Energy Strategy Reviews 57 (2025) 101638

Contents lists available at ScienceDirect

Energy Strategy Reviews

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

Capturing complex electricity load patterns: A hybrid deep learning approach with proposed external-convolution attention

Mohammad Sadegh Zare [a], Mohammad Reza Nikoo [b,*,1], Mingjie Chen [c], Amir H. Gandomi [d,e,**,1]

[a] Department of Computer Science, Engineering and IT, Shiraz University, Shiraz, Iran
[b] Department of Civil and Architectural Engineering, Sultan Qaboos University, Muscat, Oman
[c] Water Research Center, Sultan Qaboos University, Muscat, Oman
[d] Faculty of Engineering and IT, University of Technology Sydney, Ultimo, NSW 2007, Australia
[e] University Research and Innovation Center (EKIK), Óbuda University, 1034 Budapest, Hungary

A R T I C L E I N F O

Keywords:
Attention mechanism
Bidirectional Long Short-Term Memory
Convolutional neural network
Electricity load prediction
Short-term load forecasting
Temporal convolutional network

A B S T R A C T

Short-term electricity load forecasting is a critical factor in optimizing power systems, minimizing operating costs, and securing reliable energy resources. There are various approaches for short-term electricity load forecasting, but handling complex dependencies and sudden changes in load data remains challenging. This study introduces a hybrid deep learning model to improve load forecasting accuracy. The model combines the strengths of various deep learning architectures such as Convolutional Neural Network, Temporal Convolutional Network, and Bidirectional Long Short-Term Memory with a proposed attention mechanism. This approach helps to extract temporal relations and learn long-term patterns. Furthermore, the proposed External-Convolution Attention technique effectively captures global and temporal patterns within the input sequences. Three data sets are used to conduct experiments and validate the proposed model. The proposed model is compared against several machine learning and deep learning models across five evaluation metrics. The findings show the strength of the proposed model by outperforming other models. Our load forecasting method achieves improvements ranging from 2% to 21% across different evaluation metrics. The study also evaluates the effects of datasets, features, and prediction horizons. The presented hybrid deep learning model and a novel attention mechanism improve load forecasting accuracy, contributing to the advancement of artificial intelligence in energy optimization techniques. Our model demonstrates superior performance through extensive experimentation and diverse scenarios by identifying complex load patterns and adjusting to various datasets. This indicates its practical applicability in engineering for optimizing power systems and minimizing operational costs.

1. Introduction

Electrical energy is an indispensable part of modern society, and forecasting electricity loads plays a pivotal role in enhancing power systems, facilitating power generation, load balancing, and infrastructure planning. Accurate electricity load forecasting (LF) is also essential for efficient resource allocation, preventing overloading, and minimizing operational costs. Efficient power management is crucial as it has a direct impact on grid reliability and operational costs [1]. To meet the growing need for accurate and timely forecasts, researchers have developed various methods for short-term load forecasting (STLF), which is essential for managing daily energy operations. The broader field of STLF supports not only energy resource management but also promotes economic efficiency, informs policy planning, and contributes to environmental sustainability [2]. By providing accurate demand predictions, STLF enables utility companies to allocate resources more effectively, reduce the likelihood of power outages, and stabilize energy prices, ultimately benefiting both consumers and policy-makers [3]. Electricity LF is based on historical load data and can be divided into three groups: short-term (a few minutes, hours, or days), mid-term (a few weeks or months), and long-term (a few years). STLF has gained considerable attention in recent years because it can provide safe and smooth operation for modern power systems. Electricity load patterns

* Corresponding author at: Department of Civil and Architectural Engineering, Sultan Qaboos University, Muscat, Oman.
** Corresponding author at: Faculty of Engineering and IT, University of Technology Sydney, Ultimo, NSW 2007, Australia.
E-mail addresses: m.reza@squ.edu.om (M.R. Nikoo), gandomi@uts.edu.au (A.H. Gandomi).

[1] The two authors have the same contribution to this study.

https://doi.org/10.1016/j.esr.2025.101638
Received 3 August 2024; Received in revised form 17 December 2024; Accepted 5 January 2025
Available online 14 January 2025
2211-467X/© 2025 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

are highly complex, influenced by a range of factors such as temporal dependencies, seasonal demand variations, and external elements like weather conditions and economic activity [4]. Load data frequently show irregular spikes and dips that traditional models find difficult to capture, especially when these changes are driven by unexpected events or sudden shifts in usage [5]. While conventional statistical methods provide a solid foundation, they often struggle to accurately model the nonlinear and fluctuating patterns characteristic of electricity demand. In response, researchers are exploring new deep learning (DL) approaches to address these challenges in LF [6].

Over the years, a range of methods have been proposed to improve the precision and efficiency of STLF. These methods can be categorized into traditional and machine learning (ML) approaches [7]. Traditional techniques such as autoregressive integrated moving average (ARIMA) [8] and exponential smoothing [9] rely on established load patterns and statistical models to make predictions. These models are widely used in STLF due to their simplicity and efficiency in modeling linear dependencies. These methods usually struggle to learn the complex patterns and nonlinear dependencies in electricity load data, mainly when there are significant time variations or nonlinear load patterns [10–12]. Advancements in computer technology and ML have enabled LF models to extract nonlinear features. For instance, Olagoke et al. [13] employed an artificial neural network (ANN) with historical load and temperature data to predict short-term loads. They also used genetic algorithm (GA) to optimize their model. Jiang et al. [14] developed a hybrid prediction model by combining support vector regression (SVR) with two parameter-optimizing methods. Li et al. [15] used fast Fourier transformation to extract domain features, then a combination of an ensemble empirical mode decomposition (EEMD) model and a random forest (RF) model used to predict electricity consumption. Wang et al. [16] developed a model by using SVMD as a decomposition method and XGBoost as an ML algorithm to uncover connections between time-series and future loads. ML approaches offer greater flexibility in capturing nonlinear relationships in data, but they often fall short when it comes to effectively modeling long-term dependencies. Zhang and Jánošík [17] addressed this issue by developing a hybrid model that combines CatBoost and XGBoost algorithms, utilizing hourly electricity load and temperature data to enhance the accuracy of STLF. Their work underscores the value of hybrid models for capturing the complex dynamics in load behavior. However, these classical ML methods depend on a limited set of parameters and face constraints in fully capturing the complexity of time-series data [18,19]. Hybrid and ensemble models are gaining popularity as they enable the integration of different architectures to capture both feature-specific details and long-term dependencies [4]. However, these models also tend to increase computational demands, which can limit their use in real-time applications. Some researchers have applied innovative weighting methods to improve model accuracy. For instance, the entropy weight method has been successfully used to refine feature importance and enhance forecasting precision in both short- and long-term predictions, as demonstrated in studies on weighted grey relational analysis, decision-making techniques, and managing complex LF scenarios [20–22].

The popularity of employing DL models for time-series forecasting has grown significantly in recent years [23]. Their advanced capabilities have enabled DL models to outperform many traditional methods by handling complex dependencies in load data more effectively. These models have become popular due to their ability to automatically extract patterns from the data and capture temporal relationships and nonlinear patterns [24]. Among DL models, recurrent neural networks (RNNs) are frequently employed to predict time-series [25]. Modern RNNs networks including Gated Recurrent Unit (GRU) (e.g., [26,27]), Long Short-Term Memory network (LSTM) (e.g., [28–30]), and Bidirectional LSTM (BiLSTM) (e.g., [31]) achieved better results than traditional RNNs. Pavlatos et al. [32] implemented a BiLSTM model to improve predictive accuracy in electrical LF by capturing temporal patterns and interdependencies within the data. This approach leverages bidirectional processing, enabling the model to consider both past and future information, which led to higher accuracy compared to traditional RNN, LSTM, and GRU models. Although these models had considerable success compared to other classical LF methods, there are still various challenges associated with data processing, extracting in-depth features, and handling long-term data. Researchers have increasingly turned to hybrid models to address these challenges and improve LF accuracy. Hybrid models are designed to leverage the strengths of different DL architectures, providing a more comprehensive approach to complex pattern extraction in STLF [6]. Sadaei et al. [33] combined fuzzy time-series with Convolutional Neural Network (CNN) for STLF. A study by Sajjad et al. [34] revealed that a hybrid sequential CNN-GRU model outperformed linear regression (LR), SVR, CNN-LSTM, and BiLSTM models in terms of several evaluation metrics. Alhussein et al. [35] suggested using a DL framework called CNN-LSTM to predict short-term electricity consumption in households. In their framework, CNN layers employed to extract features and LSTM layers are used to capture patterns over time. Another popular approach is the Temporal Convolutional Network (TCN) proposed by Lea et al. [36]. This architecture effectively captures complex load patterns and long-range dependencies. For instance, Wang et al. [37] designed a model for STLF that combines TCN with Light Gradient Boosting Machine (LightGBM). In recent years, the attention mechanism (AM) has also gained popularity in time-series tasks. The famous Vaswani's Transformer model has been particularly effective in capturing dependencies using the Self-Attention Mechanism (SAM) [38]. This approach significantly enhances the ability to capture long-term dependencies and complex relationships in time-series data, a concept that has been widely adopted by researchers in recent studies. The use of hybrid models that incorporate TCN and AM has been helpful in improving the accuracy of STLF. Wu et al. [39] developed a CNN-LSTM-BiLSTM model for STLF by using an attention-based CNN to extract features from load data. The results showed that their model outperformed LSTM, BiLSTM, and CNN-LSTM models regarding lower errors. Another proposed model named TICN Att [40] combines a temporal inception convolutional network with a multi-head attention mechanism (MHAM). The inception structure is employed to extract information from input features, while the MHAM operates similarly to LSTM networks for extracting long-term dependencies. In another study [41], the ensemble-optimized BiGRU method combines BiGRU with AM and CNN, optimized through ensemble learning, to improve the accuracy and robustness of short-term photovoltaic generation forecasting. Similarly, the Seq2Seq model has been successfully applied to LF, integrating techniques like Bayesian Optimization (BO), TCN, and AM to increase forecast accuracy [42]. This model enhances feature extraction and improves sequence prediction by using CNNs and AM to focus on the most relevant data points in the sequence. Dai et al. [43] also proposed an optimized Seq2Seq model that includes BO and AM, demonstrating improved forecasting accuracy through effective data processing and feature selection. In recent years, transformer-based models have gained popularity in time-series forecasting. An alternative to the traditional transformer architecture, gMLP, demonstrates that multi-layer perceptrons (MLPs) with gating mechanisms can achieve competitive results, challenging the necessity of SAM in transformers [44]. Zerveas et al. [45] introduced a transformer-based framework tailored for multivariate time-series forecasting, emphasizing representation learning and achieving significant improvements in classification and regression tasks through unsupervised pre-training. Specifically for STLF, several transformer adaptations have shown promising results. Zhao et al. [46] developed a model that uses transformer networks for day-ahead LF, integrating LightGBM and k-means clustering for similar day selection. Another approach, the Temporal Fusion Transformer (TFT), combines RNNs with transformers to capture seasonal and time-based features [47]. Ran et al. [48] introduced a hybrid model that integrates Complete Ensemble Empirical Mode Decomposition with

Adaptive Noise (CEEMDAN) and transformers, effectively addressing long memory loss and enhancing LF accuracy across different time frames.

The primary objective of this study is to develop a hybrid DL model that improves the accuracy of STLF by addressing the complex temporal dependencies present in electricity load data. This model combines various DL architectures and includes an AM designed to capture both local and long-range dependencies. This helps overcome limitations in current approaches. A secondary goal of the study is to evaluate the model's adaptability across multiple datasets with diverse characteristics to ensure its robustness for a range of energy applications. Finally, this research aims to demonstrate that the proposed model can provide utility companies with a more reliable and accurate forecast. This supports operational efficiency and optimized resource management.

