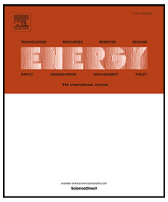
Contents lists available at ScienceDirect

Energy

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



MTRCL: A spatio-temporal multi-source data fusion method for enhancing short-term wind speed forecast capability of numerical weather prediction

Xiaolin Ju [a,b], Yiming Yue [a], Meng Xu [a,*], Xiang Chen [a], Gongzheng Lu [b,c,d,**]

[a] *School of Artificial Intelligence and Computer Science, Nantong University, Nantong, 226019, China*
[b] *State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, 210008, China*
[c] *Computing Science and Artificial Intelligence College, Suzhou City University, Suzhou, 215104, China*
[d] *Suzhou Key Lab of Multi-modal Data Fusion and Intelligent Healthcare, Suzhou City University, Suzhou, 215104, China*

ARTICLE INFO

Dataset link: https://github.com/CSNTU/MTRCL

Keywords:
Wind speed prediction
Multi-source data fusion
Time embedding
Attention mechanism
Wind energy resources

ABSTRACT

Accurate wind speed forecasting plays a pivotal role in optimizing wind power generation efficiency and advancing wind energy resource development. However, numerical weather prediction relies on global datasets and complex mathematical models, posing challenges in capturing subtle weather variations. Additionally, factors such as surface, land–sea thermal conditions, and terrain complexity further influence forecast accuracy. Existing methods generally suffer from a single data source dependency and an insufficient ability to extract spatiotemporal features. This study proposes MTRCL, a novel end-to-end approach developed to address the above challenges by integrating multi-source data fusion, time-embedded Residual Network (ResNet) with an improved Convolutional Block Attention Module (CBAM), and time-embedded Liquid Time-Constant Networks (LTCs). The proposed method leverages the time embedding (TE) technique to encode temporal information, thereby enhancing the model's capacity to capture temporal dependencies. Additionally, an improved channel-spatial-global (CBAM) three-level attention mechanism is incorporated into the ResNet framework, enabling the model to identify correlations between spatiotemporal features effectively. Finally, the efficacy of the proposed method is validated using the ERA5 hourly data on a single level. The experimental results show that the proposed model increases wind speed forecast accuracy, defined as the Fraction of Absolute Error (FA), representing the percentage of samples where the absolute wind speed error does not exceed 1 m/s, from 68.36% to 86.64%, compared to ECMWF forecast data. This improvement highlights the effectiveness of MTRCL in leveraging multi-source data to address the challenges of wind speed prediction in complex terrains and supports optimization of wind energy resource development.

1. Introduction

In recent years, wind energy has emerged as a crucial source of clean and renewable power, driven by the dual forces of the global energy transition and technological advancements [1,2]. Despite its growing significance, the inherent randomness of wind power generation and its inability to be effectively stored introduce substantial uncertainty in energy supply. Furthermore, within the low-altitude economic sector, the presence of strong convection and complex meteorological conditions in the low-altitude airspace poses a serious challenge to the flight safety of low-altitude aircraft. Consequently, accurate short-term wind speed forecasting is indispensable for optimizing the dispatching and operation of wind power systems and ensuring the safety of low-altitude aviation [3,4]. Currently, numerical weather prediction (NWP) models remain the primary approach for forecasting various meteorological variables. However, the chaotic nature and inherent uncertainties of the atmospheric system limit the capacity of numerical prediction models to fully capture the intricacies of the real atmospheric state, leading to varying degrees of deviations in forecast results. As a result, addressing the inaccuracies in NWP-based meteorological predictions has become an urgent issue to be addressed [5,6].

Error correction methods for Numerical Weather Prediction (NWP) products can be broadly categorized into two major approaches: traditional statistical techniques and machine learning-based methodologies. Traditional statistical techniques for correcting NWP outputs

* Corresponding author.
** Corresponding author at: State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, 210008, China.
E-mail addresses: Ju.xl@ntu.edu.cn (X. Ju), Yueym@stmail.ntu.edu.cn (Y. Yue), Xumeng@ntu.edu.cn (M. Xu), Xchencs@ntu.edu.cn (X. Chen), lugz@szcu.edu.cn (G. Lu).

https://doi.org/10.1016/j.energy.2025.139728
Received 26 July 2025; Received in revised form 26 November 2025; Accepted 17 December 2025
Available online 22 December 2025
0360-5442/

Nomenclature

NWP	Numerical Weather Prediction	TIGGE	THORPEX Interactive Grand Global Ensemble
RNNs	Recurrent Neural Networks	FA	Fraction of Absolute Error
LSTM	Long Short-Term Memory Network	RMSE	Root Mean Square Error
GNNs	Graph Neural Networks	MAE	Mean Absolute Error
CNNs	Convolutional Neural Networks	rRMSE	Relative Root Mean Square Error
GRU	Gate Recurrent Unit	rMAE	Relative Mean Absolute Error
ResNet	Residual Network	R	Correlation Coefficient
TE	Time Embedding	MAPE	Mean Absolute Percentage Error
CBAM	Convolutional Block Attention Module	CatBoost	Categorical Boosting
LTCs	Liquid Time-Constant Networks	ConvLSTM	Convolutional Long Short-Term Memory Network
ODEs	Ordinary Differential Equations	WRF	Weather Research and Forecasting Model
RK4	Fourth-Order Runge–Kutta Method	VMD	Variational Mode Decomposition
MLP	Multilayer Perceptron	XGBoost	eXtreme Gradient Boosting
ECMWF	European Centre for Medium-Range Weather Forecasts	TC-ResNet	Time-embedded ResNet with CBAM
TE-LTCs	Time-embedded Liquid Time-Constant Networks with RK4 Integration	RFE	Recursive Feature Elimination
C-ResNet	Residual Network with CBAM	STDGN	Spatio-temporal Data Generation Network
BPNN	Back Propagation Neural Network	DM	Diebold–Mariano

encompass a wide range of techniques, including time series analysis, Gaussian statistical models, linear regression, Kalman filtering, and nonlinear or non-parametric algorithms such as autocorrelation analysis [7,8]. Machine learning techniques, on the other hand, focus on identifying patterns and relationships within NWP data, as well as historical wind speed records to facilitate accurate predictions of future wind speeds [9]. These techniques are highly adaptable and data-driven, making them particularly suitable for handling large-scale datasets and complex nonlinear dependencies. Standard machine learning models include the Recurrent Neural Network (RNN) [10], Long Short-Term Memory Network (LSTM) [11], and Graph Neural Networks (GNNs) [12].

Traditional correction methods for numerical weather prediction (NWP) have been extensively studied, with several innovative approaches proposed to enhance prediction accuracy. Devis et al. [13] proposed a statistical regression-based downscaling technique. Their findings highlighted that incorporating temperature data significantly improved prediction accuracy. Similarly, Pearre and Swan [14] demonstrated that statistical correction techniques could reduce wind speed prediction errors within 24 h by 20%–25%. Zhao et al. [15] developed a single-day probabilistic wind speed prediction model to address inherent biases in numerical weather predictions. Homleid [16] proposed the use of the Kalman filtering, which effectively mitigates the systematic errors in the forecasts of numerical weather prediction models. Peng et al. [17] advanced the field further by introducing a pattern anomaly integral prediction correction method. Liu et al. [18] proposed a modified Taylor Kriging time series forecasting method, adapted for wind speed prediction. Fawad et al. [19] proposed a method combining multiparameter probability distributions with L-moment estimation, demonstrating its value in extreme wind speed analysis.

With the development of computational technology, machine learning has emerged as a pivotal tool in mitigating prediction biases in numerical models. However, wind speed data, influenced by complex meteorological conditions and topographical factors, frequently exhibit pronounced spatiotemporal coupling characteristics and non-stationary features. These complexities pose significant challenges for single-model architectures. To address these challenges, researchers have proposed machine learning-driven multi-model fusion prediction frameworks. Comparative studies have consistently demonstrated that hybrid models significantly outperform single-model architectures in terms of prediction efficiency [20–22]. Specific advancements in this domain include the work of Li et al. [23], who improved wind speed prediction accuracy by integrating an improved Hidden Markov Model with Fuzzy C-Means clustering algorithms. Zhou et al. [24] proposed a VMD-PCA-RF robust correction framework, demonstrating stable and high-precision prediction capabilities in annual scenarios. Empirical studies by Sun et al. [25] revealed that random forest and support vector machine models constructed using direct strategies outperform traditional statistical methods and numerical bias correction techniques. Wang et al. [26] employed random forest algorithms to correct wind forecasts derived from WRF products. Moreover, Yang et al. [27] proposed a two-stage correction prediction method for wind power. Zhang et al. [28] proposed a hybrid model based on VMD-WT and PCA-BP-RBF neural networks, achieving notable improvements in short-term wind speed prediction accuracy.

Deep learning, a specialized branch of machine learning, facilitates representation learning through multi-layer neural networks, enabling advanced pattern recognition and problem-solving capabilities. In recent years, its exceptional nonlinear fitting and feature fusion abilities have led to widespread applications in meteorology. Kim et al. [29] proposed a spatiotemporal neural network for prediction result correction, which achieves ideal correction effects. Han et al. [30] constructed a CUnet model to correct the wind speed and direction predictions for North China by the ECMWF global model. Gong et al. [31] proposed a hybrid model, VMD-PE-FCGRU, designed explicitly for short-term offshore wind speed prediction, whose prediction accuracy is significantly improved compared with traditional models. Xu et al. [32] constructed the WRF-VMD-PCA-LSTM model, which effectively enhanced multi-step wind speed prediction performance. Luo et al. [33] proposed an enhanced Stacked Extreme Learning Machine method. Zhang et al. [34] proposed a wind speed prediction model, ST-DFNet, which exhibits superior predictive accuracy across various time scales.

Despite significant advancements in the two categories of correction methods, most models struggle to simultaneously balance the correlations among temporal, spatial, and elemental features during feature extraction. This limitation prevents them from comprehensively capturing the intricate relationships within the data. Additionally, existing models often overlook the influence of complex terrain on near-surface atmospheric dynamics, which plays a critical role in meteorological phenomena. Furthermore, feature fusion across multi-source data remains a persistent challenge, as integrating diverse and relevant information from various data sources proves to be a complex and demanding task.

This study focuses on the region spanning 35.13°–47°N and 103°–126.88°E, which serves as the core development zone of the national large-scale “Three-North” wind power base. It possesses abundant and high-density wind energy resources, holding significant strategic value for ensuring national energy security and supporting the “dual carbon”

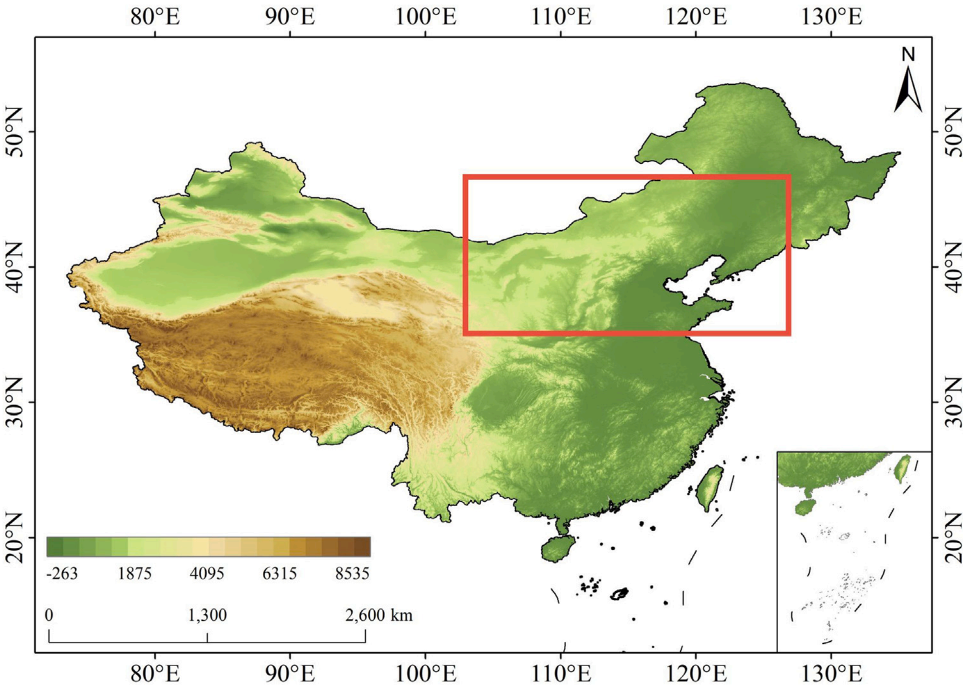


Fig. 1. The geographical distribution and topographic height in China and the study area of this paper (red solid box: 35.13°–47°N, 103°–126.88°E).

transition. However, this region spans the northeastern edge of the Qinghai-Tibet Plateau, the Loess Plateau, the Mongolian Plateau, and the North China Plain, with significant differences in terrain altitude, encompassing various geomorphic units, including gobi deserts, mountains, and river valleys. The local circulation phenomena caused by the complex terrain result in extremely uneven wind speed distributions in both time and space, posing challenges for prediction. Traditional numerical models and single machine learning models often fail to capture these intricate topographic features [35]. Therefore, addressing this complexity is essential for enabling accurate wind speed predictions. To address the challenges mentioned above, this paper proposes an end-to-end framework, named MTRCL (Multi-source data fusion, Time-embedded ResNet with CBAM, and Time-embedded Liquid Time-Constant Networks), leveraging multi-source data fusion to enhance wind speed prediction accuracy. The primary contributions of this study are as follows.

(1) This study proposes an innovative end-to-end framework for spatiotemporal multi-source data fusion, which provides the model with a meteorological physical mechanism background by integrating terrain parameters, addressing the issue that traditional models neglect the influence of geographical environment factors on wind fields. Unlike existing methods that rely on a single data source or lack physical mechanism interpretability, this framework achieves a direct mapping from raw multi-source data to wind speed prediction, significantly enhancing prediction accuracy and practicality.

(2) To address the issues of traditional CNNs lacking adaptability and temporal perception when extracting meteorological spatial features, this study integrates an improved channel-spatial-global three-level attention mechanism (CBAM) and a time embedding technique (TE) onto ResNet, constructing the TC-ResNet module. This module dynamically focuses on highly relevant parameters through the attention mechanism, while the TE technique captures the periodic patterns of time, solving the limitation of traditional CNNs' inability to adjust the importance of different meteorological parameters adaptively. Compared with traditional ResNet and other spatial feature extraction methods, this module enables synchronous capture of spatiotemporal features and significantly enhances the representation of complex meteorological spatial features.

