

Article

A Hybrid Convolutional Network for Forecasting Dynamic Carbon Emission Factors of Electricity Users

Chenchen Liu ^{1,2}, Shenqiang Gan ^{1,2} and Xiaoshun Zhang ^{1,2,*} 

¹ College of Information Science and Engineering, Northeastern University, Shenyang 110819, China; liuchenchen@mails.neu.edu.cn (C.L.); gansq@mails.neu.edu.cn (S.G.)

² Foshan Graduate School of Innovation, Northeastern University, Foshan 528311, China

* Correspondence: zhangxiaoshun@mail.neu.edu.cn

Abstract

With the accelerating transition toward low-carbon power systems, dynamic carbon emission factor (DCEF) forecasting has become increasingly important for carbon accounting and real-time carbon management. However, the complex nonlinearity, strong temporal dependency, and occasional near-zero fluctuations in DCEF sequences make accurate forecasting difficult, as existing methods often fail to simultaneously capture long-term trends and short-term variations. To address these challenges, this paper proposes a hybrid deep learning framework termed MTCN (Mamba–temporal convolutional network fusion), which integrates a Mamba-based state space model and a temporal convolutional network (TCN) for DCEF forecasting. Experiments on two practical electricity carbon factor datasets demonstrate the effectiveness of the proposed framework. MTCN achieves prediction accuracies of 95.62% and 95.56% on two case datasets, respectively, outperforming mainstream deep learning models and existing hybrid approaches. The results indicate that the proposed method effectively captures both long-term evolutionary patterns and short-term fluctuations of DCEFs, providing decision information support for real-time carbon management and power system decarbonization.

Keywords: dynamic carbon emission factors; time series forecasting; Mamba; temporal convolutional network; hybrid deep learning

1. Introduction

With the advancement of China’s carbon neutrality strategy, carbon emission assessment has become increasingly important for power system operation and energy management. In December 2021, the State Council issued the Measurement Development Plan (2021–2035), which emphasized the improvement of carbon emission measurement systems and the development of electricity-based carbon-accounting technologies [1]. In August 2022, the National Development and Reform Commission, the National Bureau of Statistics, and the Ministry of Ecology and Environment issued the Implementation Plan for Accelerating the Establishment of a Unified and Standardized Carbon Emission Statistical Accounting System, highlighting the application of dynamic carbon emission factors in greenhouse gas accounting [2]. In October 2022, the State Administration for Market Regulation and other ministries further proposed the development of carbon emission factors, online carbon monitoring, and uncertainty evaluation technologies [3]. These policies provide an institutional basis for the development and application of dynamic carbon emission accounting.



Academic Editor: Alessandro Niccolai

Received: 5 August 2026

Revised: 16 September 2026

Accepted: 18 September 2026

Published: 21 September 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

Hua et al. [4] proposed a carbon emission flow-based energy routing strategy for the Energy Internet, enabling carbon-aware energy routing by tracing carbon emissions along energy transmission paths. Zhang et al. [5] developed a hybrid time series simulation framework that integrates multifactor decomposition with deep learning to forecast provincial carbon emissions under different scenarios. Liu et al. [6] constructed a real-time carbon emission estimation framework for industrial parks based on non-intrusive load monitoring, enabling equipment-level emission estimation without additional sensors. These studies demonstrate the growing demand for high temporal resolution carbon information in power system carbon assessment.

Against this background, accurate accounting of indirect electricity-related emissions has become increasingly important for energy-intensive industries [7]. Traditional static emission factors, which rely on average values over fixed periods, cannot adequately reflect short-term changes in grid carbon intensity caused by variations in generation structure and load conditions. In contrast, dynamic carbon emission factors (DCEFs) provide time-varying information on the carbon intensity associated with electricity generation and consumption. He et al. [8] developed a dynamic carbon-accounting model based on carbon flow theory and a quasi-input–output framework, enabling carbon emissions and emission factors to be traced across transmission, substation, and distribution nodes. Lin et al. [9] incorporated DCEFs into the multi-objective optimal scheduling of integrated energy systems and demonstrated their effectiveness as low-carbon dispatch signals.

Accurate DCEF forecasting is therefore important for low-carbon scheduling, demand-side carbon management, and electricity carbon accounting [10]. Yan et al. [11] proposed a short-term forecasting method for electricity carbon emission factors with high temporal and spatial resolution, demonstrating the feasibility of fine-grained carbon factor prediction. Yang et al. [12] evaluated regional power grid carbon emission risks using DCEFs and showed that dynamic carbon factors can also support carbon risk identification. Zhang et al. [13] developed a low-carbon scheduling method for industrial parks based on dynamic carbon emission prediction, using the predicted results to optimize electricity consumption patterns. Jiao et al. [14] proposed an enhanced dispatch model for park-level integrated energy systems that explicitly considers electricity–carbon coupling. Radovanović et al. [15] developed a carbon-aware computing framework for data centers, in which day-ahead carbon intensity forecasts were used to shift flexible workloads from high-carbon to low-carbon periods. Liu et al. [16] proposed a real-time emission and cost estimation method based on unit-level dynamic carbon emission factors, providing a data basis for demand-side carbon accounting. Together, these studies indicate that future DCEF information is increasingly important for carbon-aware operational decision-making.

However, accurate DCEF forecasting remains challenging because carbon intensity sequences exhibit strong nonlinearity, temporal dependence, and volatility. Kotsompolis et al. [17] combined ARIMA, XGBoost, and LSTM for carbon price forecasting and showed that hybrid models can better capture nonlinear temporal dynamics than a single statistical model. Yu et al. [18] investigated community-level electricity carbon emission prediction using dynamic emission coefficients and demonstrated that neglecting time-varying carbon intensity can introduce prediction bias. Cai et al. [19] developed a day-ahead carbon emission factor forecasting method based on LSTM, confirming the effectiveness of deep learning for temporal dependence modeling while also highlighting the volatility and near-zero characteristics of carbon factor sequences.

Traditional statistical models such as ARIMA and regression methods remain limited when handling highly nonlinear and non-stationary temporal dynamics. Nagendra and Singh [20] compared ARIMA, linear regression, random forest, and LSTM, showing that ARIMA performs well for relatively simple linear and seasonal patterns but is less effective

for complex nonlinear sequences. Siami-Namini et al. [21] compared ARIMA, LSTM, and BiLSTM and found that recurrent neural networks provide stronger forecasting capability for complex temporal patterns. With the development of deep learning, architectures such as LSTM, GRU, TCN, and Transformers have been increasingly adopted for time series forecasting. Wu et al. [22] proposed Autoformer, which improves long-sequence forecasting through decomposition and autocorrelation mechanisms. Zhou et al. [23] proposed Informer, which reduces the computational and memory costs of long-sequence forecasting through ProbSparse self-attention and a generative decoder. Maji et al. [24] developed CarbonCast for multi-day grid carbon intensity forecasting, demonstrating the applicability of deep learning to carbon intensity prediction. Guan et al. [25] proposed a daily electricity carbon emission factor forecasting method for energy-intensive enterprises based on temporal feature optimization and hybrid intelligent algorithms. Chen et al. [26] developed a real-time carbon emission factor prediction method for power system load nodes based on a gated recurrent cell network.

Despite these advances, existing methods still face limitations in simultaneously capturing long-range temporal dependencies and short-term local fluctuations. Gu et al. [27] proposed the structured state space sequence model S4, demonstrating the effectiveness of state space modeling for long-sequence representation. Uhrig et al. [28] applied recurrent neural networks to multivariate energy demand forecasting and noted their relatively high computational cost and limited capability for modeling long-range dependencies. Yu et al. [29] proposed PI-Mamba for battery state-of-health estimation, combining Mamba with an equivalent circuit model and demonstrating the potential of selective state space modeling in energy-related time series applications. Bukhari et al. [30] developed a TCN-based fault detection and classification scheme for microgrids, in which dilated causal convolution-based residual blocks are used to capture low-level fault features and high-level temporal dependencies without an external feature extractor. Ref. [31] reviewed Transformer-based multivariate time series forecasting and noted that convolution-based approaches are effective for local temporal pattern extraction but remain limited in global dependency modeling. These limitations motivate the development of a hybrid forecasting framework that combines the complementary capabilities of state space modeling and temporal convolution.

To address these challenges, this paper proposes a hybrid deep learning framework called MTCN (Mamba–temporal convolutional network fusion), which combines a Mamba-based state space model with a TCN, focusing on the time series characteristics of carbon factors in high-energy-consuming enterprises. The main contributions of this paper are as follows:

- A correlation study was conducted, selecting appropriate time series input windows based on periodicity and trend to fully utilize the time series data of DCEFs.
- Reasonable error evaluation metrics are adopted to effectively avoid the evaluation bias caused by near-zero DCEF values, ensuring fair and stable performance assessment across different datasets.
- The dual-branch architecture can simultaneously capture the inherent long-term time dependence and multi-scale local fluctuation patterns in the DCEF sequences. Subsequently, a fusion module based on the multi-layer perceptron (MLP) was designed to adaptively integrate the complementary representations learned from the two branches, thereby enhancing nonlinear feature interactions and improving the overall prediction performance.
- A large number of comparative experiments and hyperparameter experiments conducted on two power carbon factor datasets demonstrated that the proposed algo-

rithm framework consistently outperformed mainstream deep learning models and existing hybrid methods in terms of performance.

2. Methods

This section presents the architecture and prediction procedure of the proposed MTCN model. It covers three aspects: the selection of input features, the network structure of MTCN, and the workflow of the DCEF prediction model (DCP). Figure 1 illustrates the overall DCP workflow. The input is a one-dimensional time series of DCEF values, which is directly used as a single-channel input without additional channel replication. MTCN is constructed from Mamba, TCN, and a fully connected layer. The tensor shape evolves through the network as follows: the input has shape (batch size, 240, 1); the Mamba branch and the TCN branch each produce a scalar prediction batch size (batch size, 1); the two branch outputs are concatenated into a fusion feature of shape (batch size, 2); the fusion MLP maps it to (batch size, 1); and the final output remains (batch size, 1).