Despite the availability of numerous techniques and models for STLF, accurately capturing complex temporal patterns and dependencies remains tricky. Additionally, numerous studies in STLF have faced challenges in capturing relationships in electrical load data because of weather conditions, unforeseen events, and changes in electricity demand. The continued need for accurate STLF has led to increased research into hybrid models capable of handling these dynamic and variable factors more effectively. Our proposed model employs several DL techniques, including CNN, TCN, BiLSTM, and the AM. Each architecture has unique characteristics that enhance the model to learn various aspects of electrical loads. CNN helps the model capture local patterns and short-term relations. In the TCN, long-range dependencies are captured efficiently. BiLSTM can process information in two directions simultaneously and allow the model to consider past and future contexts at the same time. Finally, External-Convolution (ExConv) Attention was proposed as an AM that enables the model to pay attention to different segments of data and capture interactions between them effectively. Integrating multiple DL techniques and the novel ExConv Attention mechanism provides a more robust solution for learning intricate dependencies in load data, adapting effectively to diverse datasets. Several experiments were implemented using historical electric load and climate data to assess the performance of the proposed approach. It is not uncommon for models to perform well in specific contexts but fail when applied to different datasets. So, this study analyzed three sets of electricity load data with a resolution of 1 h and different statistical properties. The proposed model is compared against ML and DL models commonly used in the literature. The experimental results indicate that our method performs better in terms of accuracy and ability to capture complex load patterns than other models.

STLF is a rapidly evolving field to improve forecasting accuracy and manage the dynamic patterns in load data [4]. Researchers have developed a variety of hybrid models, combining different architectures to more effectively capture spatial, temporal, and sequential dependencies [6]. Factors such as the volume of available data, the frequency of data sampling, and regional differences further complicate the challenge of accurate forecasting, underscoring the need for adaptable models. In this study, we contribute to this field by integrating CNN, TCN, BiLSTM, and ExConv Attention to create a robust forecasting model that leverages the unique strengths of each component. This approach addresses the limitations of single-method models and aligns with recent research trends that emphasize the development of flexible forecasting frameworks. As part of the ongoing evolution of electricity LF, this study contributes the following to the field:

- Incorporates a variety of DL architectures to capture various aspects of electricity load patterns.
- Uses CNN for effectively extracting localized patterns, TCN for long-range dependencies, and BiLSTM for simultaneous processing of historical information in both forward and backward directions.
- An innovative approach called ExConv Attention is introduced, drawing inspiration from External Attention Mechanism(EAM) and Depth-wise Convolution (DWC) to enhance the attention's effectiveness.
- Combines historical electricity load data with climate data more efficiently to handle different scenarios and make more accurate predictions.
- Analyzes the robustness of the approach by comparing it with various models based on different data sets.

The remaining sections of this study are structured as follows. In Section 2, we will delve into the methodology, providing information on the DL architectures employed in our model. Moving forward to Section 3, we will discuss the dataset used for evaluation and the experimental setup employed. In Section 4, Section 5, and Section 6, we will review the results and analysis along with a discussion of our findings. Finally, in Section 7, we will summarize the findings and outline avenues for further research.

2. Methodology

In this section, we introduce a model that offers an innovative approach to STLF. Our proposed method leverages DL architectures such as CNN, TCN, BiLSTM, and a proposed AM developed for accurate LF. Each module has been carefully selected to harness its strengths in extracting dependencies, spatial patterns, and contextual information from load data. Given the widespread familiarity with CNN [49,50], LSTM [51], and BiLSTM [52] architectures in the field, we refrain from providing detailed explanations here for brevity. The subsequent subsections explain each component of our proposed method in detail before describing how they are combined.

2.1. Temporal convolutional network (TCN)

TCN utilizes neural networks to analyze time-series data [36]. It employs elements such as causal convolution (CC), dilated convolution (DC), and residual block (RB) to extract long-term temporal patterns. Detailed explanations of each component can be found in the following subsections.

2.1.1. Causal convolutions (CCs)

CCs are an essential part of TCN architecture, allowing the network to produce outputs the same length as their inputs without incorporating future information. Fig. 1(a) shows the CC's structure. For time-series input $X = (x_0, x_1, \dots, x_t)$, the output y_t at time t is just based on inputs at the present time and partial past times $(x_t, x_{t-1}, x_{t-2}, x_{t-3})$, while no future inputs are used $(x_{t+1}, x_{t+2}, \dots, x_{t+T})$. Here, X represents the input sequence of time-series data, where each element x_i corresponds to a time step in the sequence. The output y_t at each time t depends only on inputs up to x_t, ensuring that no future information is incorporated.

A CC closely maintains the chronological order of the inputs and outputs so that an output at time t is just dependent on an input up to time $t-1$ in the previous layer. This structure enables the model to gradually develop an understanding of the input sequence while avoiding the inclusion of future data in current predictions. This makes it particularly well-suited for time-sensitive applications, such as STLF.

2.1.2. Dilated convolutions (DCs)

To overcome the problem of limited field receptivity associated with CCs, TCN introduces DCs. Fig. 1(b) illustrates the structure of DCs in TCN. DCs expand the receptive field by allowing the convolutional input to cover a broader range of intervals. Increasing the dilation factor d along with the filter size k allows capturing long-range dependencies in the input time-series. As a result of this design choice, the top layers can accept a broader range of input data. The DC operation in TCN is

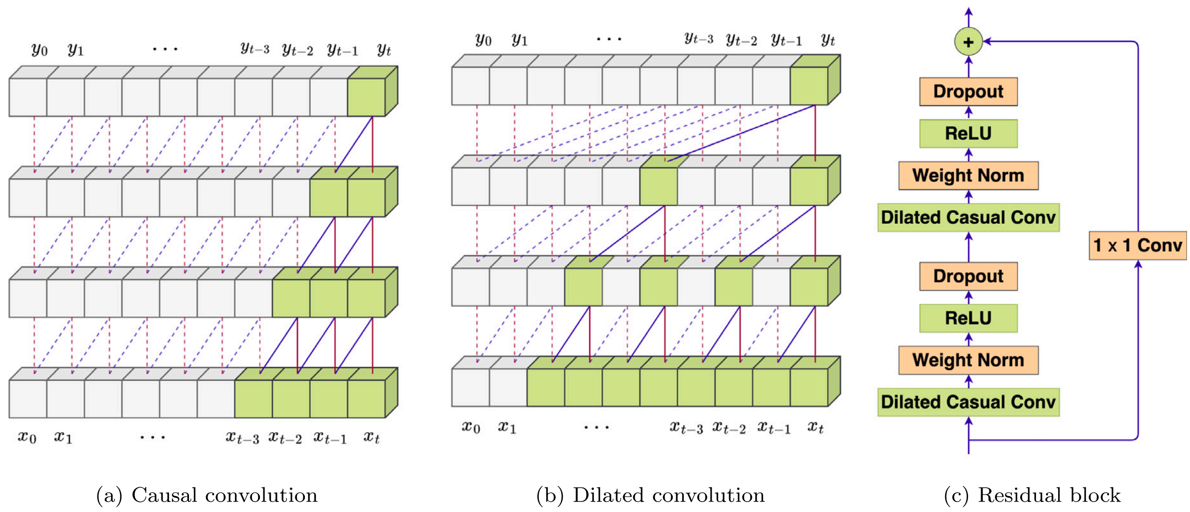

Fig. 1. Components of Temporal Convolutional Network (TCN).

as follows:

$$H_T(X,f) = \sum_{i=0}^{k-1} f(i) \cdot x_{T-d\cdot i} \tag{1}$$

where $X = (x_0, x_1, \dots, x_t, \dots, x_T)$ represents one-dimensional time-series input, $H_T(\cdot)$ denotes the DC operation on the sequence element T, f indicates the filter, d is dilation factor, k is filter size, and $T - di$ indicates the direction of the past. By adjusting d, the receptive field grows, allowing the model to capture dependencies across larger data intervals. This expanded receptive field is crucial for tasks where understanding long-term dependencies can enhance the predictive accuracy of the model.

2.1.3. Residual blocks (RBs)

RBs are employed in TCN to effectively overcome the challenges associated with training very deep networks and mitigate the problem of vanishing gradients. Fig. 1(c) illustrates the structure of revan RB. The RB consists of two branches. The first applies a transformation operation called $F(\cdot)$ on input X_{h-1}. The second one performs a 1×1 convolution operation to maintain the number of feature maps consistent. The output X_h of the hth RB is as follows:

$$X_h = \delta(F(X_{h-1}) + X_{h-1}), \tag{2}$$

where $\delta(\cdot)$ is an activation process, commonly ReLU, that introduces nonlinearity to help the network learn complex patterns and $F(\cdot)$ represents a series of transformation applied to X_{h-1}, the input to this block, and typically includes operations like CC, DC, WeightNorm normalization, and dropout. These transformations enable the residual block to extract more abstract features, with the residual connection $(+X_{h-1})$ ensuring that the original input information is preserved. The stacked RBs in TCN allow the network to extract features across a broad temporal range of input.

2.2. Attention mechanism

Attention is the fundamental part of Transformer model which is highly efficient in numerous sequence modeling tasks including time-series forecasting [38]. This approach not only enables the model to focus on different parts of the input sequence at the same time but also allows it to capture different sorts of information and enhance its representational capabilities. This idea was applied to our model, leading to the development of a method called ExConv Attention to improve the efficiency of the attention mechanism. The proposed approach draws inspiration from SAM, EAM, and DWC. These models and the proposed ExConv Attention are explained below.

2.2.1. Self-attention mechanism (SAM)

SAM is a type of AM that calculates weighted sums of values according to how a query (Q) is similar to a set of key (K) and value (V) pairs [38]. The matrices Q, K, and V are all created from the same input. In this context, Q represents the attention query, K is the key matrix used to match relevance, and V is the value matrix that is combined to create the attention output. The attention function is calculated as follows:

$$\mathrm{SAM}(Q,K,V) = \mathrm{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{3}$$

where d_k is the dimensionality of the Q and K vectors. The dot product between Q and K is scaled to prevent large values. The resulting attention weights are normalized using the softmax operation, and the weighted sum of values is obtained to represent the weight summation. The scaling factor $\sqrt{d_k}$ helps maintain numerical stability by keeping the computed values within a reasonable range, and the softmax operation normalizes these values to probabilities. Fig. 2 provides a visual representation of the SAM.

2.2.2. External attention mechanism (EAM)

SAM has been widely used for capturing relations within data samples. However, they may not adequately consider potential relationships between elements in different data samples. To address this limitation, Guo [53] introduced the EAM, a novel attention module with linear complexity. The EAM employs two fully connected networks to replace the conventional key and value matrices used in attention calculations. These networks are designed to be lightweight, learnable, and shared, effectively capturing correlations between different samples while significantly reducing computation. Building upon the AM initially presented for visual tasks, we apply the EAM to time-series sequences. The attention function of the EAM is defined as follows:

$$\mathrm{EAM}(F) = \mathrm{Norm}\left(F{M_k}^T\right)M_v \tag{4}$$

where F is the input feature map, *Norm* represents a double normalization operation, and M_k and M_v are two memory units, two linear layers without bias that replace the key and value components of traditional attention. These units capture data-level relationships across multiple samples, providing a broader context than SAM. Unlike traditional SAM that only consider relations within a single data sample, the EAM computes attention between input sequences and these two memory units. By utilizing M_k and M_v as repositories of knowledge derived from the entire training dataset, the EAM effectively models potential connections between different samples. Next, the inferred attention

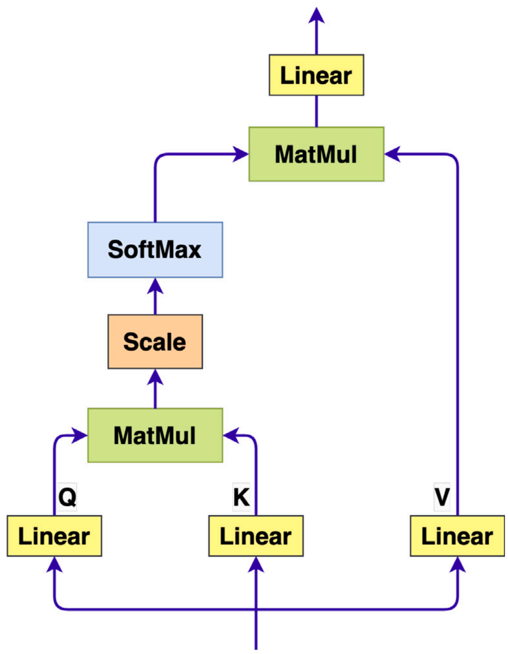

Fig. 2. Architecture of the Self-Attention Mechanism (SAM).

