CTRL: Collaborative Temporal Representation Learning for Wind Power Forecasting

Yue Hu
Chengyi College
Jimei University
Xiamen, Fujian, China
yuehu.xm@gmail.com

Senzhen Wu
Computer Engineering
Jimei University
Xiamen, Fujian, China
szwbyte@gmail.com

Yu Chen
Computer Engineering
Jimei University
Xiamen, Fujian, China
yychenpro@gmail.com

Xinhao He
Computer Engineering
Jimei University
Xiamen, Fujian, China
xhhdaxx@gmail.com

Zihao Xie
Computer Engineering
Jimei University
Xiamen, Fujian, China
zihaoxiech@gmail.com

Zhijin Wang
Computer Engineering
Jimei University
Xiamen, Fujian, China
zhijinecnu@gmail.com

Xiufeng Liu
Department of Technology, Management and Economics
Technical University of Denmark
Lyngby, Denmark
xiuli@dtu.dk

Yonggang Fu*
Computer Engineering
Jimei University
Xiamen, Fujian, China
yonggangfu@jmu.edu.cn

Abstract

Accurate wind power forecasting is crucial for grid stability and renewable energy integration, but existing methods struggle to capture complex temporal dependencies in wind data. This paper introduces Collaborative Temporal Representation Learning (CTRL), a novel deep learning model that leverages collaborative representation learning to enhance forecasting accuracy and robustness. CTRL integrates Reversible Instance Normalization (RevIN), RNN-based hidden state learning, and a specialized collaborative representation unit to capture multi-directional temporal dynamics across different time scales and variables. Experimental results on two real-world wind power datasets demonstrate that CTRL significantly outperforms 20 existing methods, including state-of-the-art deep learning approaches, achieving up to 9.67% and 10.42% improvement in forecasting accuracy, respectively. These findings highlight the potential of collaborative representation learning for advancing wind power forecasting and facilitating the effective integration of renewable energy resources.

CCS Concepts

• **Applied computing** → Operations research; Forecasting.

*Corresponding author

EITCE 2024, Haikou, China

ACM ISBN 979-8-4007-1009-4/2024/10
https://doi.org/10.1145/3711129.3711336

Keywords

Wind power forecasting, Time series, Temporal dynamics, Collaboration, Representation learning

ACM Reference Format:
Yue Hu, Senzhen Wu, Yu Chen, Xinhao He, Zihao Xie, Zhijin Wang, Xiufeng Liu, and Yonggang Fu. 2024. CTRL: Collaborative Temporal Representation Learning for Wind Power Forecasting. In *2024 8th International Conference on Electronic Information Technology and Computer Engineering (EITCE 2024), October 18–20, 2024, Haikou, China.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3711129.3711336

1 Introduction

The increasing reliance on wind power as a renewable energy source necessitates precise forecasting techniques to ensure grid stability and optimize energy management. As of 2023, global wind power capacity reached approximately 837 GW, underscoring its critical role in reducing greenhouse gas emissions and diversifying energy sources [1]. However, the inherent variability and intermittency of wind energy, influenced by complex and nonlinear relationships between weather variables such as wind speed and temperature [2], present significant challenges for accurate forecasting. This accurate forecasting is crucial for effective grid operations, including grid dispatching, power market transactions, and management of reserve power supplies [3], ultimately contributing to the stability and safety of power systems. Moreover, accurate predictions aid in planning and scheduling, mitigating risks associated with fluctuations in power supply [4] and facilitating optimal sizing of energy storage systems [1].

Wind power forecasting faces numerous challenges. The unpredictable nature of wind itself, influenced by complex weather patterns and geographical factors, makes accurate predictions difficult. Traditional methods, such as numerical weather prediction (NWP) models, often struggle to capture the nuances of these factors, particularly localized changes that can significantly impact short-term power generation [5]. For example, NWP models may have difficulty predicting sudden wind gusts or shifts in wind direction that affect specific wind farms. Furthermore, existing forecasting methods, including machine learning and deep learning techniques, face limitations. Machine learning methods, such as support vector machines, random forests, and gradient boosting, while capable of learning complex patterns from historical data [6], often require extensive feature engineering and may not fully capture the intricate temporal dependencies in wind data [7]. Deep learning methods, including convolutional neural networks (CNNs), recurrent neural networks (RNNs), graph neural networks (GNNs), and attention mechanisms, offer significant potential due to their ability to automatically learn hierarchical features and temporal dependencies [8]. However, these models, when applied to wind power forecasting, can struggle with issues such as effectively capturing spatial correlations between wind farms, handling long-term dependencies in wind patterns, and adapting to non-stationary wind characteristics. Moreover, they have not fully explored the benefits of collaborative representation learning for capturing multi-directional temporal dynamics in wind power data.