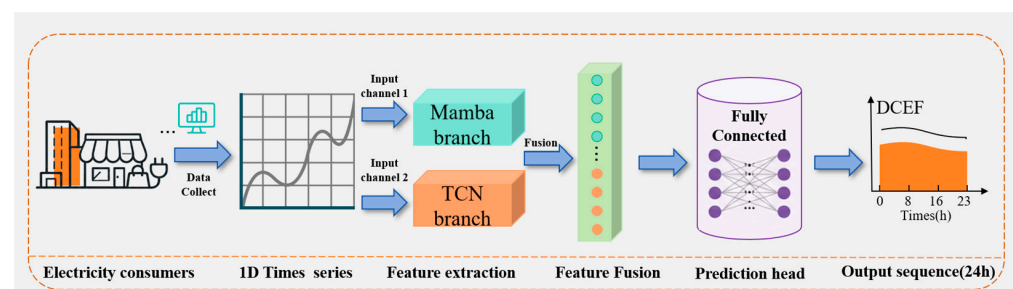


Figure 1. DCEF prediction framework diagram.

2.1. Problem Formulation

DCEF time series exhibits strong temporal autocorrelation due to the inherent physical inertia of power systems, the periodicity of human activities, and the persistence of meteorological conditions affecting renewable energy generation. Specifically, DCEFs at hour t are significantly correlated with their historical values at lags corresponding to diurnal cycles (24 h), weekly patterns (168 h), and even longer-term trends driven by seasonal load variations and carbon market dynamics. This autocorrelation structure provides the fundamental basis for using historical observations to predict future DCEF values.

Let the DCEF time series be denoted as $X = \{x_1, x_2, \dots, x_T\} \in R^T$, where x_t represents the DCEF per unit of electricity consumption at time step t with hourly resolution. The temporal autocorrelation of X can be quantified by the autocorrelation function (ACF) $\rho(k) = \frac{E[(x_t - \mu)(x_{t-k} - \mu)]}{\sigma^2}$, where μ and σ^2 are the mean and variance of the series, respectively, and k denotes the time lag. Empirical analysis of real-world DCEF data reveals that $\rho(k)$ remains significant ($|\rho(k)| > 0.3$) for lags up to $k = 240$ h (10 days), indicating substantial long-range temporal dependence that extends beyond simple daily periodicities. This observation corroborates findings in related energy forecasting literature, where electricity-related time series have been shown to possess long-memory characteristics spanning multiple days due to the compounding effects of weather patterns, industrial production cycles, and grid dispatch decisions.

Based on this autocorrelation analysis, we formulate the DCEF prediction problem as follows. Given a historical window of length $L = 240$ h, we aim to learn a mapping function: $R^L \rightarrow R^H$ that predicts the DCEF values for the next H hours: $\hat{x}_{t+1}, \dots, \hat{x}_{t+H} = f(x_{t-L+1}, x_{t-L+2}, \dots, x_t)$, where H is the prediction horizon. In this study, we focus on single-step-ahead prediction ($H = 1$), i.e., forecasting the DCEF value for the next hour

given the preceding 240 h, which serves as the fundamental building block for longer-horizon forecasts.

Given the historical window $X_t = \{x_{t-L+1}, \dots, x_t\} \in R^L$, we construct the training dataset comprising input–output pairs $\{X_t, x_{t+1}\}$. The prediction task is then to minimize the expected prediction error $loss = E[\zeta(f(X_t), x_{t+1})]$. The MAPE metric is particularly suitable for DCEF prediction in practical applications, as it directly quantifies the percentage deviation in carbon accounting, facilitating cost estimation in carbon trading markets.

2.2. Selection of Time Series Window

The selection of the historical window length L is a critical determinant of forecasting performance for DCEFs, as it directly governs the amount of temporal information available to the model. Rather than relying on heuristic or fixed-length choices, we propose a data-driven approach based on the ACF, which quantitatively measures the linear dependence between current and past observations. This method ensures that the chosen window captures statistically significant temporal dependencies while excluding weakly correlated or noisy lags, thereby balancing information sufficiency and model parsimony.

Formally, for a stationary DCEF time series $\{x_t\}$, the sample autocorrelation coefficient at lag k is defined as:

$$\hat{\rho}(k) = \frac{\sum_{t=k+1}^T (x_t - \bar{x})(x_{t-k} - \bar{x})}{\sum_{t=1}^T (x_t - \bar{x})^2} \quad (1)$$

where $\bar{x}$ is the sample mean and T is the total number of observations. The coefficient $\hat{\rho}(k)$ ranges from -1 to 1 , with values significantly deviating from zero indicating persistent serial correlation. Under the null hypothesis of no autocorrelation, $\hat{\rho}(k)$ asymptotically follows a normal distribution with variance $\frac{1}{T}$, enabling the construction of a 95% confidence interval (approximately $\pm \frac{1.96}{\sqrt{T}}$). We compute the ACF over a sufficiently long maximum lag (e.g., 30 days) and identify the smallest lag K beyond which the absolute ACF values consistently fall below the confidence band and do not exhibit any periodic resurgence. This lag K is interpreted as the decorrelation time—the point after which past observations provide negligible predictive information for the current value.

Regarding the time series autocorrelation structure of the DCEF dataset (Figure 2), we detected a slow-decaying pattern with significant long-term memory characteristics: the ACF remained above the statistically significant threshold of 0.25 for the lags $\tau \leq 240$ h (approximately 10 days), and gradually decayed below the noise baseline only after crossing this critical time scale. Further detailed analysis of the daily frequency observations revealed that the correlation evolution within this 10-day window was not a monotonic decrease but rather a non-stationary re-coupling process that “first decreased and then increased.” This indicates the existence of multiple competitive decorrelation mechanisms within the system. By integrating the correlation maps with weekly and ten-day aggregation granularities, the above nonlinear time-scale features were cross-validated. This finding strongly corroborates the selection of 10 days as the optimal embedding window for information retention and state predictability in this dynamical system.

Based on the combined ACF analysis, we set the historical window to $L = 240$ h (10 days). This choice offers three advantages: (1) it encompasses ten complete diurnal cycles and one full weekly cycle, capturing both short-term and medium-term periodicities; (2) it exploits the majority of significant autocorrelation while excluding the long-tail noise beyond 240 h; and (3) it aligns with the typical persistence of synoptic weather systems and power system dispatch horizons (day-ahead to week-ahead), providing a physically meaningful context for DCEF evolution. The subsequent experiments also confirmed that 10 days was indeed the appropriate duration.

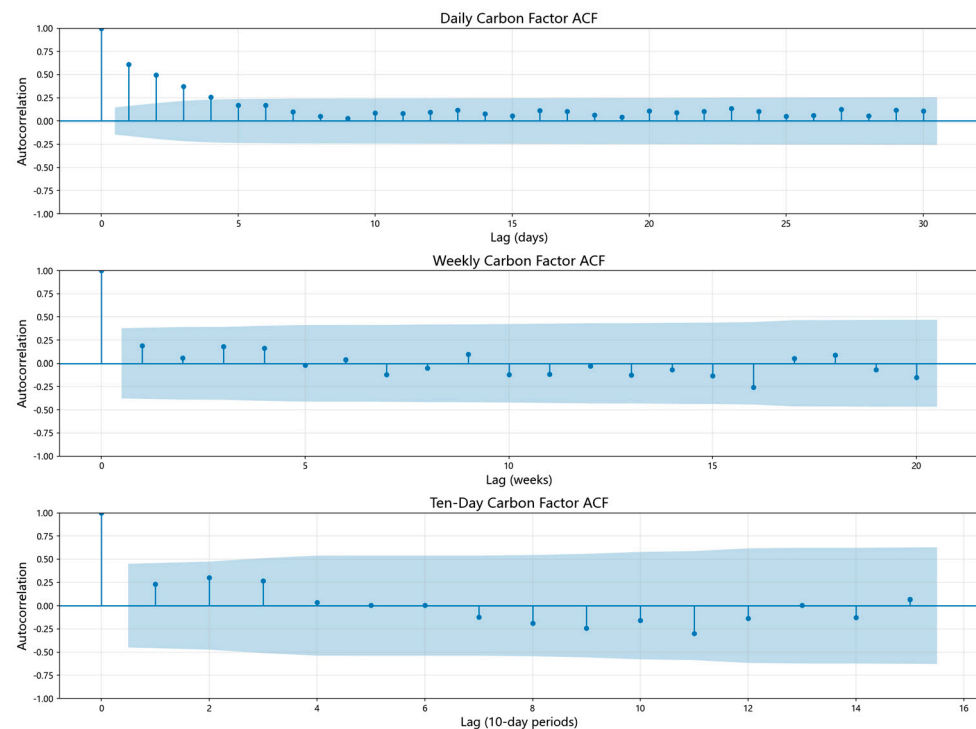


Figure 2. The daily, weekly, and 10-day autocorrelation of the concrete dataset.

2.3. Network Structure of MTCN

The MTCN network architecture comprises an organic integration of the Mamba state space model, TCN, and a fusion MLP, which synergistically combines the distinctive advantages of each component. The Mamba module operates on the embedded representation of the 240 h history and performs content-aware selective state space modeling over this representation, effectively addressing the quadratic complexity bottleneck inherent in Transformer-based architectures. The TCN module, through its hierarchically stacked dilated causal convolutions, extracts multi-scale local temporal features at varying receptive fields, capturing short-term volatility patterns induced by hourly load fluctuations and renewable energy ramping events. The fusion MLP serves as the adaptive integration module, employing a learnable gating mechanism to dynamically balance the contributions of global dependencies from Mamba and local patterns from TCN, achieving optimal feature synergy for the final regression task.

The remainder of this section elaborates on the network modules encompassed in the proposed MTCN architecture.