(3) To better model the complex dynamic characteristics of wind speed time series, this study integrates the TE technique and meta-learning mechanisms into Liquid Time-Constant Networks (LTCs), constructing the TE-LTCs module. This module introduces the Fourth-Order Runge–Kutta method (RK4) to ensure the stability of long sequence processing and to address the vanishing gradient problem in traditional RNNs. By fusing temporal context information via the TE technique and meta-learning mechanisms, while dynamically adjusting the parameters of differential equations, the model can adaptively capture wind speed change patterns at different time scales. Compared with traditional time series models such as LSTM and GRU, this module demonstrates significant advantages in handling nonlinear temporal dependencies, effectively improving prediction accuracy and stability.

The study is organized as follows: Section 2 describes the data and process, including an overview of data sources, preprocessing procedures, and the architectural design of the MTRCL model. Section 3 presents the experimental setup and the analysis of the results, including feature selection, model performance comparison, and the effects of spatial and temporal corrections. Section 4 describes the advantages of the MTRCL model, analyzes the limitations of this method and feasible future directions, and discusses the model's computational complexity and training time. Finally, Section 5 concludes the study by summarizing the key findings and discussing the significant implications and contributions of the research.

2. Data and method

2.1. Data

2.1.1. Data source

Fig. 1 illustrates the topographic map of China, with the study area outlined by a red solid box. This area spans latitudes 35.13°–47°N and longitudes 103°–126.88°E, covering parts of northern China. The numerical model prediction data utilized in this study are sourced from the TIGGE (THORPEX Interactive Grand Global Ensemble) project of ECMWF. The dataset covers a period of 1461 days, from January 1, 2021, to December 31, 2024. The TIGGE dataset features a horizontal resolution of 0.5° × 0.5° and provides forecasts twice daily, at 00:00

Table 1
ECMWF numerical forecast 26 meteorological factor fields.

Number	Element	Abbreviation	Final selection
1	2 m dew point temperature	d2m	
2	10-m U wind component	u10	Y
3	Convective available potential energy	cape	
4	Maximum temperature at 2 m in the last 6 h	mx2t6	
5	Minimum temperature at 2 m in the last 6 h	mn2t6	
6	Skin temperature	skt	
7	Snowfall water equivalent	sf	
8	Surface latent heat flux	slhf	
9	Surface net shortwave (solar) radiation	ssr	Y
10	Surface sensible heat flux	sshf	
11	Total cloud cover	tcc	
12	Total column water	tcw	
13	2-m temperature	t2m	
14	10-m V wind component	v10	Y
15	Convective inhibition	cin	
16	Land–sea mask	lsm	Y
17	Mean sea level pressure	msl	Y
18	Orography	orog	Y
19	Snow depth water equivalent	sd	
20	Soil moisture	sm	Y
21	Soil temperature	st	
22	Sunshine duration	sund	
23	Surface net long-wave (thermal) radiation	str	
24	Surface pressure	sp	
25	Top net long-wave (thermal) radiation	ttr	
26	Total precipitation	tp	

and 12:00 UTC. For this study, forecasts were initiated at 00:00 and 12:00 UTC each day, with a focus on the 6-hour and 12-hour intervals. Specifically, the forecasts correspond to 06:00, 12:00, 18:00, and 24:00 UTC(or 00:00 UTC of the subsequent day). Consequently, the data sample comprises a total of 26,929,152 samples, structured as follows: “1461” represents the number of days during the research period (2021–2024). The second dimension, “4”, refers to the four forecast times per day. The third dimension, “4608” represents the grid dimension, consisting of 48 latitude points and 96 longitude points. The TIGGE output includes 26 meteorological parameters (refer to Table 1). Additionally, the zonal wind component (u10) and meridional wind component (v10) at 10 m above ground level are combined into a single variable representing the 10-meter wind speed (TIGGE_wind_speed), which follows the formula:

$$TIGGE\_wind\_speed = \sqrt{u10^2 + v10^2} \tag{1}$$

As a result, the TIGGE dataset utilized for prediction contains a total of 27 parameters.

To describe the influence of complex terrain on near-surface atmospheric motion, existing numerical models often rely on simplified parameterization schemes. However, when high-resolution geographic terrain data is applied in such complex terrain conditions, significant gradient variations in parameters can easily lead to instability in calculating partial differential equations. Deep learning networks, particularly feature fusion techniques, have demonstrated effectiveness in addressing this issue [36]. In this study, ASTER GDEM V3 data is utilized, offering a spatial resolution of 1 arc-second (horizontal spacing of about 30 m near the equator). Four key parameters—elevation, relief, slope, and aspect—are incorporated to enhance the accuracy of terrain representation.

Due to the lack of observed data from real meteorological stations within the study area, this study utilizes the ERA5 hourly dataset on single levels, as identified through investigation. Specifically, the u10 and v10 components within this dataset are synthesized into ERA5_wind_speed to substitute for observed data. As demonstrated in the study by Zhang et al. [35], ERA5 data has been successfully employed to predict short-term hub-height wind speeds in northeastern China, verifying the feasibility and effectiveness of ERA5 data as a substitute for observed data in wind speed prediction research. The ERA5 dataset features a spatial resolution of 0.25° × 0.25° and a temporal resolution of 1 h. Provided by the European Centre for Medium-Range Weather Forecasts (ECMWF), ERA5 is a global reanalysis dataset that has been processed and generated according to uniform standards, ensuring stable and high-quality data. This study also extracted five temporal parameters from the raw data, namely Year, Month, Day, Hour, and Season.

Finally, the preprocessed dataset, spanning from January 2021 to December 2022, served as the training set for model parameter optimization. This dataset included 8 parameters (TIGGE_wind_speed, orog, sm, v10, u10, lsm, msl, ssr) selected from 27 TIGGE meteorological parameters that have the most significant correlation with ERA5_wind_speed. Additionally, three terrain parameters (elevation, relief, slope) were selected from 4 ASTER GDEM V3 terrain parameters that have the most significant correlation with ERA5_wind_speed, and 5 time parameters (Year, Month, Day, Hour, Season). All parameters were covered at four time points—UTC 06:00, 12:00, 18:00, and 24:00 (UTC 00:00 of the next day)—resulting in a total of 13,455,360 samples. The dataset for 2023, comprising the same features and totaling 6,727,680 samples, was utilized as the validation set to assess the model’s performance during training and refine hyperparameters. Similarly, the dataset for 2024, containing 6,746,112 samples with identical features, was employed as the test set to evaluate the accuracy of the model’s predictions in practical applications. To ensure fair comparison, all baseline models and comparison methods evaluated in this study utilized the same input features and dataset splits as described above.

2.1.2. Data preprocessing

A multi-step approach was employed to address missing values in the datasets, including ERA5_wind_speed, 27 meteorological parameters from TIGGE, and the four terrain parameters of ASTER GDEM V3. Specifically, linear interpolation along the time dimension was applied to meteorological parameters [37]. The interpolation process follows the formula:

$$x(t,i,j) = x(t_1,i,j) + \frac{x(t_2,i,j) - x(t_1,i,j)}{t_2 - t_1} \cdot (t - t_1) \tag{2}$$

where $x(t,i,j)$ represents the interpolation result at time t and grid point (i,j). $x(t_1,i,j)$ and $x(t_2,i,j)$ are the values of the nearest valid time points before and after the missing point, respectively. t_1 and t_2 are the valid time indices.

For the three data sources, linear interpolation and nearest-neighbor interpolation were applied in the spatial dimension, as described in [38]. The nearest-neighbor interpolation process follows the formula:

$$x(t,i,j) = x(t, i_{nearest}, j_{nearest}) \tag{3}$$

where the coordinates of the nearest-neighbor point are:

$$(i_{nearest}, j_{nearest}) = argmin_{(i',j')\in Valid} \sqrt{(i-i')^2 + (j-j')^2} \tag{4}$$

where $Valid$ represents the set of all valid data points, serves as the foundation for filling missing values. For the remaining small number of missing values that could not be resolved through this method, they are subsequently filled using the global average value of the variable. This final step ensures completeness and consistency across the dataset.

Subsequently, we employed the 3σ rule to process the outliers in all datasets. This statistically grounded approach effectively identifies extreme values that differ significantly from the normal distribution, thereby mitigating the influence of anomalous data on the model.

To ensure the consistency of the datasets in both temporal and spatial dimensions, alignment procedures were applied to the ERA5_wind_speed data and the 27 meteorological parameters from TIGGE. Temporal alignment was achieved by identifying the standard

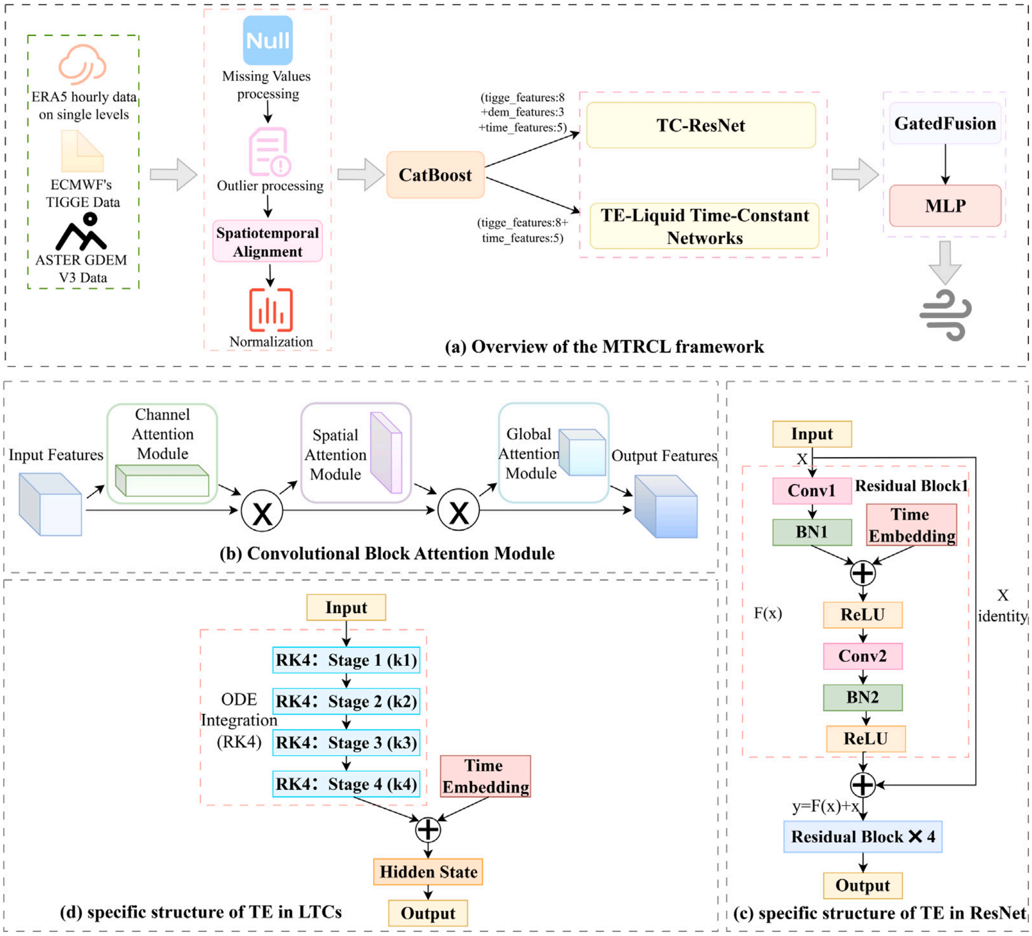


Fig. 2. The Overall Architecture and Innovative Modules of the MTRCL Model: (a) Overview of the MTRCL framework, (b) Convolutional Block Attention Module (CBAM), (c) specific structure of TE in ResNet, (d) specific structure of TE in LTCs.

timestamps (06:00, 12:00, 18:00, and 24:00 UTC) shared between the two datasets and then filtering the data to these common time points. For spatial alignment, the linear interpolation was employed to interpolate all datasets onto a unified latitude–longitude grid, effectively resolving the discrepancies in spatial resolution across different data sources.

Finally, the maximum–minimum normalization method was employed to standardize all data, scaling it to fall within the interval [0, 1]. The formula for the normalization is as follows:

$$X_{\mathrm{std}} = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \tag{5}$$

where X represents the original data, $X_{\min}$ and $X_{\max}$ are the minimum and maximum values in the dataset, respectively, and X_{std} is the standardized data.

2.2. Method

2.2.1. Model architecture of mtrcl

Accurate wind speed prediction requires the integration of spatial features while simultaneously capturing dynamic temporal variations, posing significant challenges for model design. Traditional single models often fail to effectively learn spatio-temporal features concurrently. To address this limitation, this study proposes an innovative end-to-end model architecture, called MTRCL (Multi-source data fusion, Time-embedded ResNet with CBAM, and Time-embedded Liquid Time-Constant Networks).

Fig. 2 illustrates the comprehensive architecture of the MTRCL model, including the detailed design of the Convolutional Block Attention Module (CBAM), and the implementation strategies for integrating temporal embedding (TE) into both the ResNet branch and the Liquid Time-Constant Networks (LTCs) branch.

The MTRCL model achieves high-precision wind speed prediction through the close collaboration of multiple components. It addresses the existing issues in current wind speed prediction tasks, such as the insufficient collaborative extraction of spatio-temporal features, the difficulty in eliminating the interference of complex terrain, and the unstable modeling of long-sequence temporal dependencies. As shown in Fig. 2(a), firstly, multi-source data undergoes data preprocessing and feature selection in sequence. Then, the core features are mapped into a unified tensor by the fusion layer and input into two branch modules in parallel. Subsequently, the TC-ResNet module extracts high-level, nonlinear spatial feature representations. It simultaneously identifies the geographical regions and meteorological variables most relevant to the prediction task at specific time points. Meanwhile, the TE-LTCs module is employed to model the temporal evolution process of wind speed time series. Finally, the model incorporates a gated fusion mechanism to concatenate the features extracted by the two branch modules along the channel dimension, forming a feature vector. A two-layer neural network is used to calculate the fusion weights, achieving adaptive feature fusion. The fused features are then processed by the Multilayer Perceptron (MLP) module, which maps these features through multiple nonlinear transformations to the final wind speed predictions.

