Enhanced Long-Term Prediction Based on 2D Tensorization for Building Energy Consumption

Song Deng*
Key Laboratory of Advanced Process Control for Light Industry (Ministry of Education), Jiangnan University, Wuxi, Jiangsu, China
**Corresponding Author.*

Abstract: Building energy consumption constitutes a significant contributor to greenhouse gas emissions. Precise forecasting of energy consumption in buildings plays a pivotal role in managing building energy efficiently, thereby aiding in the reduction of greenhouse gas emissions. However, existing prediction algorithms often focus on within-period variations in energy consumption data and overlook between-period variations. This limitation makes it challenging to achieve precise prediction results, especially for long-term forecasts. To address this research gap, this paper proposes an Enhanced Long-Term Prediction based on 2D Tensorization (ELP2T) for building energy consumption. First, a Frequency-Guided 2D Tensorization Network (FG-2TN) is proposed. Energy consumption data, when represented as a one-dimensional time series, faces certain limitations, and FG-2TN is employed to address these limitations. Second, a Progress-Optimized Deep Convolutional Network (PO-DCN) is proposed. It is designed to efficiently extract and learn features from the obtained two-dimensional tensors with fewer parameters and less time. Third, a modular method is proposed to transform one-dimensional time series into two-dimensional tensors using Fast Fourier transform. These tensors are processed and then concatenated into a one-dimensional sequence for output. Ultimately, a comparative analysis was carried out using several traditional forecasting algorithms to highlight the superior performance of the ELP2T model. The obtained average R^2 score for our proposed method is 0.807, representing an impressive enhancement of 11.06% compared to the most advanced alternatives. This substantial improvement firmly establishes the superiority of our approach. Especially, when considering a prediction length of 720 units, the performance gain of this metric increases to 18.09%, underscoring the pronounced advantage of our method in addressing long-term forecasting scenarios.

Keywords: Buildings electricity consumption; Long-Term prediction; Data-driven method; Deep learning; Frequency-Guided 2D Tensorization Network; Prgress-Optimized Deep Convolutional Network

1. Introduction

The world is currently facing numerous problems caused by global warming. One of the leading causes of global warming is the excessive emissions of greenhouse gases. Surprisingly, buildings represent a significant portion, approximately 40%, of the total greenhouse gas emissions[1]. To address this problem effectively, it becomes crucial to accurately predict building energy consumption, as it allows for better management and planning of energy usage[2]. Such predictions enable intelligent control decisions, help maintain a balance between energy supply and demand, provide insights into building behavior, and facilitate optimal adaptations to changing conditions[3]. They assist in achieving finer control over power systems and elevating energy utilization efficiency. Moreover, long-term prediction can help power systems better analyze electricity consumption cycles and make better plans[4]. Hence, enhancing the accuracy of predictions for building energy consumption is imperative[5].

Methods for predicting building energy consumption are generally divided into two main categories: methods based on physical models and data-driven methods.

Physical-model-based methods, also known as white-box approaches, utilize the principles of thermodynamics and heat transfer to estimate energy usage within buildings. For example, the energy consumption of heating, ventilation, and air conditioning (HVAC) systems is estimated using various approaches, including the variable base degree day method, the equivalent full load hours method, and the bin method. Popular software tools for simulating building energy include EnergyPlus, eQuest, and DOE-2, among others. These physical-model-based approaches require detailed input information about the building's physical attributes, HVAC systems, equipment, and the schedules of occupants for simulation purposes. While this detailed input can lead to precise energy consumption predictions, it also tends to make the modeling process more complex and time-consuming.

Data-driven approaches for predicting building energy consumption fall into two primary categories: statistical methods and machine learning techniques. Statistical methods, often referred to as grey box models, generally rely on regression analysis for prediction. For instance, Wen et al. [6] utilized Multiple Linear Regression (MLR) for forecasting energy use in buildings. Additionally, Kozitsin et al.[7] applied the Auto-regressive Integrated Moving Average (ARIMA) model to capture time series trends and seasonal variations in energy consumption. Though these conventional statistical techniques laid the groundwork for predicting energy use, the intricate and multifaceted nature of building systems limited their precision in forecasts.

To overcome the limitations of physical-model based and statistical methods, researchers have turned to machine learning. Ma et al.[8] employed Support Vector Machines (SVM) to address nonlinear problems and developed an SVM-based model for forecasting office lighting energy consumption. Amber et al.[9] utilized Artificial Neural Networks (ANNs) to predict power consumption in a London-based administrative building. Jiang et al.[10] used Random Forest (RF) for energy consumption prediction, improving stability and accuracy by combining multiple decision trees. Additionally, ensemble learning methods have had a significant impact on building energy consumption prediction[11]. However, existing machine learning algorithms face challenges such as inaccurate predictions, low computational efficiency, and limited applicability.

In recent years, deep learning algorithms have made significant advancements, particularly in the field of building energy consumption prediction. Cai et al.[12] successfully applied Convolutional Neural Networks (CNNs) to building energy consumption prediction, offering new insights into data processing for energy forecasting. At the same time, Recurrent Neural Networks (RNNs) and their variants, such as Long Short-Term Memory (LSTM) networks and Gated Recurrent Units (GRUs), have been widely adopted for their ability to effectively model the dependencies in sequential data. Fan et al. proposed a hybrid Convolutional-Recurrent Neural Network for residential building energy consumption prediction. The researchers conducted comparisons across a variety of deep learning models specialized for time series analysis, including LSTM, GRU, CNN-LSTM, and CNN-GRU. In their experiments, the CNN-GRU model emerged as the most effective, demonstrating superior performance. Recently, Transformers with attention mechanisms have been widely adopted for sequential modeling. Li et al.[13] improved the performance of sequence-to-sequence deep learning models by incorporating attention mechanisms. Their experiments, focusing on energy consumption prediction in 36 buildings, demonstrated an average 2% improvement in prediction accuracy. Time series models excel at capturing temporal information in building energy data but become less reliable for longer prediction horizons.