A. Mamba of MTCN

The Mamba module serves as the global dependency extraction backbone of the MTCN architecture (as shown in Figure 3), and originates from the state space model (SSM) theory that has recently demonstrated remarkable performance in long-sequence modeling tasks. Unlike Transformer-based models that suffer from $O(T^2)$ computational complexity due to the self-attention mechanism, Mamba maintains $O(T)$ complexity through its selective scan algorithm while preserving the ability to capture ultra-long-range dependencies. This characteristic makes Mamba particularly suitable for the 240 h DCEF prediction task, where historical patterns spanning multiple days must be effectively memorized and utilized.

The core innovation of Mamba lies in its time-varying state transition mechanism, which enables content-aware reasoning. For an input sequence $X = \{x_1, x_2, \dots, x_T\} \in$

$R^{T \times d}$, where $T = 240$ and d denotes the feature dimension, the Mamba module computes the hidden state evolution through:

$$h_t = \bar{A}_t h_{t-1} + \bar{B}_t x_t \quad (2)$$

$$y_t = C_t h_t + D_t x_t \quad (3)$$

where $h_t \in R^d$ represents the hidden state at time step t , $\bar{A}_t$ and $\bar{B}_t$ are input-dependent discretized state transition matrices, C_t is the output projection matrix, and D_t is the skip connection matrix. Critically, the parameters $\bar{A}_t$ and $\bar{B}_t$ are dynamically generated from the input x_t through learnable projections, enabling the model to selectively propagate or forget information based on the current input context—a property termed selective state space modeling.

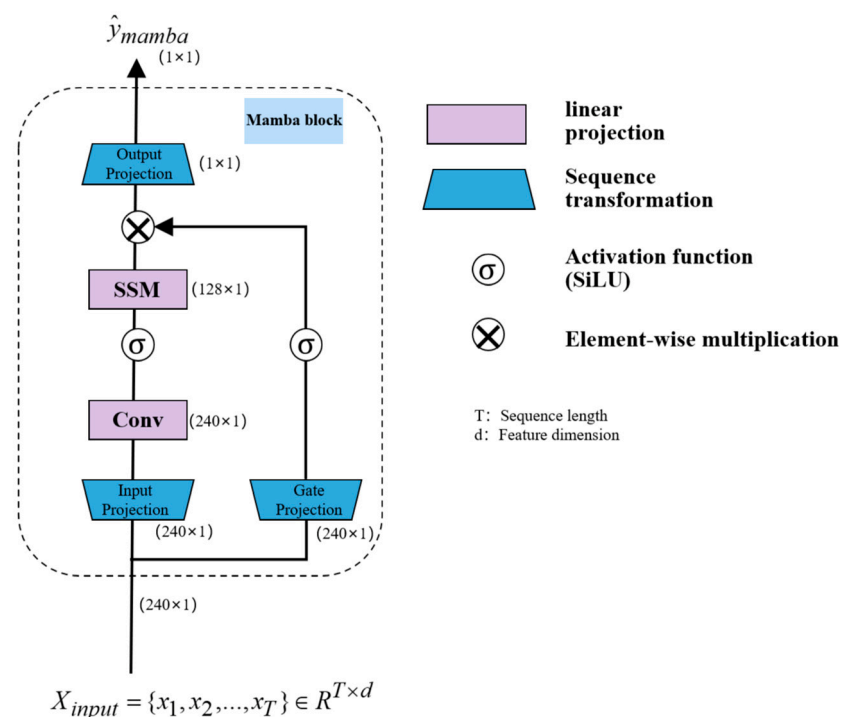


Figure 3. Mamba branch model algorithm.

Our implementation adopts the QuadMamba variant supporting multi-directional scanning, consisting of four stages: (1) input projection and depthwise convolution—linear expansion followed by a 3×1 depthwise convolution with SiLU activation to capture local context; (2) multi-directional selective scan—parallel scanning along multiple directions, completing state space transformation in $O(T)$ time; (3) gated output projection—modulating SSM outputs with the original input as gating signals; and (4) final projection and residual connection—projecting back to the original dimension with a residual connection to facilitate gradient flow.

The Mamba module stacks $L = 4$ layers with layer normalization between layers and produces a branch prediction $\hat{y}_{mamba} \in R$ through a linear projector. Its key advantages include linear complexity, selective memory, and bidirectional context capture.

The Mamba module stacks $L = 4$ layers with layer normalization between layers. The tensor shape evolves through the branch as follows: the input has shape $(B, 240, 1)$. It is first projected by the SeriesEmbedding layer into a representation of shape $(B, 1, d_{model})$, where the temporal dimension is encoded into the d_{model} —dimensional embedding. The

parameter chosen for this experiment is 128, that is, $d_{model} = 128$. The Mamba branch processes this embedded representation and produces a scalar prediction of shape $(B, 1)$.

B. TCN of MTCN

The TCN module serves as the local feature extraction branch (as shown in Figure 4), with dilated causal convolution as its core operator. This operation simultaneously ensures causality (predictions depend only on historical data) and exponential receptive field expansion. For input $x \in R^T$ and filter f , the dilated causal convolution at time t is defined as:

$$F(t) = \sum_{i=0}^{k-1} f(i) \cdot x_{t-d \cdot i} \quad (4)$$

where k is the kernel size and d is the dilation factor. The dilation factor grows exponentially with layer depth as $d = 2^i$, with the receptive field expanding as $R = 1 + \sum_{i=0}^{num-1} (k-1) \cdot 2^i$.

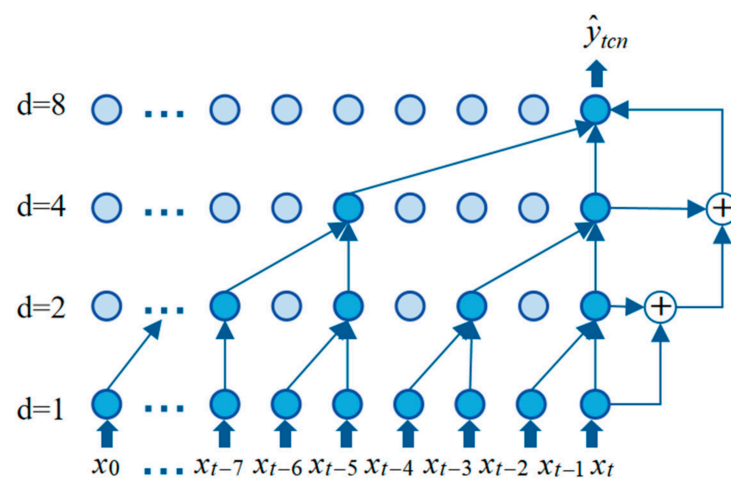


Figure 4. TCN branch model algorithm.

The module consists of three stacked residual blocks with dilation factors $d = [1, 2, 4]$ and channel configurations $[1 \rightarrow 64] \rightarrow [64 \rightarrow 64] \rightarrow [64 \rightarrow 128]$. Each residual block follows the pre-activation design:

$$out = \text{ReLU}(\text{Conv1D}(\text{ReLU}(\text{Conv1D}(x)))) + \text{Downsample}(x) \quad (5)$$

Residual connections ensure stable gradient propagation through depth. The feature vector at the last time step is extracted and projected through a linear layer to obtain branch prediction $\hat{y}_{tcn} \in R$. TCN offers advantages in parallel training, controllable receptive fields, and robust gradient flow.

Residual connections ensure stable gradient propagation through depth. The tensor shape evolves through the branch as follows: the input has shape $(B, 240, 1)$; after the first residual block it becomes $(B, 64, 240)$; after the second block, it becomes $(B, 64, 240)$; after the third block, it becomes $(B, 128, 240)$; the feature vector at the last time step is then extracted, giving a shape $(B, 128)$; and a linear layer projects it to a scalar prediction of shape $(B, 1)$, denoted as $\hat{y}_{tcn} \in \mathbb{R}$. TCN offers advantages in parallel training, controllable receptive fields, and robust gradient flow.

C. Fully connected neural network of MTCN

The fully connected neural network is implemented as the FusionMLP module, which synthesizes the complementary representations from the Mamba and TCN branches. Unlike simple concatenation or averaging, FusionMLP employs a learnable adaptive gating mechanism that dynamically balances the contributions of both branches based on the

temporal context of each input sample. This design is motivated by the heterogeneous temporal patterns of DCEF sequences: global modeling dominates during stable renewable periods, while local capture becomes more critical during sudden ramping events. The adaptive gate enables the model to automatically determine the optimal fusion strategy.

FusionMLP consists of three steps. First, feature concatenation: $\hat{y}_{mamba}$ and $\hat{y}_{tcn}$, each of shape $(B, 1)$, are concatenated along the feature dimension into $z \in \mathbb{R}^{B \times 2}$. Second, gate generation: a two-layer MLP with Sigmoid activation produces the gate weight $\alpha = \sigma(W_2 \cdot \text{ReLU}(W_1 z + b_1) + b_2) \in [0, 1]^{pred_len}$, where $W_1 \in \mathbb{R}^{h \times 2}$, $W_2 \in \mathbb{R}^{1 \times h}$, and $h = 64$. Third, weighted fusion: $\hat{y}_{final} = \alpha \odot \hat{y}_{mamba} + (1 - \alpha) \odot \hat{y}_{tcn}$. When α approaches 1, the Mamba branch dominates; when α approaches 0, the TCN branch prevails; intermediate values indicate balanced utilization.

This mechanism offers three advantages: input-adaptive fusion, end-to-end gradient transparency, and interpretable gate values. The overall loss function is the mean squared error $Loss = \frac{1}{N} \sum (\hat{y}_{final} - y_{true})^2$, with end-to-end joint optimization encouraging both branches to learn complementary representations.

3. Results and Discussion

The proposed MTCN framework is compared with 10 classical time series forecasting models on both the concrete manufacturing dataset and the glass manufacturing dataset. The number of layers of all the models involved in the experiments is detailed in Table 1. All model implementations, training, and evaluation are carried out in Python (3.10.20) using the PyTorch (2.1.1) deep learning framework. The hardware configuration utilized in this study is as follows: an Intel(R) Core(tm) i5-14400f processor (Intel Corporation, Santa Clara, United States of America), 32 GB RAM, 2 TB storage, and an NVIDIA GeForce RTX 4060 Ti desktop GPU (NVIDIA, Santa Clara, United States of America).