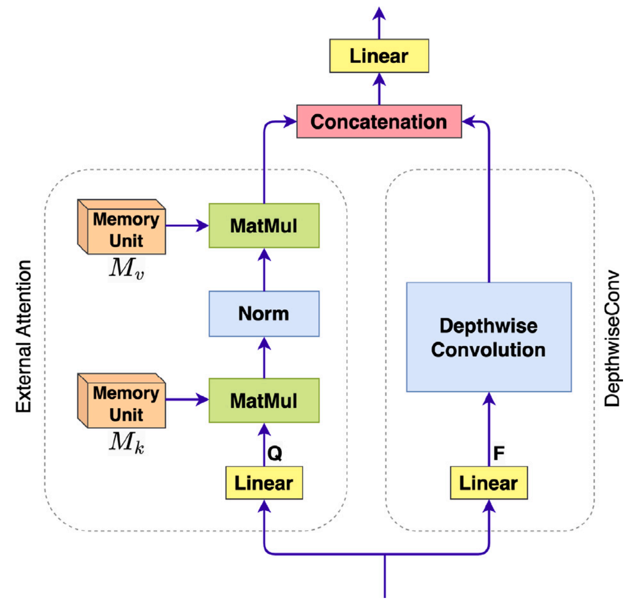

Fig. 3. Structure of the proposed External-Convolution (ExConv) Attention.

map from the prior knowledge of data-level is normalized similarly to SAM. By updating the input features using the computed similarities from M_v and the attention map, the model focuses on relevant information across the dataset while keeping things efficient. One significant difference between EAM and SAM lies in using two memory units, M_k and M_v, as key and value. This design enhances the ability of the network to learn trends and relationships in the data, resulting in improved performance. Additionally, EAM applies double normalization technique suggested in [54] which enhances representation learning and the similarity between individual sequences can be modeled better. Double normalization enhances the stability of the attention scores by normalizing the data twice, improving the model's ability to generalize to different contexts within the input sequences.

2.2.3. Depth-wise convolution (DWC)

DWC represents a type of convolutional operation frequently employed in DL models [55]. In this study, the attention model integrates convolutional operations to capture local contextual information. To learn contextual sequence representations in hidden dimensions, we have adopted a methodology that involves DWC. Point-wise projective transformation (PWPT) and contextual transformation (CT) are two primary components of DWC which makes it suitable for this task. Besides, an improved variant of DWC that dynamically adjusts the weights of contextual elements is employed [56]. This structure includes several convolution sub-modules, each containing cells with different kernel sizes. These varied kernel sizes are efficient at detecting features across different ranges. For a given input sequence X, the output $O \in R^{n\times d}$ from a DWC has learned weights $W^Q \in R^{H\times k\times d}$, where k denotes the kernel width, d represents the hidden size, and H is for weight sharing. This configuration allows DWC to apply convolutions to different input regions, capturing local context and the importance of varying sequence parts. The calculation for element i and output dimension c is as follows:

$$O_{i,c} = \mathrm{DWC}(X) = \sum_{j=1}^{k} \left(\mathrm{softmax}\left(\sum_{c=1}^{d} W^{Q}_{j,c} X_{i,c} \right) \cdot X_{i+j-\left\lceil \frac{k+1}{2} \right\rceil, c} \right) \tag{5}$$

2.2.4. ExConv attention

To enhance the AM's capabilities and effectively capture global, local, and temporal features, a new approach called ExConv Attention has been proposed. This innovative AM combines the strengths of EAM and DWC, as depicted in Fig. 3. The ExConv Attention is specifically designed to allow the model to focus on different aspects of the sequential data, particularly in STLF.

The ExConv Attention method brings together the advantages of both components to achieve better results. The DWC component plays a vital role in capturing local contextual information by employing PWPT and CT. While original convolutions do not allow for separation, DWCs can share the same PWPT with the AM because they can independently perform convolutions across every channel. This enables the model to identify and emphasize features within the input sequence effectively. By incorporating dynamic convolution, it enhances the adaptability of weights and more performance improvements in the AM. Moreover, the EAM component addresses SAM limitations by considering potential dependencies between elements of different data samples. Using two memory units and incorporating prior knowledge into its operations are the reasons behind this ability. The ExConv attention function is defined as follows:

$$\mathrm{ExConv}(X) = [\mathrm{DWC}(X); \mathrm{EAM}(X)] \tag{6}$$

where $[\cdot;\cdot]$ denotes concatenation, X is the input sequence, $DWC(X)$ and $EAM(X)$ represents the outputs of the DWC and EAM. Both DWC and EAM are applied to the input sequence. Here, X represents the full sequence of data that the model processes. At the same time, $DWC(X)$ extracts features from X based on local patterns, and $EAM(X)$ captures broader patterns by considering dependencies across the sequence. Concatenating $DWC(X)$ and $EAM(X)$ combines these two perspectives into an enriched feature representation to improve the model's ability to make accurate predictions by balancing local and global information. The ExConv Attention concatenates the output of the DWC and EAM to provide a more enriched contextualization and an improved representation of features. The local features extracted by DWC capture immediate variations within the input sequence. Meanwhile, the global and temporal dependencies identified by EAM enable the model to recognize long-term trends and relationships across samples. This combination is especially valuable for time-series tasks, where both local and long-range dependencies play a significant role in accurate predictions. So,

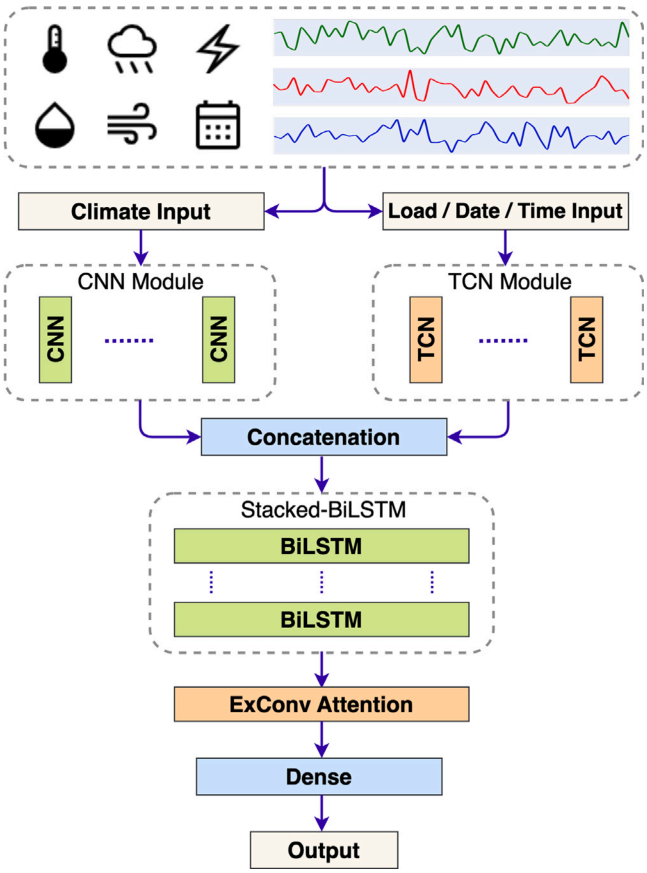

Fig. 4. The outline of the proposed model.

the proposed ExConv Attention is able to capture information at the local, global, and temporal levels of the input sequence and is suitable for tasks like STLF.

2.3. Proposed method

The proposed model includes a CNN module and a TCN module for input data. These modules are integrated with a Stacked-BiLSTM and ExConv Attention to effectively analyze and capture the dynamics and dependencies of data in the STLF task. This model aims to predict future electricity loads accurately and reliably by considering climatic and temporal aspects of electricity load data. The architecture of the proposed model is illustrated in Fig. 4.

The CNN module is specifically employed to extract relevant features from the climatic data. CNNs help this model learn linear and nonlinear dependencies in spatial data. The spatial filters in CNNs can capture local patterns in climatic features (e.g., temperature variation, wind speed, and humidity levels). These local patterns can usually influence energy consumption. The TCN module can capture the temporal dependencies and extract features from the date-time and electricity load data. As TCN previously discussed (see Section 2.1), it is effective in time-series data and capturing long-term and complex temporal patterns. Components of TCN allow the model to consider a broader range of historical information by extracting short-term fluctuations and long-term trends of electricity loads.

In the next step, the extracted features by the CNN and TCN modules concatenate. These features feed to a Stacked-BiLSTM to extract more enriched temporal dependencies and long-term patterns. Layers of BiLSTM help the model to explore dependencies in load data and learn temporal and complex relationships. Furthermore, the ExConv Attention (see Section 2.2) is introduced to combine the strengths of both EAM and DWC and augment the model's ability to identify relevant data and learn essential dependencies. EAM enables the model to focus on many segments of the input sequence at the same time, capturing a variety of interactions and patterns. DWC enhances the model to attend to relevant input segments and incorporate enriched local contextual information. The proposed ExConv Attention helps the model to learn local, global, and temporal dependencies. The integration of these components can grasp the complex interdependencies and dynamics in the data. As a result, it improves the performance, accuracy, and reliability of STLF.

The proposed model's multi-component design was chosen to meet the specific requirements of STLF. CNNs are responsible for spatial feature extraction, which is crucial for identifying localized patterns and immediate fluctuations in climate-related data. For capturing temporal dependencies, TCN provides an efficient approach to handling both short-term and long-term patterns. This makes it particularly effective for time-series data with seasonal variations. BiLSTM was selected over unidirectional RNNs due to its bidirectional functionality. It allows the model to learn from both past and future sequences simultaneously. The ExConv Attention further strengthens the model by dynamically prioritizing key segments of data. Unlike traditional AMs, ExConv combines features to emphasize critical local, global, and temporal dependencies. This makes the model highly adaptable to the complex time-series data involved in STLF.

Table 1
Statistics of the electricity load datasets in megawatt-hours (MWh).

Dataset	Load		
	Mean	Variance	Median
ADI	17.18	30.42	16.21
AKR	21.04	43.57	20.38
SNA	11.58	11.80	11.38

3. Data and experimental setup

3.1. Data

Three datasets were obtained from the Electric Distribution Company of Shiraz city in Iran, containing wholesale load records from different stations. These records, collected over three years from 2015-03-21 to 2018-03-20, are commonly used for planning and monitoring purposes. These realistic datasets are suitable for operational and planning applications in the power sector. Table 1 presents the mean, variance, and median of the electricity load values in megawatt-hours (MWh) for the three datasets: ADI, AKA, and SNA (station names are anonymized to protect the confidentiality of the data sources). All datasets have a resolution of 1 h. Different statistics and distributions of datasets can enhance the robustness of experimental analysis and provide a better understanding of the results.

As the time and load data alone may not provide sufficient information for DL models, additional features are extracted to convert from univariate to multivariate time-series forecasting. Based on the available past load and date-time information, new features are generated as shown in Table 2. In addition, hourly meteorological data, including humidity, temperature, rain index, and wind speed, are collected for the same period. The datasets used in this study capture the complex nature of electricity load patterns. Hourly load measurements show diverse statistical characteristics, such as high variance and varying seasonality. Each dataset reflects unique fluctuations in demand that correspond to different times of the day, week, and season. This challenges the model to learn and adapt to both short-term spikes and long-term trends.

A sliding window approach is adopted to prepare the data for the STLF task [39,57,58]. In this method, the non-time-series input data (tabular data) is converted to time-series features based on the

Table 2
The new extracted features from date and time.

