tokenizations.

- Introducing transposed feature-specific embeddings for various data types. This allows the model to accurately accommodate the unique characteristics of different data sources, such as weather, time, and load information, thereby improving adaptability to the fluctuations in load forecasting data.
- Adding a Linear Layer Post-Embedding after the transposed feature-specific embeddings. The linear transformation helps align and standardise features across the sequence, making it easier for the subsequent attention mechanisms in the Transformer to perform complex pattern recognition and extraction.
- Employing causal convolutions and dilations in the encoder of the Transformer structure by adding a TCN block, which enhances the ability of TFTformer to handle sequential data and capture temporal dependencies.

The contributions in this study are as follows:

- Design TFTformer, a new model based on Transformer, offering significant accuracy, adaptability, and applicability improvements across varied datasets.
- Conduct extensive numerical studies over different datasets to validate the performance of TFTformer.
- Utilise ANOVA in the ablation study to underscore the robustness of TFTformer and provide a statistically thorough evaluation of the importance of each component on TFTformer's performance.

To improve the interpretability and reliability of the model, this study employs a SHAP-based surrogate approach integrated with Random Forest regression to analyse feature influences [44]. This approach addresses the interpretability challenges posed by the TFTformer's complex architecture, which combines transformer and TCN components [45]. By leveraging Random Forest's capacity to capture nonlinear feature interactions while maintaining interpretability through its tree-based architecture [46], we quantify the contribution of individual features to the model's predictions. This approach provides a clearer understanding of the model's decision-making process while maintaining comparable prediction performance on actual data.

The remainder of this paper is structured as follows: Section 2 presents the problem formulation and details the data preprocessing techniques used in this study. Section 3 describes the components of the TFTformer, including transposed feature-specific embeddings, adding a linear layer after embedding, and incorporating the TCN within the encoder. Section 4 includes a case study using real-world data to illustrate and discuss the performance of the TFTformer. Section 5 presents an ablation study that evaluates the impact of each component and the interactions on the prediction performance of the TFTformer. Section 6 details the interpretability analysis of the TFTformer model using a SHAP-based surrogate approach, providing insights into feature importance and model decision-making processes. Section 7 concludes the paper by summarising the findings and contributions of this study and outlining potential directions for future research.

2. Problem setup and data preprocessing

This section begins with the problem formulation, defining the forecasting problem and its parameters. The section then covers the data processing techniques, including data grouping, data normalisation, and one-hot encoding.

2.1. Problem formulation

This work focuses on multivariate load forecasting using a multi-input single-output approach. The objective is to predict load values by establishing a mapping between historical data $X_{1:L} \in \mathbb{R}^{M\times L}$, consisting of M variables over a look-back window of L time steps, and the predicted load values $\hat{Y}_{L+1:L+H} \in \mathbb{R}^{1\times H}$ for the next H time steps. The observed load values $Y_{L+1:L+H}$ are equal to the corresponding component of $[X_{\text{load}}]_{L+1:L+H}$. The input data X is composed of three components: $X_{\text{load}} \in \mathbb{R}^{1\times L}$, representing load variable; $X_{\text{weather}} \in \mathbb{R}^{Q\times L}$, representing weather variables; and $X_{\text{time}} \in \mathbb{R}^{K\times L}$, representing time variables. Thus, the total number of features is $M = 1+Q+K$, where Q is the number of weather features, and K is the number of time features. The modelling framework can be extended to incorporate additional variables if more features become available, allowing for more complex modelling scenarios. The forecasting problem is formulated as:

$$\hat{Y}_{L+1:L+H} = \mathcal{F}([X_{\text{weather}}, X_{\text{time}}, X_{\text{load}}]_{1:L}; \theta) \tag{1}$$

where $\mathcal{F}$ represents the mapping function and θ indicates the set of model parameters.

2.2. Data preprocessing

The section summarises the steps of data preprocessing, which starts with a strategic grouping of the dataset, then the essential preprocessing techniques, including data normalisation and applying one-hot encoding.

2.2.1. Data grouping

The data is classified into three distinct groups: the weather group X_{weather}, which includes parameters such as temperature, humidity, wind speed, wind direction, and cloud cover; the time group X_{time}, which comprises variables such as year, month, day of the month, day of the week, hour of the day, and holidays; and the load group X_{load}, which solely encompasses the load variables. The data grouping simplifies subsequent steps for data preprocessing and feature-specific embedding, which will be detailed in the following sections.

Table 1
Summary of Transformer-based models for time series forecasting and their relation to our work.

Model	Key contributions	Limitations	Related to our work
Vanilla Transformer [26]	• Introduced self-attention for parallel sequence processing. • Significantly reduced training time compared to RNNs in NLP.	• Permutation-invariant attention can overlook strict temporal ordering. • Struggles to model complex multivariate relationships without specialised embeddings. • Quadratic complexity can be an obstacle for very long sequences.	The foundation for all variants of the Transformer mode.
Temporal Fusion Transformer [28]	• Employs gating and attention for interpretable multi-horizon forecasts. • Manages static and observed inputs separately for more flexible feature handling.	• Model complexity can increase for high multivariate data. • Short-term and long-term dependencies are handled, but can become challenging with large feature sets.	A simpler feature embedding strategy can be used to maintain interpretability and manage varied inputs without overly increasing complexity.
Informer [29]	• Proposes ProbSparse attention, reducing memory usage for long sequence forecasting. • Improves global attention efficiency over standard Transformers.	• Focus on long-term forecasting can lead to weaker short-term detail capture. • Handling intricate interactions among multiple variables is not deeply addressed.	Balancing long-term and short-term dependencies is crucial for multivariate time-series forecasting.
Reformer [30]	• Adopts Locality-Sensitive Hashing (LSH) for near-linear attention complexity. • Utilises reversible layers to reduce memory usage.	• LSH may not effectively track short-term patterns if the hashing does not align with local temporal clusters. • Primarily demonstrated on NLP tasks, with limited exploration for multivariate load series.	Using the LSH attention mechanism can reduce the memory, but capturing local dependencies should be improved.
Flowformer [31]	• Integrates normalising flows in attention to capture complex data distributions. • Targets robust global context for longer sequences.	• Complex flow-based transformations may be difficult to tune, especially for multivariate data. • Temporal ordering and short-term patterns can still be overtaken by global flow mechanisms.	Rather than normalising flows, using transposed embeddings and temporal convolutions to preserve local and global patterns in load data.
Transformer + Multi-Scale Acceleration [33]	• Proposes multi-scale acceleration feature fusion for industrial time series. • Demonstrates improved accuracy for specialised datasets with complex variations.	• Highly customised multi-scale feature engineering can be difficult to generalise. • Still needs better short- and long-term interplay for forecasting applications.	Using specific embeddings for each feature type, rather than heavy domain-specific engineering.
Autoformer [34]	• Introduces “auto-correlation” to capture cyclical patterns and trends. • Decomposes time series into seasonal and trend components for improved long-term forecasting.	• Decomposition may not be optimal for all variables when dealing with diverse features. • Local variations and short-term spikes can be overtaken by seasonal/trend decomposition.	Focusing on feature embeddings and convolutional layers to simultaneously capture local and global dependencies.
Hierarchical Transformer [35]	• Uses a hierarchical encoder–decoder with a temporal-channel attention block. • Targets multi-scale features, focusing on net-load relationships.	• Complex multi-stage designs can be expensive to train. • Handling very high-dimensional data across multiple scales remains challenging.	Addressing local/global load dependencies without introducing overly complex hierarchies.
Multi-decoder Transformer [36]	• Single encoder with multiple decoders for different types of loads. • Aims to handle parallel forecasting tasks in integrated energy systems.	• Model size grows quickly with additional decoders. • Complex interactions among different types of data across decoders remain underexplored.	Separate decoders inspire the idea of specific feature embedding.
Spatial–Temporal Attention Transformer [38]	• Learns both spatial correlations and temporal dependencies. • Useful for multivariate residential data when spatial information is available.	• Highly dependent on explicit or inferred spatial relationships, not always obtainable. • Short-term variations may still be overshadowed by overall spatiotemporal modelling.	Focusing on time-series embeddings for local/global load dependencies, especially in diverse grids where spatial information might be unavailable.
Transformer + Improved EMD [37]	• Uses enhanced empirical mode decomposition (EMD) to mitigate noise in load sequences. • Demonstrates better short-term load forecasting by separating frequency bands.	• EMD introduces extra pre-processing and complexity. • Multivariate data with multiple interdependent variables may require separate decomposition pipelines, complicating the approach.	Employing convolutional operations can handle local fluctuations in load data.
i-Transformer [40]	• Processes each variable sequence as a separate token, preserving feature-level ordering. • Improves handling of variable-specific dynamics in some multivariate tasks.	• Short and long-term coupling between variables still requires careful token fusion.	Transposed embeddings are inspired by variable sequences as a separate token.
PatchTST [41]	• Splits time series into patches for tokenization, balancing local and global insights.	• May lose fine-grained detail if patches are too large or overlap is insufficient. • Handling high-dimensional data can demand extensive tuning of patch settings.	Instead of patch-based tokenization, employing transposed embeddings for each group of features, preserving short and long-range dependencies without patch tuning.

2.2.2. Data normalisation for numerical variables

Data normalisation is a key preprocessing step that scales numerical data within a specific range to reduce scale differences among features, thereby enhancing numerical stability and ultimately improving model performance. Min–max normalisation, which adjusts data values between 0 and 1 without losing information [47], is applied to the weather group X_{weather}, load group X_{load}, and the year parameter, as normalisation is more efficient for capturing trends over time in load forecasting [48]. The holiday parameter, which is already set to 0 for weekdays and 1 for weekends or public holidays, does not require normalisation. The data normalisation can prevent features with a large range from dominating other features, thus biasing the training process.

2.2.3. One-hot encoding for categorical variables

Categorical variables such as the month, day of the month, day of the week, and hour of the day are converted into binary vectors, known as dummy variables as shown in Fig. 2. Each category is represented by a vector where one element is “hot” (encoded as 1) while all other elements are “cold” (encoded as 0), ensuring that each category is equally distant from others, thereby eliminating any sequential relationship that might be misinterpreted by the model [49]. Such technique called one-hot encoding preserves the categorical information in a format that is suitable for processing by TFTformer, allowing the model to accurately interpret and utilise the categorical data without adding any false sense of order.

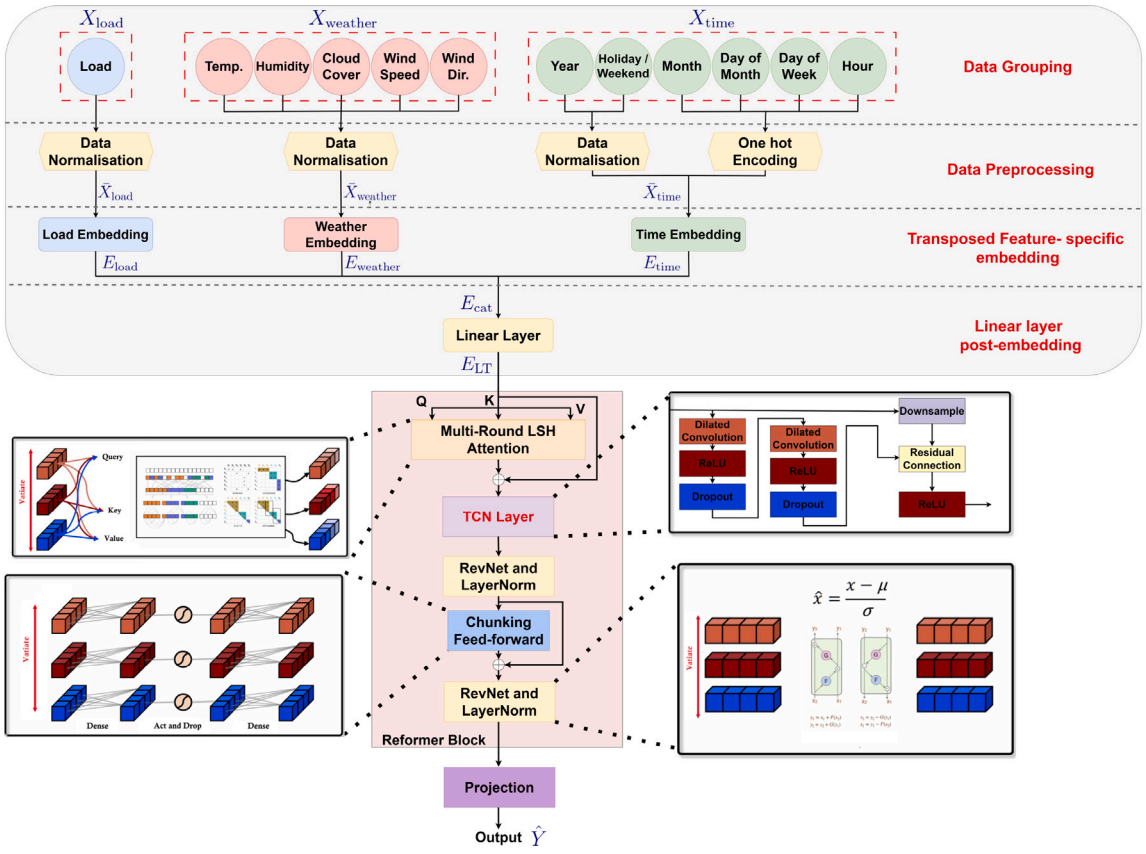


Fig. 2. A high-level overview of TFTformer framework, with certain elements adapted from [40].

3. Model formulation

This section describes the transposed feature-specific embeddings used for various data categories. The description elaborates on the linear layer utilised after the embedding process. Additionally, the section details the TCN layer concerning causal convolutions and dilations into the encoder of the Transformer structure. An overview of the TFTformer's structure is presented in Fig. 2.