2.2.2. TC-ResNet

Convolutional Neural Networks (CNNs) often encounter limitations in feature extraction and challenges such as gradient vanishing when processing spatial features in wind speed prediction [39]. To address these challenges, this study employs the ResNet architecture [40] to introduce residual connections, which are mathematically expressed as follows:

$$out = identity + F(x) \tag{6}$$

where $identity$ represents the input shortcut connection via 1 × 1 convolution, and $F(x)$ is the non-linear transformation of the main path. The TC-ResNet module consists of five convolutional layers. The first convolutional layer employs a 3 × 3 kernel with a stride of 1 and padding of 1. To ensure dimension compatibility, the shortcut connection applies a 1 × 1 convolution with a stride of 1 and no padding. The subsequent convolutional layers share identical configurations, with each employing a 3 × 3 kernel, a stride of 1, and padding of 1. This uniform parameterization preserves spatial resolution throughout the network, and the incorporation of residual connections facilitates robust deep feature learning.

The TC-ResNet module enhances the basic ResNet architecture by incorporating the channel-spatial-global (CBAM) three-level attention mechanism and the time embedding (TE) technique. These are specifically designed to extract high-quality spatial features and enhance their temporal sensitivity. ResNet provides a stable foundation for deep feature extraction through skip connections, enabling the network to learn multi-level spatial representations, from local details to large-scale meteorological field distributions. When processing the diverse, multi-source data in this study, this structure ensures that the model can deepen the network layers while maintaining a smooth information flow, thereby preventing the decline in learning ability caused by gradient issues.

Based on this, the study introduces CBAM to improve the accuracy and effectiveness of spatial feature extraction. The channel attention dynamically evaluates the importance of different meteorological and terrain parameters, automatically focusing on the most relevant channels for wind speed prediction. Spatial attention accurately locates key impact areas geographically, highlighting significant regions under complex terrain conditions and reducing interference from irrelevant areas. Global attention enhances the integration of channel and spatial information overall, ensuring the model captures correlations among multi-dimensional features comprehensively.

Simultaneously, by incorporating the TE technique, temporal parameters are encoded into high-dimensional vectors and integrated into convolutional operations through feature modulation. This grants the spatial feature extraction process the ability to perceive time, enabling it to identify periodic meteorological patterns across various time scales.

The three main components of the TC-ResNet module work together seamlessly as detailed in Algorithm 1. First, the TE technique dynamically incorporates temporal parameters into spatial features, giving the model temporal awareness. Next, ResNet performs initial feature extraction on multi-source data, creating a foundational feature map. An improved CBAM then filters key features from this map across channel, spatial, and global dimensions.

Fig. 2(b) shows the specific structure of CBAM. The channel attention mechanism evaluates feature channels through global average pooling to generate attention weights, and employs two 1 × 1 convolutional layers with a stride of 1 and no padding. The formula is as follows:

$$\alpha = \sigma(W_1 \cdot ReLU(W_0 \cdot AvgPool(x))) \quad (\alpha \in R^{11\times1\times1}) \tag{7}$$

where W_0 is the bottleneck convolutional layer that reduces the channel dimension, W_1 is the channel-expanding convolutional layer that maps back to the original channel dimension, and $ReLU$ represents the activation function, which is used to add non-linearity.

Algorithm 1 Time-embedded Residual Network with an improved CBAM

```
Inputs: $X_{spatial} \in R^{B\times11\times H\times W}$: spatial features;
        $T_{features} \in R^{B\times5}$: time features
Outputs: $F_{spatial} \in R^{B\times56\times H\times W}$: enhanced spatial features
for each input batch do
    $E_{time} \leftarrow$ TimeEmbed($T_{features}$)
    $E_{time} \leftarrow$ Reshape($E_{time}$, $[B, 56, 1, 1]$)
  // Extract initial spatial features
    $F_{conv1} \leftarrow$ Conv2d($X_{spatial}$)
    $F_{bn1} \leftarrow$ BatchNorm($F_{conv1}$) $+ E_{time}$
    $identity \leftarrow$ Conv1x1($X_{spatial}$)
  // Residual block construction
    for $i = 2$ to 5 do
        $F_{conv_i} \leftarrow$ Conv2d($F_{bn_{i-1}}$)
        $F_{bn_i} \leftarrow$ BatchNorm($F_{conv_i}$) $+ identity$
        $identity \leftarrow F_{bn_i}$
    end for
  // Apply improved CBAM attention mechanism
    $\alpha \leftarrow$ ChannelAttention($F_{bn5}$)
    $F_{channel} \leftarrow \alpha \odot F_{bn5}$
    $\beta \leftarrow$ SpatialAttention($F_{channel}$)
    $F_{spatial\_att} \leftarrow \beta \odot F_{channel}$
    $\gamma \leftarrow$ GlobalAttention($F_{spatial\_att}$)
    $F_{spatial} \leftarrow F_{spatial\_att} + \gamma \odot F_{spatial\_att}$
end for
return $F_{spatial}$
```

The spatial attention mechanism captures the localized features through channel pooling operations to generate spatial attention weights, employing a 7 × 7 convolutional layer with a stride of 1 and padding of 3. The formula is as follows:

$$\beta = \sigma\left(Conv_{7\times7}\left(Concat(F_{avg}, F_{max})\right)\right) \quad (\beta \in R^{1\times48\times96}) \tag{8}$$

where F_{avg} and F_{max} denote the average pooling and max pooling operations performed on the feature map, respectively; and $Conv_{7\times7}$ represents the convolution operation that conducts spatial feature extraction and dimension compression on the concatenated feature map. The global attention mechanism extracts a global feature vector through adaptive average pooling, thereby generating global weights:

$$\gamma = \sigma(W \cdot v_{flatten}) \quad (\gamma \in R^{11\times1\times1}) \tag{9}$$

where $v_{flatten}$ denotes the global feature vector obtained by flattening the feature map after adaptive average pooling, W refers to the weight matrix of the fully connected layer, σ is the Sigmoid activation function used to generate attention weights, and γ is the generated global attention weight vector that adaptively reweights the original feature map to emphasize globally critical information. Through this multi-level attention mechanism, CBAM enables the model to adjust feature weights from a global perspective dynamically. As a result, it focuses on the spatial features crucial for wind speed prediction, improving the model's prediction ability.

To integrate Time Embedding (TE) into the ResNet architecture, five temporal parameters are mapped to a feature space using a three-layer fully connected neural network. The mathematical formulation for this mapping is provided below.

$$TE(t) = W_4 \cdot ReLU(W_3 \cdot ReLU(W_2 \cdot t)) \tag{10}$$

where t denotes the input vector of five temporal parameters; W_2, W_3, and W_4 represent the weight matrices corresponding to the three fully connected layers in sequence; and $TE(t)$ is the final time embedding vector, which is fused with the feature maps to incorporate temporal information into spatial feature learning. As shown in Fig. 2(c), this design enables the model to perceive the temporal dependencies of the data and capture the variation patterns of wind speed at different times.

2.2.3. TE-Liquid Time-Constant Networks

In the task of wind speed prediction, traditional time series models, such as simple RNNs and their variants (LSTM, etc.), demonstrate a specific capacity to capture the dynamic characteristics of temporal data [41]. However, they are prone to issues such as vanishing or exploding gradients during the processing of long sequences. Furthermore, their ability to fuse multi-source heterogeneous data is limited, preventing them from fully leveraging the spatial and temporal information of the wind speed data. To address these challenges, this study introduces the Liquid Time-Constant Networks (LTCs) architecture [42]. LTCs are designed based on continuous-time recurrent neural networks and model the dynamic behavior of time series using ordinary differential equations (ODEs). Their adaptive time constant feature allows the network to automatically adjust response times based on the input's rate of change, making them especially effective at capturing nonlinear dynamic features in meteorological systems.

The TE-LTCs module combines LTCs, the time embedding (TE) technique, the Fourth-Order Runge–Kutta Method (RK4), the gating mechanism, and a meta-learning mechanism to deeply model the complex dynamic evolution of wind speed time series.

By incorporating the TE technique and RK4, this study improves the model's ability to capture complex temporal patterns, especially in handling non-stationary wind speed sequences. The TE encodes multi-scale temporal parameters into vectors and integrates them into the LTCs' state update process via a gating mechanism, enabling the model to recognize periodic meteorological patterns. RK4 ensures computational stability and accuracy in solving ODEs through adaptive step-size numerical integration, effectively capturing complex temporal dependencies and providing a solid foundation for time series modeling.

Additionally, the TE-LTCs module incorporates a meta-learning mechanism to adjust the parameters of ODEs dynamically. By learning from prior knowledge of historical meteorological patterns, the model is provided with optimized initial parameter values, which accelerates the adaptation process to new meteorological conditions. Meanwhile, the application of a time-varying step size function further enhances the model's flexibility. During periods when wind speed changes are relatively stable, the model can use a larger step size for calculation, thereby improving computational efficiency. In the event of sudden changes in wind speed, the model automatically reduces the step size to capture the nonlinear transient characteristics during these abrupt changes accurately.

During the actual operation, as detailed in Algorithm 2, the basic structure of LTCs is responsible for the preliminary extraction of temporal features and the modeling of long-term dependencies in the wind speed time series. The TE technique enhances the model's perception of temporal dynamics and periodicity. The gating mechanism enables intelligent parameter adjustment. The RK4 method ensures the accuracy and stability of numerical calculations. The meta-learning mechanism optimizes the model's adaptability to different wind speed change patterns. The time-varying step size function flexibly adjusts the step size of the RK4 integration, ensuring that the model can accurately capture various temporal features of wind speed while maintaining computational efficiency. These components work synergistically to build a robust temporal modeling capability that can precisely handle the nonlinear dynamic features, multi-scale periodic changes, and sudden meteorological disturbances in wind speed time series. This capability forms a spatiotemporal complementarity with the spatial features extracted by the TC-ResNet module, thereby improving the overall prediction accuracy of the model.

As shown in Fig. 2(d). Initially, the temporal embedding vector is processed by a multi-layer perceptron, yielding a temporal feature representation that is isomorphic to the hidden state. Subsequently, at each time step, a gating mechanism is applied to dynamically integrate the temporal features with the network's hidden states. The corresponding formula governing this gating mechanism is provided as follows:

Algorithm 2 Time-embedded Liquid Time-Constant Networks with RK4 Integration

```
Inputs: $X_{seq} \in R^{B\times T\times C\times H\times W}$: sequential features;
        $T_{seq} \in R^{B\times T\times 5}$: time features
Outputs: $H_{out} \in R^{B\times d\times H\times W}$: output temporal features
$h \leftarrow$ Initialize$(B \times H \times W, d)$
for $t = 1$ to $T$ do
    $e'_t \leftarrow$ MLP(TimeEmbed($T_{seq}[:, t, :]$))
  // Compute gating mechanism and feature fusion
    $g_t \leftarrow \sigma(W_g[X_{seq}[:, t, :, :, :]; e'_t] + b_g)$
    $f_t \leftarrow g_t \odot X_{seq}[:, t, :, :, :] + (1 - g_t) \odot e'_t$
  // Generate time-adaptive parameters
    $W_\theta(t), b_\theta(t) \leftarrow$ ParamNet$(e'_t)$
    $\Delta t(e'_t) \leftarrow \alpha \cdot \exp(w^T_{\Delta t} e'_t + b_{\Delta t}) + \beta$
  // Apply RK4 integration for ODE solving
    $k_1 \leftarrow$ ODEFunc$(h, f_t; W_\theta(t), b_\theta(t))$
    $k_2 \leftarrow$ ODEFunc$(h + 0.5\Delta t k_1, f_t; W_\theta(t), b_\theta(t))$
    $k_3 \leftarrow$ ODEFunc$(h + 0.5\Delta t k_2, f_t; W_\theta(t), b_\theta(t))$
    $k_4 \leftarrow$ ODEFunc$(h + \Delta t k_3, f_t; W_\theta(t), b_\theta(t))$
    $h \leftarrow h + \frac{\Delta t}{6}(k_1 + 2k_2 + 2k_3 + k_4)$
    $h \leftarrow$ Clamp$(h, -100, 100)$
end for
$H_{out} \leftarrow$ OutputLayer$(h)$
return $H_{out}$
```

$$g_t = \sigma(W_g[x_t; e'_t] + b_g), \quad f_t = g_t \odot x_t + (1 - g_t) \odot e'_t \tag{11}$$

where x_t denotes the input data, while e_t represents the temporal embedding vector. The temporal feature e'_t, isomorphic to the hidden state, is derived from e_t through a transformation using a multi-layer perceptron. The parameters W_g and b_g are learnable variables within the gating mechanism. Furthermore, the time embedding modulates the differential equation parameters of LTCs through a meta-learning mechanism. The corresponding formula for this process is provided as follows:

$$W_\theta(t), b_\theta(t) = ParamNet(e'_t) \tag{12}$$

where $ParamNet$ represents the parameter-generating network, $W_\theta(t)$ represents the time-varying weight matrix, and $b_\theta(t)$ represents the bias vector. To enhance the capacity for modeling non-stationary time series data, the RK4 integration step incorporates a time-varying step size function. The corresponding formula for the time-varying step size function is provided as follows:

$$\Delta t(e'_t) = \alpha \cdot exp(w^T_{\Delta t} e'_t + b_{\Delta t}) + \beta \tag{13}$$

where α and β are hyperparameters, and $w_{\Delta t}$ and $b_{\Delta t}$ are learnable parameters that dynamically adjust the integration step size and direction. These parameters utilize temporal embedding features, enabling the model to capture the nonlinear characteristics of wind speed changes accurately.

2.2.4. Model training

During the model training phase, Bayesian optimization is employed to conduct a global search for hyperparameters. In comparison to traditional methods such as grid search and random search, Bayesian optimization constructs a surrogate model of the objective function, leveraging prior knowledge and historical evaluation results to search for the optimal solution of hyperparameters more efficiently. All experiments were conducted on a computer equipped with an Intel Core i5-13600 K CPU and an NVIDIA GeForce RTX 4090 GPU with 24 GB VRAM. For model configuration, the input length was set to 4 time

Table 2
List of the search ranges for hyperparameters of each model.