Feature name	Ranges/Values
Season	[1:4]
Month	[1:12]
Week of year	[1:52]
Day of month	[1:31]
Day of week	[1:7]
Weekday/Holiday	[0,1]
Hour of day	[1:24]
Business hours	[0,1]

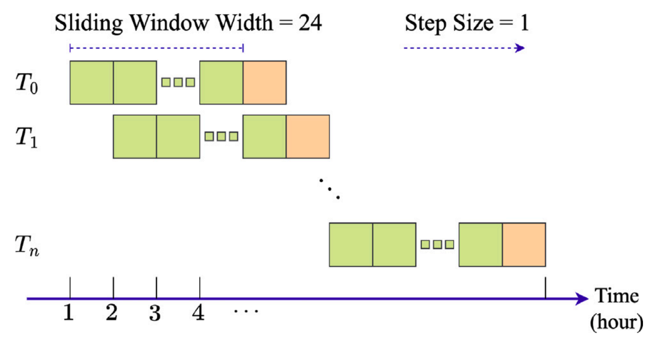

Fig. 5. The principle of the sliding window.

historical load and its associated features. A sliding window width of 24 h is chosen, and the step size is one hour, resulting in a continuous and overlapping window progression. This process ensures that the prediction model has sufficient data for learning. As shown in Fig. 5, relevant data are extracted for load prediction at every time step using the sliding window principle. By using this approach, it is possible to capture the influence of adjacent time points and select an appropriate time range for accurate LF.

The dataset is split into training set (70%), validation set (15%), and testing set (15%), a commonly used ratio in time-series forecasting that ensures balanced evaluation. Several studies in the field have similarly employed this split, supporting its effectiveness in achieving reliable results across different datasets [59,60]. As time-series should be kept in their original order and the intention is always to predict future events, no shuffling or randomization techniques are employed. The features are preprocessed using either one-hot encoding or normalization techniques. Climate features are scaled to the range of [0, 1] using min–max normalization as below:

$$x' = \frac{x - \min(X)}{\max(X) - \min(X)} \tag{7}$$

where X represents the entire feature, x denotes data before normalization, and x' is the normalized data. This process is widely used in similar studies to ensure consistent scaling of climate features.

3.2. Evaluation metrics

In this paper, R-squared (R^2), root mean squared error (RMSE), mean absolute error (MAE), mean absolute percentage error (MAPE), and median absolute percentage error (MdAPE) are chosen for evaluating models. The selection of these performance indicators was guided by their relevance and widespread use in time-series forecasting research [61–64]. Each metric provides distinct insights into model accuracy and reliability. The R^2 metric measures the proportion of variance explained by the model, with values closer to 1 indicating a better fit. RMSE and MAE evaluate errors, with RMSE placing greater emphasis on larger errors. MAPE captures relative error which is suitable for comparisons across different datasets. MdAPE reduces the impact of extreme values, which are common in load data due to sudden demand surges or drops. The formulas are listed below:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{8}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{9}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{10}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100 \tag{11}$$

$$MdAPE = \text{median}\left(\left|\frac{y_i - \hat{y}_i}{y_i}\right|\right) \times 100 \tag{12}$$

where n is the total number of test samples, y_i denotes the true load value, and $\hat{y}_i$ represents the model-forecasted load. Higher R^2 values indicate better performance and lower values of other metrics show lower errors. MdAPE is more robust than MAPE in terms of outliers.

3.3. Loss function

The Mean Squared Error (MSE) loss function is used as the optimization criterion for training models. This metric measures the squared difference between the forecasted and true values, with the goal of minimizing it during the training process to reduce the overall error. MSE loss function is used in various STLF studies, such as [27,57]. Following is the MSE loss function:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{13}$$

where n is total number of samples in train set, y_i denotes the true load value, and $\hat{y}_i$ represents the model-forecasted load value.

4. Results

A wide range of ML, DL, and transformer-based models are evaluated as the common baseline models. The ML models include SVR [65], Multi-Layer Perceptron (MLP) [66], and XGBoost [67], which are widely studied and recognized as benchmarks for STLF. The DL models include CNN [68], GRU [69], LSTM [70], BiLSTM [31], TCN [58], CNN-LSTM [71], and TCN-LSTM [72]. These diverse range of DL models can deal with sequential data in different scenarios and their effectiveness are studied in various STLF papers. These ML and DL models were selected based on their prevalence in LF literature and thorough comparative studies. There are numerous other ML and DL models, but their inclusion would exceed the scope of this study. Moreover, three other state-of-the-art DL and transformer-based models for time-series forecasting are compared with the proposed model: InceptionTime [73], gMLP [44], and TST [45]. While these three models are not specifically designed for LF, they are general methods for time-series analysis that have been applied to this problem. The following subsections present a series of experiments designed to compare the proposed method with other models, each accompanied by its respective results and discussions.

4.1. Comparing LF of various models on different datasets

LF evaluations for all models in terms of R^2, RMSE, and MAE on the testing sets of the three datasets (AKR, SNA, and ADI) are presented in Table 3. The proposed hybrid neural network model outperforms all competing models. The results reveal a considerable performance improvement over the comparing models, where the proposed model shows smaller forecasting load errors. Among the single models, TCN demonstrates the highest accuracy, while SVR and MLP models perform relatively poorly. When comparing the performance of hybrid models, it is observed that TCN exhibits superior feature extraction capabilities compared to CNN, as evidenced by the overall better performance of

Table 3
Comparative performance of models on different datasets using R^2, MAE and RMSE.

Model	AKR			SNA			ADI		
	R^2	MAE	RMSE	R^2	MAE	RMSE	R^2	MAE	RMSE
SVR	0.970	0.557	0.830	0.918	0.409	0.687	0.966	0.462	0.702
MLP	0.977	0.525	0.722	0.915	0.435	0.697	0.955	0.657	0.806
XGBoost	0.980	0.394	0.674	0.935	0.321	0.609	0.984	0.284	0.480
gMLP	0.982	0.438	0.642	0.942	0.304	0.576	0.978	0.379	0.560
IncepTime	0.987	0.361	0.544	0.940	0.309	0.589	0.984	0.317	0.485
TST	0.985	0.387	0.576	0.934	0.345	0.614	0.983	0.350	0.502
CNN	0.976	0.540	0.747	0.884	0.527	0.816	0.976	0.430	0.591
GRU	0.972	0.595	0.802	0.929	0.370	0.636	0.982	0.341	0.507
LSTM	0.984	0.413	0.603	0.936	0.322	0.603	0.983	0.317	0.494
BiLSTM	0.987	0.375	0.554	0.937	0.347	0.601	0.975	0.420	0.606
TCN	0.989	0.326	0.497	0.934	0.328	0.613	0.984	0.309	0.476
CNN-LSTM	0.987	0.349	0.553	0.933	0.360	0.619	0.982	0.325	0.514
TCN-LSTM	0.987	0.361	0.535	0.946	0.295	0.554	**0.991**	0.235	0.398
Proposed	**0.991**	**0.267**	**0.450**	**0.952**	**0.237**	**0.526**	**0.991**	**0.228**	**0.355**

Table 4
Comparison of models using MAPE and MdAPE metrics on different datasets.

Model	AKR		SNA		ADI	
	MAPE	MdAPE	MAPE	MdAPE	MAPE	MdAPE
SVR	2.836	1.929	4.339	2.580	3.391	1.900
MLP	2.695	2.164	4.549	3.355	4.689	4.060
XGBoost	2.032	1.239	3.639	2.068	2.180	1.263
gMLP	2.395	1.655	3.078	1.825	2.918	1.905
IncepTime	1.920	1.332	3.112	2.035	2.272	1.473
TST	2.083	1.443	3.250	2.173	2.648	1.780
CNN	2.936	2.273	5.304	3.502	3.150	2.334
GRU	3.124	2.412	3.961	2.639	2.602	1.629
LSTM	2.085	1.596	3.504	2.095	2.368	1.480
BiLSTM	1.986	1.473	3.566	2.429	3.241	2.201
TCN	1.755	1.290	3.093	2.129	2.466	1.435
CNN-LSTM	1.885	1.261	3.922	2.648	2.686	1.577
TCN-LSTM	1.916	1.409	3.237	2.020	2.228	1.297
Proposed	**1.429**	**0.968**	**2.446**	**1.603**	**1.865**	**1.198**

TCN-LSTM over CNN-LSTM. InceptionTime, gMLP, and TST also have comparable performance to CNN-LSTM and TCN-LSTM. Additionally, the performance of models varies across different datasets, with some models performing better on one dataset and obtaining lower results on another. This highlights the flexibility of the proposed method to adjust to different settings and datasets, which can be seen in its consistently superior performance across the AKR, SNA, and ADI datasets. A comparison of our method with other models reveals that the proposed method combines the advantages of other sub-models and achieves superior results. The proposed method achieves RMSE improvements of approximately 9%, 3%, and 13% compared to the next best models for the AKR, SNA, and ADI test sets, respectively. The respective values for MAE improvements are 17%, 17%, and 2%. These significant improvements highlight the enhanced accuracy and forecasting capabilities of the proposed method.

The proposed hybrid neural network model is also evaluated using additional metrics, MAPE and MdAPE, as summarized in Table 4. Results confirm that the proposed method is superior to all other models in terms of these metrics, demonstrating its superior forecasting accuracy. XGBoost achieves favorable MdAPE results compared to other models, which means that the model's predictions are more accurate around the median of the error distribution. However, despite XGBoost's performance, the proposed model still outperforms all models, emphasizing its exceptional forecasting capabilities. When compared to the following best models for the AKR, SNA, and ADI test sets, the proposed method achieves MAPE improvements of approximately 19%, 21%, and 15%.

Overall, the results indicate that our model has a superior forecasting capability. Additionally, the model's adaptability to different datasets demonstrates its robustness, generalization, and potential for accurate STLF applications.

4.2. Analysis of prediction curves for different models

The prediction curves of various methods at four specific dates are illustrated in Fig. 6. The load curves are plotted with a resolution of 24 h. Fig. 6(a) exhibits the load curve on 24 November 2017 in the SNA dataset, which shows relatively smaller fluctuations in the data range. Conversely, Fig. 6(d) portrays the load curve on 15 February 2018 in the ADI dataset, where higher variations in load are observed. Figs. 6(b) and 6(c) represent load curves in the AKR dataset on 31 December 2017 and 28 January 2018, respectively. These two figures illustrate higher levels of load variation within a single day compared to others.

During the forecasting process, certain individual models produce inaccurate predictions that deviate significantly from the actual load curve. However, the proposed model consistently demonstrates superior fitting performance in capturing the actual load values when compared to other models, which have limitations in capturing the underlying patterns and dynamics of the electricity load data. The proposed model is able to accurately capture load variations even when data fluctuations are significant.

In order to gain deeper insights into the performance of our method, a comparative evaluation is conducted, where it is compared with five other methods, as illustrated in Fig. 7. The evaluation period spans 240 h, specifically from 27 October 2017 to 5 November 2017, using the AKR dataset. Each figure in the grid depicts the actual load values (represented by dashed lines) and the predicted values (represented by solid lines) generated by the different models.

The proposed method demonstrates a superior fit to the actual load values, particularly at critical turning points where the power grid's load experiences a shift from decreasing to increasing or vice versa. These turning points are crucial in capturing the dynamics of the load patterns, and the proposed method effectively captures these transitions, resulting in a more accurate prediction. Moreover, the notable deviations observed in predictions made by other models underscore the superiority of our proposed method in capturing and modeling load variations in power grid data. In addition, Fig. 7(g) presents the cumulative sum of the absolute differences between predicted and actual values, illustrating prediction accuracy over time. The proposed method consistently shows lower cumulative error compared to other models. This demonstrates its ability to closely follow actual values where the level of reliability is essential for accurate forecasting. Based on the findings, it is evident that the suggested method is highly effective and demonstrates efficiency in capturing the complex patterns of STLF.