Table 1. Efficacy testing outcomes of models under concrete manufacturing company.

Model Name	RMSE	MAE	R ²	MAPE
KNN	0.0498 ± 0.0002	0.0373 ± 0.0002	0.5124 ± 0.0036	0.0645 ± 0.0002
ARIMA	0.0362 ± 0.0001	0.0273 ± 0.0001	0.6086 ± 0.0016	0.0453 ± 0.0001
Random forest	0.0361 ± 0.0002	0.0268 ± 0.0002	0.6108 ± 0.0032	0.0453 ± 0.0001
XGBoost	0.0364 ± 0.0001	0.0269 ± 0.0001	0.6049 ± 0.0025	0.0447 ± 0.0002
GRU	0.0358 ± 0.0001	0.0269 ± 0.0002	0.6168 ± 0.0030	0.0445 ± 0.0001
TCN	0.0359 ± 0.0001	0.0270 ± 0.0001	0.6145 ± 0.0029	0.0447 ± 0.0001
RSAP-DFM	0.0359 ± 0.0002	0.0269 ± 0.0001	0.6158 ± 0.0018	0.0447 ± 0.0001
TCTS	0.0358 ± 0.0001	0.0268 ± 0.0001	0.6175 ± 0.0027	0.0445 ± 0.0002
MLP	0.0364 ± 0.0001	0.0273 ± 0.0002	0.6045 ± 0.0031	0.0452 ± 0.0002
Mamba	0.0361 ± 0.0002	0.0268 ± 0.0002	0.6108 ± 0.0015	0.0453 ± 0.0001
MTCN (proposal)	0.0354 ± 0.0001	0.0265 ± 0.0001	0.6249 ± 0.0034	0.0438 ± 0.0001

The DCEF dataset used in this study is constructed as follows. The unit of DCEF is kgCO₂/kWh. The spatial boundary is at the enterprise level, and the temporal resolution is hourly. The DCEF values are provided by the participating enterprises through their internal carbon emission monitoring and accounting systems. Due to confidentiality agreements with the data providers, the detailed generation mix, emission factors, and carbon flow allocation methods cannot be disclosed. Missing data are handled by forward nearest-neighbor interpolation. Forward nearest-neighbor interpolation uses only the historical observations before the missing point and does not involve future information. The measurement uncertainty is estimated at approximately ± X%, mainly originating from the accuracy of the enterprise electricity meters and the carbon emission accounting process.

Regarding the DCEF differences between the two enterprises, although they are located in the same regional grid, they are connected to different nodes and have different

electricity consumption times and load profiles. Therefore, their corresponding node marginal carbon intensities differ. We have rigorously formulated the mapping relationship between enterprise load, grid carbon intensity, and the DCEF target variable.

Model performance is evaluated using four metrics: root mean square error (RMSE), mean absolute error (MAE), coefficient of determination (R^2), and mean absolute percentage error (MAPE). These metrics are defined as follows.

$$u_{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_{i,predict} - y_{i,true})^2} \quad (6)$$

$$u_{MAE} = \frac{1}{n} \sum_{i=1}^n |y_{i,predict} - y_{i,true}| \quad (7)$$

$$u_{R^2} = 1 - \frac{\sum_{i=1}^n (y_{i,predict} - y_{i,true})^2}{\sum_{i=1}^n (y_{i,predict} - y_{i,mean})^2} \quad (8)$$

$$u_{MAPE} = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_{i,predict} - y_{i,true}}{y_{i,true}} \right| \quad (9)$$

where $y_{i,predict}$ is the forecast value; $y_{i,true}$ is the true value; and y_{mean} is the average of the true values. In this experiment, the prediction accuracy of the model is defined as $(1 - MAPE) \times 100\%$, which represents the complement of the average absolute percentage error, rather than the accuracy metric used in classification tasks. This study aims to construct comprehensive, diverse, and representative datasets of DCEFs for training and validating the proposed MTCN network, ensuring accurate prediction of DCEF variations under different industrial user profiles, seasonal conditions, and load patterns. The data are collected from two real-world industrial enterprises, corresponding to typical high-energy-consumption sectors: concrete manufacturing and glass manufacturing. Based on the recorded time series data, multiple operating scenarios are generated by considering different production periods (e.g., weekdays/weekends, peak/valley/off-peak electricity price periods), resulting in full-year hourly sequences covering 8760 h.

In this section, a single variable—the historical DCEFs—is selected as the input feature. Autocorrelation and partial autocorrelation analyses are performed to determine the optimal time window. The results show that the lagged values up to 240 time points (corresponding to 10 days) exhibit significant correlation with the current value. Therefore, the sampling time window covers the past 10 days of DCEF data, i.e., 240 consecutive hourly points (10 days $\times$ 24 h). For both the concrete manufacturing dataset and the glass manufacturing dataset, the dimension of a single input sample is 240×1 . The model takes these 240 historical points as input to predict the DCEF value for the next time step (one hour ahead). To fit the input structure of the MTCN framework, the feature matrix is directly used as a single-channel one-dimensional sequence without additional channel replication. The final input shape is (batch size, 240, 1).

The experimental data are divided into training, validation, and test sets. The training set and validation set contain 7008 samples, with the training set accounting for 80% of the data over the period from 1 January 2025 00:00 to 24 August 2025 14:00, and the validation set accounting for 20% over the period from 24 August 2025 15:00 to 19 October 2025 23:00. The test set contains 1752 time points, of which 240 h are excluded to prevent overly tight overlap between the training and test sets, which could otherwise inflate the reported accuracy. The effective test set therefore contains 1512 samples, covering the period from 30 October 2025 00:00 to 31 December 2025 23:00. Zero values in the DCEF sequences of the dataset are first replaced with missing values before model training. This step is necessary because a DCEF value of zero is physically implausible for the electricity users

considered in this study and would cause the MAPE denominator to become zero, leading to undefined or extremely large percentage errors. The resulting missing values are then filled using nearest-neighbor interpolation, which replaces each missing entry with the nearest valid observation along the time axis. This preserves the local temporal structure of the DCEF sequence while ensuring that the test set contains no zero values, so that the MAPE denominator is never zero.

To ensure a fair comparison across all models, the same sliding window function `create_sequences` (`data`, `seq_len` = 240, `pred_len` = 1) is applied to all models (MLP, BiGRU, Mamba, MTCN). With an input window of 240 h and a prediction step of 1 h, the first predictable sample of each model's test set starts from the 241st time point of the test segment. The first 240 points are used only as input history and are not counted as prediction targets; they are discarded uniformly for all models. Consequently, the test set size of every model equals the original length of the test segment minus 240, which guarantees an identical number of test samples across models. After inverse normalization, all models compute RMSE, MAE, MAPE, and R^2 on exactly the same set of test samples. In addition, the time column in each test result is truncated from the same starting point, so that the timestamps correspond one-to-one with the prediction targets. This procedure eliminates additional errors that would otherwise arise from temporal misalignment.

The concrete and glass manufacturing datasets are used in all experiments. The mini-batch size is set to 32 and the learning rate to 0.0001. For the comparison experiments, all baseline networks adopt their own most suitable training hyperparameters.

3.1. Concrete Manufacturing Company (Case I)

This study takes a concrete manufacturing enterprise as a typical high-energy-consuming industrial user, and its power load characteristics show clear continuity and periodic fluctuations. The DCEFs change with the variation in the power grid structure and production load, presenting a complex spatiotemporal pattern that reflects the multi-scale coupling characteristics of industrial carbon emissions. Due to the availability and verifiability of the data, this case is representative of the industry and suitable for model validation, thus serving as an ideal benchmark for evaluating the effectiveness of the proposed MTCN in the prediction of DCEFs.

To quantitatively evaluate the performance advantage of the proposed MTCN architecture in DCEF prediction accuracy, this study constructs ten benchmark comparison schemes, covering classic networks and frontier models. Key hyperparameters such as batch size and learning rate are consistent with those of MTCN, and the same preprocessing process and hardware environment are adopted to ensure the fairness and reproducibility of the performance comparison. The performance of the proposed MTCN framework is evaluated through the assessment of dynamic and convergence behaviors. Figure 5 shows the trend of the loss function and validation accuracy during the training process. As shown in Figure 5a, the training loss drops rapidly in the initial few epochs and then stabilizes, indicating an efficient and stable optimization process that avoids the common oscillations or divergence phenomena in complex time series models. Correspondingly, the validation accuracy curve in Figure 5b rises rapidly in the initial stage and remains at approximately 91.9% after the 30th epoch. The stability of the validation accuracy and the lack of a significant gap between the training set and validation set performance indicate that MTCN has good predictive performance. This convergence characteristic is crucial in the context of DCEF prediction, as the model must reliably track the continuous changes in power carbon intensity while avoiding overfitting to specific production rhythms or grid fluctuations in the data of concrete plants. Table 1 provides a comprehensive quantitative comparison of all 11 models, listing four standard regression indicators: RMSE, MAE,

R^2 , and MAPE. The proposed MTCN achieves a MAPE of 0.0438 on the test set, with a prediction accuracy of up to 95.62% (Figure 6 shows the predicted and actual values for the test set).

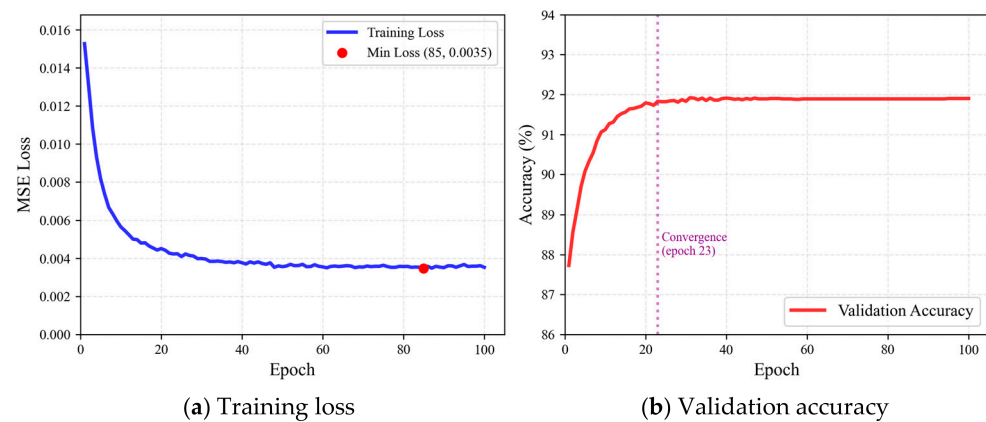


Figure 5. Convergence and generalization performance. (a) The training loss exhibits a rapid decline in the early stage and gradually flattens out, indicating effective optimization; (b) The validation accuracy improves correspondingly and maintains a stable performance around 91.9% after the 30th epoch, demonstrating robust generalization.