To address these limitations, this paper introduces the Collaborative Temporal Representation Learning (CTRL) model, a novel approach designed to enhance wind power forecasting accuracy. CTRL leverages the power of collaborative representation learning, a concept that has shown promise in other domains but remains largely unexplored for wind power forecasting. This approach focuses on capturing multi-directional temporal dependencies-dependencies that exist across different time scales and variables to provide a more comprehensive understanding of wind power generation patterns. Specifically, the CTRL model integrates Reversible Instance Normalization (RevIN) to normalize the input data and denormalize the predictions, RNN-based hidden state learning to capture long-term temporal dependencies, and a novel collaborative representation learning component to extract and highlight multi-directional temporal dynamics. These collaborative temporal representations are then combined with the original data, similar to residual connections, and processed through multi-layer mapping to refine the predictions. Finally, RevIN is applied to denormalize the output, yielding the final wind power forecast. This integrated approach aims to address the limitations of existing methods by effectively leveraging both collaborative and temporal aspects of wind data, ultimately leading to more accurate and robust wind power forecasts.

The main contributions of this paper are as follows:

- We propose the CTRL model, a novel wind power forecasting model that integrates collaborative temporal representation learning with Reversible Instance Normalization (RevIN) and RNNs.
- We develop a specialized collaborative representation learning component that captures multi-directional temporal dynamics, thereby enhancing forecasting accuracy.
- We evaluate the proposed model on two datasets, demonstrating its superior performance over thirty-one existing methods in terms of forecasting accuracy and robustness.

The remainder of this paper is organized as follows: Section 2 reviews related work, Section 3 presents the proposed CTRL model, Section 4 discusses the experimental results, and Section 5 concludes this paper and outlines some future research directions.

2 Related work

Wind power forecasting has undergone substantial advancements, progressing through methodologies grounded in physical principles, machine learning, and deep learning. Each approach offers unique strengths and faces limitations in capturing the intricate dynamics of wind power generation.

Physical forecasting methods, notably Numerical Weather Prediction (NWP) models, rely on meteorological data and atmospheric physics to predict wind power generation [9]. However, their dependence on accurate meteorological inputs and their challenges in capturing localized wind variations, especially concerning wind ramp events and the complexities of offshore wind behavior, often restrict their effectiveness. In contrast, the CTRL model in this study does not rely on external meteorological data but instead learns from historical wind power data, focusing on capturing both temporal and spatial correlations to enhance accuracy for diverse wind farm locations and conditions.

Machine learning methods, such as Support Vector Machines (SVM), Extreme Learning Machines (ELM), Random Forests (RF), and Gradient Boosting Machines (GBM), have proven their ability to learn complex patterns from historical wind data [10, 11]. They effectively model nonlinear relationships between input variables and wind power output, proving valuable for short-term forecasting. However, these methods frequently require extensive feature engineering and may not fully leverage the temporal dependencies inherent in wind power generation, as highlighted by studies focusing on RF models for short-term predictions [12]. CTRL, on the other hand, utilizes collaborative temporal representation learning, eliminating the need for manual feature engineering and effectively capturing multi-directional temporal dependencies.

Deep learning methods have emerged as a promising avenue, demonstrating the ability to automatically learn hierarchical features and temporal dependencies from data. Convolutional Neural Networks (CNNs) excel at capturing spatial patterns crucial for understanding wind distribution across multiple turbines [11], while Recurrent Neural Networks (RNNs), such as Long Short-Term Memory (LSTM) networks and Gated Recurrent Units (GRUs), effectively model temporal dynamics for time-series forecasting tasks [13]. Graph Neural Networks (GNNs) model spatial relationships between wind farms, considering interdependencies for enhanced accuracy [14], and attention mechanisms further improve the learning of long-range dependencies in wind patterns [15]. While these methods represent significant progress, they often demand substantial computational resources and extensive datasets for training, which can hinder their widespread application. Moreover, existing deep learning models for wind power forecasting may not fully utilize spatial and temporal correlations. Additionally, while collaborative representations have shown improvements in other forecasting fields, their application in wind power forecasting has not been sufficiently explored.