4.3. Evaluating the performance of models over different weeks

In this experiment, we evaluated the performance of different models over 20 consecutive weeks using the AKR dataset. Our objective was to assess the stability and accuracy of the models in LF across different

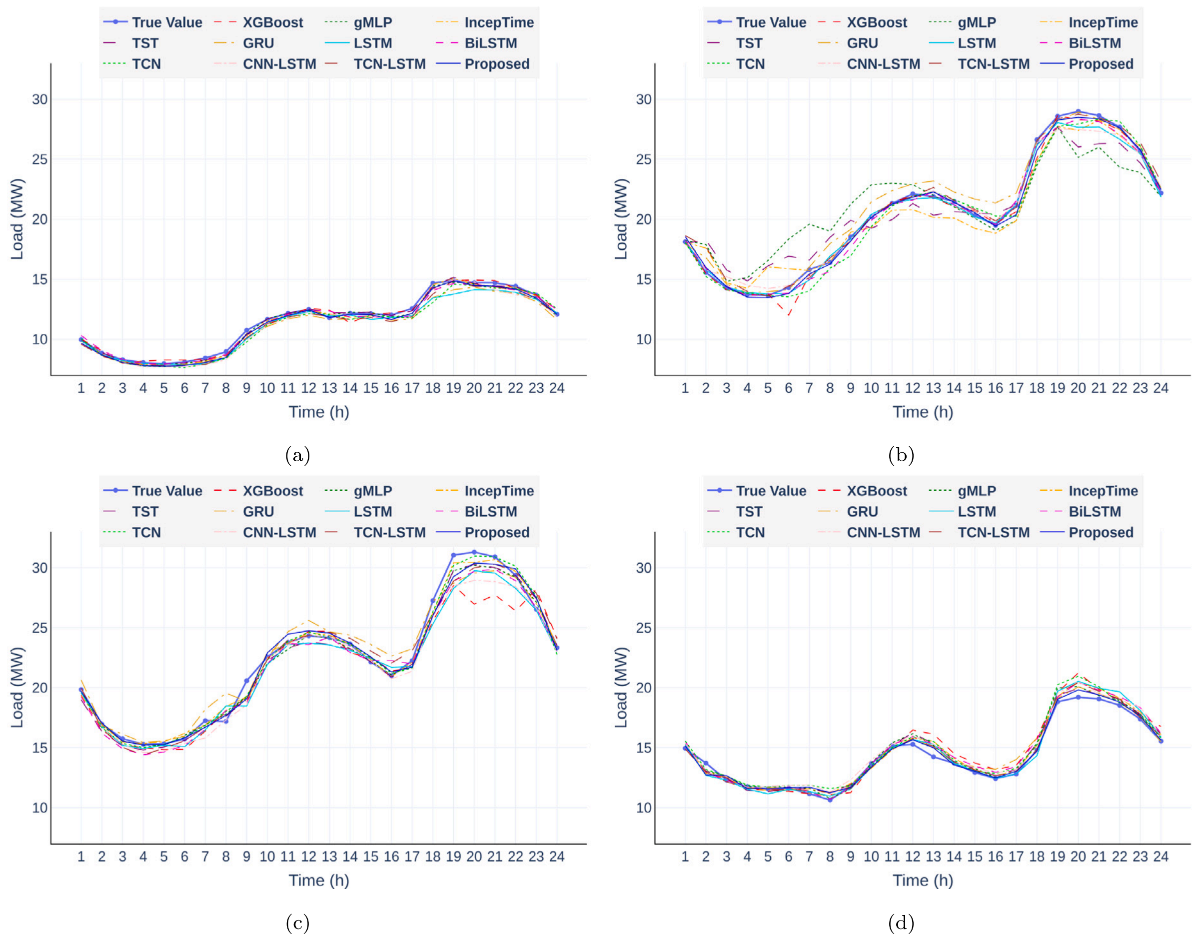

Fig. 6. Prediction curves of different models at four specific dates.

time periods. Fig. 8 displays the RMSE values obtained by the proposed model and other comparative models throughout these 20 weeks (the total RMSE for each week is calculated). Interestingly, the proposed model outperformed the comparison models in 18 out of 20 weeks. In the remaining two weeks, it performed similarly to the CNN-LSTM and TCN models.

The experiment was also conducted using MAPE, shown in Fig. 9. The proposed method clearly outperformed other models in 18 out of 20 different weeks, which shows how reliable and accurate it is in predicting load trends. Despite the fact that the proposed method performed less well in two cases than CNN-LSTM, gMLP, and TST, the results are still comparable. The outcomes demonstrate the strength of our suggested model in forecasting loads over a specific timeframe and validate its potential for real-world applications.

Table 5
Cross-dataset results as the model trained on AKR and evaluated on ADI test set.

Model	R^2	MAE	RMSE	MAPE	MdAPE
SVR	0.929	0.825	1.019	6.200	5.083
MLP	0.916	0.875	1.104	6.040	5.291
XGBoost	0.958	0.554	0.786	4.113	2.777
gMLP	0.912	0.919	1.133	6.891	5.653
IncepTime	0.949	0.665	0.864	4.734	3.729
TST	0.923	0.857	1.056	6.287	5.041
CNN	0.887	1.004	1.281	7.509	5.735
GRU	0.942	0.716	0.916	5.448	4.094
LSTM	0.965	0.533	0.719	4.296	3.046
BiLSTM	0.931	0.822	1.005	6.170	5.337
TCN	0.967	0.554	0.697	4.233	3.348
CNN-LSTM	0.930	0.853	1.007	6.720	5.425
TCN-LSTM	0.960	0.565	0.764	4.233	3.062
Proposed	**0.978**	**0.443**	**0.567**	**3.450**	**2.675**

4.4. *Cross-dataset testing for evaluating generalization performance*

The goal of this cross-dataset experiment is to examine the performance of the trained models in terms of generalization. The objective is also to examine how well the models could adapt to unseen data from different locations. So, the models were trained on the AKR train set and then evaluated on the ADI test set. Table 5 presents the LF evaluation metrics for the models on the ADI dataset.

Our proposed model achieved the best results in this experiment, which indicates its robustness and ability to generalize well. This experiment shows that the proposed method provides more precise predictions even on unfamiliar data or regions.

4.5. *Extended forecasting horizons*

The aim of this experiment is to evaluate models by extending the forecasting horizon beyond the next hour. Instead of predicting the load for the next hour, all models are assessed for predicting the load for the 5th and 24th next hours. AKR dataset is used to generate input–output pairs for training and testing using the sliding window approach. Tables 6 and 7 provide evaluation metrics for our model and other comparison models over extended forecasting horizons. Our model successfully outperforms others in terms of MAE, RMSE, MAPE, MdAPE, and R^2 for both the next 5th and 24th hour predictions. These results demonstrate the effectiveness and flexibility of the proposed model in forecasting extended horizons of load data.

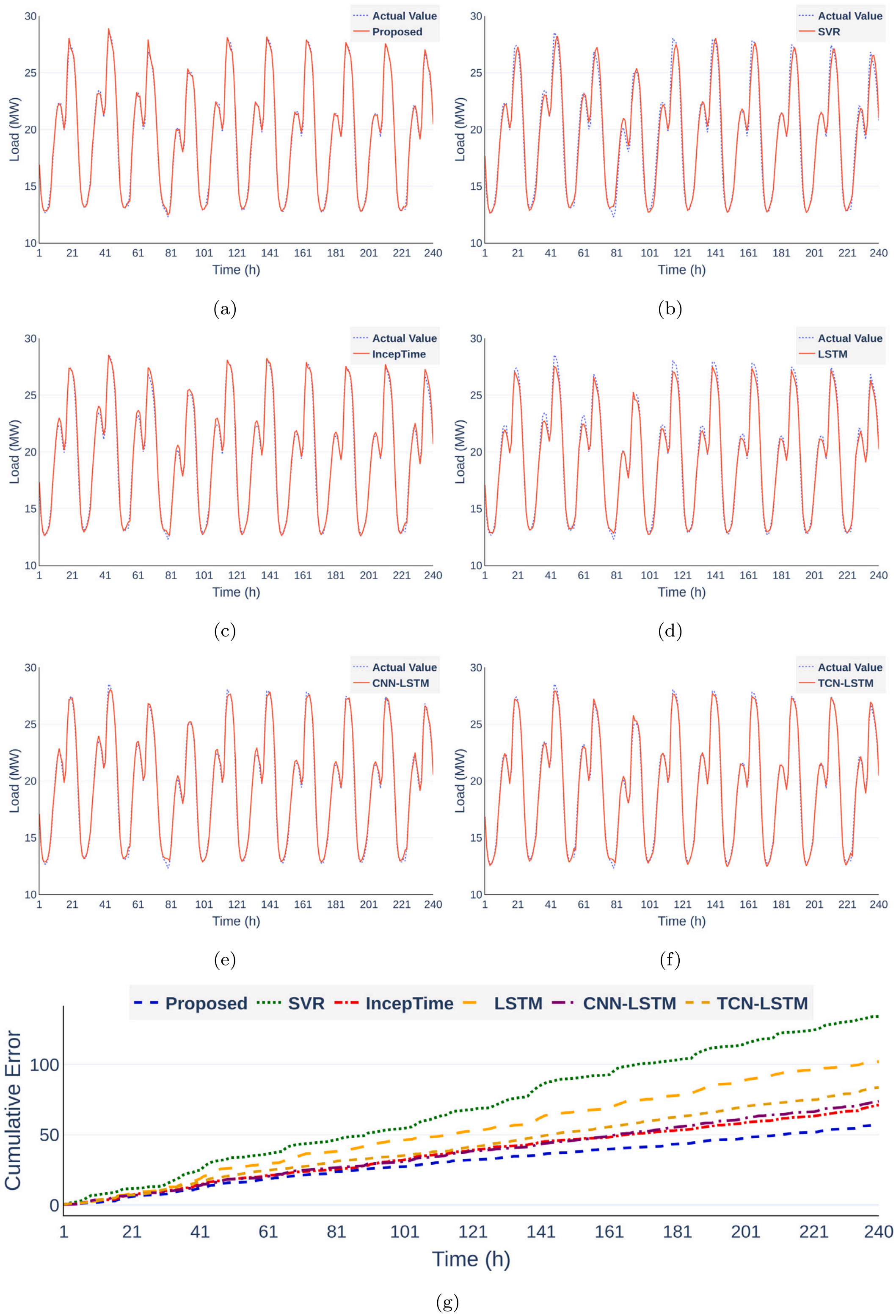

Fig. 7. Comparison of predicted values versus actual values across 6 models over a 240-h period.

4.6. Evaluating days and hours

The amount of electricity consumption can undergo different changes on different days of the week and at different hours according to the activities of users and companies. The error rates of models during the week and on weekends are worth examining. Fig. 10 compares the RMSE for this case on the ADI dataset. The first finding that can be seen is that the proposed method performed better than others. On

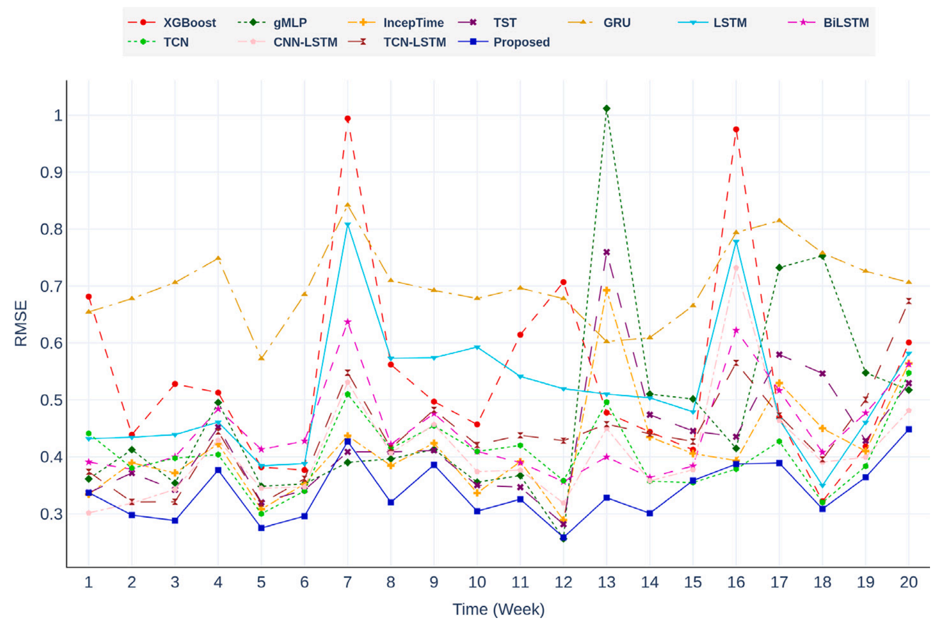

Fig. 8. RMSE comparison per week for different models using AKR dataset.