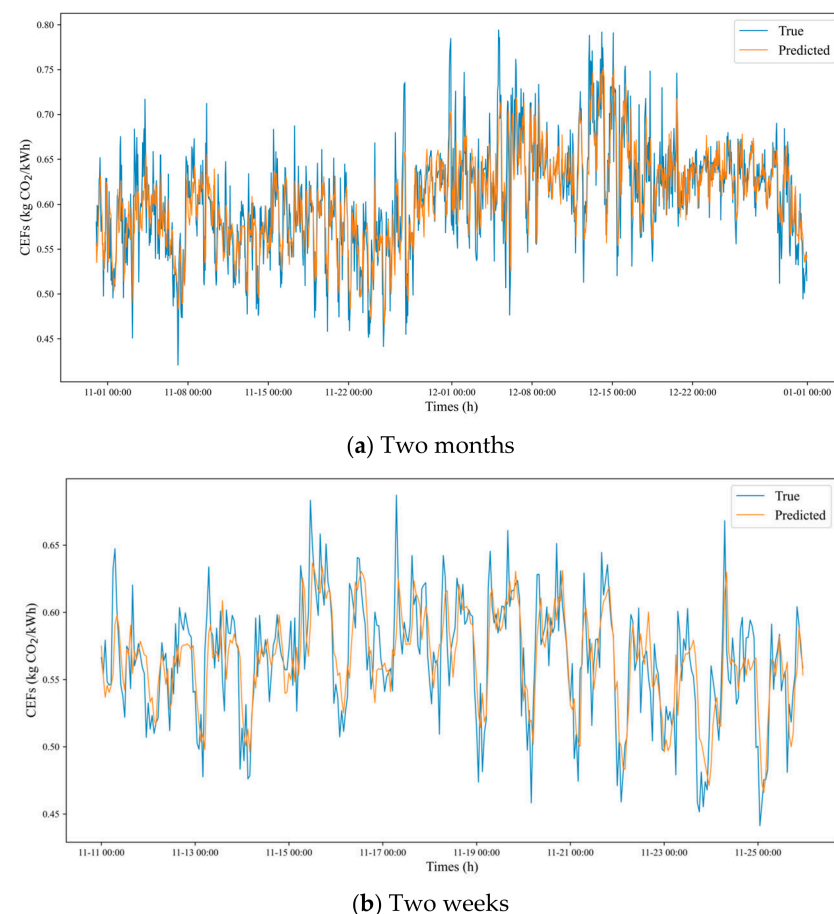


Figure 6. Comparison of actual values and predicted values in the testing set of the concrete industry. (a) The comparison over a period of two months; (b) The comparison over a period of two weeks.

Table 1 reports the MAPE mean $\pm$ standard deviation of each model over five independent random seeds. MTCN achieves the lowest MAPE of 0.0438 ± 0.0001 , followed by GRU and TCTS, XGBoost, TCN, RSAP-DFM, and KNN. To verify statistical significance, we

used the one-sided Wilcoxon signed-rank test. The results show that MTCN outperforms all baselines across all five seeds ($p < 0.05$), indicating a significant difference. Because the MAPE values of all models are highly consistent across the five seeds, the paired t-test fails due to near-zero variance of the differences. Therefore, we use the Wilcoxon test as the primary significance test. These results indicate that the improvement of MTCN in MAPE is statistically reliable.

The study shows that MTCN not only achieves state-of-the-art accuracy in DCEF prediction but also maintains excellent generalization performance when transitioning from controlled training data to real industrial operating conditions. The integrated visualization results and performance benchmarks provide a transparent and repeatable foundation for deploying this framework in concrete enterprises and other high-energy-consuming industries, ultimately facilitating more accurate carbon management and accelerating the goal of industrial carbon neutrality.

3.2. Glass Manufacturing Company (Case II)

The glass industry is another typical high-energy-consuming process industry in the building materials sector. This production process requires maintaining a continuous high temperature, resulting in a large power load and extremely strict requirements for power supply continuity. Unlike the concrete industry, the melting process in glass production has the characteristic of “non-interruption”; any power interruption could cause the melting furnace to solidify, resulting in severe equipment damage. Additionally, the unit product energy consumption and carbon emission intensity rank among the top in the building materials sub-sector. Moreover, glass production is influenced by market orders and seasonal demand, leading to more obvious fluctuations in the load curve. The dynamic carbon emission coefficient changes in a more complex way with the production rhythm and the carbon intensity of the power grid. This study uses a glass manufacturing enterprise as a second typical user case to verify the predictive ability and adaptability of the proposed MTCN in another high-energy-consuming, high-continuity, and high-carbon-emission industry. It further tests the predictive robustness of the model under different production modes and load characteristics.

The training dynamics of the MTCN framework on the glass manufacturing dataset further confirm its optimized stability and predictive ability. As shown in Figure 7a, the training loss drops sharply in the first 10 cycles and then converges steadily to a plateau level of approximately 0.0025, indicating that multi-scale time convolution can effectively capture complex load patterns and is not affected by gradient instability. Correspondingly, the validation accuracy curve in Figure 7b rises rapidly after the 30th cycle and stabilizes at 92.96%, maintaining this high level throughout the subsequent training process. This stable convergence behavior is particularly important for the glass industry, as its production process is inherently “non-interruptible”—any unexpected deviation in DCEF prediction could lead to incorrect decisions in the power-sensitive melting operation, thereby endangering furnace solidification and causing severe equipment damage. MTCN does not suffer from overfitting, and its validation accuracy remains consistently high, indicating that the model can reliably generalize to the periodic fluctuations and market order-driven load changes specific to glass manufacturing. On the actual test set, as shown in Figure 8, it can be seen that the predicted trend and values are almost consistent with the actual values.

Table 2 shows a comprehensive performance comparison of all 11 models using glass industry data. The proposed MTCN still achieved the lowest prediction error among all competitors, with an RMSE of 0.0360, MAE of 0.0271, and MAPE of 0.0446 (equivalent to a prediction accuracy of up to 95.56%). The results indicate that the fusion of multi-scale time features enables MTCN to continue outperforming traditional statistical methods

and advanced deep learning models when facing more significant periodic fluctuations and market-driven load changes in glass manufacturing. It is notable that the model maintains a low MAPE value in both the concrete and glass cases. Therefore, the second case study demonstrates that MTCN is applicable in industrial scenarios for predicting DCEFs, thereby enabling precise carbon management and facilitating the decarbonization path for all high-energy-consuming sectors in the building materials industry.

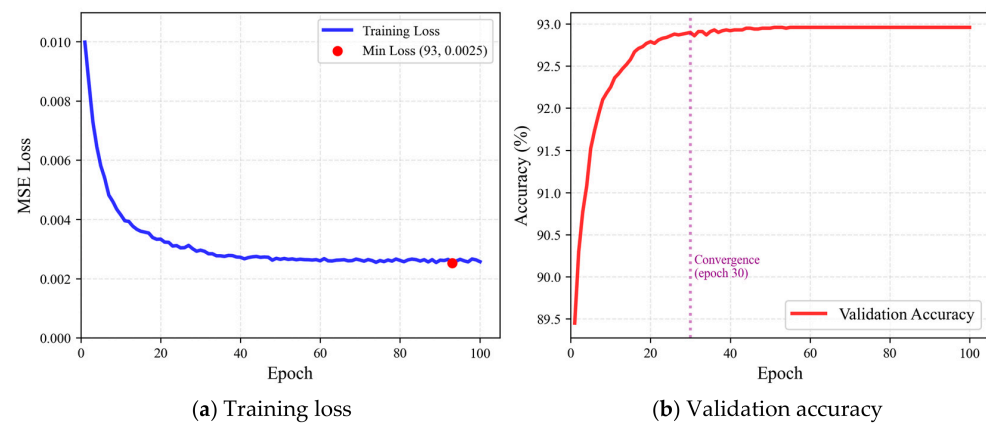


Figure 7. Training loss and test accuracy curves of the proposed model on the second dataset over 100 epochs. (a) The training loss decreases sharply in the first 10 epochs and gradually converges to approximately 0.0025; (b) The validation accuracy increases steadily and stabilizes at 92.96% after 30 epochs, demonstrating effective optimization and robust generalization.