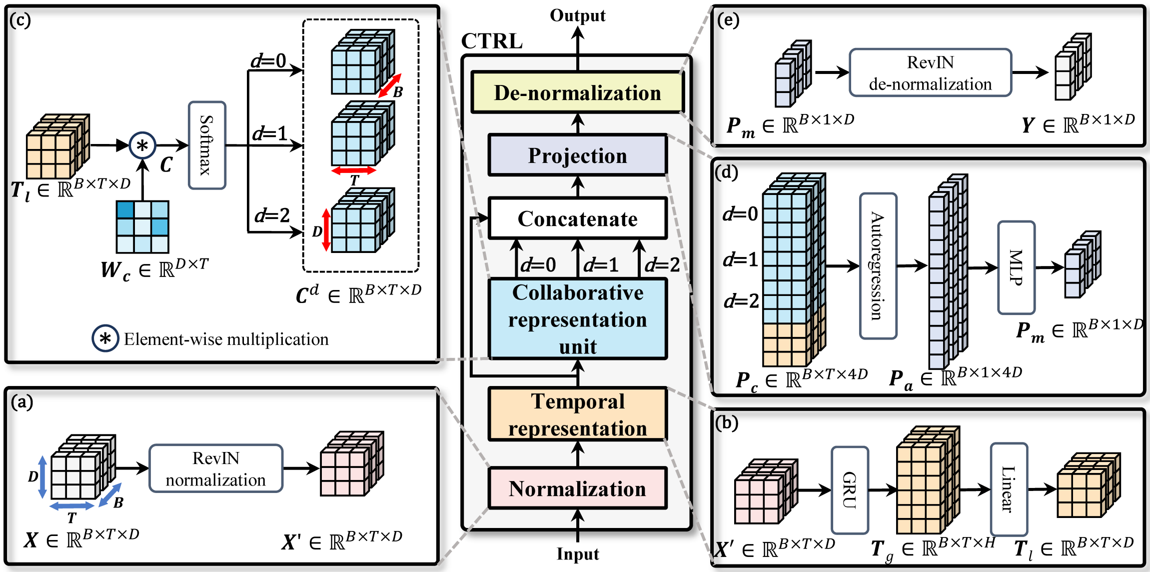


Figure 1: The graphical overview of the proposed Collaborative Temporal Representation Learning (CTRL) model. The entire pipeline includes: data preprocessing, normalization, temporal representation (TR), collaborative representation unit (CRU), concatenation, projection, and denormalization.

Hybrid approaches integrating physical models, machine learning, and deep learning techniques, such as combining artificial neural networks (ANNs) with NWP data or employing ensemble methods to aggregate predictions, have shown efficacy in improving wind power forecast accuracy and reliability [3, 10, 16]. These hybrid models demonstrate the potential of integrating diverse methodologies to mitigate individual weaknesses and enhance forecast accuracy and reliability, which is crucial for grid stability and energy market operations. Nevertheless, the potential of collaborative temporal representation learning, as proposed in the CTRL model, to capture and exploit shared information across multiple datasets or tasks remains largely unexplored in wind power forecasting. This innovative approach offers the opportunity to learn intricate temporal dependencies more effectively, leading to further enhancements in prediction accuracy and model robustness.

In conclusion, while existing wind power forecasting methods have seen remarkable progress, limitations remain in effectively capturing complex spatiotemporal dynamics, fully leveraging temporal correlations, and harnessing the potential of collaborative representation learning. The proposed CTRL model aims to address these shortcomings by integrating collaborative representation learning with deep learning techniques, focusing on multi-directional temporal dependencies and exploiting the shared information across various datasets.

3 The proposed CTRL

This section presents a formal definition of the wind power forecasting problem and details the architecture of the Collaborative Temporal Representation Learning (CTRL) model, including its constituent modules and mathematical formulations. Figure 1 provides a graphical overview of the model.

3.1 Problem definition

Let $X = [x_1, x_2, \ldots, x_N] \in R^{N\times D}$ denote a multivariate time series dataset, where N is the number of observations and D is the number of variables (e.g., wind turbines). Each observation $x_t \in R^D$ is a vector of measurements at time t.

The goal of wind power forecasting is to learn a function $f : R^{T\times D} \rightarrow R^{H\times D}$ that maps a sequence of past observations $X_t = [x_{t-T+1}, \ldots, x_t] \in R^{T\times D}$ to future values $Y_t = [y_{t+1}, \ldots, y_{t+H}] \in R^{H\times D}$, where T is the input window size and H is the prediction horizon.