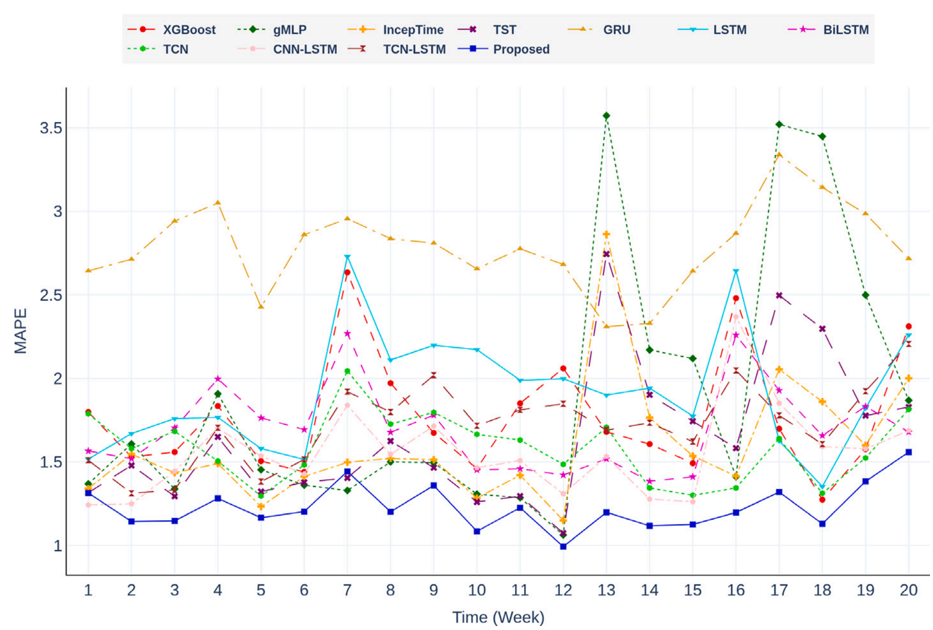

Fig. 9. MAPE comparison per week for different models using AKR dataset.

the other hand, it is not possible to find a precise relationship between days and the error rate. For example, some models have more errors on weekdays and others on weekends. Additionally, it is important to examine the error rates at different hours. Fig. 11 shows the average prediction error of the proposed method in 24 h for all three datasets. It can be seen that, for example, at midnight or during working hours, the error rate is lower, and this is because electricity use follows a certain pattern. It can be seen that the error rates between 17:00 and 20:00 on all three datasets is higher than in other hours. These hours fall outside of working hours and, depending on electricity consumption behavior, can exhibit varying fluctuations; hence, the error rate during these times is higher.

5. Discussion

In the results section, the proposed method was discussed and analyzed in comparison with other methods and different scenarios. However, there are still more aspects that will be discussed in more detail. The experimental studies of this paper show that the proposed method clearly has a more impressive and better performance than other methods. Different data and various experiments that were done concerning STLF can prove the validity of this conclusion. The findings reveal that our hybrid model does a great job of capturing both short- and long-term patterns, which aligns perfectly with our goal of improving forecasting accuracy across diverse datasets. This

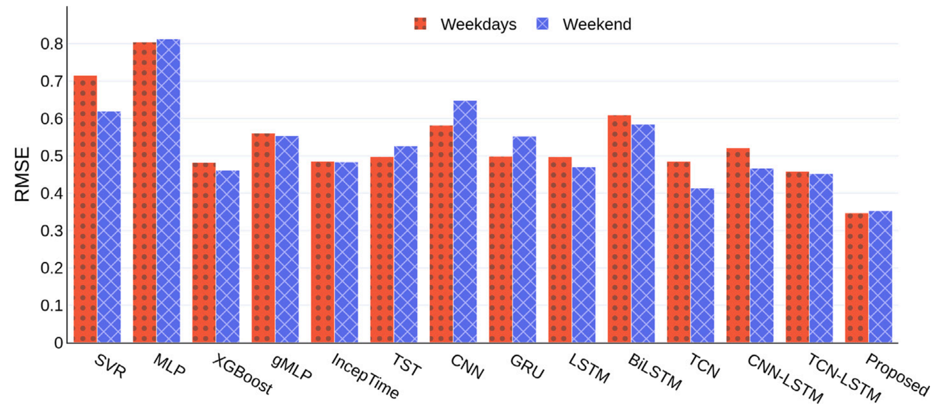

Fig. 10. Comparing RMSE values of weekdays and weekend.

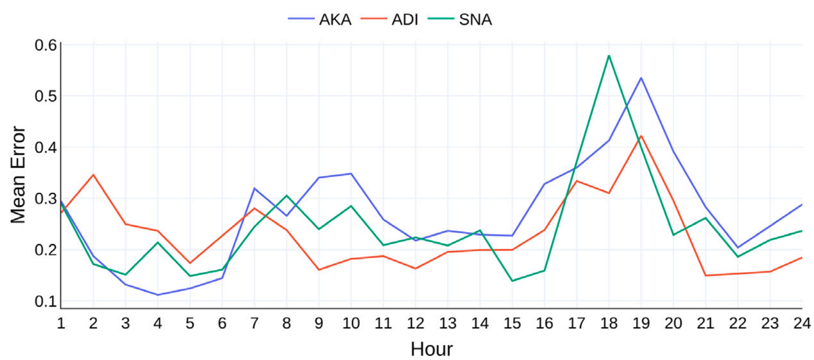

Fig. 11. Comparing error in different hours of day for different sets of data.

Table 6
Performance comparison of LF models for 5 h extended horizon.

Model	R^2	MAE	RMSE	MAPE	MdAPE
SVR	0.939	0.846	1.186	4.458	3.114
MLP	0.889	1.294	1.594	6.825	5.833
XGBoost	0.959	0.607	0.966	3.215	2.067
gMLP	0.942	0.810	1.151	4.451	3.292
IncepTime	0.953	0.714	1.041	3.865	2.729
TST	0.945	0.777	1.117	4.215	3.003
CNN	0.946	0.795	1.115	4.325	3.065
GRU	0.941	0.817	1.163	4.457	3.147
LSTM	0.930	0.866	1.264	4.793	3.379
BiLSTM	0.948	0.754	1.086	4.137	2.905
TCN	0.956	0.677	1.008	3.653	2.430
CNN-LSTM	0.937	0.846	1.199	4.490	3.275
TCN-LSTM	0.948	0.734	1.086	3.944	2.676
Proposed	**0.966**	**0.561**	**0.887**	**2.992**	**2.005**

Table 7
Performance comparison of LF models for 24 h extended horizon.

Model	R^2	MAE	RMSE	MAPE	MdAPE
SVR	0.918	0.948	1.368	4.925	3.330
MLP	0.925	0.919	1.309	4.897	3.607
XGBoost	0.936	0.777	1.211	4.078	2.608
gMLP	0.901	1.027	1.505	5.537	3.801
IncepTime	0.903	0.965	1.487	5.176	3.261
TST	0.911	0.936	1.428	5.039	3.376
CNN	0.885	1.173	1.624	5.907	4.624
GRU	0.917	0.942	1.375	5.105	3.678
LSTM	0.922	0.888	1.331	4.812	3.201
BiLSTM	0.908	0.981	1.448	5.388	3.772
TCN	0.918	0.989	1.373	5.440	3.991
CNN-LSTM	0.911	0.946	1.423	5.076	3.514
TCN-LSTM	0.925	0.871	1.311	4.734	3.211
Proposed	**0.956**	**0.636**	**1.016**	**3.900**	**2.503**

supports conclusions found in recent research, including ensemble models [17,61], as well as more recent hybrid DL models that showed strong improvements in STLF accuracy [42,72,74]. Our model's unique structure combines CNN, TCN, and BiLSTM, which helps it capture both local and sequential dependencies more effectively. This addresses some of the limitations seen in purely ML-based models, as pointed out in studies by Olagoke et al. [13] and Jiang et al. [14].

As expected, the proposed method has a very impressive performance compared to ML methods and performs much better than them. For example, in the AKA dataset, the XGBoost method has performed the best among the ML methods, while the proposed method has achieved better results by 32%, 33%, and 30% based on MAE, RMSE, and MAPE, respectively. This can be seen in relation to other ML methods and different data. These results are consistent with the results of other articles that have presented different hybrid DL methods and compared them with ML methods [57,75,76]. CNN-LSTM and TCN-LSTM hybrid methods have also performed better than ML methods. These methods have achieved better accuracy than individual DL methods in most cases. For example, in the ADI dataset, the TCN-LSTM method outperforms the best individual DL methods, LSTM and TCN, in terms of MAE by 26% and 24%, respectively. These results are consistent with other articles in this field, suggesting that the findings are valid [57,58]. The ExConv Attention combines depth-wise convolution with memory attention. This significantly enhances the model's ability to focus on the most relevant features in a data sequence. This aligns with findings from Wu et al. [39], who demonstrated that attention-based models like CNN and BiLSTM can reduce prediction errors. With this mechanism, the model is better at capturing complex interactions within load and climate data. This addresses a challenge that has often hindered previous models due to noise in the data and sudden shifts in patterns. Another part of the architecture of this article is the use of the

AM, the advantages of which have been mentioned in different fields and articles. For example, in the field of STLF, several articles such as [39,40] have obtained better results by using this mechanism than the basic methods, which is consistent with the approach and results of this article.

The structure of hybrid DL methods is more complicated than basic DL and ML methods, and therefore more parameters and hyperparameters should be tuned for them. Therefore, their training time is much longer and this also applies to the proposed method in this article. However, the complexity of the hybrid architecture does present certain trade-offs, particularly in terms of computational cost during training. This limitation is also noted in related studies on hybrid models [77,78]. Fine-tuning the model can be time-intensive, which complicates real-time applications, although the inference phase remains efficient. Moving forward, it will be necessary to balance model complexity with operational efficiency. It is also worth noting that, while the model performed well on the datasets tested, further validation is required to ensure consistent performance across different regions and data types. However, hybrid DL methods are prevalent, and more models are always presented in different articles as these methods often yield much better results. The reason is that when these models are trained, their weights and hyperparameters can be saved and used for future predictions without the need to retrain them. Updating the model can be done at different intervals, for example, in each season, to learn seasonal patterns and trends. However, developing new methods with more accuracy and a more straightforward network structure is always valuable. Research has shown that while hybrid DL models demand greater computational resources during training, they often deliver higher accuracy. This trade-off is acceptable in applications where precise forecasts are essential for operational efficiency [63,74,77,78]. For online applications, this approach remains viable because the inference phase used for live predictions is fast and efficient. It allows real-time forecasting without the need for constant retraining. This efficiency ensures the model's suitability for practical deployment in situations where timely and accurate predictions are critical.

6. Managerial and societal implications

The enhanced accuracy of the proposed model provides strategic and economic benefits for utility companies, policy-makers, and consumers. For power companies, more accurate LF enables better demand management and reduces the risk of outages. It also allows for optimized resource allocation, which can help lower costs. Accurate forecasting supports stable pricing and more efficient infrastructure planning, ultimately contributing to reduced operational expenses. With fewer backup systems required, companies can reduce emissions, thereby supporting sustainability goals. For policy-makers, improved forecasting informs long-term infrastructure planning and enhances grid stability. These factors are essential for modernizing energy grids to make them smarter and more adaptive. Consumers also benefit from stable energy prices, fewer disruptions, and improved service reliability, which ultimately supports broader societal well-being.

7. Conclusion

This research paper presents a DL model designed for predicting short term electricity loads. The method proposed in this study combines various techniques such as CNN, TCN, and Stacked-BiLSTM, and introduces a novel AM called ExConv Attention. This AM aims to enhance the model's ability to focus on specific segments of load data and effectively detect their interactions. The proposed approach demonstrates efficient processing of both climatic and load data, capturing spatial and temporal dependencies, resulting in accurate load forecasts. Comparative experiments against ML, DL, and transformer-based models reveal that the proposed method outperforms other models in terms of metrics like R^2, MAE, RMSE, MAPE, and MdAPE. Our proposed model demonstrated average improvements across key metrics on multiple datasets, achieving an 8% reduction in RMSE, 12% in MAE, and 18% in MAPE compared to existing models. These results highlight the model's effectiveness under diverse conditions. The statistically significant improvements across various datasets confirm the robustness and applicability of the model for practical STLF applications.