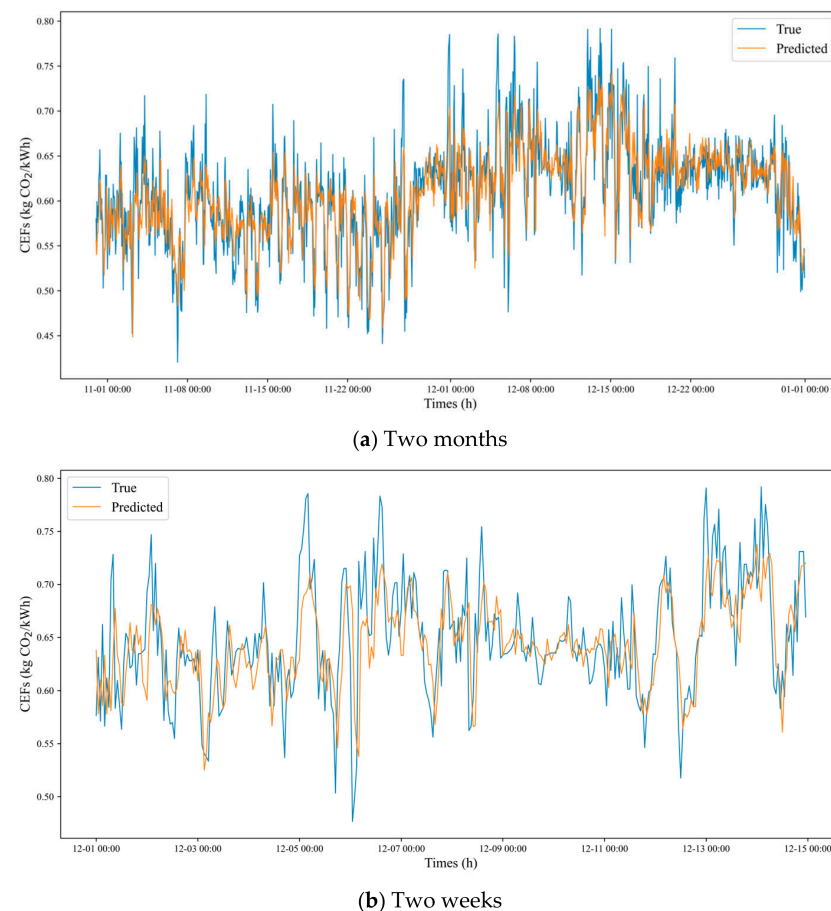


Figure 8. Comparison of actual values and predicted values in the testing set of the glass industry. (a) The comparison over a period of two months; (b) The comparison over a period of two weeks.

Table 2. Efficacy testing outcomes of models under glass manufacturing company.

Model Name	RMSE	MAE	R ²	MAPE
KNN	0.0495 ± 0.0002	0.0369 ± 0.0002	0.5188 ± 0.0032	0.0624 ± 0.0002
ARIMA	0.0366 ± 0.0001	0.0274 ± 0.0001	0.5867 ± 0.0017	0.0454 ± 0.0001
Random Forest	0.0370 ± 0.0002	0.0274 ± 0.0002	0.5782 ± 0.0034	0.0448 ± 0.0002
XGBoost	0.0390 ± 0.0002	0.0286 ± 0.0002	0.5319 ± 0.0028	0.0464 ± 0.0001
GRU	0.0362 ± 0.0001	0.0272 ± 0.0001	0.5958 ± 0.0015	0.0448 ± 0.0001
TCN	0.0363 ± 0.0002	0.0273 ± 0.0002	0.5931 ± 0.0026	0.0448 ± 0.0002
RSAP-DFM	0.0363 ± 0.0001	0.0272 ± 0.0002	0.5941 ± 0.0023	0.0449 ± 0.0002
TCTS	0.0365 ± 0.0001	0.0272 ± 0.0001	0.5888 ± 0.0013	0.0450 ± 0.0001
MLP	0.0370 ± 0.0002	0.0274 ± 0.0002	0.5782 ± 0.0027	0.0448 ± 0.0002
Mamba	0.0370 ± 0.0001	0.0274 ± 0.0001	0.5782 ± 0.0025	0.0448 ± 0.0001
MTCN (proposal)	0.0361 ± 0.0002	0.0269 ± 0.0002	0.6002 ± 0.0023	0.0444 ± 0.0002

The graphical and tabular evidence together demonstrate the capacity of the MTCN architecture to model DCEFs in an industrial scenario markedly different from concrete production. Figure 7 verifies that the model can be efficiently trained to convergence on glass industry data, learning the tighter coupling between production rhythms and electricity carbon intensity that arises from the melting furnace's continuous, temperature-critical operation. The quantitative results in Table 2 reveal that the fusion of multi-timescale temporal features enables MTCN to consistently outperform both traditional statistical approaches and advanced deep learning models, even when faced with the more pronounced periodic fluctuations and market-driven load variability of glass manufacturing. Notably, the model maintains low MAPE values across both the concrete and glass cases, underscoring its generalization capability across heterogeneous high-energy-consumption processes. This second case study thus confirms that MTCN is not merely tailored to a single industrial profile; rather, it provides a robust, transferable framework for DCEF forecasting, supporting precise carbon management and facilitating the decarbonization pathway of diverse energy-intensive sectors within the building materials industry. As shown in Table 2, for the glass manufacturing company dataset, the MAPE of each model is reported as the mean ± standard deviation over five independent random seeds. MTCN obtains the lowest MAPE. As in Case 1, we conducted a one-sided Wilcoxon signed-rank test and found that MTCN outperformed all baseline models across all five seeds ($p < 0.05$). These results are consistent with the conclusion for the concrete dataset in Table 1.

3.3. Discussion

The proposed MTCN was quantitatively evaluated on two real-world DCEF datasets representing two energy-intensive industries, namely concrete manufacturing (User 1) and glass manufacturing (User 2). As summarized in Table 3, MTCN achieved consistently low prediction errors on both datasets. For User 1, the RMSE, MAE, and MAPE were 0.0354, 0.0265, and 0.0438, respectively, whereas for User 2, the corresponding values were 0.0360, 0.0271, and 0.0446. The comparable performance across the two datasets indicates that the proposed framework can maintain stable forecasting accuracy under different industrial operating patterns and load characteristics. These results suggest that MTCN has the potential to provide reliable predictive information for demand-side carbon management in energy-intensive industries.

Table 3. Efficacy testing outcomes of MTCN under two datasets.

Data	RMSE	MAE	R ²	MAPE
User 1	0.0354 ± 0.0001	0.0265 ± 0.0001	0.6249 ± 0.0034	0.0438 ± 0.0001
User 2	0.0361 ± 0.0002	0.0269 ± 0.0002	0.6002 ± 0.0023	0.0444 ± 0.0002

To further examine the sensitivity of MTCN to training configurations, the effects of the historical window length, learning rate, and mini-batch size were investigated, as shown in Figures 9–11. The historical window length had a noticeable influence on forecasting performance. Among the tested settings, a window of 240 h yielded the best overall performance, which is consistent with the autocorrelation analysis presented in Section 2.2. The learning rate mainly affected the convergence trajectory and final prediction error. A relatively large learning rate accelerated the initial decrease in training loss but resulted in greater fluctuations in the validation metrics, whereas an excessively small learning rate produced smoother convergence at the cost of substantially longer training time and only marginal performance improvement. A learning rate of 1×10^{-4} therefore provided a favorable balance between convergence stability and prediction accuracy. The mini-batch size also influenced the training dynamics. Smaller batches introduced stronger stochastic gradient variations and slightly improved generalization in some cases, whereas larger batches led to more stable and faster convergence. Based on these observations, a mini-batch size of 32 was adopted in the final experiments.

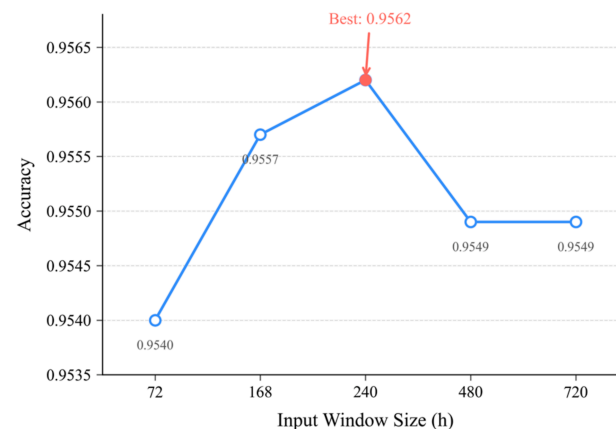


Figure 9. Values of training metrics for MTCN networks at different window sizes.

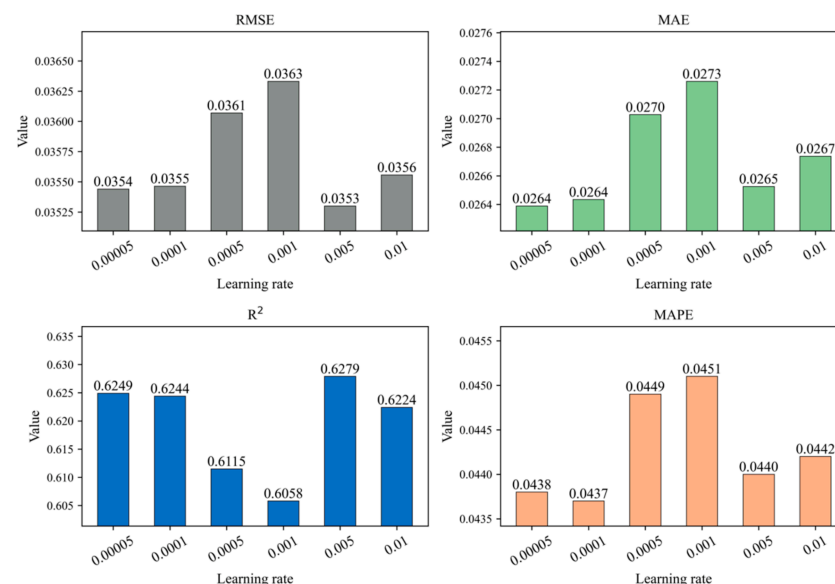


Figure 10. Values of training metrics for MTCN networks at different learning rates.

The computational cost of MTCN was also comparable to that of the other deep-learning-based methods. Under the same hardware environment, the training time of MTCN was 90.57 s, compared with 87.58 s for GRU, 89.96 s for TCN, 86.42 s for MLP,

88.14 s for Mamba, 90.65 s for RSAP-DFM, and 93.27 s for TCTS; XGBoost and Random Forest required 92.83 s and 94.71 s, respectively. In contrast, ARIMA and KNN required substantially longer computation times of 9 min 56.5 s and 10 min 23.6 s, respectively. These results indicate that the improved forecasting performance of MTCN does not introduce a substantial additional computational burden relative to the other deep learning models considered in this study.