Extensive research has shown that time series models are effective in making accurate predictions for short-term periods, typically less than 48 hours. However, their reliability and accuracy decrease when it comes to long-term forecasts that span beyond 48 hours. The majority of studies in this field focus on predicting outcomes over short to medium-term durations, largely ignoring long-term prediction due to the limitations of one-dimensional time series models. Existing prediction algorithms primarily consider the impact of each time point on neighboring point. However, they tend to overlook the interdependencies and connections between adjacent regions with in the same timeframe.

This constraint restricts the identification and extraction of closely connected two-dimensional local features, which are crucial for achieving highly accurate long-term predictions. Consequently, the task of accurately predicting outcomes over extended periods of time remains a challenging endeavor.

In time series, variations at individual time points are not only influenced by nearby temporal points but also strongly correlated with changes in neighboring periods. To make this distinction clearer, we categorize these temporal variations into two types: 'within-period variation' and 'between-period variation'. The former refers to changes between neighboring time points within the same cycle, while the latter refers to changes between time instances corresponding to identical phases across various cycles. According to these findings, this paper proposes an Enhanced Long-Term Prediction based on 2D Tensorization (ELP2T) model. The ELP2T model consists of two main components: The Frequency-Guided 2D Tensorization Network (FG-2TN) and the Progress-Optimized Deep Convolutional Network (PO-DCN). The FG-2TN module utilizes the Fast Fourier Transform (FFT) to transform 1D time series into multiple sets of 2D tensors at different frequencies. This expansion of the analysis into a 2D space allows for a more comprehensive examination of temporal changes. The PO-DCN module is designed to extract temporal features from the transformed 2D tensors and facilitate effective feature learning. The contributions of this study are outlined as follows:

1) A modular method is proposed to transform 1D time series into 2D space, enabling the simultaneous presentation of both within-period and between-period variations.

2) The FG-2TN module is proposed to obtain different frequencies and convert 1D energy consumption data into 2D tensors.

3) The PO-DCN module is proposed for feature extraction and learning by parameter-efficient blocks in 2D space.

4) The introduction of the ELP2T framework for forecasting building energy consumption, which achieves state-of-the-art performance in predominant analytical tasks.

2. Preliminary

2.1 Definition

When talking about energy consumption data, it belongs to the category of time series data. This classification comes from the fact that energy consumption data is organized in chronological order, where each data point is linked to a specific time instance. Typically, these measurements are taken at hourly, daily, or other time intervals. Time series models have the ability to uncover various time-dependent patterns within energy consumption data, such as seasonal, trending, and periodic patterns. Consequently, they allow for predicting future consumption trends and conducting detailed analyses. Energy consumption data is represented using the definition of a time series as follows:

$$X_{1D} = \{x_1, x_2, \cdots, x_{l_x} | x_i \in \mathbb{R}^{d_x}\} \quad (1)$$

Where X_{1D} represents the collection of original 1D time series data, symbolizing the input energy consumption data sequences used for prediction. l_x denotes the sequence length, and d_x signifies the number of variables recorded at each time point.

2.2 Data Preparation

2.2.1 Dataset

The publicly available dataset 'Informer-ELC' was utilized, collecting hourly electricity consumption data from 321 consumers in a city in the western United States spanning the years 2012 to 2014. A partial visualization of the dataset is shown in Figure 1.

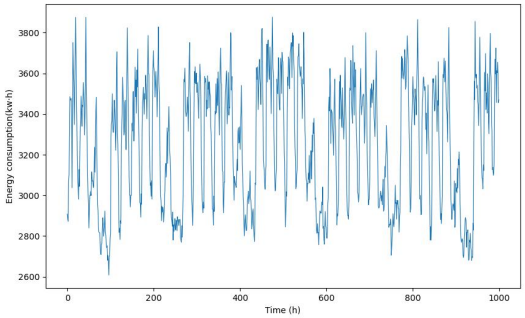


Figure 1. Partial Dataset Presentation

The construction of a data-driven model requires three distinct datasets[14]:

1) Training data, which is used to train and fit the data-driven model;

2) Validation data, utilized for unbiased assessment while fine-tuning the model's hyperparameters;

3) Testing data, employed to evaluate the

performance of the final model.

The dataset was partitioned into training, validation, and testing data following a ratio of 7:1:2.

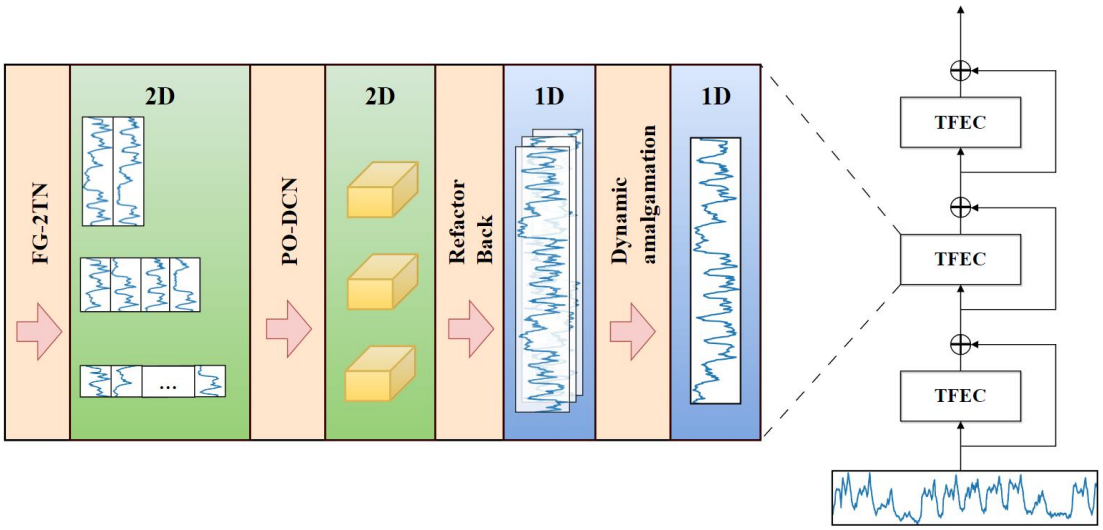


Figure 2. The Overview of the ELP2T Model Framework

2.2.2 Normalization

When the features in a dataset exhibit substantial discrepancies or the data distribution is uneven, this can lead to a significant degradation in predictive performance. Normalization plays a pivotal role in data preprocessing by adjusting the scales of features, eliminating dimensional influences, and enhancing model performance and algorithm convergence speed. Its function is to bring disparate features to a comparable scale, thus reducing the impact of outliers, improving algorithmic robustness, and concurrently emphasizing data patterns during feature engineering. This not only promotes data comparability and interpretability but also establishes a resilient foundation for machine learning and data analysis. The Min-Max normalization approach has been employed, individually applied to each data point, as elucidated below:

$$\mathrm{Norm}(x_i^m) = \frac{x_i^m - x_{min}^m}{x_{max}^m - x_{min}^m} \quad (2)$$