Evaluating its performance on three distinct datasets confirms its robustness across different data sources, suggesting its suitability for STLF in various scenarios and geographical regions. By using datasets that reflect real-world power distribution scenarios, this model demonstrates its potential for diverse energy management applications.

Despite its strong forecasting accuracy, the proposed model has certain limitations. The hybrid architecture involves higher computational costs during training, which may restrict its deployment in environments with limited resources. Moreover, the model's performance relies on large, high-quality datasets, posing challenges in regions where data availability is limited. The model's effectiveness may vary when applied to datasets with significantly different statistical characteristics. This underscores the need for further testing across a wider range of data types to enhance its robustness. We acknowledge that the findings of this study are specific to the datasets used and may require additional validation before application in other regions. Nonetheless, the model's consistent performance across multiple datasets indicates a strong potential for broader applicability.

For future studies, exploring ensemble learning techniques, incorporating external factors such as economic indicators, and integrating transfer learning methods could potentially enhance forecasting accuracy. Additional research could also focus on integrating real-time parameter estimation techniques, as described in [79], to enhance forecasting accuracy in real-time applications. Such adaptive methods may yield significant advancements in managing rapidly changing conditions within distribution grids. In addition, further exploration of advanced DL architectures and improving model interpretability can provide valuable insights for informed decision-making regarding energy systems. Future research could also benefit from optimizing the proposed hybrid model to reduce computational demands. This would make it more feasible for real-time applications. Enhancing the model's interpretability through explainable AI techniques would provide stakeholders with a clearer understanding of load patterns, thereby supporting more informed decision-making in energy management.

CRediT authorship contribution statement

Mohammad Sadegh Zare: Writing – original draft, Visualization, Software, Investigation, Formal analysis, Methodology, Data curation. **Mohammad Reza Nikoo:** Writing – review & editing, Validation, Supervision, Resources, Project administration, Methodology, Conceptualization. **Mingjie Chen:** Writing – review & editing, Validation, Resources. **Amir H. Gandomi:** Writing – review & editing, Validation, Supervision, Resources, Project administration, Methodology, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

[1] S. Khan, Short-term electricity load forecasting using a new intelligence-based application, Sustainability 15 (16) (2023) 12311, http://dx.doi.org/10.3390/su151612311.

[2] B. Magalhães, P. Bento, J. Pombo, M.d.R. Calado, S. Mariano, Short-term load forecasting based on optimized random forest and optimal feature selection, Energies 17 (8) (2024) 1926, http://dx.doi.org/10.3390/en17081926.

[3] T. Bashir, C. Haoyong, M.F. Tahir, Z. Liqiang, Short term electricity load forecasting using hybrid prophet-LSTM model optimized by BPNN, Energy Rep. 8 (2022) 1678–1686, http://dx.doi.org/10.1016/j.egyr.2021.12.067.

[4] S. Akhtar, S. Shahzad, A. Zaheer, H.S. Ullah, H. Kilic, R. Gono, M. Jasiński, Z. Leonowicz, Short-term load forecasting models: A review of challenges, progress, and the road ahead, Energies 16 (10) (2023) 4060, http://dx.doi.org/10.3390/en16104060.

[5] I.K. Nti, M. Teimeh, O. Nyarko-Boateng, A.F. Adekoya, Electricity load forecasting: a systematic review, J. Electr. Syst. Inf. Technol. 7 (2020) 1–19, http://dx.doi.org/10.1186/s43067-020-00021-8.

[6] Y. Eren, İ. Küçükdemiral, A comprehensive review on deep learning approaches for short-term load forecasting, Renew. Sustain. Energy Rev. 189 (2024) 114031, http://dx.doi.org/10.1016/j.rser.2023.114031.

[7] N. Ahmad, Y. Ghadi, M. Adnan, M. Ali, Load forecasting techniques for power system: Research challenges and survey, IEEE Access 10 (2022) 71054–71090, http://dx.doi.org/10.1109/ACCESS.2022.3187839.

[8] C. Tarmanini, N. Sarma, C. Gezegin, O. Ozgonenel, Short term load forecasting based on ARIMA and ANN approaches, Energy Rep. 9 (2023) 550–557, http://dx.doi.org/10.1016/j.egyr.2023.01.060.

[9] J.W. Taylor, Short-term electricity demand forecasting using double seasonal exponential smoothing, J. Oper. Res. Soc. 54 (8) (2003) 799–805, http://dx.doi.org/10.1057/palgrave.jors.2601589.

[10] G. Dudek, Pattern-based local linear regression models for short-term load forecasting, Electr. Power Syst. Res. 130 (2016) 139–147, http://dx.doi.org/10.1016/j.epsr.2015.09.001.

[11] M. Khodayar, J. Wang, Spatio-temporal graph deep neural network for short-term wind speed forecasting, IEEE Trans. Sustain. Energy 10 (2) (2018) 670–681, http://dx.doi.org/10.1109/TSTE.2018.2844102.

[12] J. Li, D. Deng, J. Zhao, D. Cai, W. Hu, M. Zhang, Q. Huang, A novel hybrid short-term load forecasting method of smart grid using MLR and LSTM neural network, IEEE Trans. Ind. Inform. 17 (4) (2020) 2443–2452, http://dx.doi.org/10.1109/TII.2020.3000184.

[13] M.D. Olagoke, A. Ayeni, M.A. Hambali, Short term electric load forecasting using neural network and genetic algorithm, Int. J. Appl. Inf. Syst. 10 (4) (2016) 22–28, http://dx.doi.org/10.5120/ijais2016451490.

[14] H. Jiang, Y. Zhang, E. Muljadi, J.J. Zhang, D.W. Gao, A short-term and high-resolution distribution system load forecasting approach using support vector regression with hybrid parameters optimization, IEEE Trans. Smart Grid 9 (4) (2016) 3341–3350, http://dx.doi.org/10.1109/TSG.2016.2628061.

[15] C. Li, Y. Tao, W. Ao, S. Yang, Y. Bai, Improving forecasting accuracy of daily enterprise electricity consumption using a random forest based on ensemble empirical mode decomposition, Energy 165 (2018) 1220–1227, http://dx.doi.org/10.1016/j.energy.2018.10.113.

[16] Y. Wang, S. Sun, X. Chen, X. Zeng, Y. Kong, J. Chen, Y. Guo, T. Wang, Short-term load forecasting of industrial customers based on SVMD and XGBoost, Int. J. Electr. Power Energy Syst. 129 (2021) 106830, http://dx.doi.org/10.1016/j.ijepes.2021.106830.

[17] L. Zhang, D. Jánošík, Enhanced short-term load forecasting with hybrid machine learning models: CatBoost and XGBoost approaches, Expert Syst. Appl. 241 (2024) 122686, http://dx.doi.org/10.1016/j.eswa.2023.122686.

[18] K. Zor, O. Timur, A. Teke, A state-of-the-art review of artificial intelligence techniques for short-term electric load forecasting, in: 2017 6th International Youth Conference on Energy, IYCE, IEEE, 2017, pp. 1–7, http://dx.doi.org/10.1109/IYCE.2017.8003734.

[19] M. Afrasiabi, M. Mohammadi, M. Rastegar, L. Stankovic, S. Afrasiabi, M. Khazaei, Deep-based conditional probability density function forecasting of residential loads, IEEE Trans. Smart Grid 11 (4) (2020) 3646–3657, http://dx.doi.org/10.1109/TSG.2020.2972513.

[20] L. An, Y. Teng, C. Li, Q. Dan, Combined grey model based on entropy weight method for long-term load forecasting, in: 2020 5th Asia Conference on Power and Electrical Engineering, ACPEE, IEEE, 2020, pp. 149–153, http://dx.doi.org/10.1109/ACPEE48638.2020.9136269.

[21] L. Ruxue, L. Shumin, Y. Miaona, L. Jican, Load forecasting based on weighted grey relational degree and improved ABC-svm, J. Electr. Eng. Technol. 16 (2021) 2191–2200, http://dx.doi.org/10.1007/s42835-021-00727-3.

[22] P. Gao, Y. Yang, F. Li, J. Ge, Q. Yin, R. Wang, Research on integrated decision making of multiple load combination forecasting for integrated energy system, Energy (2024) 133390, http://dx.doi.org/10.1016/j.energy.2024.133390.

[23] N.M.M. Bendaoud, N. Farah, Using deep learning for short-term load forecasting, Neural Comput. Appl. 32 (2020) 15029–15041, http://dx.doi.org/10.1007/s00521-020-04856-0.

[24] A. Almalaq, G. Edwards, A review of deep learning methods applied on load forecasting, in: 2017 16th IEEE International Conference on Machine Learning and Applications, ICMLA, IEEE, 2017, pp. 511–516, http://dx.doi.org/10.1109/ICMLA.2017.0-110.

[25] A. Gasparin, S. Lukovic, C. Alippi, Deep learning for time series forecasting: The electric load case, CAAI Trans. Intell. Technol. 7 (1) (2022) 1–25, http://dx.doi.org/10.1049/cit2.12060.

[26] W. Wu, W. Liao, J. Miao, G. Du, Using gated recurrent unit network to forecast short-term load considering impact of electricity price, Energy Procedia 158 (2019) 3369–3374, http://dx.doi.org/10.1016/j.egypro.2019.01.950.

[27] Y. Tian, S. Zhou, M. Wen, J. Li, A short-term electricity forecasting scheme based on combined GRU model with STL decomposition, in: IOP Conference Series: Earth and Environmental Science, Vol. 701, IOP Publishing, 2021, 012008, http://dx.doi.org/10.1088/1755-1315/701/1/012008.

[28] W. Kong, Z.Y. Dong, Y. Jia, D.J. Hill, Y. Xu, Y. Zhang, Short-term residential load forecasting based on LSTM recurrent neural network, IEEE Trans. Smart Grid 10 (1) (2017) 841–851, http://dx.doi.org/10.1109/tsg.2017.2753802.

[29] S. Muzaffar, A. Afshari, Short-term load forecasts using LSTM networks, Energy Procedia 158 (2019) 2922–2927, http://dx.doi.org/10.1016/j.egypro.2019.01.952.

[30] M.S. Hossain, H. Mahmood, Short-term load forecasting using an LSTM neural network, in: 2020 IEEE Power and Energy Conference At Illinois, PECI, IEEE, 2020, pp. 1–6, http://dx.doi.org/10.1109/PECI48348.2020.9064654.

[31] J. Du, Y. Cheng, Q. Zhou, J. Zhang, X. Zhang, G. Li, Power load forecasting using bilstm-attention, in: IOP Conference Series: Earth and Environmental Science, Vol. 440, IOP Publishing, 2020, 032115, http://dx.doi.org/10.1088/1755-1315/440/3/032115.

[32] C. Pavlatos, E. Makris, G. Fotis, V. Vita, V. Mladenov, Enhancing electrical load prediction using a bidirectional LSTM neural network, Electronics 12 (22) (2023) 4652, http://dx.doi.org/10.3390/electronics12224652.

[33] H.J. Sadaei, P.C.d. e Silva, F.G. Guimaraes, M.H. Lee, Short-term load forecasting by using a combined method of convolutional neural networks and fuzzy time series, Energy 175 (2019) 365–377, http://dx.doi.org/10.1016/j.energy.2019.03.081.

[34] M. Sajjad, Z.A. Khan, A. Ullah, T. Hussain, W. Ullah, M.Y. Lee, S.W. Baik, A novel CNN-GRU-based hybrid approach for short-term residential load forecasting, IEEE Access 8 (2020) 143759–143768, http://dx.doi.org/10.1109/ACCESS.2020.3009537.

[35] M. Alhussein, K. Aurangzeb, S.I. Haider, Hybrid CNN-LSTM model for short-term individual household load forecasting, IEEE Access 8 (2020) 180544–180557, http://dx.doi.org/10.1109/access.2020.3028281.

[36] C. Lea, R. Vidal, A. Reiter, G.D. Hager, Temporal convolutional networks: A unified approach to action segmentation, in: Computer Vision–ECCV 2016 Workshops: Amsterdam, the Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part III 14, Springer, 2016, pp. 47–54, http://dx.doi.org/10.1007/978-3-319-49409-8_7.