3.1. Transposed feature-specific embedding

In traditional embedding methods, each time stamp across all variables is treated as an input vector, with the variables at that time forming the input features. For a multivariate time series dataset with L time steps and d variables, the embedding process treats the input as $X \in \mathbb{R}^{L\times d}$, where each row corresponds to the values of all variables at a single time step. The embedding then maps these d variables into a high-dimensional space N using a shared linear transformation, such that $E = X \cdot W^T + b$, where $E \in \mathbb{R}^{L\times N}$, $W \in \mathbb{R}^{N\times d}$ is the weight matrix, and $b \in \mathbb{R}^{1\times N}$ is the bias vector. The uniform embedding technique benefits time series forecasting when using only historical load data. However, accurate predictions require handling diverse data types, as incorporating additional data has become a common approach to improving forecasting performance.

To address this, the transposed data embedding method, as described by Liu et al. [40], involves embedding the entire series as a token using a linear transformer for the entire input, broadening the sequence length from L to N, where N is the length of the embedded features. $E = \tilde{X} \cdot W^T + b$, where $E \in \mathbb{R}^{d\times N}$, $\tilde{X} = X^T \in \mathbb{R}^{d\times L}$, $W \in \mathbb{R}^{N\times L}$ is the weight matrix, and $b \in \mathbb{R}^{1\times N}$ is the bias vector. Transposed embedding enhances temporal representation by learning the dynamics of each variable over time. However, it still applies a uniform transformation across all variables, which limits its ability to capture the unique characteristics of individual variables.

Building upon this, the purpose of feature-specific embeddings is to identify and disentangle underlying explanatory factors, thereby enhancing the representation of each group [50]. The transposed feature-specific embedding maps the input features to a higher-dimensional space, enabling a richer representation that allows the TFTformer to capture temporal dynamics effectively and improves its ability to generalise and capture the unique characteristics of each data group. After preprocessing each group, as illustrated in the previous section, the embedding method is applied to the dataset as follows:

$$E_{\text{weather}} = \tilde{X}_{\text{weather}} \cdot W^T_{\text{weather}} + b_{\text{weather}} \tag{2}$$

$$E_{\text{time}} = \tilde{X}_{\text{time}} \cdot W^T_{\text{time}} + b_{\text{time}} \tag{3}$$

$$E_{\text{load}} = \tilde{X}_{\text{load}} \cdot W^T_{\text{load}} + b_{\text{load}} \tag{4}$$

where $\tilde{X}_{\text{weather}} \in \mathbb{R}^{Q\times L}$, $\tilde{X}_{\text{time}} \in \mathbb{R}^{K\times L}$, and $\tilde{X}_{\text{load}} \in \mathbb{R}^{1\times L}$ represent the input features for weather, time, and load features after data preprocessing, respectively. $W_{\text{weather}} \in \mathbb{R}^{N\times L}$, $W_{\text{time}} \in \mathbb{R}^{N\times L}$, and $W_{\text{load}} \in \mathbb{R}^{N\times L}$ are the weight matrices, and $b_{\text{weather}} \in \mathbb{R}^{1\times N}$, $b_{\text{time}} \in \mathbb{R}^{1\times N}$, and $b_{\text{load}} \in \mathbb{R}^{1\times N}$ are the bias vectors of their respective embedding layers. $E_{\text{weather}} \in \mathbb{R}^{Q\times N}$, $E_{\text{time}} \in \mathbb{R}^{K\times N}$, and $E_{\text{load}} \in \mathbb{R}^{1\times N}$ are the outputs of feature-specific embeddings.

3.2. Linear layer post embedding

A linear layer follows the feature embedding, drawing inspiration from Zeng et al. [39], where a linear model outperformed the Transformer model. The linear layer enhances representation by aligning and standardising features, simplifying attention mechanisms in the Transformer and enabling seamless handling of heterogeneous data by transforming different categories into a common space [26]. The linear transformation's adaptability is particularly important in scenarios

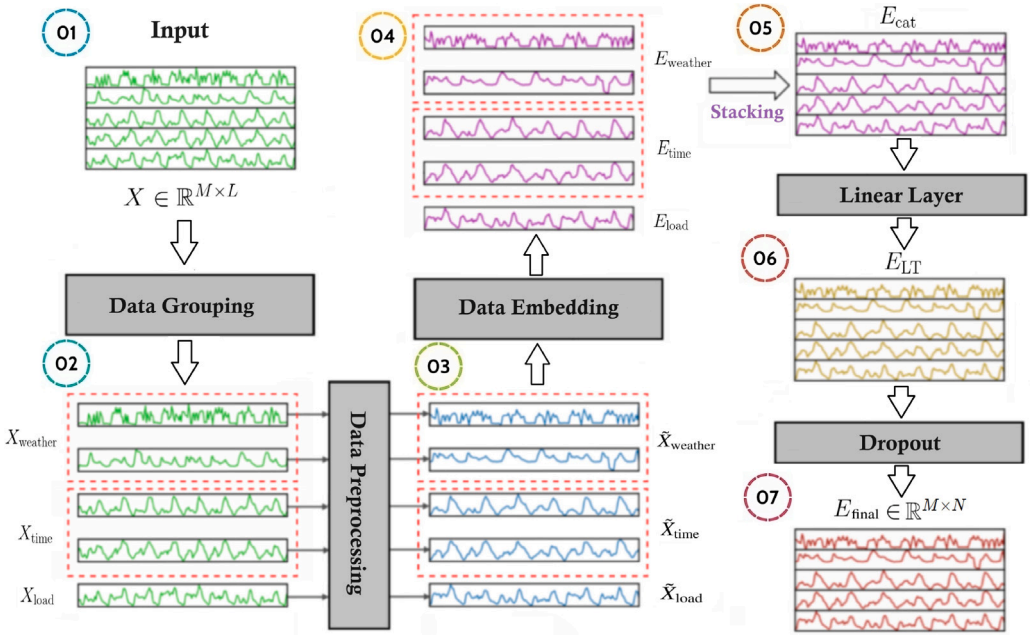


Fig. 3. The complete journey of data through preprocessing, feature-specific embedding, and the linear layer.

where the input data may vary significantly in scale and distribution. Furthermore, the added layer applies a dropout mechanism for regularisation, further enhancing the generalisation capability of the model.

The initial process involves combining the embedded data generated by the transposed feature-specific embeddings into a single matrix by vertically stacking each individual matrix, corresponding to different features, along the row dimension. Consequently, a new, larger matrix is formed, with each row representing features from the respective feature embeddings. The concatenated matrix E_{cat} is defined as:

$$E_{\text{cat}} = \begin{pmatrix} E_{\text{weather}} \\ E_{\text{time}} \\ E_{\text{load}} \end{pmatrix} \tag{5}$$

where $E_{\text{cat}} \in \mathbb{R}^{M \times N}$. Subsequently, the concatenated matrix E_{cat} undergoes a linear transformation following stacking, defined by a weight matrix $W_{\text{LT}} \in \mathbb{R}^{N \times N}$ and a bias vector $b_{\text{LT}} \in \mathbb{R}^{1 \times N}$, where the dimensions of W_{LT} are designed to map the concatenated embedding dimension. The operation is represented as:

$$E_{\text{LT}} = E_{\text{cat}} \cdot W_{\text{LT}}^{T} + b_{\text{LT}} \tag{6}$$

where $E_{\text{LT}} \in \mathbb{R}^{M \times N}$. The last step involves adding a dropout layer for regularisation purposes. The final E_{final} is expressed as:

$$E_{\text{final}} = \text{Dropout}(E_{\text{LT}}) \tag{7}$$

where $E_{\text{final}} \in \mathbb{R}^{M \times N}$. The entire process, starting from data preprocessing through feature-specific embedding and the linear layer, is illustrated in Fig. 3.

3.3. Integrating TCN within the transformer encoding

TFTformer employs the Reformer as a backbone model with separate variate data embedding techniques, a combined approach referred to as iReformer. The Reformer introduces locality-sensitive hashing (LSH) to approximate the calculation of attention scores, which lowers the complexity of the attention mechanism. The LSH attention replaces exhaustive attention in the traditional transformer with an approximate method. The LSH maps vectors to hash buckets such that nearby vectors are hashed together with high probability. For each query q_i, the attention is limited to keys k_j in the same bucket, resulting in the following computation:

$$o_i = \sum_{j \in P_i} \exp(q_i \cdot k_j - z(i, P_i)) v_j, \quad \text{where } P_i = \{j : h(q_i) = h(k_j)\} \tag{8}$$

where, o_i represents the attended value for the query q_i. P_i is the set of indices j where the keys k_j share the same hash bucket as the query q_i and z is the partition function. The computation aggregates contributions from only those keys k_j that belong to the same hash bucket as q_i, as determined by the hashing function $h(\cdot)$. The final embedding E_{final} is passed through the LSH attention layer, resulting in the output Y_{LSH}, where $Y_{\text{LSH}} \in \mathbb{R}^{M \times N}$.

To improve the ability of iReformer to analyse and interpret sequential data, the encoding process incorporates a TCN. The TCN is essential for capturing temporal dependencies, enhancing the iReformer model's ability to encode sequential information deeply and efficiently, and adapt to diverse data sources and sequence complexities.

In the TCN, two consecutive convolutional layers are employed, each followed by a rectified linear unit ReLU$(\cdot)$ activation and Dropout$(\cdot)$ for regularisation technique to prevent overfitting by randomly setting a fraction of input units to zero during training. The operations for each layer are described as:

$$Y_{\text{conv}} = \text{Conv}(Y_{\text{LSH}}), \tag{9}$$

$$Y_{\text{relu}} = \text{ReLU}(Y_{\text{conv}}), \tag{10}$$

$$Y_{\text{dropout}} = \text{Dropout}(Y_{\text{relu}}). \tag{11}$$

where Conv$(\cdot)$ represents a convolution operation that applies a filter to the input sequence Y_{LSH}, producing a transformed feature map $Y_{\text{conv}} \in \mathbb{R}^{M \times N}$. After ReLU activation, the feature map $Y_{\text{relu}} \in \mathbb{R}^{M \times N}$ is generated, followed by $Y_{\text{dropout}} \in \mathbb{R}^{M \times N}$ after applying dropout. The sequential operations repeat in the second layer, maintaining the same order of convolution, ReLU activation, and dropout as shown in Eqs. (9) to (11).

Furthermore, a residual connection is incorporated in the TCN to address the vanishing gradient problem [51]. The residual connection facilitates smoother training and improved gradient flow throughout the network:

$$Y_{\text{res}} = Y_{\text{dropout}} + Y_{\text{LSH}} \tag{12}$$

where $Y_{\text{res}} \in \mathbb{R}^{M \times N}$. The final output of the TCN layer is obtained after applying a ReLU activation to the residual connection output Y_{res}:

$$Y_{\text{TCN}} = \text{ReLU}(Y_{\text{res}}) \tag{13}$$

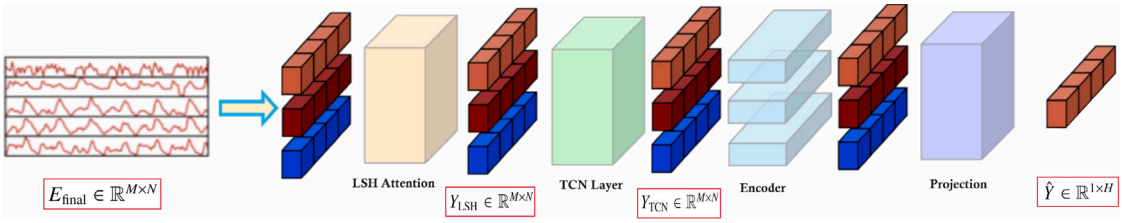


Fig. 4. An overview of TFTformer.

where $Y_{\text{TCN}} \in \mathbb{R}^{M\times N}$. The sequential TCN operations enable the encoder to efficiently process temporal sequences, capturing long-range dependencies with high accuracy. The TCN output, Y_{TCN}, proceeds through the reversible transformation which integrates RevNet principles and layer transformation (LayerNorm) by combining attention and feed-forward layers into reversible residual blocks. In this setup, layer normalisation is moved inside the residual blocks to enhance modularity and stability. The reversible transformations is defined as:

$$Y_1 = X_1 + \text{Attention}(X_2), \quad Y_2 = X_2 + \text{FeedForward}(Y_1) \tag{14}$$

The reversible blocks eliminate the need to store intermediate activations during backpropagation, significantly reducing memory usage. Finally, chunking is applied in the feed-forward layers to optimise memory usage, splitting computations into smaller sequential chunks (c) for efficient processing. The chunking feed-forward computation is defined as:

$$\begin{aligned} Y_2 &= \left[Y_2^{(1)}; \ldots ; Y_2^{(c)}\right] \\ &= \left[X_2^{(1)} + \text{FeedForward}(Y_1^{(1)}); \ldots ; X_2^{(c)} + \text{FeedForward}(Y_1^{(c)})\right] \end{aligned} \tag{15}$$

The final processed output undergoes a projection layer, resulting in a projection that produces the final load prediction $\hat{Y} \in \mathbb{R}^{1\times H}$. The structure of the TFTformer is illustrated in Fig. 4.

The integration of TCN with Reformer addresses both short and long-term dependency modelling in traditional Transformers. The Reformer's LSH attention mechanism aims to capture global dependencies; it may lose some temporal patterns due to its approximation nature. The TCN component compensates for this by using causal convolutions with dilation to directly capture both short and long-term dependencies, creating an exponentially growing receptive field. The combination of LSH and TCN enables the processing of temporal features through TCN's hierarchical convolutions while complementing the Reformer's approximate attention mechanism. The TCN structure with ReLU activation and residual connections enhances the learning of non-linear temporal relationships, while its causal padding prevents information leakage. Together, this architecture creates a more robust temporal feature representation that effectively addresses the Transformer's limitations in modelling dependencies at different time scales.