Where x_i^m and $\mathrm{Norm}(x_i^m)$ represent the original and normalized data for the m-th feature of the sequence, respectively. x_{max}^m and x_{min}^m denote the maximum and minimum values of the m-th dataset.

3. Method

This section offers a detailed overview of the ELP2T model, including its architecture depicted in Figure 2.

3.1 Overall Structure

As shown in Fig 2, ELP2T is composed of multiple Tensorized Feature Extraction Cell (TFEC) connected in a residual way. Each TFEC comprises the FG-2TN and PO-DCN modules. FG-2TN takes the 1D time series of building energy consumption as input and transforms it into a 2D representation. Subsequently, PO-DCN performs 2D feature extraction and learning. It then refactors back and combines the obtained 1D sequences dynamically to form the final output.

For the h-th layer of ELP2T, where the input is denoted as X_{1D}^{h-1}, this process can be formally expressed as:

$$X_{1D}^{h} = \mathrm{TFEC}\left(X_{1D}^{h-1}\right) + X_{1D}^{h-1} \quad (3)$$

To provide a more intuitive and clear representation of the model's structure, we have presented an algorithm flowchart, as shown in Figure 3.

In the upcoming subsections of this chapter, we will delve into detailed explanations of the FG-2TN and PO-DCN modules.

3.2 Frequency-Guided 2D Tensorization Network

Figure 4 illustrates that each time point encompasses two distinct temporal variations: variations within its immediate vicinity (within-period variation) and variations across different periods (between-period variation). However, the traditional 1D time series format is limited to capturing changes between consecutive time points. To overcome this limitation, we have explored the

implementation of a 2D temporal structure. This structure is explicitly designed to represent both within- and between-period variations. Such an improvement in the representation significantly enhances the potential for more effective feature learning.

To be more specific, for a time series of length l_x recording d_x variables, the original 1D arrangement is denoted as $X_{1D} \in \mathbb{R}^{l_x \times d_x}$. In order to capture in between-period variation, the identification of frequencies and periods is necessary. This is achieved through FFT analysis in the frequency domain on the time series, as shown below:

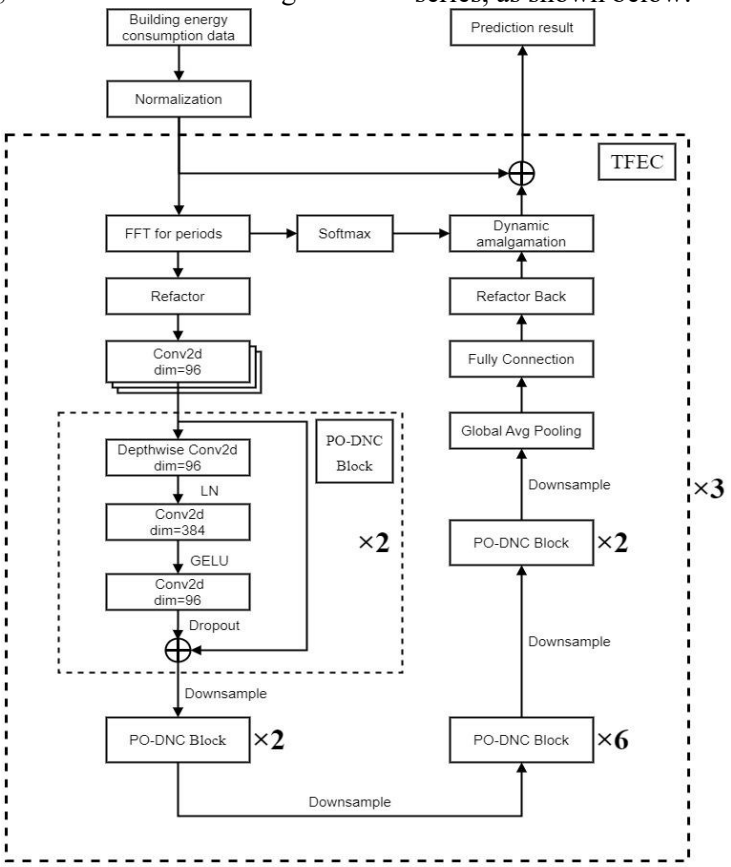


Figure 3. The Algorithm Flowchart of ELP2T

$$A = Avg\left(Amp(FFT(X_{1D}))\right) \quad (4)$$

$$\{f_1, f_2, \cdots, f_k\} = arg\,\mathrm{Topk}(A) \quad (5)$$

$$\tau_i = \left\lceil {}^{l_x}/_{f_i} \right\rceil, i \in \{1,2,\cdots,k\} \quad (6)$$

Where $\mathrm{FFT}(\cdot)$ and $\mathrm{Amp}(\cdot)$ represent Fast Fourier Transform and amplitude calculation, respectively. The symbol $A \in \mathbb{R}^{l_x}$ represents the computed amplitude values of various frequencies, obtained by averaging the values across the d_x dimensions using $\mathrm{Avg}(\cdot)$. Notably, the *i*-th value of amplitude A_i denotes the amplitude strength corresponding to the *i*-th frequency, which corresponds to the periodic length $\tau_i = \left\lceil {}^{l_x}/_{f_i} \right\rceil$. To account for the sparsity in the frequency domain and mitigate noise from irrelevant high frequencies, we select only the top-k amplitudes and obtain the most significant frequency $\{f_1, f_2, \cdots, f_k\}$ along with the non-normalized amplitude $\{A_1, A_2, \cdots, A_k\}$, where k is a hyperparameter. These selected frequencies also correspond to k periodic lengths $\{\tau_1, \tau_2, \cdots, \tau_k\}$. Considering the conjugate symmetry of the Fourier transform for real signals, we focus solely on frequencies within $\left\{1,2,\cdots,\left\lceil {}^{l_x}/_{2} \right\rceil\right\}$. We summarize Eq (4-6) as follows:

$$\{f_1, f_2, \cdots, f_k\}, \{\tau_1, \tau_2, \cdots, \tau_k\} = \mathrm{FreqPick}(X_{1D}) \quad (7)$$