[37] Y. Wang, J. Chen, X. Chen, X. Zeng, Y. Kong, S. Sun, Y. Guo, Y. Liu, Short-term load forecasting for industrial customers based on TCN-LightGBM, IEEE Trans. Power Syst. 36 (3) (2020) 1984–1997, http://dx.doi.org/10.1109/TPWRS.2020.3028133.

[38] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L.u. Kaiser, I. Polosukhin, Attention is all you need, in: Advances in Neural Information Processing Systems, Vol. 30, Curran Associates, Inc., 2017, pp. 5998–6008.

[39] K. Wu, J. Wu, L. Feng, B. Yang, R. Liang, S. Yang, R. Zhao, An attention-based CNN-LSTM-BiLSTM model for short-term electric load forecasting in integrated energy system, Int. Trans. Electr. Energy Syst. 31 (1) (2021) e12637, http://dx.doi.org/10.1002/2050-7038.12637.

[40] C. Tong, L. Zhang, H. Li, Y. Ding, Temporal inception convolutional network based on multi-head attention for ultra-short-term load forecasting, IET Gener. Transm. Distrib. 16 (8) (2022) 1680–1696, http://dx.doi.org/10.1049/gtd2.12394.

[41] Y. Dai, W. Yu, M. Leng, A hybrid ensemble optimized BiGRU method for short-term photovoltaic generation forecasting, Energy 299 (2024) 131458, http://dx.doi.org/10.1016/j.energy.2024.131458.

[42] Y. Dai, W. Yu, Short-term power load forecasting based on Seq2Seq model integrating Bayesian optimization, temporal convolutional network and attention, Appl. Soft Comput. 166 (2024) 112248, http://dx.doi.org/10.1016/j.asoc.2024.112248.

[43] Y. Dai, X. Yang, M. Leng, Optimized Seq2Seq model based on multiple methods for short-term power load forecasting, Appl. Soft Comput. 142 (2023) 110335, http://dx.doi.org/10.1016/j.asoc.2023.110335.

[44] H. Liu, Z. Dai, D. So, Q.V. Le, Pay attention to mlps, Adv. Neural Inf. Process. Syst. 34 (2021) 9204–9215, http://dx.doi.org/10.48550/arXiv.2105.08050.

[45] G. Zerveas, S. Jayaraman, D. Patel, A. Bhamidipaty, C. Eickhoff, A transformer-based framework for multivariate time series representation learning, in: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, 2021, pp. 2114–2124, http://dx.doi.org/10.1145/3447548.3467401.

[46] Z. Zhao, C. Xia, L. Chi, X. Chang, W. Li, T. Yang, A.Y. Zomaya, Short-term load forecasting based on the transformer model, Information 12 (12) (2021) 516, http://dx.doi.org/10.3390/info12120516.

[47] P.C. Huy, N.Q. Minh, N.D. Tien, T.T.Q. Anh, Short-term electricity load forecasting based on temporal fusion transformer model, IEEE Access 10 (2022) 106296–106304, http://dx.doi.org/10.1109/ACCESS.2022.3211941.

[48] P. Ran, K. Dong, X. Liu, J. Wang, Short-term load forecasting based on CEEMDAN and transformer, Electr. Power Syst. Res. 214 (2023) 108885, http://dx.doi.org/10.1016/j.epsr.2022.108885.

[49] B. Bohara, R.I. Fernandez, V. Gollapudi, X. Li, Short-term aggregated residential load forecasting using BiLSTM and CNN-bilstm, in: 2022 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies, 3ICT, IEEE, 2022, pp. 37–43, http://dx.doi.org/10.1109/3ICT56508.2022.9990696.

[50] P.N. Thanh, M.-Y. Cho, C.-L. Chang, M.-J. Chen, Short-term three-phase load prediction with advanced metering infrastructure data in smart solar microgrid based convolution neural network bidirectional gated recurrent unit, IEEE Access 10 (2022) 68686–68699, http://dx.doi.org/10.1109/ACCESS.2022.3185747.

[51] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (8) (1997) 1735–1780, http://dx.doi.org/10.1162/neco.1997.9.8.1735.

[52] A. Graves, J. Schmidhuber, Framewise phoneme classification with bidirectional LSTM and other neural network architectures, Neural Netw. 18 (5–6) (2005) 602–610, http://dx.doi.org/10.1016/j.neunet.2005.06.042.

[53] M.-H. Guo, Z.-N. Liu, T.-J. Mu, S.-M. Hu, Beyond self-attention: External attention using two linear layers for visual tasks, IEEE Trans. Pattern Anal. Mach. Intell. 45 (5) (2022) 5436–5447, http://dx.doi.org/10.1109/TPAMI.2022.3211006.

[54] M.-H. Guo, J.-X. Cai, Z.-N. Liu, T.-J. Mu, R.R. Martin, S.-M. Hu, Pct: Point cloud transformer, Comput. Vis. Media 7 (2021) 187–199, http://dx.doi.org/10.1007/s41095-021-0229-5.

[55] F. Chollet, Xception: Deep learning with depthwise separable convolutions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1251–1258, http://dx.doi.org/10.1109/CVPR.2017.195.

[56] F. Wu, A. Fan, A. Baevski, Y.N. Dauphin, M. Auli, Pay less attention with lightweight and dynamic convolutions, in: International Conference on Learning Representations, 2019.

[57] H. Shi, L. Wang, R. Scherer, M. Woźniak, P. Zhang, W. Wei, Short-term load forecasting based on adabelief optimized temporal convolutional network and gated recurrent unit hybrid neural network, IEEE Access 9 (2021) 66965–66981, http://dx.doi.org/10.1109/ACCESS.2021.3076313.

[58] M. Liu, H. Qin, R. Cao, S. Deng, Short-term load forecasting based on improved TCN and DenseNet, IEEE Access 10 (2022) 115945–115957, http://dx.doi.org/10.1109/ACCESS.2022.3218374.

[59] M. Aliyari, Y.Z. Ayele, Application of artificial neural networks for power load prediction in critical infrastructure: A comparative case study, Appl. Syst. Innov. 6 (6) (2023) 115, http://dx.doi.org/10.3390/asi6060115.

[60] H. Sun, Y. Ren, S. Wang, B. Zhao, R. Yin, Smart grid power load type forecasting: research on optimization methods of deep learning models, Front. Energy Res. 11 (2023) 1321459, http://dx.doi.org/10.3389/fenrg.2023.1321459.

[61] M.Y. Junior, R.Z. Freire, L.O. Seman, S.F. Stefenon, V.C. Mariani, L. dos Santos Coelho, Optimized hybrid ensemble learning approaches applied to very short-term load forecasting, Int. J. Electr. Power Energy Syst. 155 (2024) 109579.

[62] S. Smyl, G. Dudek, P. Pełka, Contextually enhanced ES-dRNN with dynamic attention for short-term load forecasting, Neural Netw. 169 (2024) 660–672, http://dx.doi.org/10.1016/j.neunet.2023.11.017.

[63] X. Wang, H. Wang, B. Bhandari, L. Cheng, AI-empowered methods for smart energy consumption: A review of load forecasting, anomaly detection and demand response, Int. J. Precis. Eng. Manuf.-Green Technol. 11 (3) (2024) 963–993, http://dx.doi.org/10.1007/s40684-023-00537-0.

[64] W. Zha, Y. Ji, C. Liang, Short-term load forecasting method based on secondary decomposition and improved hierarchical clustering, Res. Eng. 22 (2024) 101993, http://dx.doi.org/10.1016/j.rineng.2024.101993.

[65] P. Pełka, Pattern-based forecasting of monthly electricity demand using support vector machine, in: 2021 International Joint Conference on Neural Networks, IJCNN, IEEE, 2021, pp. 1–8, http://dx.doi.org/10.1109/IJCNN52387.2021.9534134.

[66] G. Dudek, Neural networks for pattern-based short-term load forecasting: A comparative study, Neurocomputing 205 (2016) 64–74, http://dx.doi.org/10.1016/j.neucom.2016.04.021.

[67] R.A. Abbasi, N. Javaid, M.N.J. Ghuman, Z.A. Khan, S. Ur Rehman, Amanullah, Short term load forecasting using XGBoost, in: Web, Artificial Intelligence and Network Applications: Proceedings of the Workshops of the 33rd International Conference on Advanced Information Networking and Applications (WAINA-2019) 33, Springer, 2019, pp. 1120–1131, http://dx.doi.org/10.1007/978-3-030-15035-8_108.

[68] S. Khan, N. Javaid, A. Chand, A.B.M. Khan, F. Rashid, I.U. Afridi, Electricity load forecasting for each day of week using deep CNN, in: Web, Artificial Intelligence and Network Applications: Proceedings of the Workshops of the 33rd International Conference on Advanced Information Networking and Applications (WAINA-2019) 33, Springer, 2019, pp. 1107–1119, http://dx.doi.org/10.1007/978-3-030-15035-8_107.

[69] L. Kuan, Z. Yan, W. Xin, C. Yan, P. Xiangkun, S. Wenxue, J. Zhe, Z. Yong, X. Nan, Z. Xin, Short-term electricity load forecasting method based on multilayered self-normalizing GRU network, in: 2017 IEEE Conference on Energy Internet and Energy System Integration, EI2, IEEE, 2017, pp. 1–5, http://dx.doi.org/10.1109/EI2.2017.8245330.

[70] P. Pełka, G. Dudek, Pattern-based long short-term memory for mid-term electrical load forecasting, in: 2020 International Joint Conference on Neural Networks, IJCNN, IEEE, 2020, pp. 1–8, http://dx.doi.org/10.1109/IJCNN48605.2020.9206895.

[71] T.-Y. Kim, S.-B. Cho, Predicting residential energy consumption using CNN-LSTM neural networks, Energy 182 (2019) 72–81, http://dx.doi.org/10.1016/j.energy.2019.05.230.

[72] K. Zuo, Integrated forecasting models based on LSTM and TCN for short-term electricity load forecasting, in: 2023 9th International Conference on Electrical Engineering, Control and Robotics, EECR, IEEE, 2023, pp. 207–211, http://dx.doi.org/10.1109/EECR56827.2023.10149951.

[73] H. Ismail Fawaz, B. Lucas, G. Forestier, C. Pelletier, D.F. Schmidt, J. Weber, G.I. Webb, L. Idoumghar, P.-A. Muller, F. Petitjean, Inceptiontime: Finding alexnet for time series classification, Data Min. Knowl. Discov. 34 (6) (2020) 1936–1962, http://dx.doi.org/10.1007/s10618-020-00710-y.

[74] H.A. Al-Jamimi, G.M. BinMakhashen, M.Y. Worku, M.A. Hassan, Advancements in household load forecasting: Deep learning model with hyperparameter optimization, Electronics 12 (24) (2023) 4909, http://dx.doi.org/10.3390/electronics12244909.

[75] Z. Sheng, H. Wang, G. Chen, B. Zhou, J. Sun, Convolutional residual network to short-term load forecasting, Appl. Intell. 51 (4) (2021) 2485–2499, http://dx.doi.org/10.1007/s10489-020-01932-9.

[76] C. Cai, Y. Li, Z. Su, T. Zhu, Y. He, Short-term electrical load forecasting based on VMD and GRU-TCN hybrid network, Appl. Sci. 12 (13) (2022) 6647, http://dx.doi.org/10.3390/app12136647.

[77] Y. Chen, D. Zhang, Theory-guided deep-learning for electrical load forecasting (TgDLF) via ensemble long short-term memory, Adv. Appl. Energy 1 (2021) 100004, http://dx.doi.org/10.1016/j.adapen.2020.100004.

[78] A. Groß, A. Lenders, F. Schwenker, D.A. Braun, D. Fischer, Comparison of short-term electrical load forecasting methods for different building types, Energy Inform. 4 (Suppl 3) (2021) 13, http://dx.doi.org/10.1186/s42162-021-00172-6.

[79] T.S. Khan, H. Nazaripouya, Adaptive-extremum seeking method for impedance identification in distribution grids, in: 2023 IEEE PES Grid Edge Technologies Conference & Exposition, Grid Edge, IEEE, 2023, pp. 1–5, http://dx.doi.org/10.1109/GridEdge54130.2023.10102728.