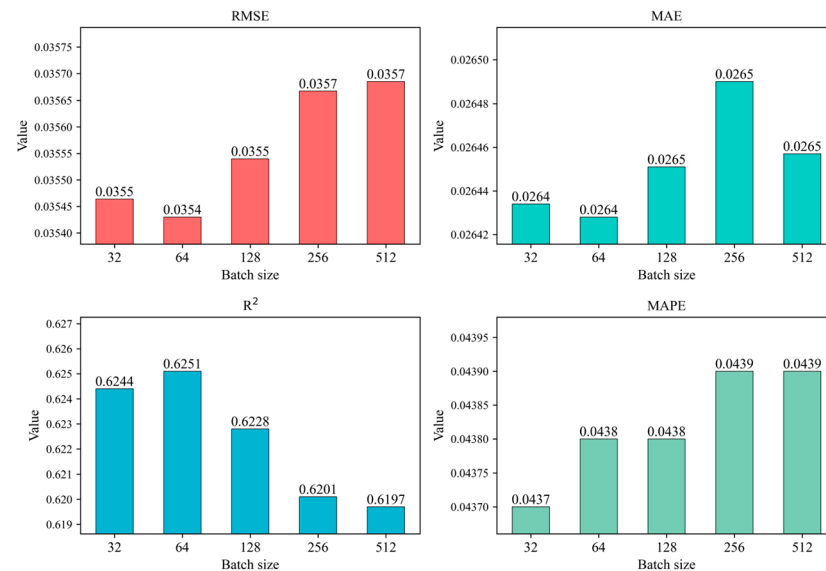


Figure 11. Training metrics values for MTCN networks with different mini-batch sizes.

Despite its favorable overall performance, MTCN exhibits reduced prediction capability during extreme fluctuations in DCEF. As shown in Table 4, the overall R^2 on the concrete manufacturing dataset is 0.6249. In relatively stable subsets, including weekday, weekend, autumn, and winter periods, the R^2 values range from approximately 0.47 to 0.66. In contrast, substantially lower values are observed for the peak, near-zero, and ramp/abrupt change segments, with R^2 values of -0.0611 , -6.9945 , and 0.0890 , respectively. Peak DCEF is defined as samples with true DCEF $\geq$ the 75th percentile of the training set DCEF distribution. Near-zero DCEF is defined as samples with true DCEF $\leq$ the 5th percentile of the training set DCEF distribution. Ramp/abrupt changes are defined as samples with $|\Delta \text{DCEF}| \geq$ the 90th percentile of the absolute first differences computed on the training set. The three segments may overlap, and samples are counted in each segment to which they belong. These results indicate that the relatively low overall R^2 is mainly associated with difficult operating segments characterized by abrupt fluctuations, low variance, low signal-to-noise ratios, or limited sample sizes. In particular, the near-zero and ramp/abrupt change segments contain only 67 and 138 samples, respectively, making the variance-based R^2 metric highly sensitive to individual prediction errors and outliers.

Nevertheless, the deterioration in R^2 does not necessarily imply uncontrolled absolute prediction errors. For example, the MAPE of the ramp/abrupt change segment is 10.74%, while the overall MAPE is 4.38%. Similarly, the peak DCEF segment achieves a MAPE of 4.08%. These results indicate that MTCN maintains bounded prediction errors even when its ability to explain short-term variance decreases. Time-period assessment also indicates that the RMSE of the model in the climbing/sudden change periods is twice the overall level, and the R^2 in the near-zero periods is negative, which shows that the model's prediction reliability is limited in extreme fluctuation periods. Therefore, MAE, RMSE, MAPE, and R^2 should be interpreted jointly when evaluating the forecasting performance of DCEF models, particularly under highly volatile conditions. The current model performs

more reliably under regular operating conditions than during abrupt carbon intensity transitions, which remains an important limitation of the present study.

Table 4. Segment-wise performance evaluation of MTCN on the concrete manufacturing company dataset.

Segment	N	RMSE	MAE	R ²	MAPE
All test samples	1512	0.0354	0.0265	0.6249	0.0438
Peak DCEF	692	0.0375	0.0273	−0.0611	0.0408
Ramp/abrupt changes	138	0.0724	0.0649	0.089	0.1074
Near-zero DCEF	67	0.0541	0.0434	−6.9945	0.0906
Weekday	1080	0.0352	0.0262	0.5887	0.0438
Weekend	432	0.0369	0.027	0.6642	0.0441
Autumn	768	0.0344	0.0261	0.5378	0.0453
Winter	744	0.037	0.0268	0.472	0.0424

From an application perspective, accurate short-term DCEF forecasts can provide useful decision support for carbon-aware electricity management. For energy-intensive enterprises, predicted carbon intensity trajectories may assist operators in shifting flexible electricity-consuming processes toward periods with relatively low expected carbon intensity, while maintaining continuously operated core processes when necessary. Such forecasts can therefore serve as an information input for subsequent low-carbon scheduling, carbon accounting, and carbon management decisions. However, the present study focuses on DCEF forecasting rather than directly optimizing industrial production schedules; therefore, the actual emission reduction benefits of forecast-driven scheduling require further investigation.

Another limitation is that the current MTCN is a purely data-driven autoregressive forecasting model that uses only historical DCEF values as input and does not explicitly incorporate physical constraints or external explanatory variables. This limitation becomes particularly evident during abrupt changes in grid carbon intensity. Recent physics-informed modeling studies provide a possible direction for further improvement. For example, He et al. [32] incorporated physical constraints into a neural operator for dynamic energy flow calculation. Inspired by this idea, future work may integrate prior knowledge such as carbon flow allocation relationships, nodal carbon potential balance constraints, generation-side operating information, and other physically meaningful variables into the forecasting framework. Such information could be incorporated through additional input features, regularization terms, or physics-informed loss functions to improve the model's responsiveness to operating conditions that are poorly represented in historical data. Further validation on a broader range of industrial sectors and electricity consumption patterns is also needed to assess the generalizability of the proposed framework.

4. Conclusions

This study has constructed a hybrid deep learning framework (MTCN) for predicting dynamic electricity carbon factors. The experimental results on two real industrial datasets show that the model, capable of fully capturing the long-term evolutionary trend and short-term random fluctuations of carbon emission factors, achieved prediction accuracies of 95.62% and 95.56%, respectively. This high-precision prediction capability can effectively support high-energy-consuming enterprises in flexibly adjusting their electricity plans and optimizing carbon trading strategies based on the predicted carbon intensity signals. It also provides reliable information support for low-carbon dispatch of the power grid, facilitating the achievement of deep decarbonization. The main contributions and innovations are fourfold:

- (1) The combination of Mamba and TCN with a learnable fusion mechanism for DCEF forecasting effectively addresses both global sequence modeling and local fluctuation capture;
- (2) Through correlation analysis, the periodic and trend characteristics of DCEF data are utilized to determine the appropriate time series input window, enabling the model to fully extract the temporal information contained in the historical carbon factor sequence;
- (3) The introduction of a MAPE metric to mitigates the distortion caused by near-zero DCEF values, ensuring fair and stable performance assessment across datasets;
- (4) Extensive validation on two real-world hourly datasets demonstrates that the proposed model achieves high prediction accuracies of 95.62% and 95.56%, respectively. This provides reliable decision support for real-time carbon management and power system decarbonization.

Author Contributions: Conceptualization, C.L.; methodology, C.L.; validation, C.L.; formal analysis, C.L. and S.G.; investigation, C.L. and S.G.; resources, C.L.; data curation, C.L. and S.G.; writing—original draft preparation, C.L.; writing—review and editing, C.L. and X.Z.; supervision, X.Z.; project administration, X.Z.; funding acquisition, X.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Due to confidentiality agreements with the data providers, the original data cannot be released.

Conflicts of Interest: The authors declare no conflicts of interest.

Nomenclature

The following Variables and abbreviations are used in this manuscript:

Variables

x_t	Carbon emission factor value at time step t
X	Carbon emission factor time series
$\hat{x}_t$	Predicted carbon emission factor value
L	Historical window length
H	Prediction horizon
k	Time lag for autocorrelation
$p(k)$	Autocorrelation function at lag k
h_t	Hidden state at time step t
$\bar{A}_t, \bar{B}_t$	Input-dependent discretized state transition matrices
α	Adaptive gate weight for feature fusion
τ	Lag time scale
u_{RMSE}	Root mean square error
u_{MAE}	Mean absolute error
u_{R^2}	Coefficient of determination
u_{MAPE}	Mean absolute percentage error

Abbreviation

CEF	Carbon emission factor
DCEF	Dynamic carbon emission factor
DCP	Dynamic electricity carbon factor prediction model
MTCN	Mamba–temporal convolutional network fusion
MLP	Multilayer Perceptron
SSM	State space model
ACF	Autocorrelation function

RMSE	Root mean square error
MAE	Mean absolute error
MAPE	Mean absolute percentage error
R ²	Coefficient of determination