In brief, X_{1D} gets k sets of frequencies with the largest amplitude $\{f_1, f_2, \cdots, f_k\}$ and corresponding period lengths $\{\tau_1, \tau_2, \cdots, \tau_k\}$ after the operation of $\mathrm{FreqPick}(\cdot)$.

Based on the selected frequencies $\{f_1, f_2, \cdots, f_k\}$ and their corresponding period lengths $\{\tau_1, \tau_2, \cdots, \tau_k\}$, we can reshape the 1D time series $X_{1D} \in \mathbb{R}^{l_x \times d_x}$ into k sets of 2D tensors using the following equation:

$$X_{2D}^i = \mathrm{Refactor}_{f_i, \tau_i}\left(\mathrm{Padding}(X_{1D})\right),\ i \in \{1,2,\cdots,k\} \quad (8)$$

Where $\text{Padding}(\cdot)$ expands the time series by padding zeros along the time dimension to ensure compatibility with the $\text{Refactor}_{f_i,\tau_i}(\cdot)$, where f_i and τ_i represent the number of rows and columns of the transformed 2D tensor.

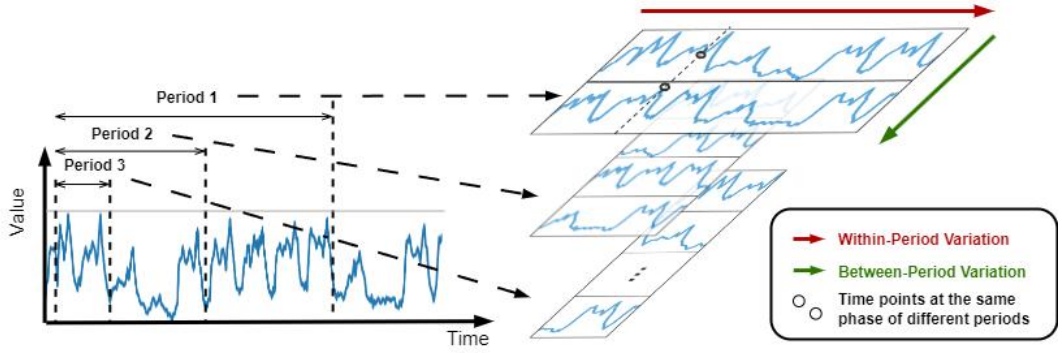


Figure 4. A Univariate Example to Illustrate 2D Structure in Time Series

It's worth noting that $X_{2D}^{i} \in \mathbb{R}^{f_i \times \tau_i \times d_x}$ represents the i-th time series reshaped according to frequency - f_i. Ultimately, as illustrated in Figure 4, through the application of the selected frequency and estimated period, we derive a collection of 2D tensors $\{X_{2D}^{1}, X_{2D}^{2}, \cdots, X_{2D}^{k}\}$, which signifies k unique temporal variations originating from various periods. Furthermore, it's important to highlight that this transformation brings forth two kinds of localities within the transformed 2D tensors: localities among neighboring time points (columns, indicating within-period variation) and localities among successive periods (rows, indicating between-period variation). As a result, the 2D temporal variations can be efficiently analyzed using 2D kernels.

3.3 Progress-Optimized Deep Convolutional Network

Convolutional Neural Networks (CNNs) have demonstrated remarkable performance in 2D image processing. Through convolutional layers and pooling layers, CNNs can effectively capture local features of images and progressively learn higher-level abstract features. However, in order to make CNNs applicable to structured data and get more accurate predictions, it need some improvements. We have incorporated some of its macroscopic and microscopic design elements that are beneficial for model performance into traditional CNNs, resulting in the proposal of the Progress-Optimized Deep Convolutional Network (PO-DCN). The PO-DCN module encompasses a range of novel design choices aimed at enhancing both the efficiency and effectiveness of tensor processing. We will delve into these design aspects in the following subsections. The framework of PO-DCN is illustrated in Figure 5.

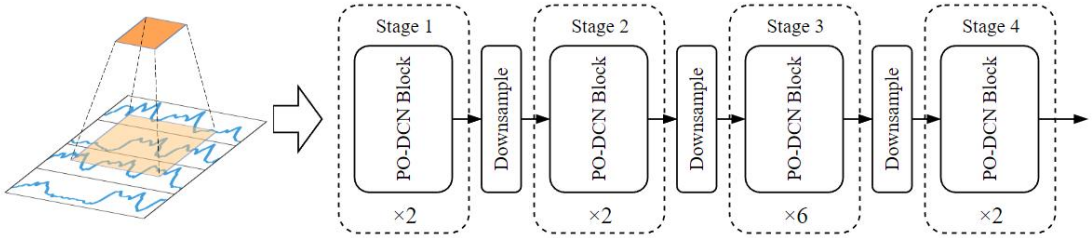


Figure 5. The Framework of PO-DCN

3.3.1 Depthwise separable convolution

Traditional convolutional neural network consists of two main steps: spatial convolution and channel-wise mixing. However, Depthwise Separable Convolution breaks these steps down and utilizes different approaches: depthwise convolution and pointwise convolution, to replace the conventional convolution operations. In the depthwise convolution phase, processing is performed separately on each channel of the input feature map, without any interaction between the channels. In other words, each channel employs an independent convolution kernel for feature extraction. The advantage of this strategy lies in reducing the number of model parameters while still capturing unique

information from each channel.