4. Case study

The study utilises load datasets from various geographical locations, including Belgium, New Zealand, and five Australian states: New South Wales (NSW), Victoria (VIC), Queensland (QLD), South Australia (SA), and Tasmania (TAS). The datasets span from January 1, 2015, to December 31, 2022. The data source and sampling rate are presented in Table 2. Additionally, all relevant weather parameters, such as temperature, humidity, cloud cover, wind speed, and wind direction, are sourced from the Open-Meteo website, with hourly availability. To ensure consistency across the datasets, an hourly average is computed for all load data and integrated with the hourly weather data. The dataset is divided into a training set (2015 to 2021) and a testing set (2022), resulting in a training-to-testing ratio of 7:1.

To evaluate the performance of the TFTformer, a comparative analysis is conducted with several state-of-the-art models, including FEDformer [52], Crossformer [53], FITS [54], iTransformer [40], iFlashformer [31], iFlowformer [32], iReformer [30], and CARD [55]. The hyperparameter selection is guided by iRefromer's findings that transformer model performance is robust to most hyperparameters, with learning rate being the most critical parameter requiring tuning. Extensive testing validated that the TFTformer achieves optimal performance with the same hyperparameters as iReformer. The Mean Absolute Percentage Error (MAPE) is used to evaluate the accuracy of the prediction, defined as:

$$\text{MAPE} = \frac{1}{H}\sum_{t=1}^{H}\frac{|Y(t) - \hat{Y}(t)|}{Y(t)} \times 100\% \tag{16}$$

where, Y is the observed data, $\hat{Y}$ is the predicted data, and H is the length of data.

Fig. 5 compares the MAPE for various forecasting methods across different seasonal periods for the seven regions. For Belgium, located in the northern hemisphere, seasons follow the conventional pattern (Spring: March–May, Summer: June–August, Autumn: September–November, Winter: December–February). The remaining regions in the southern hemisphere (New Zealand, NSW, VIC, QLD, SA, and TAS) experience opposite seasonal patterns. The results reveal significant variations in forecasting accuracy across different seasons and regions. Higher MAPE values during spring can be attributed to the most weather-fluctuated season globally [56], resulting in significant energy demand and supply variability that complicates accurate forecasting. Differences in higher MAPE values in summer for some and winter for others can be explained by the average load curves for these regions, as shown in Fig. 6. New Zealand, QLD, and TAS have mild summers and relatively flat load curves, with most variability driven by increased heating demand in winter. In contrast, regions with more pronounced seasonal differences, such as NSW, VIC, and SA, have their load curves greatly varied by the combined impacts of cooling demand during summer and heating demand during winter. In addition, the high error rates could be attributed to the high installations of rooftop photovoltaic systems, the dynamic nature of peak demand profiles, and the integration of distributed energy resources, as observed in SA [57]. However, TFTformer consistently exhibits the lowest MAPE values across all regions and seasons, demonstrating its robustness and reliability in maintaining low forecasting errors under diverse conditions.

Fig. 7 compares the MAPE across different forecasting methods and regions, analysed by day of the week. Across all areas, a general trend of increased MAPE on weekends is observed, highlighting the challenges of maintaining forecasting accuracy during these days. Among the methods compared, TFTformer consistently exhibits the lowest MAPE values across all regions and days of the week. TFTformer demonstrates superior performance and reliability, maintaining lower error rates during weekends when forecasting accuracy typically decreases.

The TFTformer consistently achieves the lowest MAPE values in both seasonal and daily analyses, demonstrating its robustness and adaptability across various temporal contexts, which highlights its capability to deliver accurate and reliable predictions, making it a valuable method for diverse and dynamic forecasting environments.

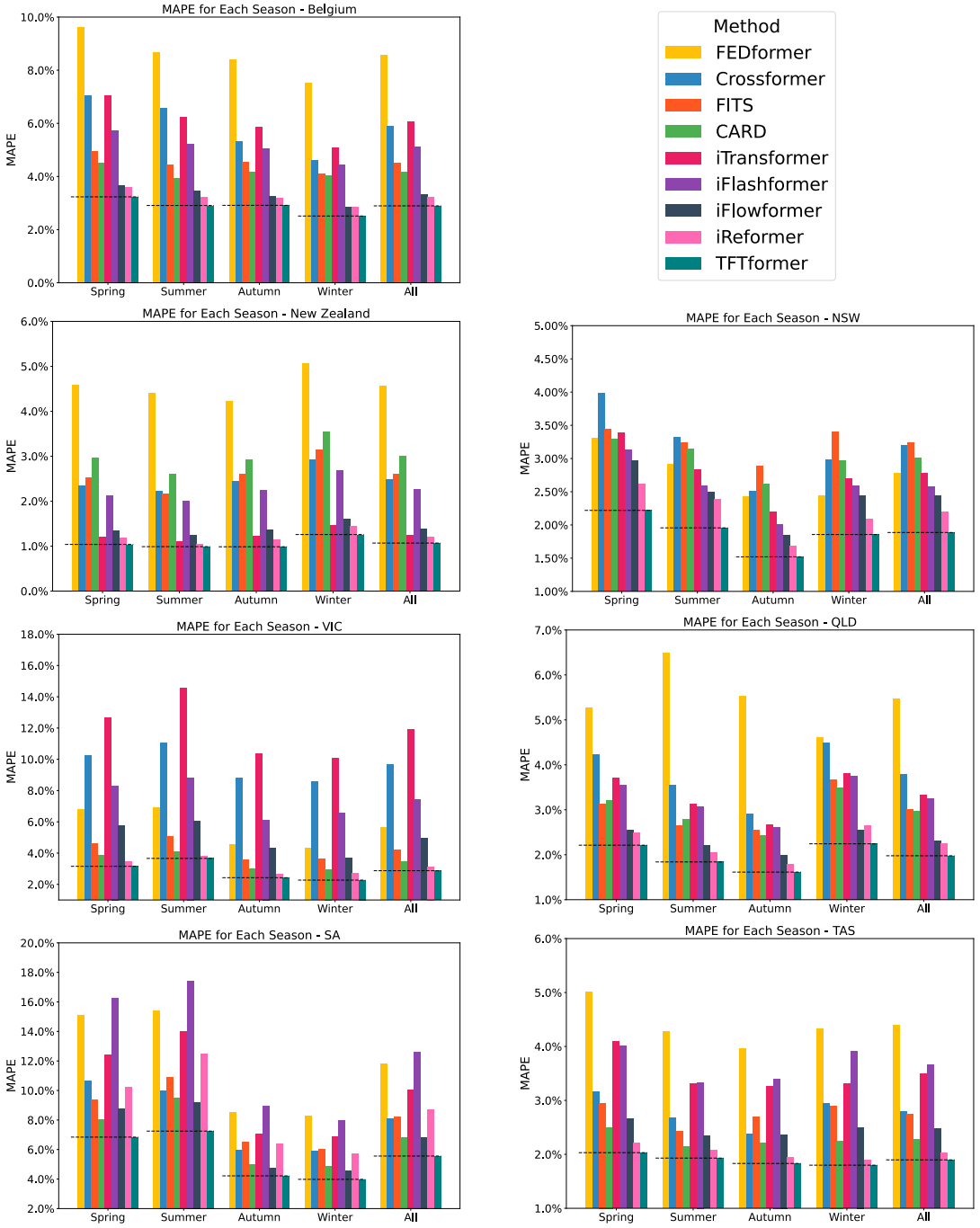


Fig. 5. MAPE for different methods for load forecasting for each season in Belgium, New Zealand, NSW, VIC, QLD, SA, and TAS.

Table 2
Data Summary.

Location	Data source	Sampling rate
Belgium	Elia TSO	Hourly
New Zealand	Electricity Authority	30 min
Australia (NSW, VIC, QLD, SA, TAS)	AEMO	Jan 1, 2015–Sep 30, 2021: 30 min Oct 1, 2021–Dec 31, 2022: 5 min

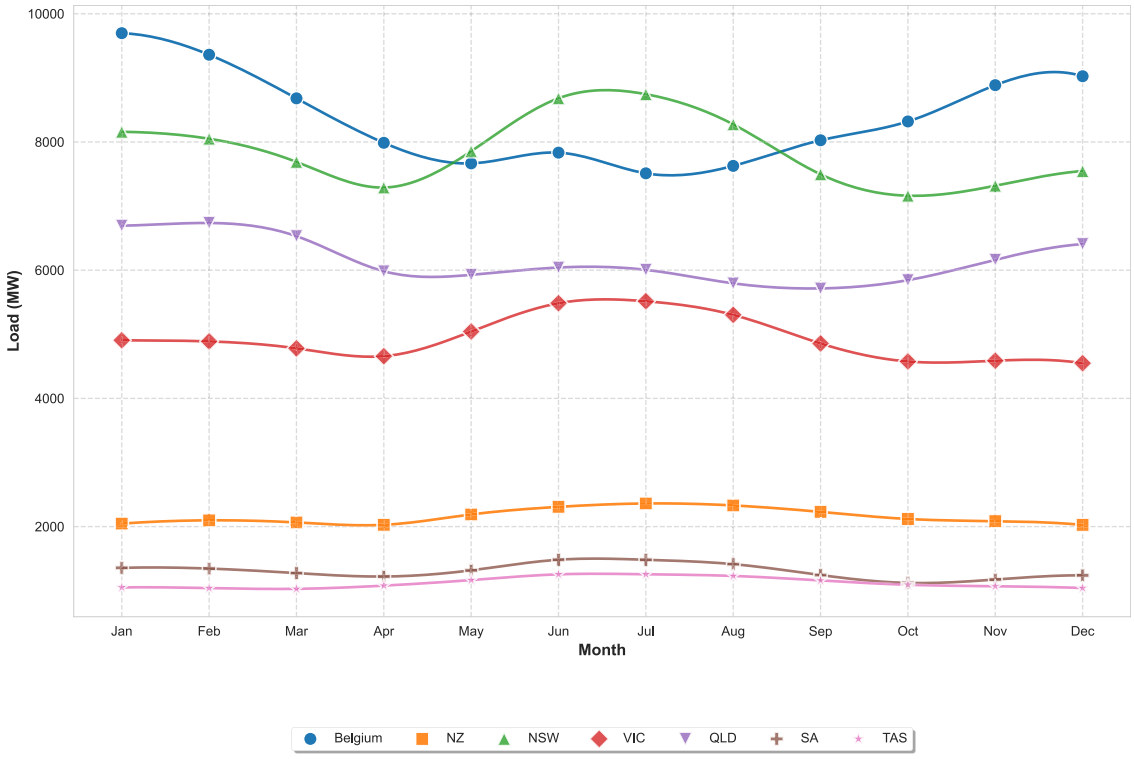


Fig. 6. Monthly Average Electricity Load by Region.

Table 3
Performance comparison of different models across locations using MAPE, MSE, MAE, and RMSE metrics (average and standard deviation of five runs per model per location).