Parameters	Model									
	MTRCL	ConvLSTM	TE-LTCs	ResNet	TE-ResNet	TC-ResNet	LSTM	GRU	Trans former	XGBoost
Learning rate	[1e−4,5e−4]	[5e−5,3e−4]	[5e−5,5e−4]	[1e−5,1e−3]	[1e−5,1e−3]	[1e−5,1e−3]	[5e−5,3e−4]	[5e−5,3e−4]	[1e−5,1e−4]	[1e−2,3e−1]
Weight decay	[1e−4,1e−2]	[1e−4,1e−2]	[1e−4,1e−2]	[1e−4,1e−2]	[1e−4,1e−2]	[1e−4,1e−2]	[1e−4,5e−3]	[1e−4,5e−3]	[1e−4,5e−3]	–
Dropout rate	[5e−2,3e−1]	[2e−1,5e−1]	[5e−2,3e−1]	[1e−1,4e−1]	[1e−1,4e−1]	[1e−1,4e−1]	[1e−1,6e−1]	[1e−1,6e−1]	[1e−1,6e−1]	–
ResNet hidden dim	{32,48,56, 64,128}	–	–	{32,48,56, 64,128}	{32,48,56, 64,128}	{32,48,56, 64,128}	–	–	–	–
LTCs hidden dim	{192,200, 216,224}	–	{192,200, 216,224}	–	–	–	–	–	–	–
ConvLSTM hidden dim	–	{32,64,128}	–	–	–	–	–	–	–	–
CBAM reduction	{8,12,16, 20,24}	–	–	–	–	{8,12,16, 20,24}	–	–	–	–
Batch size	{8,16,24}	{8,16,24}	{8,16,24}	{8,16,24}	{24,28,32}	{16,24,32}	{4,8,16}	{4,8,16}	{2,4,8}	–
Early stopping round	[3,8]	[3,8]	[3,8]	[3,8]	[3,8]	[3,8]	[3,8]	[3,8]	[3,8]	–
Accumulation steps	{1,2,4}	{1,2,4}	{1,2,4}	{1,2,4}	{1,2,4}	{1,2,4}	{1,2,4}	{1,2,4}	{1,2,4}	–
LSTM hidden dim	–	–	–	–	–	–	{64,96, 128,160}	–	–	–
GRU hidden dim	–	–	–	–	–	–	–	{64,96, 128,160}	–	–
d_model	–	–	–	–	–	–	–	–	{128,192, 256,320}	–
nhead	–	–	–	–	–	–	–	–	{4,8,16}	–
num_layers	–	–	–	–	–	–	{1,2,3,4}	{1,2,3,4}	{2,3,4,6}	–
n_estimators	–	–	–	–	–	–	–	–	–	[1e2,1e3]
STDGN hidden dim	–	–	–	–	–	–	–	–	–	–

Parameters	Model		
	STDGN	BPNN	C-ResNet
Learning rate	[1e−4,1e−3]	[1e−4,1e−3]	[1e−4,1e−3]
Weight decay	[1e−4,5e−3]	[1e−4,5e−3]	[1e−4,5e−3]
Dropout rate	[1e−1,4e−1]	[1e−1,4e−1]	[1e−1,4e−1]
ResNet hidden dim	–	–	{32,48,56,64,128}
LTCs hidden dim	–	–	–
LSTM hidden dim	–	–	–
CBAM reduction	–	–	{8,12,16,20,24}
Batch size	{8,16,24}	{8,16,24}	{8,16,24}
Early stopping round	[3,8]	[3,8]	[3,8]
Accumulation steps	{1,2,4}	{1,2,4}	{1,2,4}
LSTM hidden dim	–	–	–
GRU hidden dim	–	–	–
d_model	–	–	–
nhead	–	–	–
num_layers	–	–	–
n_estimators	–	–	–
STDGN hidden dim	{64,96,128,160}	–	–

steps, corresponding to 24 h of historical data. The prediction horizon was configured as 1 time step, which equals a 6-hour advance prediction. The time step interval was 6 h, aligning with the four daily observation times (06:00, 12:00, 18:00, and 24:00 UTC). The specific search ranges of hyperparameters for each model are detailed in Table 2. After conducting 75 rounds of experiments, the optimal hyperparameter combinations for each model are finally determined and are presented in Table 3. It should be noted that, as the baseline model selected for this study, the Pangu model directly adopts the default parameter configuration publicly implemented in its paper [43]; while for the other comparison model, Statistical regression, due to its simple model structure, its core parameters are conventional settings recognized within the field, and thus are not listed separately. The AdamW optimizer is employed for parameter updates, in conjunction with a cosine annealing learning rate scheduler to ensure smooth convergence. The loss function is SmoothL1Loss. Additionally, gradient accumulation techniques are employed to accommodate limited video memory, ensuring efficient utilization of computational resources.

2.2.5. Evaluation metric

The accuracy of wind speed forecasts is assessed through a comprehensive set of evaluation metrics. This study employs the Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Correlation Coefficient (R), Relative Root Mean Square Error (rRMSE), Relative Mean Absolute Error (rMAE), Mean Absolute Percentage Error (MAPE), and the Fraction of Absolute Error (FA), which represents the percentage of samples where the absolute wind speed error does not exceed 1 m/s. These indicators collectively provide a robust framework for evaluating the predictive performance of the forecasting models [44]. The calculation formulas for these indicators are as follows:

$$RMSE = \sqrt{\frac{1}{N}\sum_{k=1}^{N}(V_{P,k} - V_{M,k})^2} \tag{14}$$

$$MAE = \frac{1}{N}\sum_{k=1}^{N}\left|V_{P,k} - V_{M,k}\right| \tag{15}$$

$$R = \frac{\sum_{k=1}^{N}\left[(V_{P,k} - \overline{V_P})(V_{M,k} - \overline{V_M})\right]}{\sqrt{\sum_{k=1}^{N}(V_{P,k} - \overline{V_P})^2 \sum_{k=1}^{N}(V_{M,k} - \overline{V_M})^2}} \tag{16}$$

$$rRMSE = \frac{\sqrt{\frac{1}{N}\sum_{k=1}^{N}(V_{P,k} - V_{M,k})^2}}{\left(\frac{1}{N}\sum_{k=1}^{N}V_{M,k}\right)} \times 100\% \tag{17}$$

$$rMAE = \frac{\frac{1}{N}\sum_{k=1}^{N}\left|V_{P,k} - V_{M,k}\right|}{\left(\frac{1}{N}\sum_{k=1}^{N}V_{M,k}\right)} \times 100\% \tag{18}$$

$$FA = \frac{N_r}{N_f} \times 100\% \tag{19}$$

$$MAPE = \frac{1}{N}\sum_{k=1}^{N}\left|\frac{V_{P,k} - V_{M,k}}{V_{M,k}}\right| \times 100\% \tag{20}$$

In the formulas mentioned above, V_M represents the measured wind speed, $V_{M,k}$ represents the measured wind speed at time k after transformation, $\overline{V_M}$ represents the average value of the measured wind speed within the assessment period after transformation, V_P represents the predicted wind speed, $V_{P,k}$ represents the predicted wind speed at time k after transformation, $\overline{V_P}$ represents the average value of the predicted

Table 3
List of the selected parameters for each model.

Parameters	Model												
	MTRCL	ConvLSTM	TE-LTCs	ResNet	TE-ResNet	TC-ResNet	LSTM	GRU	Transformer	XGBoost	STDGN	BPNN	C-ResNet
Learning rate	3.9e−4	2e−4	3e−4	1e−4	2e−4	4e−4	1e−4	1e−4	5e−5	0.1	5e−4	5e−4	4e−4
Weight decay	1e−4	1e−3	1e−4	1e−3	8e−3	1e−3	1e−3	1e−3	1e−3	–	1e−3	1e−3	1e−3
Dropout rate	0.25	0.4	0.2	0.3	0.3	0.3	0.4	0.4	0.4	–	0.3	0.3	0.3
ResNet hidden dim	56	–	–	56	56	56	–	–	–	–	–	–	56
LTCs hidden dim	216	–	216	–	–	–	–	–	–	–	–	–	–
ConvLSTM hidden dim	–	64	–	–	–	–	–	–	–	–	–	–	–
CBAM reduction	16	–	–	–	–	16	–	–	–	–	–	–	16
Batch size	16	16	16	8	28	24	8	8	4	–	8	16	24
Early stopping round	5	5	5	3	5	5	5	5	5	–	5	5	5
Accumulation steps	1	1	1	1	1	1	2	2	4	–	2	1	1
LSTM hidden dim	–	–	–	–	–	–	128	–	–	–	–	–	–
GRU hidden dim	–	–	–	–	–	–	–	128	–	–	–	–	–
d_model	–	–	–	–	–	–	–	–	256	–	–	–	–
nhead	–	–	–	–	–	–	–	–	8	–	–	–	–
num_layers	–	–	–	–	–	–	2	2	4	–	–	–	–
n_estimators	–	–	–	–	–	–	–	–	–	500	–	–	–
STDGN hidden dim	–	–	–	–	–	–	–	–	–	–	128	–	–

wind speed within the assessment period after transformation, N refers to the sample size, N_r represents the number of samples with an absolute wind-speed error no greater than 1 m/s, and N_f represents the number of predicted samples.

3. Results

3.1. Feature selection

Feature selection methods are generally classified into three categories: filter selection, wrapper selection, and embedded selection. To verify the effectiveness of different feature selection methods, this study selected typical algorithms for each of the three categories to conduct comparative experiments. Among them, for filter selection, the Pearson correlation coefficient and mutual information methods were used together for screening, and ultimately, 8 meteorological parameters (TIGGE_wind_speed, u10, lsm, sund, sm, ssr, str, sshf) and 3 terrain parameters (relief, slope, elevation) were selected. For wrapper selection, the recursive feature elimination (RFE) method was adopted, and all selected parameters were: TIGGE_wind_speed, mn2t6, u10, t2m, msl, lsm, skt, orog, slope, aspect, and relief. For embedded selection, the random forest algorithm was used, and all selected parameters were: TIGGE_wind_speed, u10, v10, lsm, sm, orog, sund, ssr, elevation, relief, and slope. After feature selection by different methods, dedicated datasets were constructed, and the performance was evaluated based on various indicators in the testing phase.

Categorical Boosting(CatBoost), a gradient-boosting decision tree model, can compute the importance scores of each feature during the model training process. The comparison results show (see Table 4) that its performance in all indicators in the final testing phase was superior to other feature selection methods, fully demonstrating the advantages of CatBoost in feature selection tasks. Similarly, Zeng et al. [44] employed the CatBoost selection method to screen key meteorological features in their research on short-term wind speed prediction, and verified the effectiveness of this method in identifying critical variables for meteorological prediction. Building on the comparative experiment results of this study and research foundation, we employ the CatBoost algorithm to conduct feature importance assessment and selection.

Fig. 3(a) illustrates the significance of TIGGE features obtained through CatBoost training, highlighting several meteorological parameters that strongly influence wind speed. Terrain influences airflow speed via altitude differences, thereby affecting wind speed; variations in land cover roughness directly regulate near-surface wind speed [35]. Soil moisture changes can modify energy exchange between the surface and atmosphere [45], while the level of mean sea level pressure determines airflow rate. Collectively, these factors impact the accuracy of wind speed predictions, confirming the validity of feature selection using the embedded CatBoost method.

To identify the optimal number of features for wind speed prediction, we further analyzed the relationship between feature count and model performance. This analysis involved gradually increasing the number of features and evaluating the model using error metrics, specifically the RMSE and the MAE, for each subset of features. As illustrated in Fig. 3(b), both RMSE and MAE decrease as the number of features increases. Notably, when the number of meteorological parameters reaches eight, the error metrics plateau, indicating that these eight parameters sufficiently improve the ECMWF model's predictions. Beyond this threshold, the inclusion of additional features yields negligible improvement in model performance. Consequently, the eight most influential meteorological parameters were selected as the final feature set, ensuring an optimal balance between model complexity and predictive accuracy.

Fig. 4(a) illustrates the importance of the ASTER GDEM V3 terrain features derived from CatBoost training. These terrain parameters exhibit a strong correlation with ERA5_wind_speed, highlighting their influence on atmospheric dynamics. As altitude increases, atmospheric pressure and air density decrease, leading to increased wind speeds. Additionally, terrain relief is closely related to wind speed. In regions with extensive topographic variations, such as mountainous areas, airflow is influenced by terrain-forced effects, resulting in complex wind

Table 4
Evaluation of forecast results of different feature selection methods.

Method	FA (%)	RMSE (m/s)	MAE (m/s)	rRMSE (%)	rMAE (%)	R	MAPE (%)
Filter selection	80.63	0.83	0.62	23.14	17.14	0.91	20.57
Wrapper selection	81.38	0.82	0.61	22.94	17.01	0.92	20.41
Embedded selection	83.04	0.80	0.60	21.78	16.54	0.92	19.94
CatBoost	**86.64**	**0.70**	**0.52**	**19.57**	**14.55**	**0.94**	**14.85**

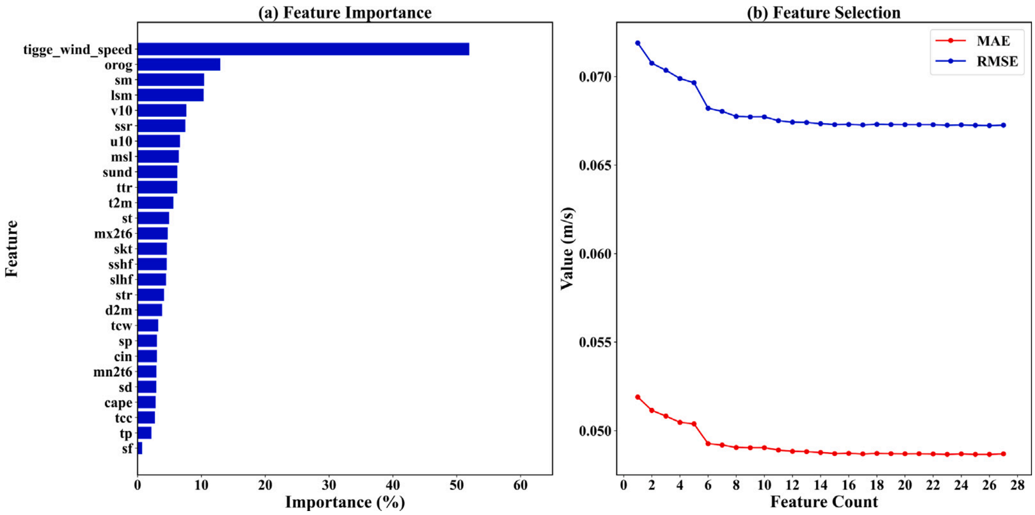


Fig. 3. Feature selection of ECMWF-TIGGE based on the CatBoost model: (a) the importance of different characteristics, (b) the MAE and RMSE between ERA5_wind_speed and the wind speed predicted by the MTRCL model after training with different numbers of input features.

patterns. Additionally, the slope of the terrain impacts airflow velocity; steep slopes tend to accelerate airflow, while gentle slopes exert a comparatively minor influence. These findings underscore the importance of terrain features in determining the magnitude and direction of wind speed, thereby validating the effectiveness of feature selection using the embedded methods within CatBoost [46].