Following the depthwise convolution stage, pointwise convolution utilizes 1×1 convolution kernel to fuse features across channels. It is used to aggregate the information contained in all channels, and generate the final output feature map. It is noteworthy that pointwise convolution does not perform spatial sliding operations, as it solely undertakes linear combinations between channels. The key advantage of Depthwise Separable Convolution lies in significantly reducing the model's parameter count and computational cost.

Through this decomposed convolution approach, Depthwise Separable Convolution successfully segregates spatial and channel-related feature extraction processes, thereby enhancing model efficiency and lightweightness while preserving information richness. Novel avenues in the design of deep learning models, aiding in achieving remarkable performance even under limited computational resources. Fig 6 shows the comparison of traditional convolution and depthwise separable convolution.

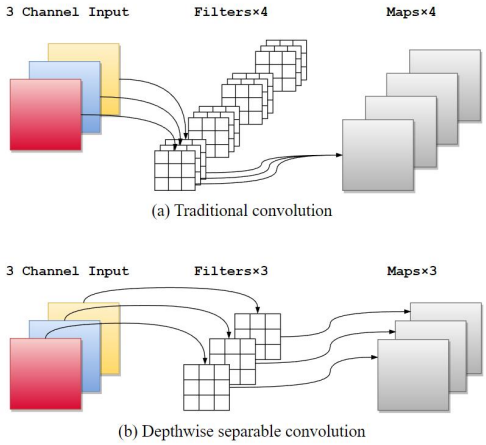


Figure 6. Comparison of Traditional Convolution and Depthwise Separable Convolution

3.3.2 Inverted bottleneck

In the design of the Transformer block, a pivotal innovation is the introduction of the inverted bottleneck structure, which involves expanding the hidden dimension of the Multi-Layer Perceptron (MLP) block to four times the input dimension. We incorporate this design into the CNN framework. The essence of the inverted bottleneck structure aims to enhance the network's feature extraction capability without introducing excessive computational burden and memory overhead, thus better capturing critical information within input data. The comparison of traditional bottleneck structure and PO-DNC's inverted bottleneck structure is shown is Figure 7.

To provide a more comprehensive depiction of this structure and its enhancements, we have compared the structure of ResNeXT with the PO-DCN block. Firstly, by applying smaller convolution kernels for dimension reduction, the input feature map is compressed, effectively reducing computational demands and memory requirements. Subsequently, larger convolution kernels are employed to capture richer feature information, enabling the network to precisely comprehend subtle variations within input data. Lastly, smaller convolution kernels are once again employed to elevate the dimension, restoring features to a higher dimension. This multi-layer convolutional operation strategy empowers the network to process input data more intricately and comprehensively without sacrificing crucial features.

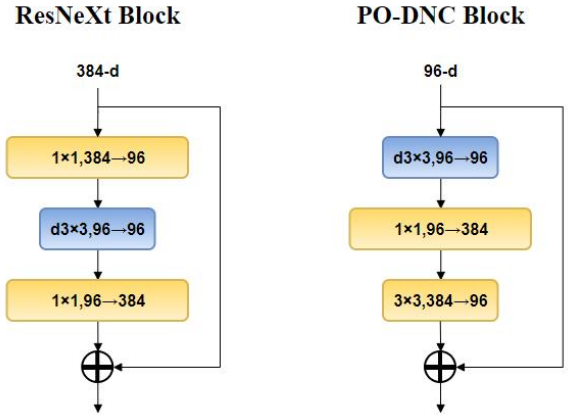


Figure 7. Traditional Bottleneck Structure (left) and PO-DNC's Inverted Bottleneck Structure (right)

Such a design holds significance when integrated into CNNs. The introduction of the inverted bottleneck structure enables the network to adapt better to features of varying scales and hierarchies, thereby enhancing feature extraction capabilities. Using distinct-sized convolution kernels at critical stages, the network becomes adept at capturing the diversity and complexity of input data, showcasing superior performance across diverse tasks.

In summary, the integration of the inverted bottleneck design from the Transformer block into CNNs brings innovation to the network's feature extraction process. This design strategy enhances the network's expressive capacity

while maintaining computational efficiency, allowing the network to better comprehend and process input data, resulting in improved performance across various applications.

3.3.3 Changing stage compute ratio

In traditional CNNs, as the network depth increases, issues like vanishing or exploding gradients may arise, rendering the training of deep networks challenging. To address this, ResNet introduced a clever approach by dividing the network into multiple consecutive stages, each containing several residual blocks. The output of each stage serves as the input for the next, forming a cascaded architecture. During training, the model gradually approximates the ideal mapping by learning from the residual blocks in each stage, enabling the construction of deeper networks.

Recently, the Swin Transformer (ST) has showcased remarkable performance in tasks like 2D image processing. Although ST does not employ the structure of CNN, it still adopts the multi-stage concept. In the original ResNet, the stacking of blocks across stages (from stage1 to stage4) followed a ratio of (3, 4, 6, 3), approximately 1:1:2:1. In contrast, in the ST, for example, Swin-T has a ratio of 1:1:3:1, and Swin-L has a ratio of 1:1:9:1. It's evident from these examples that the ST emphasizes a higher proportion of block stacking in the third stage. Drawing inspiration from this, we attempt to modernize the model's parameters by applying a similar stage ratio, incorporating the essence of the ST's approach. We set the stacking of residual blocks in stages 1 to 4 of our model to (3, 3, 9, 3), aiming to further optimize its performance. This not only maintains the model's lightweight and efficient nature but also better uncovers the inherent structure of features. This modernized design holds the potential to enhance the model's performance and achieve superior results across various tests.

4. Experimental Results and Analysis

In this section, we conducted a comprehensive set of experiments to assess the effectiveness of our proposed model. We begin by introducing the general performance evaluation metrics employed. Subsequently, we compare and analyze the experimental outcomes of our model with those of other models for a comparative study. Finally, we conducted ablation experiments to delve into the individual contributions of various modules within the model.