Location	Metric	Models								
		FEDformer	Crossformer	FITS	iTransformer	iFlashformer	iFlowformer	iReformer	CARD	TFTformer
Belgium	MAPE	8.17 ± 0.64	5.66 ± 0.29	6.06 ± 0.11	5.32 ± 0.27	6.36 ± 1.03	3.16 ± 0.10	3.20 ± 0.27	4.00 ± 0.15	**3.15 ± 0.15**
	MSE	499036 ± 71179	239606 ± 19109	294259 ± 7867	224,235 ± 21,535	322,358 ± 94,295	80764 ± 4644	84052 ± 13964	130497 ± 10668	**80291 ± 6541**
	MAE	555.9 ± 43.1	356.0 ± 20.5	413.3 ± 7.0	364.2 ± 17.1	437.3 ± 71.8	216.7 ± 7.0	218.2 ± 20.0	272.5 ± 11.9	**215.7 ± 10.8**
	RMSE	704.6 ± 50.5	489.1 ± 19.8	542.4 ± 7.2	473.0 ± 22.3	561.4 ± 84.7	284.1 ± 8.2	289.0 ± 23.2	360.9 ± 14.8	**283.1 ± 11.7**
New Zealand	MAPE	4.54 ± 0.47	2.41 ± 0.20	3.77 ± 0.60	2.71 ± 1.85	2.90 ± 0.96	1.29 ± 0.06	1.22 ± 0.08	2.90 ± 0.15	**1.07 ± 0.01**
	MSE	15629 ± 2884	4654 ± 584	11869 ± 2388	8666 ± 11,692	7898 ± 6125	1322 ± 92	1233 ± 154	7450 ± 755	**962 ± 14**
	MAE	97.0 ± 9.7	52.0 ± 4.0	80.6 ± 11.3	58.9 ± 40.5	62.8 ± 20.8	27.9 ± 1.2	26.5 ± 1.7	63.1 ± 3.4	**23.4 ± 0.1**
	RMSE	124.4 ± 12.0	68.1 ± 4.3	108.5 ± 10.3	76.8 ± 52.7	83.8 ± 29.5	36.3 ± 1.3	35.0 ± 2.2	86.2 ± 4.4	**31.0 ± 0.2**
NSW	MAPE	6.02 ± 1.27	3.28 ± 0.32	4.02 ± 0.05	3.75 ± 0.09	3.96 ± 0.76	2.18 ± 0.15	2.16 ± 0.12	2.85 ± 0.15	**2.01 ± 0.08**
	MSE	330616 ± 136112	112745 ± 22271	168826 ± 2097	140,649 ± 7377	160,316 ± 61,944	49846 ± 4856	50168 ± 7411	89345 ± 9727	**39170 ± 2444**
	MAE	446.1 ± 94.5	243.6 ± 27.9	303.6 ± 3.4	278.5 ± 7.4	294.0 ± 54.6	162.4 ± 10.9	159.5 ± 7.9	213.6 ± 11.7	**148.6 ± 5.4**
	RMSE	564.4 ± 109.7	334.3 ± 31.6	410.9 ± 2.6	374.9 ± 9.8	394.2 ± 69.9	223.0 ± 10.7	223.4 ± 15.9	298.5 ± 16.5	**197.8 ± 6.3**
VIC	MAPE	6.75 ± 0.84	4.34 ± 0.09	5.86 ± 0.02	8.55 ± 1.95	9.28 ± 4.05	3.69 ± 0.46	3.97 ± 0.42	3.71 ± 0.19	**2.99 ± 0.11**
	MSE	166501 ± 39976	72688 ± 1775	134683 ± 708	274,568 ± 104,033	356,158 ± 265,928	53840 ± 11072	62645 ± 12378	58825 ± 5784	**38838 ± 3781**
	MAE	313.3 ± 38.4	194.5 ± 4.7	275.8 ± 1.0	386.1 ± 85.6	431.4 ± 193.6	170.1 ± 20.1	186.1 ± 18.2	174.3 ± 9.1	**139.4 ± 5.2**
	RMSE	405.0 ± 49.5	269.6 ± 3.3	367.0 ± 1.0	512.0 ± 111.5	554.0 ± 222.1	230.8 ± 23.7	249.1 ± 24.8	242.3 ± 11.8	**196.8 ± 9.7**
QLD	MAPE	5.04 ± 0.31	3.37 ± 0.28	3.72 ± 0.01	3.63 ± 0.59	3.26 ± 0.18	2.36 ± 0.20	2.17 ± 0.13	3.10 ± 0.09	**1.99 ± 0.06**
	MSE	145727 ± 17252	68857 ± 9437	90440 ± 494	87,942 ± 27,358	69,964 ± 7434	35055 ± 5309	29357 ± 2817	64634 ± 3659	**25036 ± 1534**
	MAE	293.8 ± 21.4	191.3 ± 16.2	217.1 ± 0.8	210.6 ± 33.8	188.6 ± 10.7	138.1 ± 11.6	125.9 ± 7.4	179.3 ± 5.0	**115.8 ± 3.7**
	RMSE	381.1 ± 22.4	261.8 ± 17.9	300.7 ± 0.8	293.3 ± 44.0	264.3 ± 14.3	186.7 ± 14.5	171.1 ± 8.4	254.1 ± 7.2	**158.2 ± 4.8**
SA	MAPE	12.16 ± 1.31	8.12 ± 0.79	10.68 ± 0.04	13.15 ± 2.54	13.91 ± 2.26	7.27 ± 0.28	7.66 ± 0.66	6.88 ± 0.14	**6.79 ± 0.26**
	MSE	30828 ± 8074	13198 ± 1705	23061 ± 126	33,522 ± 10,293	35,630 ± 7771	11402 ± 904	13217 ± 1520	**10831 ± 421**	12802 ± 1640
	MAE	132.0 ± 17.0	81.8 ± 4.0	108.0 ± 0.4	128.0 ± 23.3	135.6 ± 19.0	75.7 ± 3.0	78.7 ± 6.1	**69.6 ± 1.4**	69.7 ± 3.3
	RMSE	174.2 ± 221.4	114.6 ± 7.4	151.9 ± 0.4	181.0 ± 27.7	187.6 ± 20.8	106.7 ± 4.3	114.8 ± 6.6	**104.1 ± 2.0**	112.9 ± 7.1
TAS	MAPE	4.41 ± 0.19	2.66 ± 0.12	3.28 ± 0.06	3.62 ± 0.97	4.21 ± 1.12	2.13 ± 0.26	2.04 ± 0.14	2.35 ± 0.10	**1.87 ± 0.02**
	MSE	4068 ± 339	1580 ± 157	2556 ± 63	3040 ± 1590	4122 ± 1987	1043 ± 237	957 ± 120	1280 ± 105	**801 ± 17**
	MAE	50.6 ± 2.1	31.0 ± 1.4	38.5 ± 0.6	41.7 ± 11.2	48.7 ± 12.9	24.6 ± 3.0	23.6 ± 1.6	27.3 ± 1.2	**21.6 ± 0.3**
	RMSE	63.7 ± 2.7	39.7 ± 2.0	50.6 ± 0.6	53.4 ± 13.9	62.1 ± 16.2	32.1 ± 3.6	30.9 ± 1.9	35.8 ± 1.4	**28.3 ± 0.3**

Table 3 presents the detailed performance metrics MAPE, Mean Squared Error (MSE), Mean Absolute Error (MAE), and Root Mean Squared Error (RMSE) for each model across all locations, with each value representing the average and standard deviation of five independent runs. The repeated experiments for each model and location help demonstrate the consistency and reliability of our findings while also allowing for a thorough comparison of model performance across different geographical contexts and evaluation metrics.

Seven scatter plots for each location and an additional error metric are presented in Appendix. In the scatter plots, the x-axis typically represents the observed or actual values, while the y-axis represents the predicted values by each model, providing a clear representation of each model's predicting accuracy. The figures show three metrics: Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and the coefficient of determination (R^2). The results highlight TFTformer's remarkable accuracy and minimal deviation in its predictions from the actual values, demonstrating its superior performance and minimum error compared to other models.

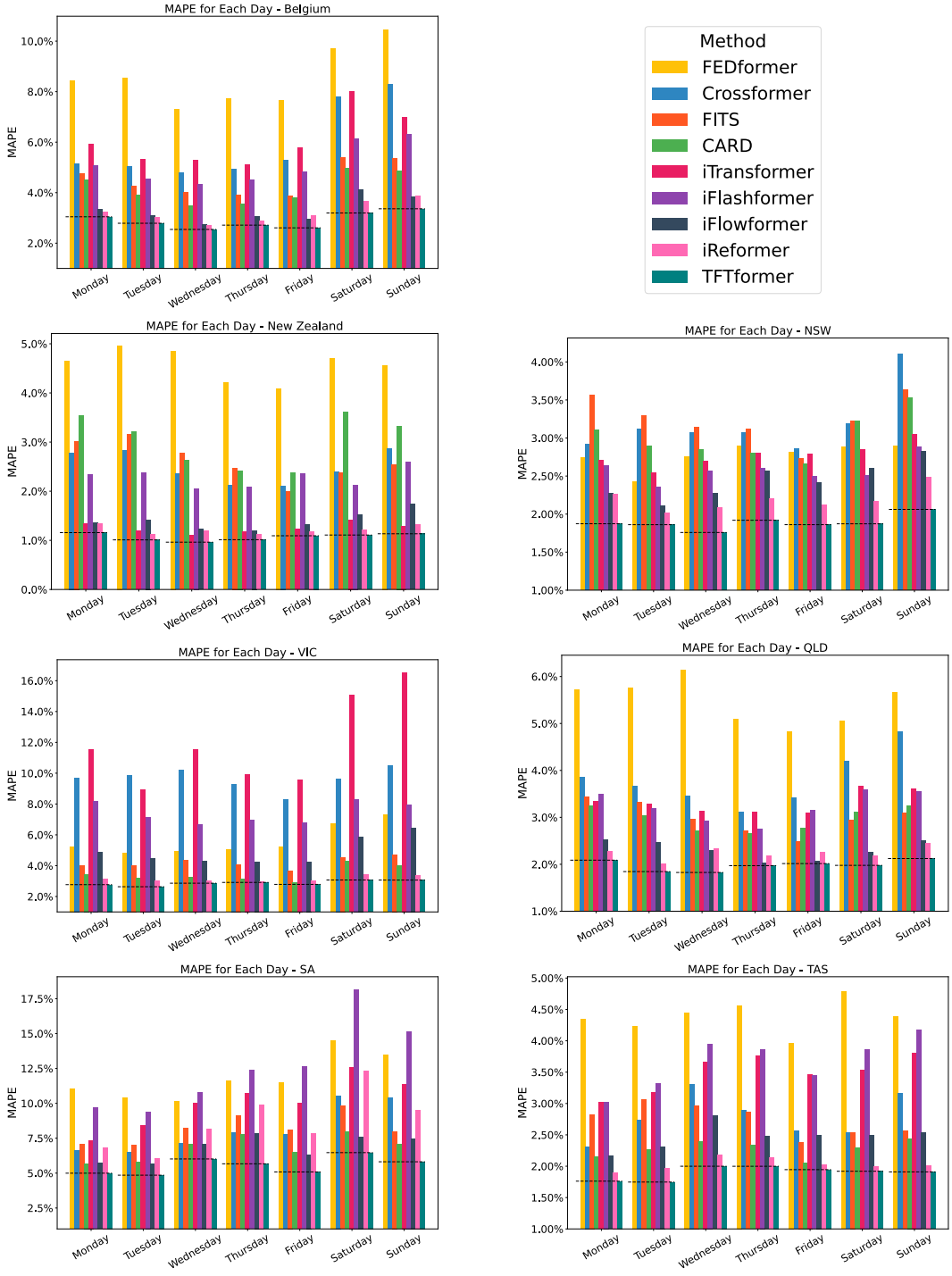


Fig. 7. MAPE for different methods for load forecasting for each day in Belgium, New Zealand, NSW, VIC, QLD, SA, and TAS.

Table 4
Impact of incremental enhancements on MSE across different datasets.

Component				Location						
Data preprocessing	Feature-specific embedding	Linear layer post-embedding	TCN layer	NSW	VIC	QLD	SA	TAS	NZ	ELIA
×	×	×	×	46 873	50 439	31 638	15 758	953	1191	84 816
×	×	×	✓	47 396	59 466	28 698	12 251	860	1185	82 385
×	×	✓	×	**37 366**	70 062	33 377	12 083	871	1012	86 235
×	×	✓	✓	46 107	89 781	32 317	12 382	836	1007	86 822
×	✓	×	×	53 900	50 701	29 258	13 225	1021	1436	122 522
×	✓	×	✓	45 408	53 771	30 134	19 277	966	1296	115 998
×	✓	✓	×	51 877	41 783	25 806	17 032	909	1167	109 253
×	✓	✓	✓	47 029	37 463	27 806	13 839	926	1353	113 369
✓	×	×	×	39 843	**36 837**	**24 117**	15 934	903	1002	**72 875**
✓	×	×	✓	43 631	44 797	25 518	14 304	888	1036	78 905
✓	×	✓	×	38 709	53 917	38 265	14 903	**788**	867	85 162
✓	×	✓	✓	40 600	46 770	**23 950**	14 906	**816**	**903**	79 356
✓	✓	×	×	38 840	42 791	24 201	14 056	942	**962**	77 973
✓	✓	×	✓	50 082	40 300	28 503	**10 623**	903	986	77 270
✓	✓	✓	×	**35 619**	34066	23062	**11 120**	779	967	**75 484**
✓	✓	✓	✓	35167	**34 316**	24 867	7355	**816**	969	69473

Table 5
Impact of incremental enhancements on MAPE across different datasets.

Component				Location						
Data preprocessing	Feature-specific embedding	Linear layer post-embedding	TCN layer	NSW	VIC	QLD	SA	TAS	NZ	ELIA
×	×	×	×	2.19%	3.53%	2.25%	8.69%	2.03%	1.21%	3.22%
×	×	×	✓	2.15%	3.96%	2.12%	6.78%	1.93%	1.20%	3.17%
×	×	✓	×	**1.97%**	4.34%	2.30%	6.67%	1.96%	1.10%	3.27%
×	×	✓	✓	2.03%	4.95%	2.26%	6.77%	1.90%	1.10%	3.28%
×	✓	×	×	2.37%	3.60%	2.16%	7.07%	2.13%	1.31%	3.89%
×	✓	×	✓	2.05%	3.55%	2.19%	9.47%	2.08%	1.27%	3.80%
×	✓	✓	×	2.34%	3.27%	2.04%	9.34%	2.01%	1.21%	3.69%
×	✓	✓	✓	2.19%	**2.98%**	2.13%	8.41%	2.04%	1.29%	3.77%
✓	×	×	×	2.01%	3.00%	**1.94%**	6.89%	2.00%	1.09%	**3.00%**
✓	×	×	✓	2.13%	3.19%	2.02%	7.08%	1.98%	1.11%	3.12%
✓	×	✓	×	1.99%	3.36%	2.54%	6.68%	**1.84%**	1.01%	3.31%
✓	×	✓	✓	2.02%	3.11%	**1.93%**	**6.66%**	1.91%	**1.02%**	3.17%
✓	✓	×	×	1.99%	3.22%	1.95%	7.01%	2.04%	**1.07%**	3.06%
✓	✓	×	✓	2.28%	3.15%	2.07%	**6.25%**	2.01%	1.08%	3.08%
✓	✓	✓	×	**1.91%**	2.84%	1.88%	6.84%	1.83%	**1.07%**	**3.05%**
✓	✓	✓	✓	1.89%	**2.87%**	1.98%	5.56%	**1.90%**	**1.07%**	2.90%

5. Ablation study

A comprehensive ablation study is conducted to evaluate the contributions of each element of the TFTformer model to load forecasting accuracy. The study begins by computing the MSE and MAPE for every possible combination of elements. The systematic analysis assesses the impact of each component, both individually and in combination. TFTformer has four different elements, resulting in sixteen different combinations.

Tables 4 and 5 present the MAPE and MSE values across seven datasets with the progressive integration of model enhancements. The best results are underlined in red, the second-best in blue, and the third-best in green. The results show that the best performance generally comes from combining data preprocessing, transposed feature-specific embedding, and linear layer post-embedding, as observed in regions such as VIC, QLD, and TAS. Adding the TCN layer shows the best results in some locations, such as NSW, SA, and ELIA. Data preprocessing consistently provides significant improvements, but the impact of each technique and its combinations varies by region.