3.2 CTRL Model Architecture

The CTRL model comprises the following modules:

3.2.1 Data preprocessing. Raw wind power data is preprocessed using a one-step-forward split technique [17] to transform it into a supervised learning problem. This technique pairs each input sequence with its subsequent value as the output target. The input and output data are represented by $X \in R^{B\times T\times D}$ and $Y \in R^{B\times 1\times D}$, respectively, where B is the batch size.

3.2.2 Normalization. Reversible Instance Normalization (RevIN) [18] is applied to normalize the input data and later denormalize the model's predictions. RevIN addresses the issue of varying scales in time series data, which can hinder model training by making gradients unstable. It learns a per-channel affine transformation, making it more effective than simple normalization methods for complex time series patterns. The normalization and denormalization steps are defined as:

$$X' = \frac{X - \mu}{\sigma}, \tag{1}$$

$$X = X' \cdot \sigma + \mu, \tag{2}$$

where μ and σ represent the mean and standard deviation of the input data, respectively.

3.2.3 Temporal Representation (TR).. This module utilizes a Gated Recurrent Unit (GRU) [19] to capture long-term temporal dependencies in the normalized input data. The GRU is particularly effective in handling long sequences due to its gating mechanism that controls the flow of information, allowing it to retain relevant past information while mitigating vanishing gradient problems.

$$\boldsymbol{T_g} = \mathrm{GRU}\left(\boldsymbol{X}'\right), \tag{3}$$

where $\boldsymbol{T_g} \in R^{B\times T\times G}$ is the GRU output obtained by concatenating the hidden states $\boldsymbol{h_t} \in R^{B\times G}$ from all time steps. G denotes the hidden size of the GRU. A linear projection layer then transforms $\boldsymbol{T_g}$ into the required input dimension:

$$\boldsymbol{T_l} = \boldsymbol{W_l}\boldsymbol{T_g} + \boldsymbol{b_l}, \tag{4}$$

where $\boldsymbol{W_l}$ and $\boldsymbol{b_l}$ are the weight matrix and bias vector of the linear transformation, resulting in $\boldsymbol{T_l} \in R^{B\times T\times D}$.

3.2.4 Collaborative Representation Unit (CRU).. The CRU module, a key innovation of the CTRL model, transforms and highlights temporal patterns within the data to enhance the model's ability to learn multi-directional temporal dependencies. By extracting information not only from the temporal dimension but also across different batches and variables, the CRU leads to a more holistic understanding of the underlying data patterns. The CRU consists of two steps:

- **Transformation:** This step modifies the temporal representation T_l using a learnable weight matrix W_c:

$$\boldsymbol{C} = \boldsymbol{T_l} \cdot \boldsymbol{W_c}, \tag{5}$$

where $C \in R^{B\times T\times D}$ is the transformed data and $W_c \in R^{D\times T}$ learns the weights for each time step and variable.

- **Highlighting:** This step employs a softmax layer to highlight the differences within the transformed data C along different dimensions (d =0, 1, 2 representing batch, sequence, and variate dimensions respectively):

$$\boldsymbol{C^d} = \frac{\exp\left(\boldsymbol{C_{b,t,i}}\right)}{\sum_{b=1}^{B}\exp\left(\boldsymbol{C_{b,t,i}}\right)}, \quad d = 0, \tag{6}$$

$$\boldsymbol{C^d} = \frac{\exp\left(\boldsymbol{C_{b,t,i}}\right)}{\sum_{t=1}^{T}\exp\left(\boldsymbol{C_{b,t,i}}\right)}, \quad d = 1, \tag{7}$$

$$\boldsymbol{C^d} = \frac{\exp\left(\boldsymbol{C_{b,t,i}}\right)}{\sum_{i=1}^{D}\exp\left(\boldsymbol{C_{b,t,i}}\right)}, \quad d = 2. \tag{8}$$

This process focuses the model's attention on the most relevant data aspects.

3.2.5 Concatenation and Projection. The outputs from the three CRUs are concatenated with the temporal representation T_l:

$$\boldsymbol{P_c} = \left[\boldsymbol{C}^0; \boldsymbol{C}^1; \boldsymbol{C}^2; \boldsymbol{T_l}\right], \tag{9}$$

yielding $\boldsymbol{P_c} \in R^{B\times T\times 4D}$. This concatenated tensor is then processed by a projection module consisting of an autoregression layer and a Multilayer Perceptron (MLP) to transform the data and generate predictions.