4.1 Evaluation Metrics

In this research, we evaluated the efficacy of various prediction methods by comparing them across several metrics, namely Mean Squared Error (*MSE*), Mean Absolute Error (*MAE*), and the Coefficient of Determination (R^2). These metrics provide a comprehensive assessment of prediction accuracy and model performance. The formulas for calculating these metrics are outlined in Equations (9-11), where i represents the sample index, n denotes the total number of samples, y represents the measured value, $\hat{y}$ represents the predicted value, and $\bar{y}$ is the mean output. The equations are as follows:

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

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

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y}_i)^2} \quad (11)$$

MAE and *MSE* are common metrics in regression analysis. *MAE* measures the average absolute difference between predicted and actual values, squared differences, emphasizing larger errors. indicating model accuracy. *MSE* is a statistical measure that quantifies the average squared difference between the estimated values and the actual value, widely used in regression analysis to assess model accuracy. Both metrics assess regression model fit by quantifying prediction errors. Smaller values of these metrics indicate more accurate predictions.

R^2 is a widely used metric in statistics to measure the goodness of fit of a regression model. It represents the proportion of the variance in the dependent variable that can be explained by the independent variables, indicating the degree to which the regression model fits the actual data. R^2 values range between 0 and 1, where a value closer to 1 indicates a better explanatory power of the model for the data.

4.2 Experimental Performance and ComParative Analysis

To better show the model's predictive performance across various sequence lengths, we adopted different prediction sequence lengths. Keeping the input length l_x fixed at 96, we evaluated the model's predictive accuracy for various prediction sequence lengths $l_y \in$

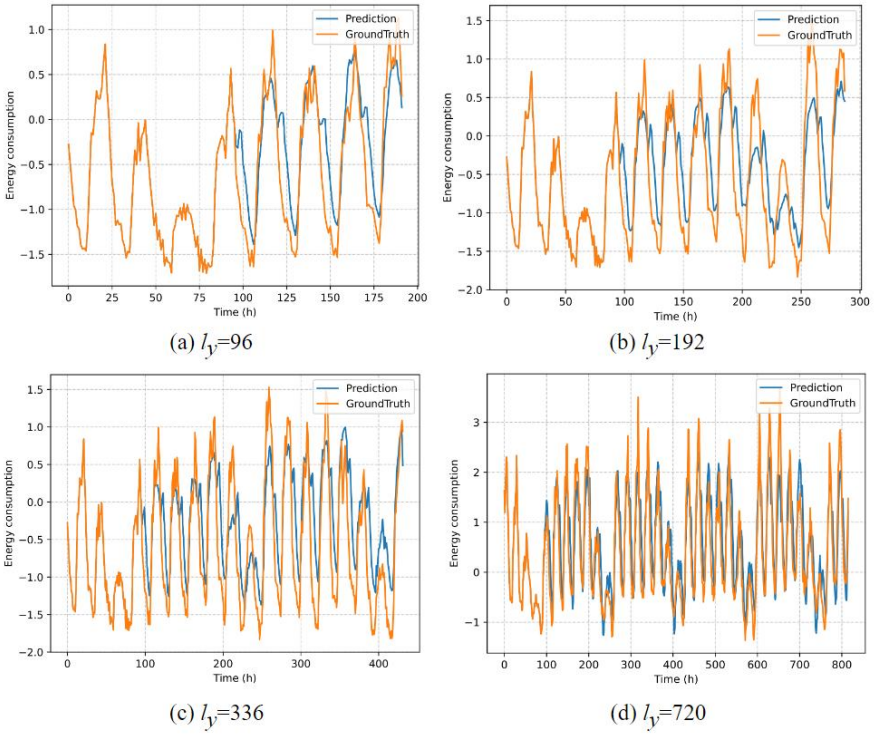
{96,192,336,720}. The visual representation of the experimental results is depicted in Fig 8.

To further validate the superior predictive performance of our proposed ELP2T model compared to other models, we established six commonly used time series forecasting algorithms for comparison. These include traditional methods like ARIMA, Temporal Convolutional Network (TCN), LSTM, as well as state-of-the-art Transformer-based approaches such as LogTrans, Informer, and Fedformer. It's important to note that due to Fedformer exhibiting the best performance across all benchmark tests, it was chosen as the primary benchmark model for comparison.

Table 1 presents the prediction results of ELP2T in comparison with other benchmark algorithms. All algorithms were evaluated using the same training, validation, and testing datasets. Fig 9 illustrates the radar chart of the R^2 metrics, and Fig 10 illustrates the results of the *MSE* and *MAE* metrics for different models under various prediction lengths, providing an intuitive overview of the algorithm's prediction performance.

Table 1. The Results of Comparative Experimental

Method	96			192			336			720		
	MSE	MAE	R^2	MSE	MAE	R^2	MSE	MAE	R^2	MSE	MAE	R^2
ELP2T	**0.171**	**0.277**	**0.819**	**0.189**	**0.292**	**0.806**	**0.207**	**0.311**	**0.786**	**0.223**	**0.331**	**0.753**
Fedformer	0.183	0.297	0.783	0.195	0.308	0.753	0.212	0.323	0.712	0.231	0.353	0.657
Informer	0.274	0.368	0.654	0.296	0.386	0.643	0.300	0.394	0.612	0.373	0.439	0.578
LogTrans	0.258	0.357	0.726	0.266	0.368	0.654	0.280	0.380	0.596	0.359	0.416	0.544
LSTM	0.375	0.437	0.537	0.442	0.473	0.504	0.639	0.676	0.439	0.980	0.814	0.286
TCN	0.985	0.813	-	0.996	0.821	-	1.012	0.824	-	1.438	0.784	-
ARIMA	0.879	0.764	-	1.032	0.833	-	1.136	0.876	-	1.136	0.933	-



(a) l_y=96 (b) l_y=192

(c) l_y=336 (d) l_y=720

Figure 8. Visualization of Some Prediction Results Under Various Prediction Lengths

It is evident that ELP2T consistently achieves the best performance across all prediction sequence lengths. Compared to Fedformer, the proposed ELP2T demonstrates an overall relative improvement of 11.06% in terms of R^2. Notably, this improvement becomes more pronounced for longer prediction sequence lengths, such as 336 and 720, where R^2 is relatively enhanced by over 12.50% and 18.10%, respectively.