ANOVA is a statistical method used to identify significant differences among group means by analysing variations within and between groups, which is essential for evaluating the impact of various factors on a dependent variable [58]. This study applies ANOVA to assess the influence of individual components within the TFTformer model on prediction accuracy metrics, MSE and MAPE. Blocking is employed to mitigate variability caused by regional variations that could influence ANOVA performance. A logarithmic transformation is also applied to stabilise variance and ensure adherence to the normal distribution assumptions critical for the validity of ANOVA results.

The analysis, detailed in Tables 6 and 7, shows that data preprocessing has a consistently significant impact on both MSE and MAPE, with p-values less than 0.001, highlighting the optimal use of data preprocessing in improving prediction accuracy [59]. Furthermore, a significant interaction between data preprocessing and transposed feature-specific embedding underscores the importance of managing different data effectively to enhance model performance [60]. The results also reveal a meaningful interaction between transposed feature-specific embedding and the linear layer post-embedding, indicating that the linear layers adeptly capture the unique dynamics of each data type by independently embedding the time series of each variable [39,40]. However, the non-significant effect of the TCN layer suggests that geographical and climatic variations at different locations may influence the performance of TCN models, indicating a potential need for fine-tuning the TCN layer to suit each specific location [61].

Table 6
The ANOVA on MSE after the log-transformation.

Effects	Source	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Main Effects	**Data Preprocessing (DP)**	**1**	**0.90**	**0.880**	**40.718**	**7.46×10^{-9}**
	Transposed Feature-Specific Embedding (FE)	1	0.00	0.010	0.572	0.4514
	Linear Layer Post-Embedding (LL)	**1**	**0.10**	**0.080**	**3.677**	**0.0583**
	TCN Layer (TC)	1	0.00	0.000	0.107	0.7448
Two-Way Interactions	**DP × FE**	**1**	**0.20**	**0.160**	**7.306**	**0.0082**
	DP × LL	1	0.00	0.000	0.077	0.7824
	FE × LL	**1**	**0.10**	**0.130**	**5.817**	**0.0179**
	DP × TC	1	0.00	0.000	0.148	0.7011
	FE × TC	1	0.00	0.000	0.028	0.8680
	LL × TC	1	0.00	0.010	0.275	0.6013
Three-Way Interactions	DP × FE × LL	1	0.00	0.010	0.482	0.4891
	DP × FE × TC	1	0.00	0.000	0.001	0.9728
	DP × LL × TC	1	0.00	0.030	1.380	0.2432
	FE × LL × TC	1	0.00	0.000	0.094	0.7593
Four-Way Interaction	DP × FE × LL × TC	1	0.00	0.020	0.813	0.3697
Residuals		90	1.90	0.020		

Table 7
The ANOVA on MAPE after the log-transformation.

Effects	Source	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Main Effects	**Data Preprocessing (DP)**	**1**	**0.35**	**0.350**	**48.507**	**5.22×10^{-10}**
	Transposed Feature-Specific Embedding (FE)	1	0.00	0.001	0.114	0.7364
	Linear Layer Post-Embedding (LL)	1	0.02	0.016	2.287	0.1340
	TCN Layer (TC)	1	0.00	0.002	0.215	0.6442
Two-Way Interactions	**DP × FE**	**1**	**0.03**	**0.034**	**4.774**	**0.0315**
	DP × LL	1	0.00	0.002	0.279	0.5987
	FE × LL	1	0.01	0.012	1.627	0.2055
	DP × TC	1	0.00	0.000	0.000	0.9970
	FE × TC	1	0.00	0.000	0.012	0.9147
	LL × TC	1	0.00	0.003	0.405	0.5263
Three-Way Interactions	DP × FE × LL	1	0.00	0.004	0.524	0.4709
	DP × FE × TC	1	0.00	0.000	0.015	0.9031
	DP × LL × TC	1	0.01	0.009	1.303	0.2567
	FE × LL × TC	1	0.00	0.001	0.086	0.7698
Four-Way Interaction	DP × FE × LL × TC	1	0.01	0.008	1.113	0.2943
Residuals		90	0.65	0.007		

On the other hand, the analysis of three-way and four-way interactions provides important insights. Firstly, there are no significant interactions between any combinations of three or four components, as all p-values exceed the 0.05 threshold. This indicates that the components primarily work independently or through simple two-way interactions, rather than through complex multi-way interactions. The results suggest that each component addresses shortcomings in the Transformer model without negatively impacting the others.

The histograms presented in Fig. 8 demonstrate the distribution of MSE and MAPE residuals before and after applying a logarithmic transformation. Before the transformation, both MSE and MAPE residuals exhibit noticeable skewness, deviating from a normal distribution. After the logarithmic transformation, the residuals for both metrics become more symmetrically distributed, closely aligning with a normal distribution. The improved distribution symmetry is reflected in higher R^2 values for both MSE (from 0.811 to 0.849) and MAPE (from 0.653 to 0.834). The logarithmic transformation effectively normalises the residuals, enhancing the robustness and reliability of ANOVA analysis.

6. Interpretability analysis of TFTformer model using SHAP-based surrogate approach

This study presents an interpretability analysis of the TFTformer model using a surrogate modelling approach integrated with SHAP (Shapley Additive exPlanations) analysis [44]. Modern transformer-based architectures feature high-dimensional parameter spaces and complex attention mechanisms, making them computationally intensive to interpret using direct methods [45]. This interpretability challenge is particularly significant for the TFTformer, as it combines transformer architecture and TCN, resulting in increased model complexity. Recent research has demonstrated that surrogate modelling provides an effective approach for interpreting deep neural networks by analysing their behaviour through more tractable representations [62, 63]. The surrogate model implementation approximates the complex model behaviour using more interpretable models while maintaining the underlying input–output mappings [64]. Particularly, the study leverages Random Forest regression as the surrogate model, exploiting its capacity to capture non-linear feature interactions while providing inherent interpretability through its tree-based architecture [46]. This approach performs well, as the Random Forest surrogate effectively replicates the base model's predictions while handling complex decision boundaries [65]. Through training the surrogate to approximate the TFTformer's behaviour, the SHAP analysis extracts feature importance rankings and interprets the model's decision-making process.

Random Forest regressor is employed with hyperparameters tuned through grid search to optimise model performance while preventing overfitting. The final configuration included 200 estimators, a maximum depth of 12, minimum samples per leaf of 1, minimum samples

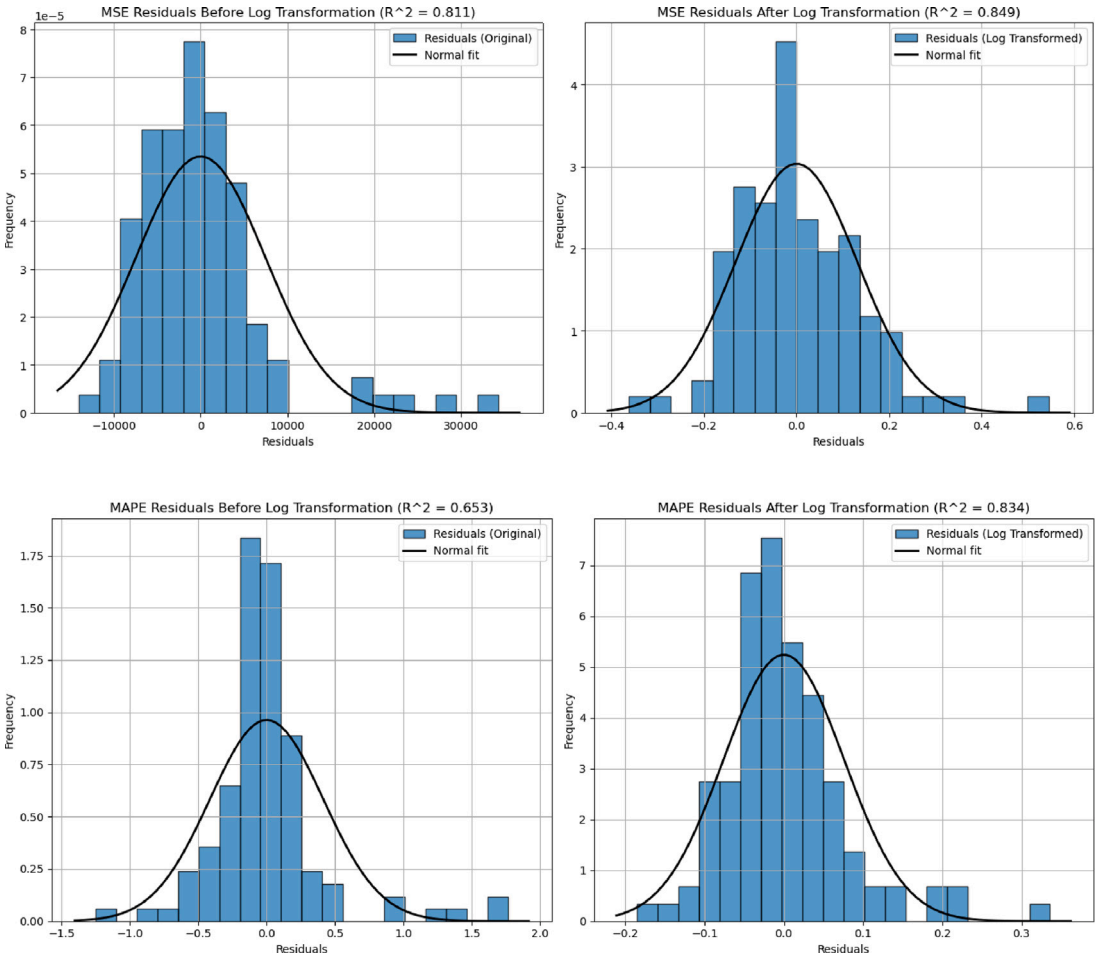


Fig. 8. Residuals Before and After Log Transformation for MSE and MAPE.

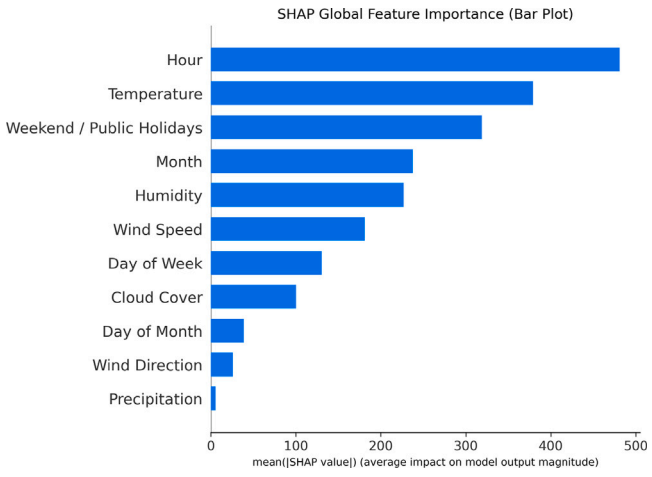


Fig. 9. SHAP Feature Importance Rankings for TFTformer Model Predictions.

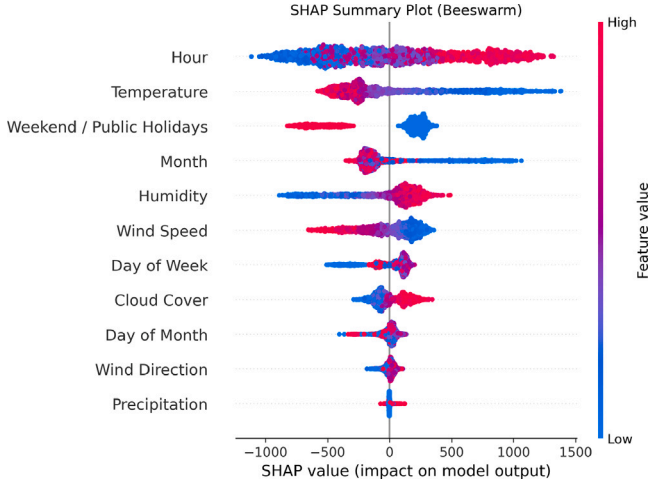


Fig. 10. SHAP Summary (Beeswarm) Plot.

for a split of 2, and the square root of features considered for splits. The model is trained on an 80/20 data split. Performance metrics demonstrate strong fidelity, with the surrogate achieving an R^2 of 0.8875 in replicating the TFTformer's predictions. Additionally, the surrogate achieves an R^2 of 0.8892 against the actual target values, indicating that it not only captures the TFTformer's behaviour but also maintains comparable prediction performance on actual data.

SHAP is used to understand how different features influence the surrogate of TFTformer's predictions. The analysis produces two key visualisations that help interpret the TFTformer's behaviour at both global and individual prediction levels.