References

1. State Council of the People's Republic of China. Notice on the Issuance of the Metrology Development Plan (2021–2035). Guo Fa [2021] No. 37, 31 December 2021. Available online: https://www.gov.cn/zhengce/content/2022-01/28/content_5670947.htm (accessed on 13 May 2026).
2. National Development and Reform Commission; National Bureau of Statistics; Ministry of Ecology and Environment of the People's Republic of China. Notice on the Issuance of the Implementation Plan for Accelerating the Establishment of a Unified and Standardized Carbon Emission Statistical Accounting System (Fa Gai Huan Zi [2022] No. 622). 2022. Available online: https://www.gov.cn/zhengce/zhengceku/2022-08/19/content_5706074.htm (accessed on 13 May 2026).
3. State Administration for Market Regulation; National Development and Reform Commission; Ministry of Industry and Information Technology; Ministry of Natural Resources; Ministry of Ecology and Environment; Ministry of Housing and Urban-Rural Development; Ministry of Transport; China Meteorological Administration; National Forestry and Grassland Administration. Notice on the Issuance of the Implementation Plan for Establishing and Improving the Standard and Metrology System for Carbon Peak and Carbon Neutrality (Guo Shi Jian Ji Liang Fa [2022] No. 92). 2022. Available online: https://www.gov.cn/zhengce/zhengceku/2022-11/01/content_5723071.htm (accessed on 13 May 2026).
4. Hua, H.; Shi, J.; Chen, X.; Qin, Y.; Wang, B.; Yu, K.; Naidoo, P. Carbon Emission Flow Based Energy Routing Strategy in Energy Internet. *IEEE Trans. Ind. Inform.* **2024**, *20*, 3974–3985. [\[CrossRef\]](#)
5. Zhang, L.; Ye, Y.; Ren, X.; Qiu, X.; Sun, Z.; Zhou, W.; Han, D. A Hybrid Time-Series Simulation Framework for Provincial Carbon Emissions Using Multi-Factor Decomposition and Deep Learning. *Sustainability* **2026**, *18*, 3108. [\[CrossRef\]](#)
6. Liu, J.; Liu, G.; Zhao, H.; Zhao, J.; Qiu, J.; Dong, Z.Y. A Real-Time Carbon Emission Estimation Framework for Industrial Parks with Non-Intrusive Load Monitoring. *Sustain. Energy Technol. Assess.* **2023**, *60*, 103482. [\[CrossRef\]](#)
7. Jin, Y.; Xu, S.; Guo, Q.; Xiong, N. Analysis of the Impact of Electricity Emission Factor Selection on High Energy-Consuming Industries under the National Carbon Market Expansion. In *2024 4th International Conference on Energy, Power and Electrical Engineering (EPEE)*; IEEE: Piscataway, NJ, USA, 2024; pp. 1050–1053. [\[CrossRef\]](#)
8. He, H.; Zhou, S.; Zhang, L.; Zhao, W.; Xiao, X. Dynamic Accounting Model and Method for Carbon Emissions on the Power Grid Side. *Energies* **2023**, *16*, 5016. [\[CrossRef\]](#)
9. Lin, Y.; Lu, Z.; Qin, H.; Liu, Z.; Li, X.; Li, G.; Wang, Q.; Wang, P.; Yang, J.; Yao, S. Multi-Objective Optimal Dispatch of Integrated Energy Systems Guided by Dynamic Carbon Emission Factors. *ACS Omega* **2026**, *11*, 12308–12320. [\[CrossRef\]](#)
10. Zhang, Y.; Ge, X.; Li, M.; Li, N.; Wang, F.; Wang, L.; Sun, Q. Demand Response Potential Day-Ahead Forecasting Approach Based on LSSA-BPNN Considering the Electricity-Carbon Coupling Incentive Effects. *IEEE Trans. Ind. Appl.* **2024**, *60*, 4505–4516. [\[CrossRef\]](#)
11. Yan, Z.; Li, Y.; Liu, Y.; Zhang, S.; Du, E.; Zhang, N.; Lu, Z.; Leng, Y. A Short-Term Forecast Method for Electricity Carbon Emission Factor with High Temporal and Spatial Resolution. *J. Clean. Prod.* **2025**, *533*, 147018. [\[CrossRef\]](#)
12. Yang, H.; Tang, A.; Zhang, K.; Wang, T.; Zhou, H.; Li, X.; Li, H.; Wang, W.; Peng, J. Regional Power Grid Carbon Emission Change Risk Assessment Based on Dynamic Carbon Emission Factors. *CSEE J. Power Energy Syst.* **2026**, *12*, 437–447. [\[CrossRef\]](#)
13. Zhang, P.; Kong, X.; Duan, M.; Zhuang, Z.; Fang, K.; Zhou, H. Low-Carbon Optimization Scheduling for Industrial Parks Based on Dynamic Carbon Emission Prediction. In *2025 44th Chinese Control Conference (CCC)*; IEEE: Piscataway, NJ, USA, 2025; pp. 6476–6481. [\[CrossRef\]](#)
14. Jiao, M.; Yang, M.; Li, M.; Zhang, Y.; Li, P. Enhanced Dispatch Model for Park-Level Integrated Energy System Considering Electricity-Carbon Collaboration. *Electr. Power Syst. Res.* **2025**, *246*, 111674. [\[CrossRef\]](#)
15. Radovanović, A.; Koningstein, R.; Schneider, I.; Chen, B.; Duarte, A.; Roy, B.; Xiao, D.; Haridasan, M.; Hung, P.; Care, N.; et al. Carbon-Aware Computing for Datacenters. *IEEE Trans. Power Syst.* **2023**, *38*, 1270–1280. [\[CrossRef\]](#)
16. Liu, J.; Zhao, H.; Wang, S.; Liu, G.; Zhao, J.; Dong, Z.Y. Real-Time Emission and Cost Estimation Based on Unit-Level Dynamic Carbon Emission Factor. *Energy Convers. Econ.* **2023**, *4*, 47–60. [\[CrossRef\]](#)
17. Kotsompolis, G.; Cheilas, P.; Konstantakis, K.N.; Sfakianakis, E.; Goutte, S.; Michaelides, P.G. Smart Forecasting of Carbon Prices Using Machine Learning and Neural Networks: When ARIMA Meets XGBoost and LSTM. *J. Forecast.* **2026**, *45*, 47–60. [\[CrossRef\]](#)
18. Yu, H.; Yang, Y.; Li, B.; Liu, B.; Guo, Y.; Wang, Y.; Guo, Z.; Meng, R. Research on the community electric carbon emission prediction considering the dynamic emission coefficient of power system. *Sci. Rep.* **2023**, *13*, 5568. [\[CrossRef\]](#)
19. Cai, M.; Huang, L.; Zhang, Y.; Liu, C.; Li, C. Day-Ahead Forecast of Carbon Emission Factor Based on Long and Short-Term Memory Networks. In *2023 5th Asia Energy and Electrical Engineering Symposium (AEEES)*; IEEE: Piscataway, NJ, USA, 2023; pp. 1568–1573. [\[CrossRef\]](#)

20. Nagendra, B.; Singh, G. Comparing ARIMA, Linear Regression, Random Forest, and LSTM for Time Series Forecasting: A Study on Item Stock Predictions. In *2023 4th IEEE Global Conference for Advancement in Technology (GCAT)*; IEEE: Piscataway, NJ, USA, 2023; pp. 1–8. [\[CrossRef\]](#)
21. Siami-Namini, S.; Tavakoli, N.; Namin, A.S. A Comparative Analysis of Forecasting Financial Time Series Using ARIMA, LSTM, and BiLSTM. *arXiv* **2019**, arXiv:1911.09512.
22. Wu, H.; Xu, J.; Wang, J.; Long, M. Autoformer: Decomposition Transformers with Auto-Correlation for Long-Term Series Forecasting. In *Advances in Neural Information Processing Systems*; Ranzato, M., Ed.; Curran Associates, Inc.: Red Hook, NY, USA, 2021; Volume 34, pp. 22419–22430. Available online: https://proceedings.neurips.cc/paper_files/paper/2021/file/bcc0d400288793e8bdcd7c19a8ac0c2b-Paper.pdf (accessed on 13 May 2026).
23. Zhou, H.; Zhang, S.; Peng, J.; Zhang, S.; Li, J.; Xiong, H.; Zhang, W. Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting. *arXiv* **2021**, arXiv:2012.07436. [\[CrossRef\]](#)
24. Maji, D.; Shenoy, P.; Sitaraman, R.K. CarbonCast: Multi-Day Forecasting of Grid Carbon Intensity. In *Proceedings of the 9th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation, Boston, MA, USA, 9–10 November 2022*; ACM Digital Library: New York, NY, USA, 2022; pp. 198–207. [\[CrossRef\]](#)
25. Guan, K.; Tan, D.; Zhong, Q.; Li, P.; Zhong, J.; Cao, W.; Zhang, H.; Liang, C. Prediction of Daily Electricity Carbon Emission Factors for High Energy-Consuming Enterprises Based on Temporal Feature Optimization and Hybrid Intelligent Algorithms. *AIP Adv.* **2025**, *15*, 085314. [\[CrossRef\]](#)
26. Chen, S.; Zheng, Y.; Wen, X.; Peng, Z.; Li, C. Research on Real-time Carbon Emission Factor Prediction Method for Load Nodes of Power System based on Gated Recurrent Cell Network. In *Proceedings of the 2023 3rd Power System and Green Energy Conference (PSGEC), Shanghai, China, 24–26 August 2023*; IEEE: Piscataway, NJ, USA, 2023; pp. 683–688. [\[CrossRef\]](#)
27. Gu, A.; Goel, K.; Ré, C. Efficiently Modeling Long Sequences with Structured State Spaces. *arXiv* **2022**, arXiv:2111.00396. [\[CrossRef\]](#)
28. Uhrig, M.N.; Vignolo, L.D.; Müller, O.V. Multivariate Forecasting of Energy Demand Using Recurrent Neural Networks. *Adv. Eng. Inform.* **2026**, *71*, 104419. [\[CrossRef\]](#)
29. Yu, G.; Cui, Y.; Guo, S.; Shan, R.; Liu, Y.; Zhao, L.; Wang, Y.; Li, J.; Wang, Z. Physics-informed Sim-to-Real transfer framework for lithium-ion battery state of health estimation via Mamba architecture. *J. Energy Storage* **2026**, *181*, 124487. [\[CrossRef\]](#)
30. Bukhari, S.B.A.; Albalawi, H.; Wadood, A.; Alatwi, A.M. Deep learning-driven fault detection and classification in microgrids using Temporal Convolutional Network. *Comput. Electr. Eng.* **2026**, *129*, 110777. [\[CrossRef\]](#)
31. Saravana, M.K.; Roopa, M.S.; Arunalatha, J.S.; Venugopal, K.R. Transformers for Multivariate Time Series Forecasting: Comprehensive Analysis, Challenges, Research Opportunities, and Future Prospects. *IEEE Access* **2026**, *14*, 11424–11457. [\[CrossRef\]](#)
32. He, S.; Zhang, S.; Li, Y.; Gu, W.; Chung, C.-Y. Physics-informed neural network for dynamic energy flow calculation in integrated electricity and gas systems. *Appl. Energy* **2026**, *407*, 127374. [\[CrossRef\]](#)

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.