To determine the optimal number of terrain features for wind speed prediction, we utilized the same approach as for TIGGE features. As illustrated in Fig. 4(b), the error metrics exhibit stabilization when the number of terrain parameters reaches three. This finding suggests that three parameters—elevation, relief, and slope—significantly enhance the predictive accuracy of the ECMWF model. Consequently, the top three-ranked terrain features were selected for further analysis.

Meanwhile, to further verify the advantages of the multivariate wind speed prediction model proposed in this study, we also conducted additional comparative experiments to compare its performance with scenarios of univariate and single-category feature inputs. Specifically, we constructed four different input configurations for performance evaluation, namely the complete multivariate feature set (integrating all 8 meteorological parameters and 3 terrain parameters), the meteorological feature set (using only the 8 selected meteorological parameters), the terrain feature set (using only the 3 selected terrain parameters), and the univariate input (using only the core benchmark parameter TIGGE_wind_speed directly related to wind speed). To eliminate the interference caused by differences in model settings or evaluation criteria, the same prediction model and indicators were used for evaluation in all four groups of experiments. As shown in Table 5, the multivariate wind speed prediction model proposed in this study performed optimally in all indicators, fully demonstrating the advantages of multivariate fusion.

Furthermore, this study also conducted sensitivity experiments on the 11 selected parameters to clarify the impact of individual input parameters on the model's predictive performance. We sequentially removed one parameter from the complete multivariate input set, used the remaining 10 parameters as the model input, and evaluated the model's performance using the same metrics in the testing phase. This approach was adopted to determine the variation patterns of prediction accuracy and error levels after removing each feature. As shown in Table 6, the removal of core features had the most significant impact on the model's performance. For instance, removing the TIGGE_wind_speed parameter resulted in a 25.17% decrease in FA and a 0.73 m/s increase in RMSE.

3.2. Model performance comparisons

The final prediction performance of the model was evaluated using the 2024 dataset as the test set, ensuring that it did not overlap with the training and validation sets. Furthermore, the model's performance was compared with that of various representative models in the current field, specifically including the Pangu-Weather model [43], Convolutional Long Short-Term Memory Network (ConvLSTM), Long Short-Term Memory network (LSTM), Gated Recurrent Unit (GRU), Transformer, STDGN [47], Back Propagation Neural Network (BPNN), eXtreme Gradient Boosting (XGBoost), and traditional statistical methods (Statistical regression). The comparative prediction results of all the aforementioned models are presented in Table 7.

Ablation experiments were also conducted to evaluate the effectiveness of each branch and component within the MTRCL model. In this study, several configurations were tested. TE-LTCs refer to introducing TE into the traditional LTCs architecture, ResNet denotes the traditional model without adding any components, TE-ResNet refers to introducing TE into the ResNet architecture, C-ResNet refers to introducing CBAM into the ResNet architecture, and TC-ResNet refers to introducing TE and CBAM into the ResNet architecture. As detailed in Table 7, the MTRCL model demonstrated significant improvements across all evaluation metrics compared to alternative models. Specifically, the MTRCL model achieved an FA value of 86.64%, RMSE of 0.70, MAE of 0.52, rRMSE of 19.57%, rMAE of 14.55%, MAPE of 14.85%, and an R-value of 0.94.

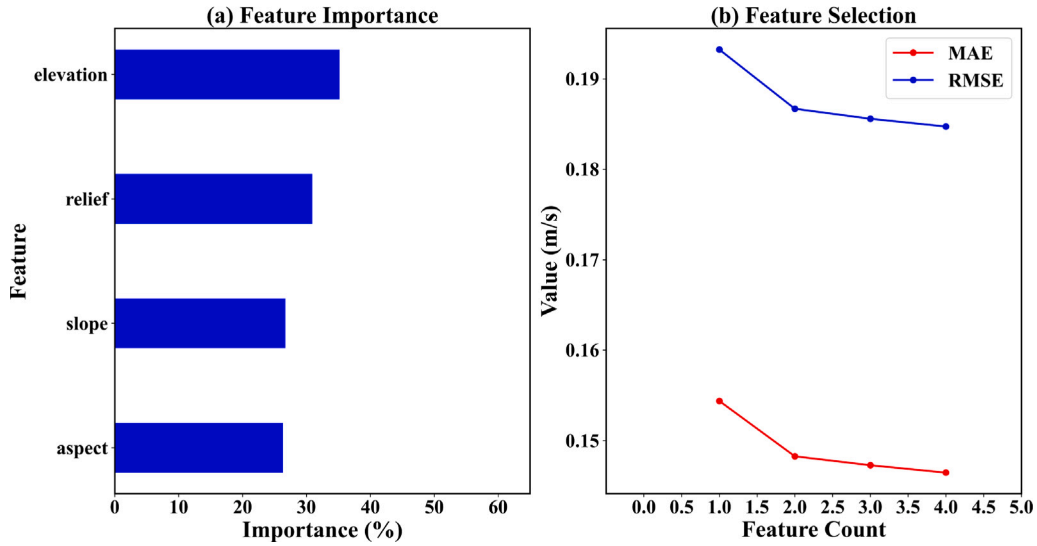


Fig. 4. Feature selection of ASTER GDEM V3 based on the CatBoost model: (a) the importance of different characteristics, (b) the MAE and RMSE between ERA5_wind_speed and the wind speed predicted by the MTRCL model after training with different numbers of input features.

Table 5
Evaluation of forecast results of different input feature types.

Input feature type	FA (%)	RMSE (m/s)	MAE (m/s)	rRMSE (%)	rMAE (%)	R	MAPE (%)
Meteorological features	76.81	0.95	0.72	25.12	18.21	0.91	21.85
Terrain features	45.10	1.81	1.42	49.18	37.26	0.52	44.71
Univariate	70.78	1.12	0.83	29.24	21.87	0.88	24.06
Multivariate	**86.64**	**0.70**	**0.52**	**19.57**	**14.55**	**0.94**	**14.85**

Table 6
Results of sensitivity analysis of input parameters.

Sensitivity analysis group	FA (%)	RMSE (m/s)	MAE (m/s)	rRMSE (%)	rMAE (%)	R	MAPE (%)
Without TIGGE_wind_speed	61.47	1.43	1.05	39.13	28.28	0.74	33.94
Without orog	73.68	1.01	0.76	26.93	20.26	0.89	23.09
Without sm	74.94	0.98	0.74	26.13	20.14	0.90	22.45
Without lsm	75.22	0.96	0.73	25.66	19.76	0.91	22.07
Without v10	78.86	0.89	0.66	23.93	17.99	0.92	21.38
Without u10	77.45	0.94	0.69	25.07	18.11	0.91	21.63
Without ssr	81.61	0.82	0.60	22.38	16.87	0.92	20.34
Without msl	79.32	0.86	0.65	23.57	17.90	0.92	20.86
Without elevation	68.91	1.17	0.92	29.36	22.61	0.82	24.87
Without relief	69.04	1.15	0.87	29.34	22.55	0.83	24.81
Without slope	71.36	1.03	0.79	27.67	21.56	0.88	23.72
Full parameters	**86.64**	**0.70**	**0.52**	**19.57**	**14.55**	**0.94**	**14.85**

To further confirm the statistical significance of the forecasting results, we perform Diebold–Mariano (DM) tests [48] to assess whether the MTRCL model shows statistically superior forecasting performance compared to baseline models. The DM test results across different seasons and the whole year are shown in Table 8. At a 1% significance level, the results reveal that the MTRCL model significantly outperforms all baseline models during each seasonal period and for the entire year. The lowest annual DM value observed is 4.77, which substantially exceeds the critical threshold at the 1% significance level, indicating notable performance differences even with the strongest baseline model. These DM test results provide statistical evidence that the MTRCL model outperforms baseline models in forecasting accuracy across different time periods.

Fig. 5 presents a radar chart of the performance of different models in key evaluation dimensions for wind energy prediction, including four core indicators: mean consistency, distribution consistency, seasonal consistency, and high-energy capture. These indicators are all calculated based on the following wind power density formula [49]:

$$WPD = \frac{1}{2}\rho V^3 \tag{21}$$

where WPD is the wind power density (unit: W m^{-2}), V is the wind speed (unit: m s^{-1}), and ρ is the air density (unit: kg m^{-3}). The wind speed data of each model at four time points every day (06:00, 12:00, 18:00, and 24:00 UTC) are used for calculation.

Mean Consistency is evaluated by comparing the mean difference between the predicted wind energy of each model and the observed wind energy. Its formula is as follows:

$$MC = 1 - \frac{|\overline{WPD_{pred}} - \overline{WPD_{obs}}|}{\overline{WPD_{obs}}} \tag{22}$$

where $\overline{WPD_{pred}}$ is the spatiotemporal average of the predicted wind power density, and $\overline{WPD_{obs}}$ is the spatiotemporal average of the observed wind power density. Distribution Consistency is quantified by the KL divergence, which measures the similarity between the predicted

Table 7
Evaluation of forecast results of each method.

Method	FA (%)	RMSE (m/s)	MAE (m/s)	rRMSE (%)	rMAE (%)	R	MAPE (%)
ECMWF-TIGGE	68.36	1.06	0.82	29.42	22.74	0.89	27.29
TE-LTCs	75.05	0.96	0.72	26.66	20.10	0.91	22.11
ResNet	64.44	1.28	0.95	35.54	26.28	0.87	31.64
TE-ResNet	76.54	1.07	0.73	29.64	20.38	0.88	22.41
TC-ResNet	79.14	0.86	0.65	23.87	17.95	0.92	26.44
ConvLSTM	71.11	1.02	0.78	28.24	21.60	0.89	25.92
LSTM	70.29	1.11	0.82	30.95	22.68	0.89	28.28
GRU	77.49	0.98	0.65	27.30	18.28	0.90	25.49
Transformer	78.61	0.94	0.69	26.14	19.19	0.91	26.03
Statistical regression	74.65	1.02	0.71	28.25	20.58	0.89	25.84
XGBoost	81.77	0.81	0.59	21.99	15.78	0.90	19.75
Pangu	85.41	0.85	0.64	21.03	17.95	0.94	15.19
STDGN	81.16	0.82	0.62	22.01	15.83	0.90	19.79
BPNN	67.31	1.12	0.85	31.15	23.68	0.84	27.93
C-ResNet	74.51	1.01	0.72	27.98	21.56	0.89	25.86
MTRCL	**86.64**	**0.70**	**0.52**	**19.57**	**14.55**	**0.94**	**14.85**

Table 8
The result of DM test.

Method	All year	Spring	Summer	Autumn	Winter
ECMWF-TIGGE	11.79*	5.62*	6.08*	5.81*	5.70*
TE-LTCs	6.47*	3.30*	3.32*	3.03*	3.66*
ResNet	14.44*	7.45*	7.04*	7.20*	7.37*
TE-ResNet	8.72*	4.80*	4.05*	4.28*	4.35*
TC-ResNet	6.70*	3.37*	3.55*	3.19*	3.47*
ConvLSTM	10.87*	6.11*	6.44*	5.68*	4.84*
LSTM	12.86*	6.55*	5.90*	6.58*	6.80*
GRU	6.74*	3.79*	2.67*	3.25*	3.78*
Transformer	9.98*	5.14*	4.95*	4.87*	5.08*
Statistical regression	7.44*	4.96*	5.35*	4.25*	4.25*
XGBoost	6.59*	3.26*	3.25*	2.59*	3.21*
Pangu	4.77*	2.59*	2.63*	2.35**	2.33**
STDGN	7.18*	4.55*	4.11*	3.87*	3.36*
BPNN	13.76*	7.99*	6.81*	6.64*	6.04*
C-ResNet	9.53*	3.62*	4.08*	3.81*	3.70*

Note:
* 1% significance level.
** 5% significance level.

wind energy distribution and the actual distribution. Its formula is as follows:

$$DC = 1 - KL(P_{obs} \parallel P_{pred}) \tag{23}$$

where P_{obs} and P_{pred} are the probability distributions of the observed and predicted wind power densities, respectively. Seasonal Consistency is calculated by grouping wind energy data by season to assess the prediction accuracy of each season. Its formula is as follows:

$$SC = \frac{1}{4}\sum_{s=1}^{4}\left(1 - \frac{|\overline{WPD_{pred,s}} - \overline{WPD_{obs,s}}|}{\overline{WPD_{obs,s}}}\right) \tag{24}$$

where $\overline{WPD_{pred,s}}$ and $\overline{WPD_{obs,s}}$ are the average wind power densities of the predicted and observed values for the sth season. High Energy Capture categorizes wind energy into 7 grades based on a threshold and assesses the model's predictive ability for high wind energy grade events. Its formula is as follows:

$$HEC = \frac{TP}{TP + FN} \tag{25}$$

where TP is the number of grid points where both the prediction and observation indicate high wind energy, and FN is the number of grid points where the observation indicates high wind energy but the prediction indicates low wind energy.

The MTRCL model significantly outperforms other benchmark models in all evaluation dimensions, especially in high energy capture and mean consistency. In terms of distribution consistency, MTRCL achieves 0.92, indicating that it can more accurately characterize the probability distribution of wind speed, providing a reliable basis for risk assessment and grid dispatch. Additionally, MTRCL achieves a score of 0.90 in seasonal consistency, significantly exceeding the 0.65 of the numerical weather prediction model ECMWF-TIGGE, demonstrating its superior ability to capture seasonal wind energy variations. Compared with other models with varying biases, MTRCL provides more comprehensive and reliable technical support for wind energy resource assessment based on ERA5 wind speed data.

3.3. *The result of space correction by mtrcl model*

Fig. 6 illustrates the spatial distribution of evaluation indicators for grid points within the study area, both before and after correction. The difference in land–sea thermal conditions affects atmospheric circulation and wind speed, making it challenging to accurately capture the resulting complex wind field changes. Due to this limitation, the ECMWF model exhibits significant prediction errors. In addition, the intricate terrain in coastal regions introduces effects such as wind blocking and acceleration, which the ECMWF model does not fully account for. In contrast, the MTRCL model leverages multi-source data and captures spatio-temporal features through an innovative architecture. This approach mitigates the impact of terrain-induced interference on wind speed predictions, thereby enhancing the model's accuracy and reliability in complex environments.