Fig. 9 shows the SHAP global feature importance as a bar plot, which ranks features by their overall impact on predictions. The plot measures feature importance using mean absolute SHAP values — the

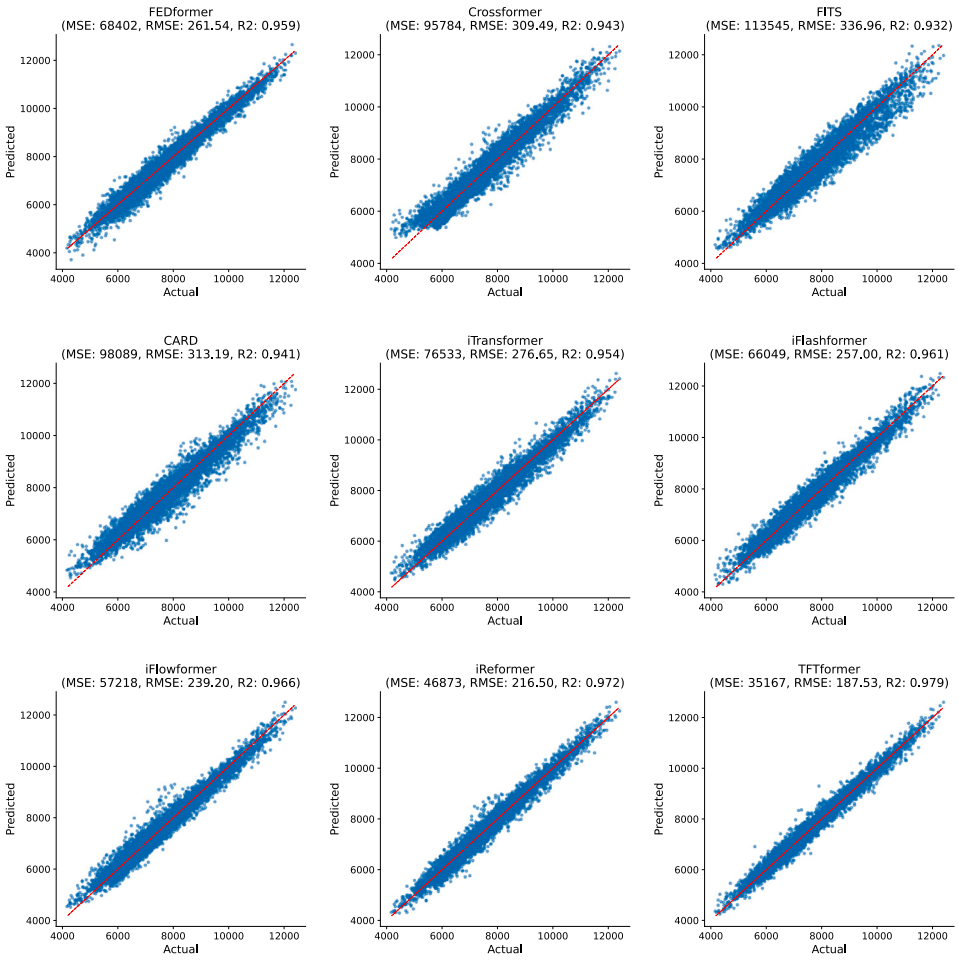


Fig. A.11. Scatter plot for different methods for load forecasting for NSW with different error metrics.

higher the value, the more that feature influences the TFTformer's decisions. Hour emerges as the most influential feature, followed closely by temperature and weekend/holiday status. Month, humidity, and wind speed show moderate importance.

Fig. 10 presents a SHAP summary plot in beeswarm format, where the x-axis shows SHAP values ranging from −1000 to 1500, indicating the size and direction of each feature's impact. When a SHAP value is positive, that feature increases the prediction; when negative, it decreases the prediction. The y-axis lists all the features, with hour at the top as the most influential. Each dot represents a single instance, and its colour (from blue to red) shows whether that feature had a low or high value. The visualisation reveals that later hours consistently drive predictions higher, while temperature shows a clear positive correlation with predicted values. Weekend and holiday periods form distinct clusters indicating higher predictions than regular weekdays, and seasonal factors captured by Month show moderate but consistent effects.

The interpretability analysis reveals how the TFTformer model weighs different input features. The findings show that the hour of the day, weekend/holiday status, and temperature consistently have the strongest influence on predictions. The results align with established patterns in grid load behaviour, where daily usage cycles, weekend and holiday routines, and weather conditions primarily drive electricity consumption. The high correlation between the surrogate model's outputs and both the TFTformer's predictions and actual values validate that the TFTformer has effectively learned to prioritise these key features for accurate forecasting.

7. Conclusion

This study introduced TFTformer, a transformer-based model enhancing electrical load forecasting through technical contributions, comprehensive validation across different regions, and detailed statistical analysis of each component's contribution. The methodological developments directly address key limitations in existing Transformer architectures for time series forecasting:

- Methodological contributions:
  - Transposed feature-specific embedding: Independently embeds weather, time, and load data, optimising feature space mappings and enhancing the capture of variable-specific temporal dynamics.
  - Linear layer post-embedding: Aligns and standardises features across different data types, enhancing pattern recognition and model responsiveness through refined attention mechanisms.
  - Temporal convolutional network integration: While the LSH attention struggles to capture sequential dependencies, integrating TCN by employing causal convolutions and dilation in the encoder enables better capture of both short and long-term patterns in load data.
- Model validation and performance:

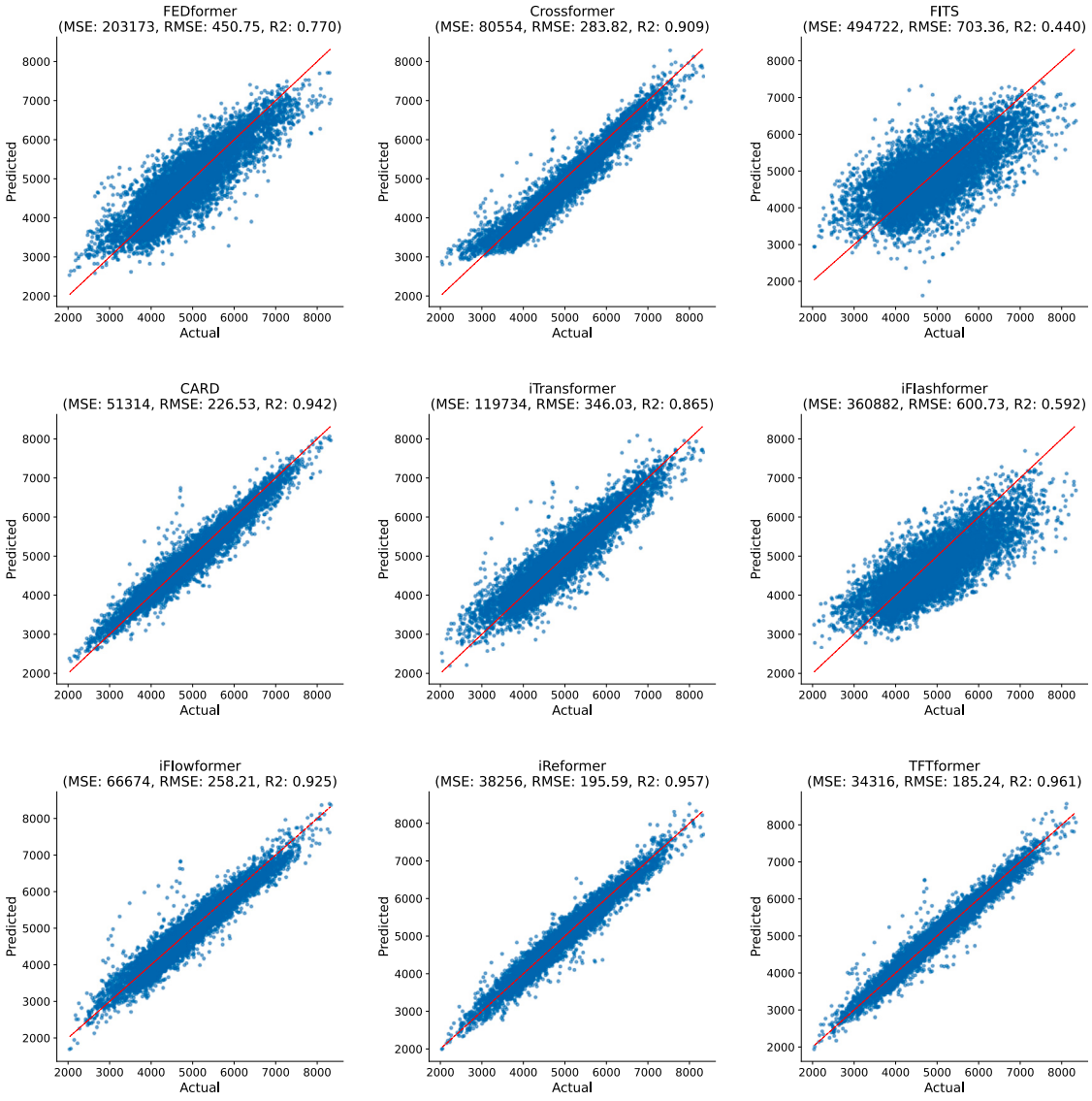


Fig. A.12. Scatter plot for different methods for load forecasting for VIC with different error metrics.

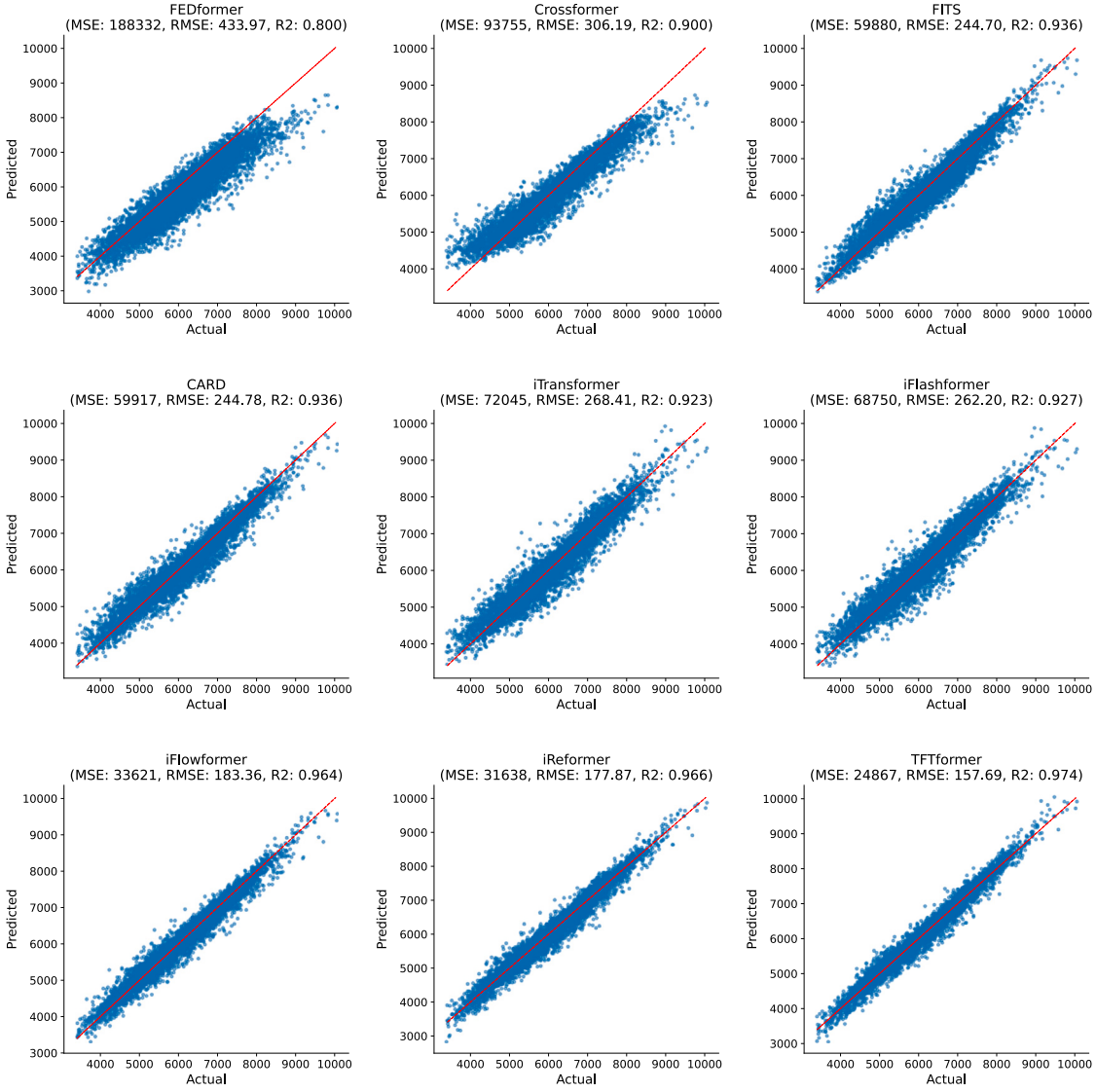


Fig. A.13. Scatter plot for different methods for load forecasting for QLD with different error metrics.