In summary, ELP2T consistently delivers exceptional performance across different prediction sequence lengths, with its advantages becoming particularly evident as the sequence

length increases. This suggests that ELP2T exhibits a comparative advantage over other sequence forecasting models, especially when dealing with longer prediction sequences.

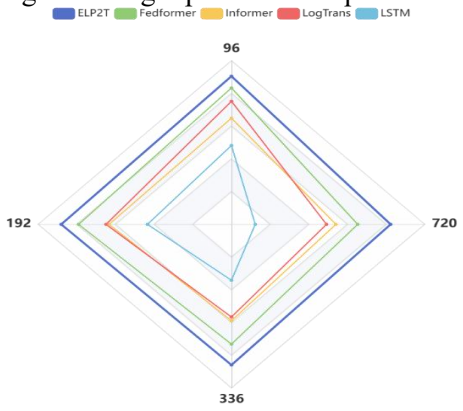


Figure 9. Radar Chart of Comparative ExPeriments for R^2 Metrics

4.3 Ablation Experiments Analysis

In this subsection, we conduct ablation experiments to explore the impact of key components in our approach on the final performance. By gradually removing or adjusting these components, we aim to investigate their influence on the overall results. Taking the state-of-the-art (SOTA) results achieved by utilizing FG-2TN and PO-DCN modules as our baseline, we tested three ablation variants of ELP2T:

1) PO-DCN: In this variant, we removed the FG-2TN module from ELP2T.

2) FG-2TN + TCN: We replaced the PO-DCN module with a traditional TCN.

3) TCN: This variant involves the removal of both the FG-2TN module and the PO-DCN module, with TCN serving as their replacement.

4) By comparing the performance of these ablation variants to the baseline which uses FG-2TN and PO-DCN modules, we can gain insights into the contributions of each component to the final performance of ELP2T.

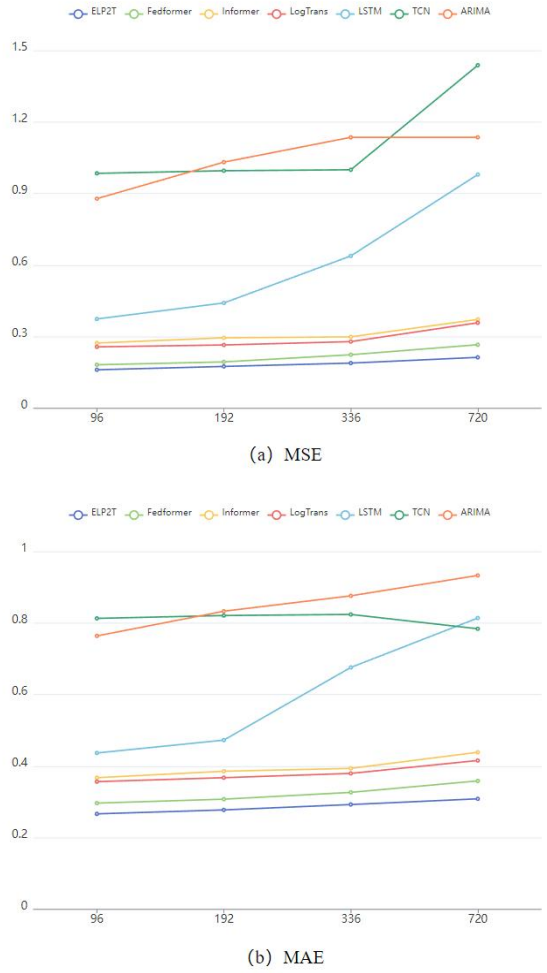


Figure 10. Performance of Comparative ExPeriments Under Different Prediction Lengths

Table 2. The Results of Ablation Experiments

Method	96			192			336			720		
	MSE	MAE	R^2	MSE	MAE	R^2	MSE	MAE	R^2	MSE	MAE	R^2
ELP2T	**0.169**	**0.273**	**0.819**	**0.186**	**0.288**	**0.806**	**0.203**	**0.306**	**0.786**	**0.224**	**0.325**	**0.753**
PO-DCN	0.288	0.394	0.713	0.387	0.486	0.607	0.593	0.657	0.493	0.822	0.758	0.328
FG-2TN+TCN	0.224	0.328	0.759	0.263	0.357	0.688	0.307	0.386	0.601	0.413	0.414	0.536
TCN	0.985	0.813	-	0.996	0.821	-	1.012	0.824	-	1.438	0.784	-

The results of the ablation experiments are presented in Table 2. Additionally, a radar chart depicting the R^2 scores of the ablation experiments can be found in Fig 11. Analyzing the prediction results of PO-DCN and ELP2T, it's evident that the FG-2TN module within ELP2T, which transforms data into a 2D space for feature extraction and learning, effectively enhances prediction accuracy. This effect becomes more pronounced as the prediction length increases. Furthermore, by comparing the results of 'FG-2TN+TCN' with those of ELP2T, it demonstrates the excellent feature learning capability of PO-DCN in the 2D space. In addition, when comparing PO-DCN with some traditional prediction algorithms as shown in Table 1, it's clear that PO-DCN achieves better prediction results, highlighting the

effectiveness of the improvements made to traditional CNNs.

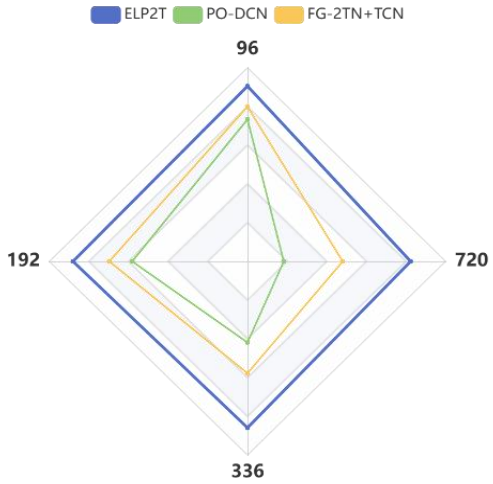


Figure 11. Radar Chart of Ablation ExperiMents for R² Metrics

4.4 Generalization Analysis

To evaluate the generalization capabilities of the ELP2T algorithm, it was implemented on a real-world dataset. This dataset comprised electricity consumption records over three years from 93 buildings within a university located in Wuxi, Jiangsu Province, China. For benchmarking purposes, these records were also analyzed using the state-of-the-art Fedformer model. The outcomes of training both ELP2T and Fedformer on this dataset are illustrated in Figure 12. The results clearly show that the ELP2T model consistently delivers robust prediction performance, even when applied to diverse datasets. This experimentation underscores the ELP2T model's strong generalization capabilities, affirming its effectiveness beyond the initial conditions it was designed under.