The gray histograms on the left side of each subplot in Fig. 6 represent the quantitative frequency distribution characteristics of each evaluation index within the study area. For the FA index, the histogram of the MTRCL model indicates that the false alarm rates of most grid points are concentrated in the lower value interval. In contrast, the ECMWF model exhibits a more dispersed distribution, with notably higher frequencies in regions characterized by elevated false alarm rates. Regarding RMSE and MAE indices, the histogram peaks of the MTRCL model are located within a narrower error range, indicating that the prediction errors are more concentrated and of lower magnitude. Conversely, the ECMWF model exhibits a broader error distribution, with higher frequencies in intervals corresponding to larger errors. For the MAPE index, the histogram of the MTRCL model shows that most grid points are concentrated in the lower-value interval. In contrast, the ECMWF model exhibits a more dispersed distribution, with higher frequencies in regions characterized by elevated MAPE values. The spatial distribution of various evaluation indicators reveals that the MTRCL model is markedly superior to the ECMWF model. This superiority is particularly evident in its ability to deliver more accurate and stable wind speed predictions.

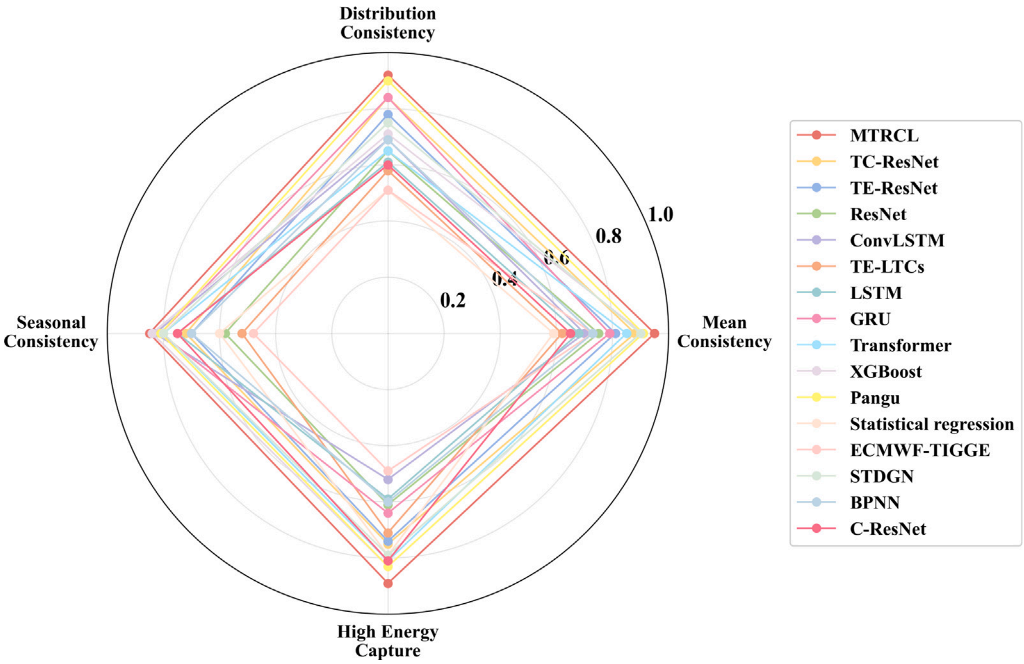


Fig. 5. Radar chart of wind energy prediction performance metrics across each model.

Fig. 7 illustrates the corrected grid wind speed forecast produced by the MTRCL model, alongside the grid wind speed forecast from the ECMWF model, and the grid wind speed data from ERA5. Generally, the corrected wind speed values are always lower than the forecast values before correction and are closer to the observed values. This indicates the improved accuracy and reliability of the MTRCL model in refining wind speed predictions.

The gray histograms on the left side of each subplot in Fig. 7 provide a quantitative display of the frequency distribution characteristics of wind speed values within the study area. The histogram corresponding to the MTRCL model indicates that wind speeds exhibit a relatively narrow distribution and a distinct peak. This suggests that the corrected wind speed forecasts generated by the MTRCL model exhibit strong consistency. In contrast, the histogram of ECMWF-TIGGE presents a more dispersed distribution, with relatively higher frequencies in the higher wind speed interval. The histogram distribution of ERA5 wind speed data is intermediate between the two but closer to the distribution pattern of the MTRCL model, further verifying the effectiveness of the MTRCL model's correction. As shown in Fig. 7, notable variations in wind speed are observed across various geographical locations within the two datasets. This figure effectively highlights the spatial distribution of wind speed predictions, showcasing both the similarities and differences between the two models. It provides valuable insights into the geographical variability of wind speed and the associated sample-related information, contributing to a comprehensive evaluation of the models' predictive capabilities.

3.4. The result of time correction by mtrcl model

To evaluate the correction effects of different models on wind speed across various time dimensions, we conducted a systematic experimental analysis, which primarily involved comparisons by season, month, and at four specific time points each day (06:00, 12:00, 18:00, and 24:00 UTC). Fig. 8 presents the seasonal comparisons of the predicted wind speeds from various models for the year 2024. Fig. 9 provides a comprehensive assessment of the monthly predicted wind speeds of each model in the year 2024. These results demonstrate that the MTRCL model has significant advantages over other models in wind speed prediction across different seasons and months.

To demonstrate the model's correction more effectively, Fig. 10 presents time-series plots and scatter plots of the forecast results of the ECMWF model with the correction results generated by the MTRCL model for four specific time points each day throughout 2024. The time-series plots demonstrate that the corrected forecast results of the MTRCL model closely align with the ERA5 wind speed data, maintaining a consistent trend of change. Additionally, the scatter plots indicate that the wind speeds forecasted after correction by the MTRCL model are more concentrated and distributed along the 1:1 diagonal line. In contrast, the ECMWF model exhibits relatively large wind speed forecast errors at various times, with notable deviations in predicting wind speed change trends.

4. Discussion

4.1. Advantages of the mtrcl model

Accurate short-term wind speed prediction is essential for the deployment, operation, and maintenance of wind power facilities, as well as ensuring the safety of low-altitude aircraft. In this work, we introduce an innovative end-to-end model architecture called MTRCL. The model constructs a dual-branch architecture that decouples and fuses spatiotemporal features, combining a multi-source data framework to demonstrate excellent capabilities in spatiotemporal collaborative modeling. With an improved CBAM integrated into the TC-ResNet module, the model emphasizes key features relevant to wind speed prediction. Meanwhile, in the TE-LTCs module, its adaptive mechanism effectively minimizes the influence of data noise on temporal modeling, leading to more stable predictions.

Furthermore, the MTRCL model excels in dynamic temporal modeling due to the deep integration of the time embedding (TE) technique within the dual-branch structure. In the TC-ResNet module, TE enables the model to detect differences in the temporal background while extracting terrain and meteorological correlations, thereby preventing deviations that result from neglecting the temporal dimension. In the TE-LTCs module, TE adjusts the parameters of ODEs and the RK4 integration step size via a gating mechanism, allowing the model to adaptively modulate the granularity of its forecasts according to

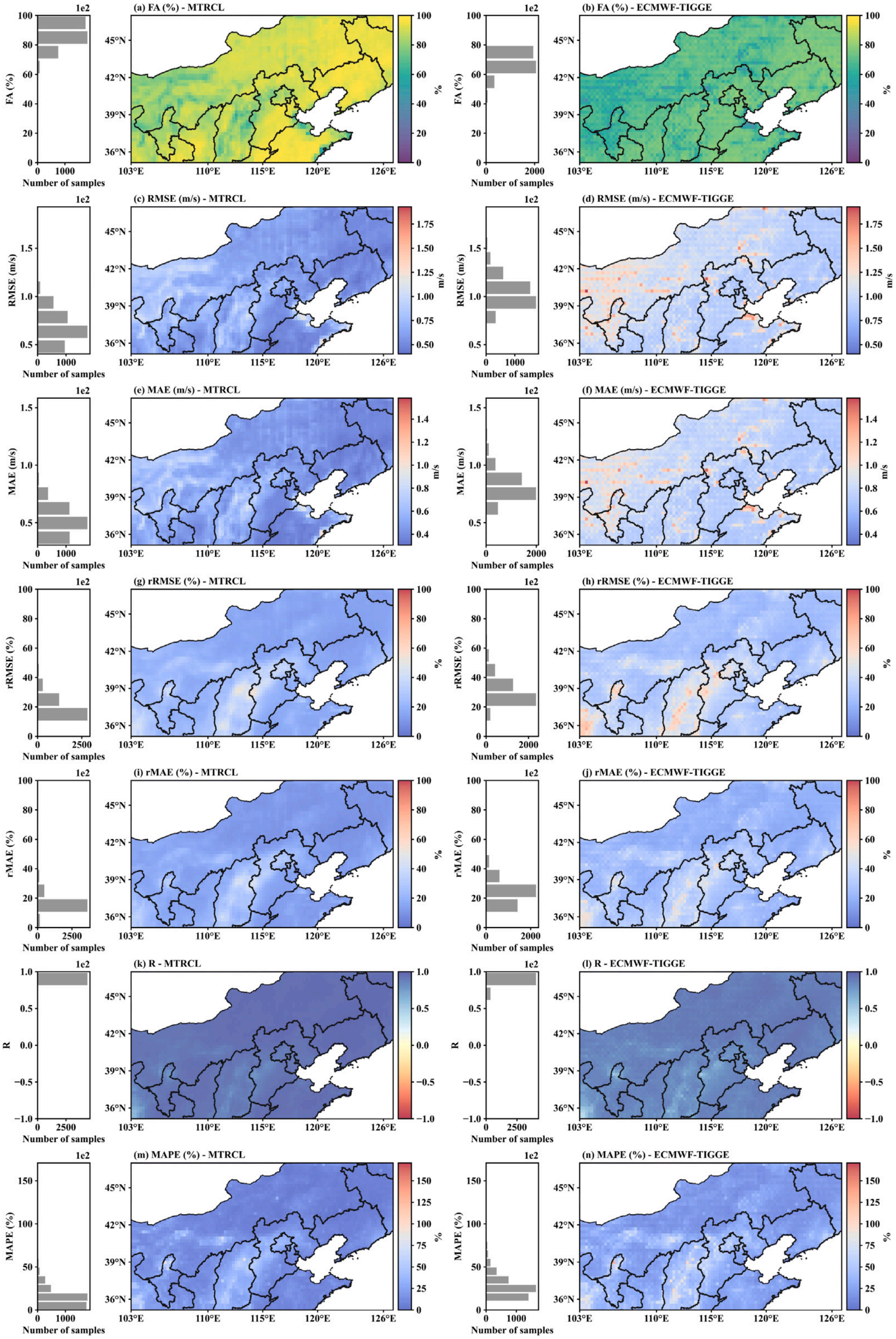


Fig. 6. Comparison of the ERA5_wind_speed in 2024 with spatial distribution maps for wind speed forecasts by the ECMWF-TIGGE model and the MTRCL model: (a–b) FA index difference maps, (c–d) RMSE index difference maps, (e–f) MAE index difference maps, (g–h) rRMSE index difference maps, (i–j) rMAE index difference maps, (k–l) R index difference maps, (m–n) MAPE index difference maps.

wind speed variations at different time scales, thereby improving the accuracy of capturing wind speed evolution over time.

Additionally, the MTRCL model demonstrates strong adaptability and robustness. In the face of the complex terrain within the study area, which features alternating highlands, canyons, and plains, the model can successfully analyze the terrain dynamic phenomena that are difficult for NWP models to represent. At the same time, this adaptability and robustness extend to the temporal dimension, enabling

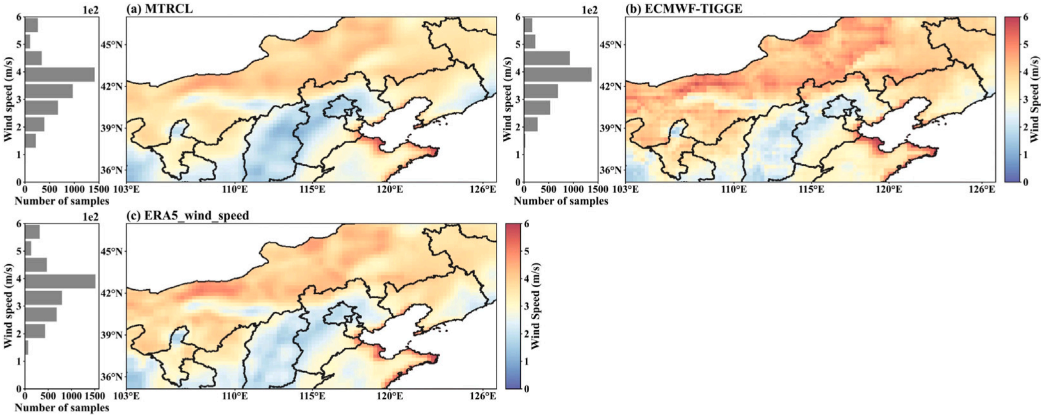


Fig. 7. Spatial distribution of the annual average wind speed in the study area in 2024: (a) the MTRCL model, (b) the ECMWF-TIGGE model, (c) the ERA5_wind_speed.