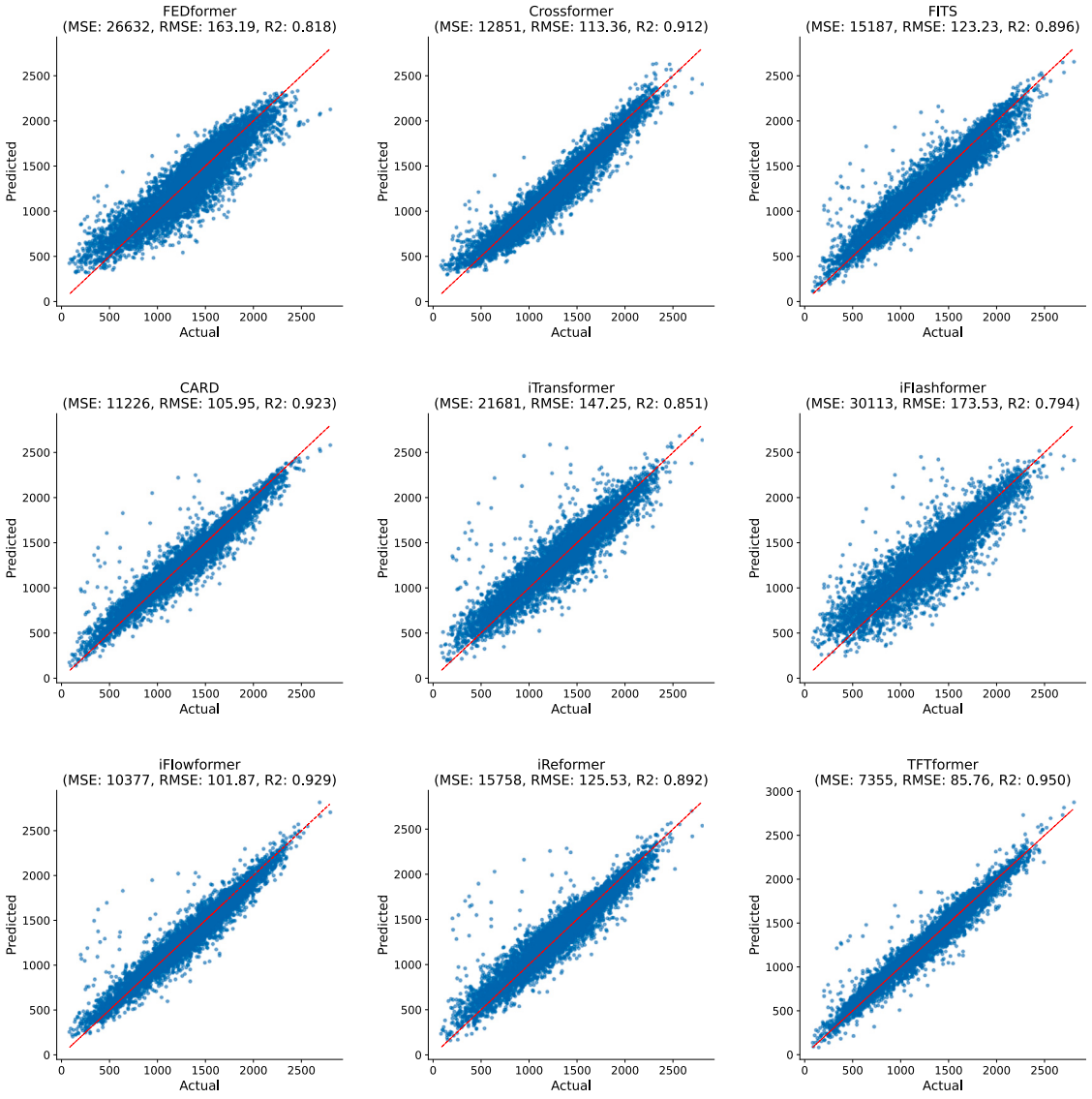


Fig. A.14. Scatter plot for different methods for load forecasting for SA with different error metrics.

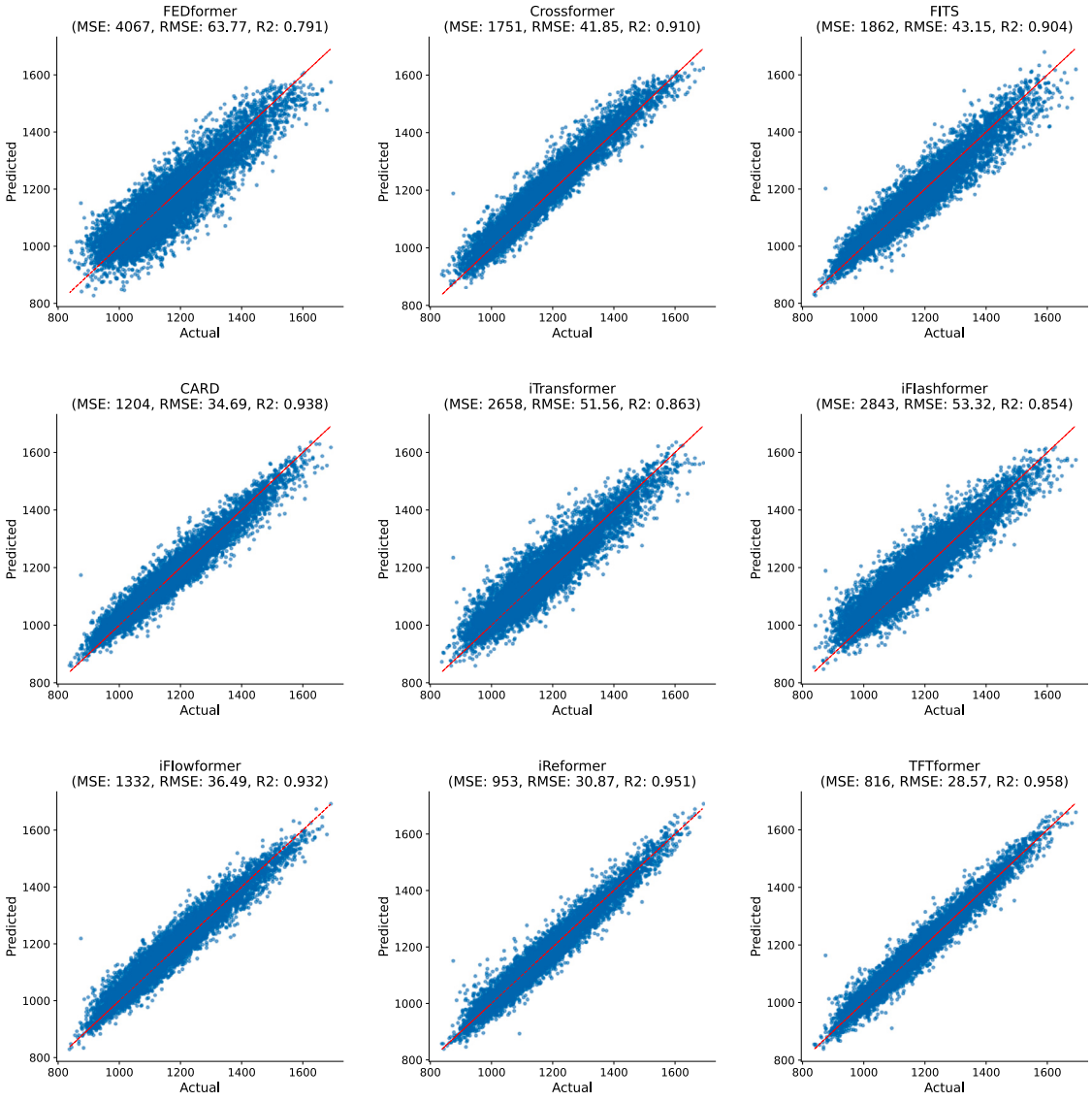


Fig. A.15. Scatter plot for different methods for load forecasting for TAS with different error metrics.

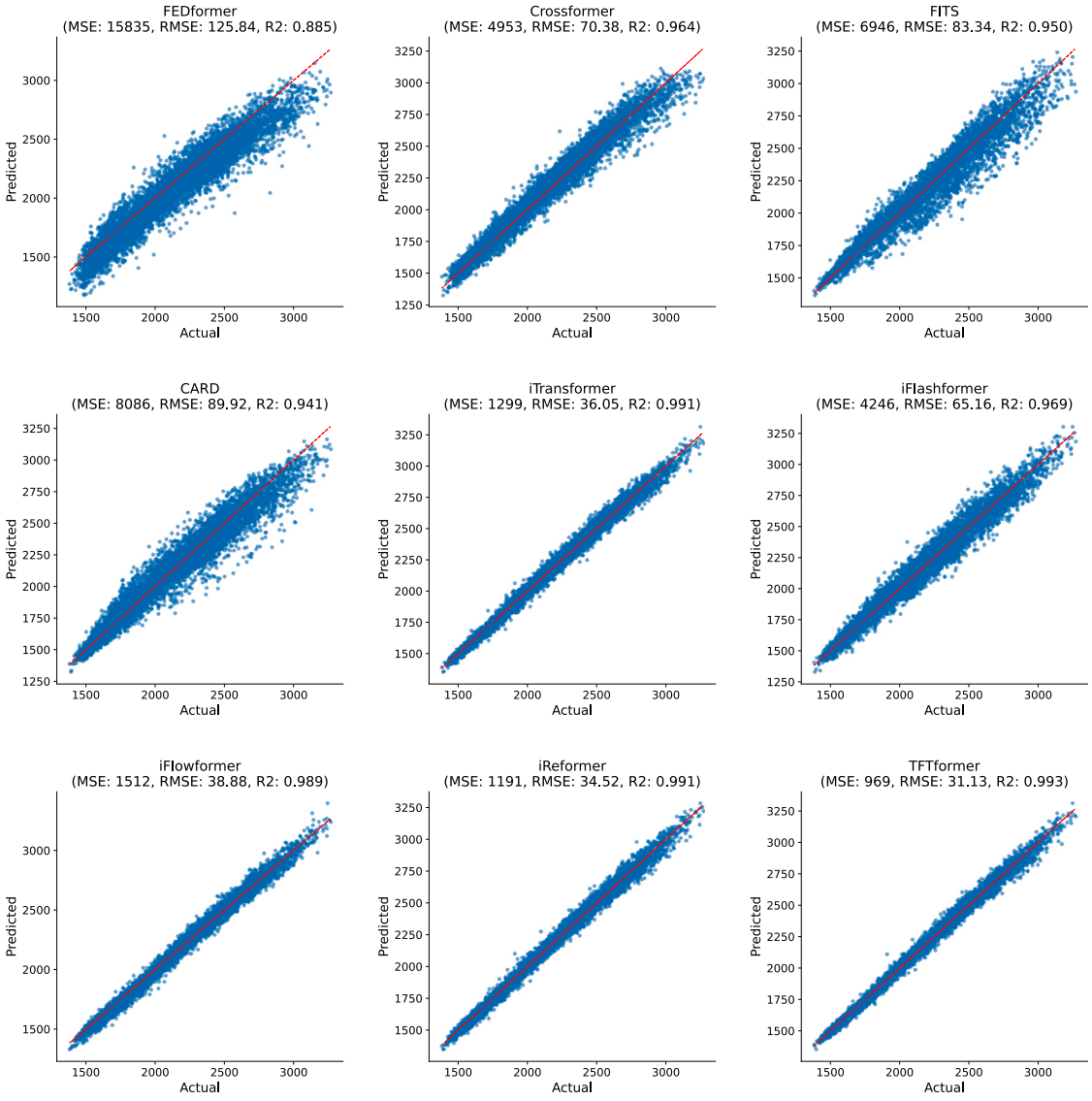


Fig. A.16. Scatter plot for different methods for load forecasting for New Zealand with different error metrics.

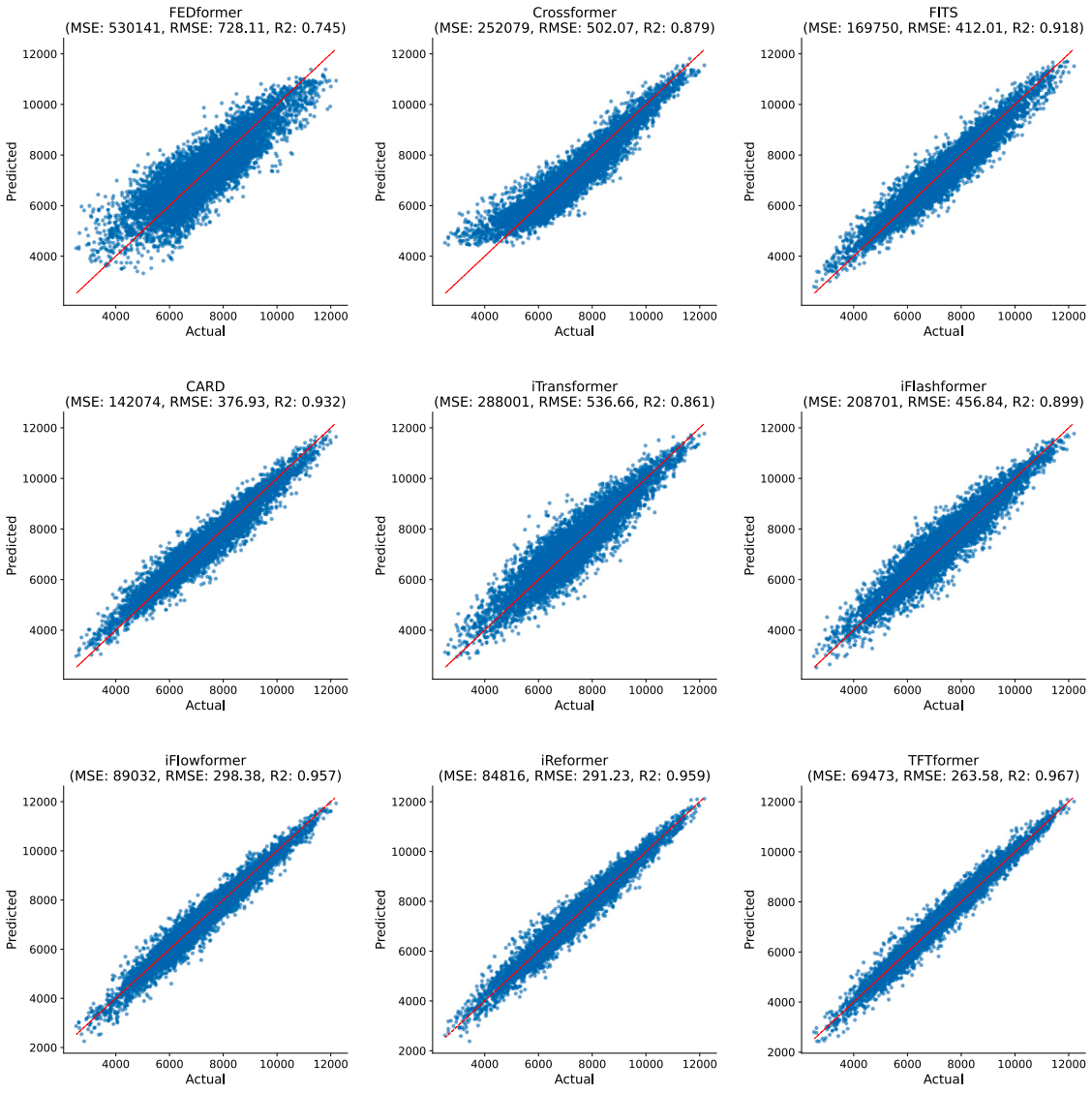


Fig. A.17. Scatter plot for different methods for load forecasting for Belgium with different error metrics.