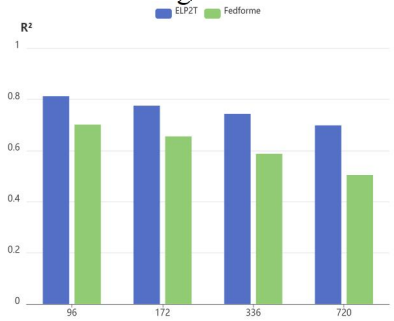


Figure 12. Experimental Results of GeneralIzation Analysis

5. Conclusion

In this study, we have introduced the ELP2T model, aiming to contribute to the improvement of accurate long-term building energy consumption prediction. Our approach is guided by the unique insights into temporal variations within and between periods, leading to the development of a comprehensive framework. Firstly, we introduced the FG-2TN module, which leverages the FFT to transform 1D time series into 2D tensors. This innovative approach allows for the extraction of spatial and frequency features, effectively capturing short-term within-period variations and long-term between-period trends. Secondly, we introduced the PO-DCN module, inspired by both CNNs and Transformer architectures. This module facilitates efficient feature extraction from the derived 2D tensors, further enhancing the accuracy and efficiency of our model. Finally, the proposed ELP2T model demonstrated its superiority in predicting building energy consumption. Experimental results demonstrate that our model outperforms other algorithms in terms of accuracy and stability. Furthermore, the experiments highlight the model's potential for generalization and its applicability to diverse datasets.

The ELP2T model proposed in this paper is particularly suited for predicting building energy consumption with strong periodic patterns. However, real-world scenarios often involve non-periodic or weakly periodic energy consumption data. Therefore, future research should focus on enhancing the accuracy of predictions for non-periodic data. We aim to explore additional deep learning models, such as the Diffusion model, to achieve accurate predictions for a wider range of complex energy consumption data.

References

[1] X. Liang, S. Chen, X. Zhu, X. Jin, Z. Du, Domain knowledge decomposition of building energy consumption and a hybrid data-driven model for 24-h ahead predictions, Appl. Energy. 344 (2023) 121244. https://doi.org/10.1016/j.apenergy.2023.121244.

[2] X. Niu, Z. Ma, W. Ma, J. Yang, T. Mao, The spatial spillover effects and equity of carbon emissions of digital economy in China, J. Clean. Prod. 434 (2024) 139885. https://doi.org/10.1016/j.jclepro.2023.1398

85.
[3] L. Lei, W. Chen, B. Wu, C. Chen, W. Liu, A building energy consumption prediction model based on rough set theory and deep learning algorithms, Energy Build. 240 (2021) 110886. https://doi.org/10.1016/j.enbuild.2021.110886.
[4] Luis Pérez-Lombard, José Ortiz, Christine Pout, A review on buildings energy consumption information, Energy and Buildings, volume 40, issue 3, 2008, pages 394-398, ISSN 0378-7788. https://doi.org/10.1016/j.enbuild.2007.03.007.
[5] Y. Chen, M. Guo, Z. Chen, Z. Chen, Y. Ji, Physical energy and data-driven models in building energy prediction: A review, Energy Rep. 8 (2022) 2656–2671. https://doi.org/10.1016/j.egyr.2022.01.162.
[6] L. Wen, K. Zhou, S. Yang, Load demand forecasting of residential buildings using a deep learning model, Electr. Power Syst. Res. 179 (2020) 106073. https://doi.org/10.1016/j.epsr.2019.106073.
[7] V. Kozitsin, I. Katser, D. Lakontsev, Online Forecasting and Anomaly Detection Based on the ARIMA Model, Appl. Sci. 11 (2021) 3194. https://doi.org/10.3390/app11073194.
[8] H. Zhong, J. Wang, H. Jia, Y. Mu, S. Lv, Vector field-based support vector regression for building energy consumption prediction, Appl. Energy. 242 (2019) 403–414. https://doi.org/10.1016/j.apenergy.2019.03.078.
[9] K.P. Amber, R. Ahmad, M.W. Aslam, A. Kousar, M. Usman, et. al, Intelligent techniques for forecasting electricity consumption of buildings, Energy. 157 (2018) 886–893. https://doi.org/10.1016/j.energy.2018.05.155.
[10] B. Jiang, Z. Cheng, Q. Hao, N. Ma, A Building Energy Consumption Prediction Method Based on Random Forest and ARMA, in: 2018 Chinese Automation Congress (CAC), 2018. https://doi.org/10.1109/cac.2018.8623540.
[11] R. Wang, B. Rong, S. Ma, D. Ma, L. Wu, et. al, Prediction of ash fusion temperatures of municipal solid waste incinerator ash based on support vector regression, J. Energy Inst. 111 (2023) 101438. https://doi.org/10.1016/j.joei.2023.101438.
[12] Y.-J. Lee, H.-J. Choi, Forecasting Building Electricity Power Consumption Using Deep Learning Approach, in: 2020 IEEE International Conference on Big Data and Smart Computing (BigComp), 2020. https://doi.org/10.1109/bigcomp48618.2020.000-8.
[13] Taik, S. Cherkaoui, Electrical Load Forecasting Using Edge Computing and Federated Learning, in: ICC 2020 - 2020 IEEE International Conference on Communications (ICC), 2020. https://doi.org/10.1109/icc40277.2020.9148937.
[14] T. Zhou, Z. Ma, Q. Wen, X. Wang, L. Sun, et. al, FEDformer: Frequency Enhanced Decomposed Transformer for Long-term Series Forecasting, International Conference on Machine Learning. (2022).