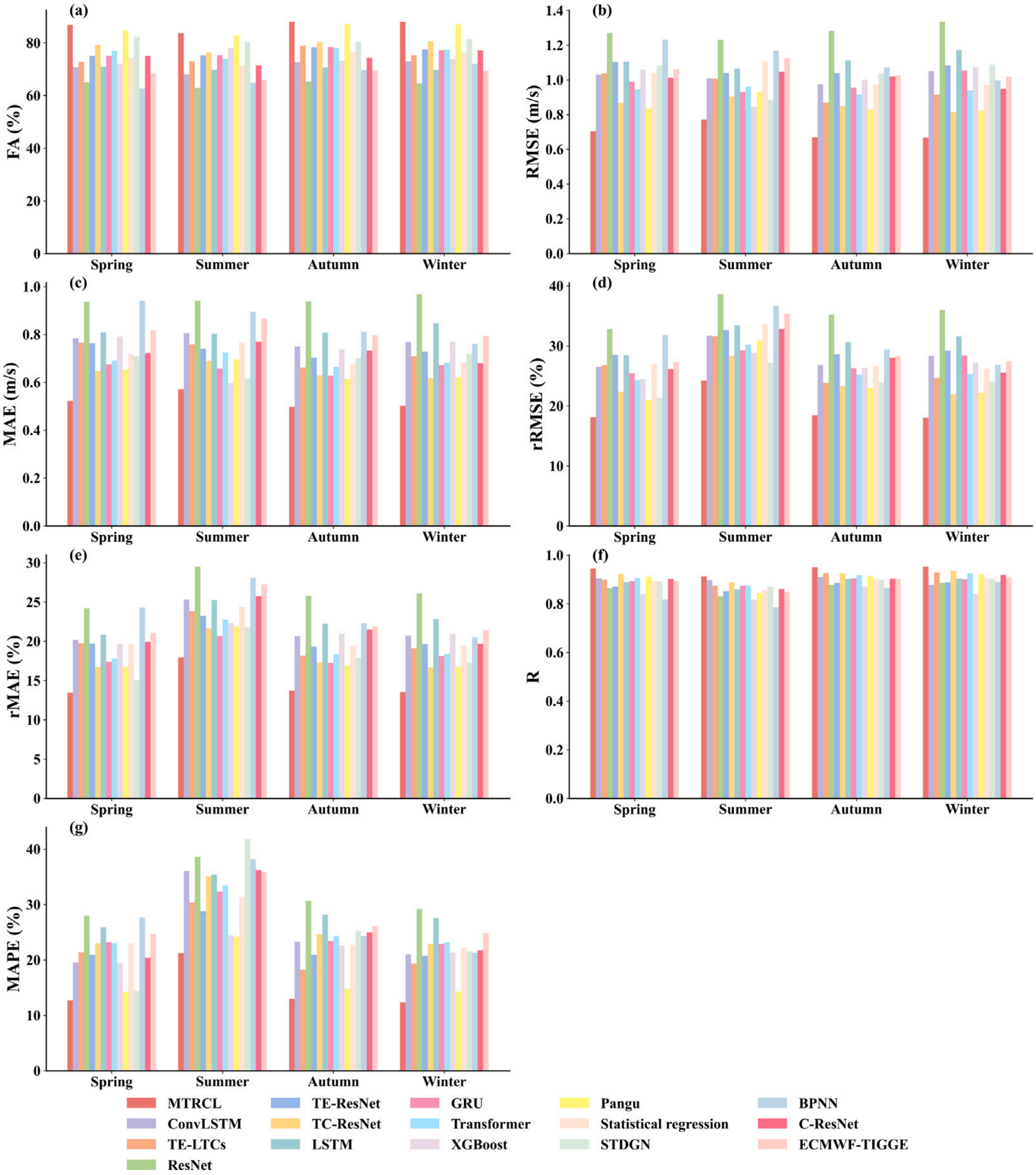


Fig. 8. (a–g) Seasonal comparison of wind speed forecasts in terms of FA, RMSE, MAE, rRMSE, rMAE, R, and MAPE indicators for each method in the year 2024.

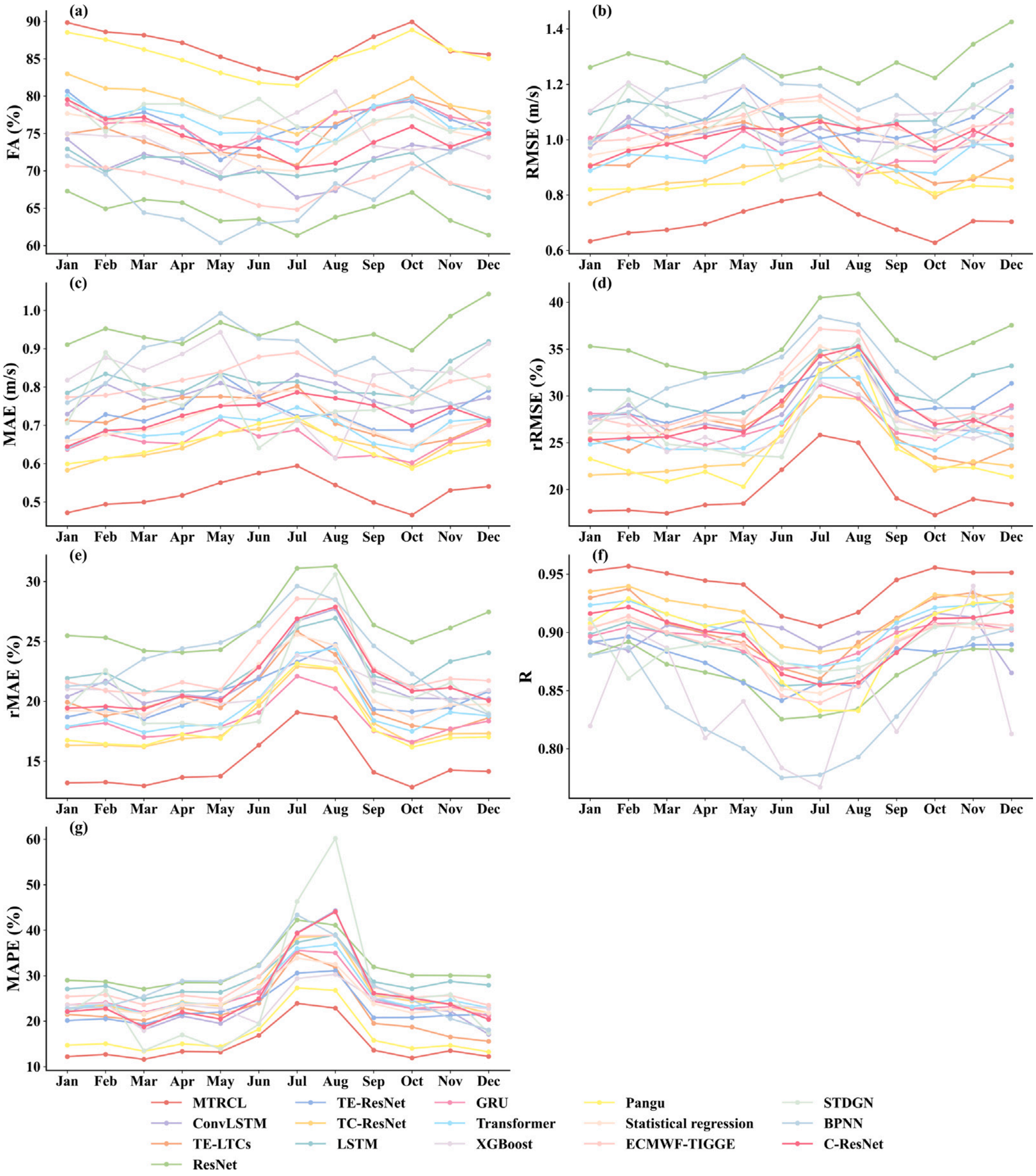


Fig. 9. (a–f) Monthly comparison of wind speed forecasts in terms of FA, RMSE, MAE, rRMSE, rMAE, R, and MAPE indicators for each method in 2024.

the model to stably cope with the dynamic changes in wind speed across different seasons and time periods.

4.2. Limitations and future directions

The proposed model has certain limitations, which can be addressed in future research. Currently, the MTRCL model mainly focuses on deterministic (point) wind speed prediction, providing specific wind speed values. However, in practical wind power generation planning and operation scenarios, assessing prediction uncertainty is equally essential. Probabilistic prediction can quantify the uncertainty of the prediction and provide more comprehensive information. Therefore, future research can explore the realization of probabilistic wind speed prediction. By combining methods such as confidence intervals for prediction results and quantile regression, the model can provide a more comprehensive basis for decision-making in wind power operations.

Meanwhile, the MTRCL model may face challenges in terms of stability and accuracy when dealing with extreme weather events. In such scenarios, the proportion of sample data in the training set is low, leading to insufficient learning of these meteorological features by the model and an increased risk of prediction bias. To address this limitation, future research could design new loss functions to enhance the model's focus on extreme events, while also exploring data augmentation techniques to simulate and generate sample data that conform to the characteristics of real extreme weather events. Additionally, when applying the model to other regions, scenarios, or data of different resolutions, fine-tuning or re-training may be necessary.

4.3. Analysis of computational complexity and training time

The computational complexity and training time of a model are key factors influencing its practical deployment. To comprehensively evaluate the computational efficiency of the proposed MTRCL model, we compare it with the baseline models discussed in this paper, focusing on four leading indicators: Gflops, forecast time, training time, and parameters. As shown in Table 9, the MTRCL model has high prediction accuracy but requires more computing resources. Its Gflops is 109, which exceeds most baselines, but its forecast time is only 19 ms. Regarding training time, the total training duration for the MTRCL model is 72 min, less than that of most baseline models. In terms of model size, the MTRCL model contains 1.68 M parameters, which is lower than several baseline models, demonstrating its parameter

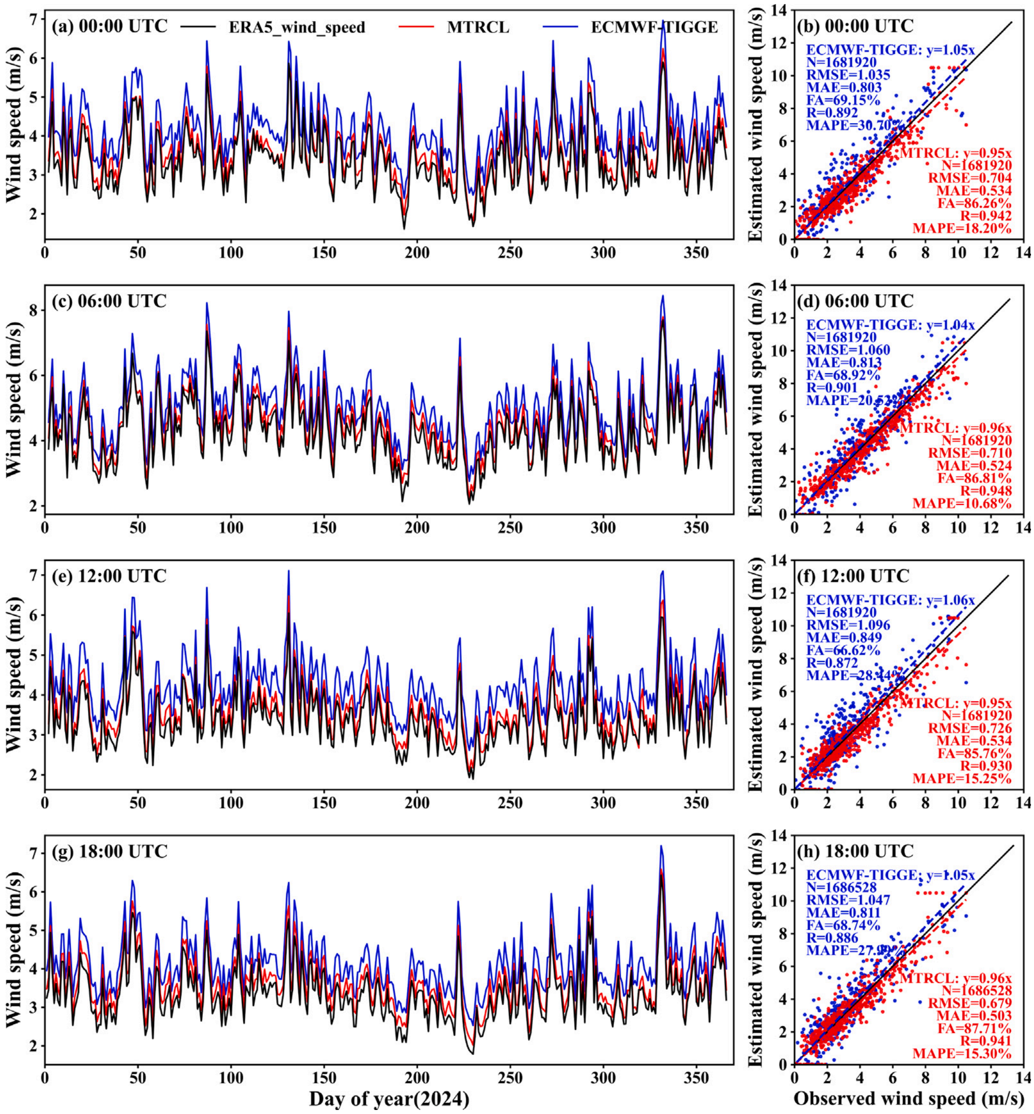


Fig. 10. (a–h) Time-series plots and scatter plots of the prediction results of the ECMWF-TIGGE model and the MTRCL model at 06:00, 12:00, 18:00, and 24:00 (i.e., 00:00) UTC each day in 2024.

efficiency. Overall, the MTRCL model strikes a good balance between computational resource needs and performance, demonstrating strong potential for practical deployment.

5. Conclusion

This study proposes an innovative method designed to enhance the accuracy of short-term wind speed predictions. The proposed model was evaluated using multi-source meteorological data from the study area from January 1, 2021, to December 31, 2024. Furthermore, comparative analyses involving multiple models, along with ablation studies, were conducted to evaluate the performance and robustness of the proposed approach. The key findings of this study are summarized as follows:

(1) The proposed MTRCL model adopts a dual-branch spatiotemporal feature fusion architecture. Specifically, the TC-ResNet module combines ResNet with the improved Convolutional Block Attention Module (CBAM) and time embedding (TE) to extract spatial features while enhancing sensitivity to temporal dimensions. Additionally, the TE-LTCs module, built upon the Liquid Time-Constant networks (LTCs), incorporates time embedding (TE) and utilizes a gating mechanism alongside meta-learning to dynamically modulate the parameters of differential equations, thereby uncovering intricate dynamic patterns in time series. This innovative design offers a robust framework for modeling and predicting complex temporal phenomena.

(2) When compared with all the baseline models in this paper, the proposed MTRCL model achieved more outstanding and consistently stable short-term wind speed predictions. The FA, RMSE, MAE, rRMSE, rMAE, R, and MAPE values obtained by the proposed method were 86.64%, 0.70, 0.52, 19.57%, 14.55%, 0.94, and 14.85%, respectively. These results indicate that the proposed model can reduce errors and more accurately capture the trends of wind speed changes.

(3) A series of ablation studies and comparative experiments were conducted to assess the contributions of individual components within the MTRCL model to its overall performance. The results demonstrate that incorporating the improved Convolutional Block Attention Module (CBAM) and the time embedding (TE) technique significantly improves

Table 9
Comparison of the computational complexity and training time of each method.

Method	Gflops	Forecast time (ms)	Training time (min)	Parameters (M)
TC-ResNet	83	36	69	0.46
TE-ResNet	74	23	70	0.44
ResNet	68	20	51	0.43
ConvLSTM	101	66	104	10.80
TE-LTCs	81	29	64	1.02
LSTM	94	58	98	0.31
GRU	105	93	111	6.58
Transformer	114	108	122	18.98
Statistical regression	47	19	73	0.003
XGBoost	77	46	119	5.47
STDGN	78	50	85	8.82
BPNN	91	37	71	1.12
C-ResNet	79	24	80	0.45
MTRCL	**109**	**19**	**72**	**1.68**

model accuracy and stability, with their combined integration yielding superior performance compared to variants lacking these components.