- Several region validations: Tests the TFTformer across seven different locations (Belgium, New Zealand, and five Australian states), showing consistent performance improvements across different grid systems and operating conditions.
- Performance measurements: Achieves the lowest MAPE, MSE, MAE, and RMSE values compared to different models across seasonal fluctuations, with MSE improvements of over 50% against most comparing models and 16%–17% against the nearest competitors.
- Statistical validation: Uses ANOVA testing to verify the effectiveness of each component, confirms the importance of the methodological elements, and demonstrates how different parts of TFTformer work together effectively.
- Interpretability analysis: Implements a SHAP-based surrogate model approach that achieves high fidelity ($R^2 > 0.88$) in replicating TFTformer's behaviour while providing transparent insights into feature importance and decision-making processes.

In future work, the TFTformer model can be extended in several promising directions to enhance its applicability and functionality. Incorporating diverse data types, such as economic indicators or detailed environmental metrics, could refine the prediction capacity and sensitivity of the model to external influences. Beyond traditional load forecasting, testing the TFTformer in varied contexts, such as energy consumption of electric vehicles or battery health monitoring, could reveal its potential in other critical areas. The expanded applications could showcase the model's versatility in managing different types of time-series data and its adaptability to distinct operational needs.

CRediT authorship contribution statement

Ahmad Ahmad: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Xun Xiao:** Writing – review & editing, Supervision, Resources, Project administration, Methodology, Investigation, Conceptualization. **Huadong Mo:** Writing – review & editing, Supervision, Resources, Project administration, Funding acquisition, Conceptualization. **Daoyi Dong:** Writing – review & editing, Supervision.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

Huadong Mo reports financial support was provided by Australian Research Council. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

The paper is supported by the ARC Linkage Project LP210200473 and ARC Industrial Transformation Research Hub for Integrated Energy Storage Solutions IH180100020.

Appendix. Scatter plots and additional error metrics to evaluate the TFTformer in various locations

See Figs. A.11–A.17.

Data availability

Data will be made available on request.

References

[1] Australia CaEN electricity network transformation roadmap: final report, Commonwealth of Australia; 2017.
[2] Qiao W, Li Z, Liu W, Liu E. Fastest-growing source prediction of US electricity production based on a novel hybrid model using wavelet transform. Int J Energy Res 2022;46:1766–88.
[3] Wei N, Yin C, Yin L, Tan J, Liu J, Wang S, et al. Short-term load forecasting based on WM algorithm and transfer learning model. Appl Energy 2024;353:122087.
[4] Kuster C, Rezgui Y, Mourshed M. Electrical load forecasting models: A critical systematic review. Sustain Cities Soc 2017;35:257–70.
[5] Li K, Huang W, Hu G, Li J. Ultra-short term power load forecasting based on CEEMDAN-SE and LSTM neural network. Energy Build 2023;279:112666.
[6] Tarmanini C, Sarma N, Gezegin C, Ozgonenel O. Short term load forecasting based on ARIMA and ANN approaches. Energy Rep 2023;9:550–7.
[7] Dubey A, Kumar A, García-Díaz V, Sharma A, Kanhaiya K. Study and analysis of SARIMA and LSTM in forecasting time series data. Sustain Energy Technol Assess 2021;47:101474.
[8] Mado I, Rajagukguk A, Triwiyatno A, Fadllullah A. Short-term electricity load forecasting model based dsarima. Int J Electr Energy Power Syst Eng 2022;5:6–11.
[9] He Y, Cao C, Wang S, Fu H. Nonparametric probabilistic load forecasting based on quantile combination in electrical power systems. Appl Energy 2022;322:119507.
[10] Moradzadeh A, Mansour-Saatloo A, Nazari-Heris M, Mohammadi-Ivatloo B, Asadi S. Introduction and literature review of the application of machine learning/deep learning to load forecasting in power system. In: Application of machine learning and deep learning methods to power system problems. 2021, p. 119–35.
[11] Dudek G, Pełka P, Smyl S. A hybrid residual dilated LSTM and exponential smoothing model for midterm electric load forecasting. IEEE Trans Neural Netw Learn Syst 2021;33:2879–91.
[12] Rafati A, Joorabian M, Mashhour E. An efficient hour-ahead electrical load forecasting method based on innovative features. Energy 2020;201:117511.
[13] Dai Y, Zhao P. A hybrid load forecasting model based on support vector machine with intelligent methods for feature selection and parameter optimization. Appl Energy 2020;279:115332.
[14] Li G, Li Y, Roozitalab F. Midterm load forecasting: A multistep approach based on phase space reconstruction and support vector machine. IEEE Syst J 2020;14:4967–77.
[15] Aprillia H, Yang H, Huang C. Statistical load forecasting using optimal quantile regression random forest and risk assessment index. IEEE Trans Smart Grid 2020;12:1467–80.
[16] Khodayar M, Wang J. Probabilistic time-varying parameter identification for load modeling: A deep generative approach. IEEE Trans Ind Inform 2020;17:1625–36.
[17] Wazirali R, Yaghoubi E, Abujazar M, Ahmad R, Vakili A. State-of-the-art review on energy and load forecasting in microgrids using artificial neural networks, machine learning, and deep learning techniques. Electr Power Syst Res 2023;225:109792.
[18] Hafeez G, Alimgeer K, Khan I. Electric load forecasting based on deep learning and optimized by heuristic algorithm in smart grid. Appl Energy 2020;269:114915.
[19] Xiao X, Mo H, Zhang Y, Shan G. Meta-ANN–a dynamic artificial neural network refined by meta-learning for short-term load forecasting. Energy 2022;246:123418.
[20] Zhou M, Wang L, Hu F, Zhu Z, Zhang Q, Kong W, et al. ISSA-LSTM: A new data-driven method of heat load forecasting for building air conditioning. Energy Build 2024;114698.
[21] Turkoglu MO, D'Aronco S, Wegner JD, Schindler K. Gating revisited: Deep multi-layer RNNs that can be trained. IEEE Trans Pattern Anal Mach Intell 2021;44:4081–92.
[22] Lin W, Wu D, Boulet B. Spatial–temporal residential short-term load forecasting via graph neural networks. IEEE Trans Smart Grid 2021;12:5373–84.
[23] Liao W, Bak-Jensen B, Pillai JR, Wang Y, Wang Y. A review of graph neural networks and their applications in power systems. J Mod Power Syst Clean Energy 2021;10:345–60.
[24] Tang X, Chen H, Xiang W, Yang J, Zou M. Short-term load forecasting using channel and temporal attention based temporal convolutional network. Electr Power Syst Res 2022;205:107761.
[25] Türkoğlu A, Erkmen B, Eren Y, Erdinç O, Küçükdemiral İ. Integrated approaches in resilient hierarchical load forecasting via TCN and optimal valley filling based demand response application. Appl Energy 2024;360:122722.
[26] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez A, et al. Attention is all you need. Adv Neural Inf Process Syst 2017;30.
[27] Qu K, Si G, Shan Z, Kong X, Yang X. Short-term forecasting for multiple wind farms based on transformer model. Energy Rep 2022;8:483–90.
[28] Lim B, Arık S, Loeff N, Pfister T. Temporal fusion transformers for interpretable multi-horizon time series forecasting. Int J Forecast 2021;37:1748–64.
[29] Zhou H, Zhang S, Peng J, Zhang S, Li J, Xiong H, et al. Informer: Beyond efficient transformer for long sequence time-series forecasting. In: Proceedings of the AAAI conference on artificial intelligence. vol. 35, 2021, p. 11106–15.
[30] Kitaev N, Kaiser L, Levskaya A. Reformer: The efficient transformer. In: International conference on learning representations. 2020.
[31] Dao T, Fu D, Ermon S, Rudra A, Ré C. Flashattention: Fast and memory-efficient exact attention with io-awareness. Adv Neural Inf Process Syst 2022;35:16344–59.
[32] Huang Z, Shi X, Zhang C, Wang Q, Cheung K, Qin H, et al. FlowFormer: a transformer architecture for optical flow. ECCV; 2022.
[33] Chen Z, Liu Q, Ding Z, Liu F. Automated structural resilience evaluation based on a multi-scale transformer network using field monitoring data. Mech Syst Signal Process 2025;222:111813.
[34] Wu H, Xu J, Wang J, Long M. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. Adv Neural Inf Process Syst 2021;34:22419–30.
[35] Zhang Q, Zhou S, Xu B, Li X. TCAMS-trans: Efficient temporal-channel attention multi-scale transformer for net load forecasting. Comput Electr Eng 2024;118:109415.
[36] Wang C, Wang Y, Ding Z, Zheng T, Hu J, Zhang K. A transformer-based method of multienergy load forecasting in integrated energy system. IEEE Trans Smart Grid. 2022;13:2703–14.
[37] Ran P, Dong K, Liu X, Wang J. Short-term load forecasting based on CEEMDAN and transformer. Electr Power Syst Res 2023;214:108885.
[38] Zhao H, Wu Y, Ma L, Pan S. Spatial and temporal attention-enabled transformer network for multivariate short-term residential load forecasting. IEEE Trans Instrum Meas 2023.
[39] Zeng A, Chen M, Zhang L, Xu Q. Are transformers effective for time series forecasting? In: Proceedings of the AAAI conference on artificial intelligence. vol. 37, 2023, p. 11121–8.
[40] Liu Y, Hu T, Zhang H, Wu H, Wang S, Ma L, et al. Itransformer: Inverted transformers are effective for time series forecasting. In: The twelfth international conference on learning representations. 2024.
[41] Nie Y, Nguyen N, Sinthong P, Kalagnanam J. A time series is worth 64 words: Long-term forecasting with transformers. 2022, CoRR. abs/2211.14730.
[42] Trindade A. ElectricityLoadDiagrams20112014. UCI Machine Learning Repository; 2015, http://dx.doi.org/10.24432/C58C86.
[43] Xu D, Ruan C, Korpeoglu E, Kumar S, Achan K. Self-attention with functional time representation learning. Adv Neural Inf Process Syst 2019;32.
[44] Song Z, Cao S, Yang H. An interpretable framework for modeling global solar radiation using tree-based ensemble machine learning and Shapley additive explanations methods. Appl Energy 2024;364:123238.
[45] Scott M, Lee S-I, et al. A unified approach to interpreting model predictions. Adv Neural Inf Process Syst 2017;30:4765–74.
[46] Parimbelli E, Buonocore TM, Nicora G, Michalowski W, Wilk S, Bellazzi R. Why did AI get this one wrong?—Tree-based explanations of machine learning model predictions. Artif Intell Med 2023;135:102471.
[47] Mukhametzyanov I. Normalization of multidimensional data for multi-criteria decision making problems: Inversion, displacement, asymmetry. Springer Nature; 2023.
[48] Hong T, Wilson J, Xie J. Long term probabilistic load forecasting and normalization with hourly information. IEEE Trans Smart Grid 2013;5:456–62.
[49] Yu L, Zhou R, Chen R, Lai K. Missing data preprocessing in credit classification: One-hot encoding or imputation? Emerg Mark Financ Trade 2022;58:472–82.

[50] Bengio Y, Courville A, Vincent P. Representation learning: A review and new perspectives. IEEE Trans Pattern Anal Mach Intell 2013;35:1798–828.
[51] Liu F, Ren X, Zhang Z, Sun X, Zou Y. Rethinking skip connection with layer normalization in transformers and resnets. 2021, ArXiv Preprint arXiv:2105.07205.
[52] Zhou T, Ma Z, Wen Q, Wang X, Sun L, Jin R. Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting. In: International conference on machine learning. 2022, p. 27268–86.
[53] Zhang Y, Yan J. Crossformer: Transformer utilizing cross-dimension dependency for multivariate time series forecasting. In: The eleventh international conference on learning representations. 2023.
[54] Xu Z, Zeng A, Xu Q. FITS: Modeling time series with $10k$ parameters. In: The twelfth international conference on learning representations. 2024.
[55] Wang X, Zhou T, Wen Q, Gao J, Ding B, Jin R. CARD: Channel aligned robust blend transformer for time series forecasting. In: The twelfth international conference on learning representations. 2024.
[56] Lee C. Weather whiplash: Trends in rapid temperature changes in a warming climate. Int J Clim 2022;42:4214–22.
[57] Operator A. South Australian electricity report, 2017.
[58] Goto Y, Nagahata H, Taniguchi M, Monti A, Xu X. ANOVA with dependent errors. Springer; 2023.
[59] Ahmad A, Xiao X, Mo H, Dong D. Tuning data preprocessing techniques for improved wind speed prediction. Energy Rep 2024;11:287–303.
[60] Yin Q, Wu S, Wang L. Multiview clustering via unified and view-specific embeddings learning. IEEE Trans Neural Netw Learn Syst 2018;29:5541–53.
[61] Yao J, Cai Z, Qian Z, Yang B. A novel approach based on TCN-LSTM network for predicting waterlogging depth with waterlogging monitoring station. PLOS ONE 2023;18:e0286821.
[62] Ancona M, Oztireli C, Gross M. Explaining deep neural networks with a polynomial time algorithm for Shapley value approximation. In: International conference on machine learning. 2019, p. 272–81.
[63] Nam W, Choi J, Lee S. Interpreting deep neural networks with relative sectional propagation by analyzing comparative gradients and hostile activations. In: Proceedings of the AAAI conference on artificial intelligence. vol. 35, 2021, p. 11604–12.
[64] Zhao H, Chen H, Yang F, Liu N, Deng H, Cai H, et al. Explainability for large language models: A survey. ACM Trans Intell Syst Technol 2024;15:1–38.
[65] Tolomei G, Silvestri F. Generating actionable interpretations from ensembles of decision trees. IEEE Trans Knowl Data Eng 2019;33:1540–53.