The proposed MTRCL model represents a promising advancement in correcting short-term wind speed forecasts. With more accurate wind speed forecast results, meteorological researchers and relevant departments can optimize meteorological service plans in advance. In the energy sector, wind power dispatch strategies can be adjusted according to the forecast results, allowing for planned power output when the predicted wind speed is high, thereby enhancing the efficiency of wind energy utilization.

CRediT authorship contribution statement

Xiaolin Ju: Writing – review & editing, Software, Methodology. **Yiming Yue:** Writing – original draft, Validation, Software, Investigation. **Meng Xu:** Writing – review & editing, Supervision, Conceptualization. **Xiang Chen:** Writing – review & editing, Supervision. **Gongzheng Lu:** Writing – review & editing, Supervision.

Declaration of competing interest

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

Acknowledgments

The TIGGE (THORPEX Interactive Grand Global Ensemble) project of ECMWF was downloaded from https://apps.ecmwf.int/datasets/data/tigge/levtype=sfc/type=cf/. The ERA5 hourly data on single levels were obtained from the Copernicus Climate Change Service Climate Data Store for this analysis [50]. The ASTER GDEM V3 [51], provided by NASA EOSDIS Land Processes DAAC, was utilized to analyze topographic characteristics of the study area. This work is supported by the open research fund of Suzhou Key Lab of Multi-modal Data Fusion and Intelligent Healthcare (No. 25SZZD11).

Data availability

The data utilized in this study, along with the implementation code for the proposed model, are publicly accessible via the following GitHub repository: https://github.com/CSNTU/MTRCL.

References

[1] Kumar Y, Ringenberg J, Depuru SS, Devabhaktuni VK, Lee JW, Nikolaidis E, Andersen B, Afjeh A. Wind energy: Trends and enabling technologies. Renew Sustain Energy Rev 2016;53:209–24. http://dx.doi.org/10.1016/j.rser.2015.07.200.

[2] McKenna R, Ostman v.d. Leye P, Fichtner W. Key challenges and prospects for large wind turbines. Renew Sustain Energy Rev 2016;53:1212–21. http://dx.doi.org/10.1016/j.rser.2015.09.080.

[3] Duan J, Zuo H, Bai Y, Duan J, Chang M, Chen B. Short-term wind speed forecasting using recurrent neural networks with error correction. Energy 2021;217:119397. http://dx.doi.org/10.1016/j.energy.2020.119397.

[4] Zheng J, Wang J. Short-term wind speed forecasting based on recurrent neural networks and levy crystal structure algorithm. Energy 2024;293:130580. http://dx.doi.org/10.1016/j.energy.2024.130580.

[5] Li H, Yu C, Xia J, Wang Y, Zhu J, Zhang P. A model output machine learning method for grid temperature forecasts in the Beijing area. Adv Atmospheric Sci 2019;36(10):1156–70. http://dx.doi.org/10.1007/s00376-019-9023-z.

[6] Półrolniczak M, Kolendowicz L, Czernecki B, Taszarek M, Tóth G. Determination of surface precipitation type based on the data fusion approach. Adv Atmospheric Sci 2021;38(3):387–99. http://dx.doi.org/10.1007/s00376-020-0165-9.

[7] Monahan AH, Culver AMR. The statistical predictability of surface winds over western and central Canada. J Clim 2013;26(21):8305–22. http://dx.doi.org/10.1175/JCLI-D-12-00425.1.

[8] Erdem E, Shi J. ARMA based approaches for forecasting the tuple of wind speed and direction. Appl Energy 2011;88(4):1405–14. http://dx.doi.org/10.1016/j.apenergy.2010.10.031.

[9] Hanifi S, Liu X, Lin Z, Lotfian S. A critical review of wind power forecasting methods—Past, present and future. Energies 2020;13(15). http://dx.doi.org/10.3390/en13153764.

[10] Jain LC, Medsker LR. Recurrent neural networks: Design and applications. first ed.. Inc. USA: CRC Press; 1999.

[11] Graves A, Graves A. Supervised sequence labelling. Springer; 2012.

[12] Pan Z, Yu W, Yi X, Khan A, Yuan F, Zheng Y. Recent progress on generative adversarial networks (GANs): A survey. IEEE Access 2019;7:36322–33. http://dx.doi.org/10.1109/ACCESS.2019.2905015.

[13] Devis A, van Lipzig NPM, Demuzere M. A new statistical approach to downscale wind speed distributions at a site in northern europe. J Geophys Res: Atmospheres 2013;118(5):2272–83. http://dx.doi.org/10.1002/jgrd.50245.

[14] Pearre NS, Swan LG. Statistical approach for improved wind speed forecasting for wind power production. Sustain Energy Technol Assessments 2018;27:180–91. http://dx.doi.org/10.1016/j.seta.2018.04.010.

[15] Zhao X, Liu J, Yu D, Chang J. One-day-ahead probabilistic wind speed forecast based on optimized numerical weather prediction data. Energy Convers Manage 2018;164:560–9. http://dx.doi.org/10.1016/j.enconman.2018.03.030.

[16] Homleid M. Diurnal corrections of short-term surface temperature forecasts using the Kalman filter. Weather Forecast 1995;10(4):689–707.

[17] Peng X, Che Y, Chang J. A novel approach to improve numerical weather prediction skills by using anomaly integration and historical data. J Geophys Res: Atmospheres 2013;118(16):8814–26. http://dx.doi.org/10.1002/jgrd.50682.

[18] Liu H, Shi J, Erdem E. Prediction of wind speed time series using modified Taylor Kriging method. Energy 2010;35(12):4870–9. http://dx.doi.org/10.1016/j.energy.2010.09.001, The 3rd International Conference on Sustainable Energy and Environmental Protection, SEEP 2009.

[19] Fawad M, Yan T, Chen L, Huang K, Singh VP. Multiparameter probability distributions for at-site frequency analysis of annual maximum wind speed with L-moments for parameter estimation. Energy 2019;181:724–37. http://dx.doi.org/10.1016/j.energy.2019.05.153.

[20] Li G, Shi J. On comparing three artificial neural networks for wind speed forecasting. Appl Energy 2010;87(7):2313–20. http://dx.doi.org/10.1016/j.apenergy.2009.12.013.

[21] Santamaría-Bonfil G, Reyes-Ballesteros A, Gershenson C. Wind speed forecasting for wind farms: A method based on support vector regression. Renew Energy 2016;85:790–809. http://dx.doi.org/10.1016/j.renene.2015.07.004.

[22] Ata R. RETRACTED: Artificial neural networks applications in wind energy systems: a review. Renew Sustain Energy Rev 2015;49:534–62. http://dx.doi.org/10.1016/j.rser.2015.04.166.

[23] Li M, Yang M, Yu Y, Lee W-J. A wind speed correction method based on modified hidden Markov model for enhancing wind power forecast. IEEE Trans Ind Appl 2022;58(1):656–66. http://dx.doi.org/10.1109/TIA.2021.3127145.

[24] Zhou S, Gao CY, Duan Z, Xi X, Li Y. A robust error correction method for numerical weather prediction wind speed based on Bayesian optimization, variational mode decomposition, principal component analysis, and random forest: VMD-pca-RF (version 1.0.0). Geosci Model Dev 2023;16(21):6247–66. http://dx.doi.org/10.5194/gmd-16-6247-2023.

[25] Sun L, Lan Y, Sun X, Liang X, Wang J, Su Y, He Y, Xia D. Deterministic forecasting and probabilistic post-processing of short-term wind speed using statistical methods. J Geophys Res: Atmospheres 2024;129(7):e2023JD040134. http://dx.doi.org/10.1029/2023JD040134, e2023JD040134 2023JD040134.

[26] Wang A, Xu L, Li Y, Xing J, Chen X, Liu K, Liang Y, Zhou Z. Random-forest based adjusting method for wind forecast of WRF model. Comput Geosci 2021;155:104842. http://dx.doi.org/10.1016/j.cageo.2021.104842.

[27] Yang M, Guo Y, Fan F, Huang T. Two-stage correction prediction of wind power based on numerical weather prediction wind speed superposition correction and improved clustering. Energy 2024;302:131797. http://dx.doi.org/10.1016/j.energy.2024.131797.

[28] Zhang Y, Chen B, Pan G, Zhao Y. A novel hybrid model based on VMD-WT and PCA-BP-RBF neural network for short-term wind speed forecasting. Energy Convers Manage 2019;195:180–97. http://dx.doi.org/10.1016/j.enconman.2019.05.005.

[29] Kim J, Kwon M, Kim SD, et al. Spatiotemporal neural network with attention mechanism for el Niño forecasts. Sci Rep 2022;12(1):7204. http://dx.doi.org/10.1038/s41598-022-10839-z.

[30] Han L, Chen M, Chen KK, Chen H, Zhang YB, Lu B, Song L, Qin R. A deep learning method for bias correction of ecmwf 24–240 h forecasts. Adv Atmospheric Sci 2021;38(9):1444–59. http://dx.doi.org/10.1007/s00376-021-0215-y.

[31] Gong Z, Wan A, Ji Y, AL-Bukhaiti K, Yao Z. Improving short-term offshore wind speed forecast accuracy using a VMD-PE-FCGRU hybrid model. Energy 2024;295:131016. http://dx.doi.org/10.1016/j.energy.2024.131016.

[32] Xu W, Liu P, Cheng L, Zhou Y, Xia Q, Gong Y, Liu Y. Multi-step wind speed prediction by combining a WRF simulation and an error correction strategy. Renew Energy 2021;163:772–82. http://dx.doi.org/10.1016/j.renene.2020.09.032.

[33] Luo X, Sun J, Wang L, Wang W, Zhao W, Wu J, Wang J-H, Zhang Z. Short-term wind speed forecasting via stacked extreme learning machine with generalized correntropy. IEEE Trans Ind Inform 2018;14(11):4963–71. http://dx.doi.org/10.1109/TII.2018.2854549.

[34] Zhang J, Zhang Y, Liu K, Zhao C. Multi-step prediction of spatio-temporal wind speed based on the multimodal coupled ST-dfnet model. Energy 2025;137670. http://dx.doi.org/10.1016/j.energy.2025.137670.

[35] Zhang Z, Lin L, Gao S, Wang J, Zhao H, Yu H. A machine learning model for hub-height short-term wind speed prediction. Nat Commun 2025;16(3195). http://dx.doi.org/10.1038/s41467-025-58456-4.

[36] Wang H, Song S, Huang G. GRIDMIX: Exploring spatial modulation for neural fields in pde Modeling. In: International conference on learning representations. ICLR, Beijing, China: Department of Automation, BNRist, Tsinghua University; 2025.

[37] Donk S, Wagner L, Skidmore E, Tatarko J. Comparison of the Weibull model with measured wind speed distributions for stochastic wind generation. Trans Am Soc Agric Eng 2005;48. http://dx.doi.org/10.13031/2013.18324.

[38] Shukur O, Lee MH. Imputation of missing values in daily wind speed data using hybrid AR-ANN method. Mod Appl Sci 2015;9:1. http://dx.doi.org/10.5539/mas.v9n11p1.

[39] Hong Y-Y, Satriani TRA. Day-ahead spatiotemporal wind speed forecasting using robust design-based deep learning neural network. Energy 2020;209:118441. http://dx.doi.org/10.1016/j.energy.2020.118441.

[40] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: 2016 IEEE conference on computer vision and pattern recognition. CVPR, 2016, p. 770–8. http://dx.doi.org/10.1109/CVPR.2016.90.

[41] Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput 1997;9(8):1735–80. http://dx.doi.org/10.1162/neco.1997.9.8.1735.

[42] Hasani RM, Lechner M, Amini A, Rus D, Grosu R. Liquid time-constant networks. 2020, CoRR, abs/2006.04439.

[43] Bi K, Xie L, Zhang H, et al. Accurate medium-range global weather forecasting with 3D neural networks. Nature 2023;619(7970):533–8. http://dx.doi.org/10.1038/s41586-023-06185-3.

[44] Zeng Z, Wu H, Liu Z, Zhao L, Liang Z, Liang Z, Wang Y. Enhancing short-term wind speed prediction capability of numerical weather prediction through machine learning methods. J Geophys Res: Atmospheres 2024;129(24):e2024JD041822. http://dx.doi.org/10.1029/2024JD041822, e2024JD041822 2024JD041822.

[45] Ishizuka M, Mikami M, Yamada Y, Zeng F, Gao W. An observational study of soil moisture effects on wind erosion at a gobi site in the taklimakan desert. J Geophys Res: Atmospheres 2005;110(D18). http://dx.doi.org/10.1029/2004JD004709.

[46] Helbig N, Mott R, van Herwijnen A, Winstral A, Jonas T. Parameterizing surface wind speed over complex topography. J Geophys Res: Atmospheres 2017;122(2):651–67. http://dx.doi.org/10.1002/2016JD025593.

[47] Wang J, Lin L, Gao S, Zhang Z. Deep generation network for multivariate spatio-temporal data based on separated attention. Inform Sci 2023;633:85–103. http://dx.doi.org/10.1016/j.ins.2023.03.062.

[48] Diebold F, Mariano R. Comparing predictive accuracy. J Bus Econom Statist 2002;20(1):134–44. http://dx.doi.org/10.1198/073500102753410444.

[49] Miao H, Dong D, Huang G, Hu K, Tian Q, Gong Y. Evaluation of northern hemisphere surface wind speed and wind power density in multiple reanalysis datasets. Energy 2020;200:117382. http://dx.doi.org/10.1016/j.energy.2020.117382.

[50] Hersbach H, Bell B, Berrisford P, Biavati G, Horányi A, Muñoz Sabater J, Nicolas J, Peubey C, Radu R, Rozum I, Schepers D, Simmons A, Soci C, Dee D, Thépaut J-N. ERA5 hourly data on single levels from 1940 to present. 2023, http://dx.doi.org/10.24381/cds.adbb2d47, [dataset]. [Accessed 21 March 2025].

[51] NASA/METI/AIST/Japan Spacesystems and US/Japan ASTER Science Team. ASTER global digital elevation model V003. 2019, http://dx.doi.org/10.5067/ASTER/ASTGTM.003, [dataset]. Date [Accessed: 21 March 2025].