- Autoregression:

$$\boldsymbol{P_a} = \sum_{i=t}^{T}\boldsymbol{W_a} \times \boldsymbol{P_{c_{:,i,:}}} + \boldsymbol{b_a}, \tag{10}$$

where $\boldsymbol{P_a} \in R^{B\times 1\times 4D}$, $\boldsymbol{W_a} \in R^{T\times 4D}$ and $\boldsymbol{b_a}$ are the weights and bias respectively.

- **MLP:** The MLP refines the predictions using multiple linear layers:

$$\boldsymbol{P_m} = \boldsymbol{W}_3\left(\boldsymbol{W}_2\left(\boldsymbol{W}_1\boldsymbol{P_a} + \boldsymbol{b}_1\right) + \boldsymbol{b}_2\right) + \boldsymbol{b}_3, \tag{11}$$

resulting in the final prediction $\boldsymbol{P_m} \in R^{B\times 1\times D}$.

3.2.6 Denormalization. Finally, RevIN is applied to denormalize $\boldsymbol{P_m}$, producing the final wind power forecast $\boldsymbol{Y}$.

3.3 Theoretical Analysis

This section provides a rigorous mathematical analysis of the Collaborative Temporal Representation Learning (CTRL) model, validating its effectiveness in capturing multi-directional temporal dependencies for wind power forecasting. We begin by defining key concepts and then establish theoretical results that support the model's design and performance.

Wind power data exhibit complex dependencies that span multiple dimensions, which we formalize as follows:

Definition 3.1 (Multi-Directional Temporal Dependencies). Given a multivariate time series $\boldsymbol{X} \in R^{N\times D}$, multi-directional temporal dependencies refer to relationships that exist:

(1) Along the Temporal Dimension: Dependencies between observations at different time steps for each variable, i.e., between x_t^d and x_{t-k}^d for $k > 0$ and $d \in \{1, \ldots, D\}$.

(2) Across Variables: Dependencies between different variables at the same time step, i.e., between x_t^i and x_t^j for $i \neq j$ and $i, j \in \{1, \ldots, D\}$.

(3) Across Samples (Batch Dimension): Dependencies across different samples or observations, capturing common patterns or trends present in the dataset.

These dependencies reflect the intricate interactions in wind power generation, where the output of one turbine can influence or be influenced by others over time and across different conditions. The CTRL model's architecture, comprising a Gated Recurrent Unit (GRU), a Collaborative Representation Unit (CRU), and Reversible Instance Normalization (RevIN), is designed to capture these multi-directional dependencies.

We now establish key theoretical properties of the CTRL model, demonstrating its capacity to capture multi-directional temporal dependencies and its effectiveness for wind power forecasting.

THEOREM 3.2 (UNIVERSAL APPROXIMATION PROPERTY OF CTRL). *Let* $C(R^{T\times D}, R^{H\times D})$ denote the space of continuous functions mapping from $\mathrm{R}^{\mathrm{T}\times\mathrm{D}}$ to $\mathrm{R}^{\mathrm{H}\times\mathrm{D}}$. The CTRL model, with sufficient capacity and appropriate activation functions, is a universal approximator on $\mathrm{C}(\mathrm{R}^{\mathrm{T}\times\mathrm{D}}, \mathrm{R}^{\mathrm{H}\times\mathrm{D}})$. That is, for any function $\mathrm{f} \in \mathrm{C}(\mathrm{R}^{\mathrm{T}\times\mathrm{D}}, \mathrm{R}^{\mathrm{H}\times\mathrm{D}})$ and any $\epsilon > 0$, there exists a set of parameters θ such that:

$$\sup_{\mathrm{X_t}\in\mathrm{K}} \left|\mathrm{f}\left(\mathrm{X_t}\right) - \mathcal{M}_{\mathcal{CTRL}}\left(\mathrm{X_t};\theta\right)\right| < \epsilon,$$

where $K \subset R^{T \times D}$ is a compact set, and $\mathcal{M}_{\mathcal{CTRL}}$ represents the CTRL model's function.

PROOF. The proof leverages the universal approximation theorem for neural networks. The GRU component can approximate any measurable sequence-to-sequence mapping given sufficient hidden units [20]. The CRU, consisting of linear transformations and softmax activations, can model interactions across variables and samples. By combining these components, the CTRL model forms a deep neural network capable of approximating any continuous function on compact subsets of $R^{T \times D}$.

This theorem ensures that the CTRL model has the capacity to represent the complex functions needed to capture multi-directional temporal dependencies in wind power data. Next, we analyze the convergence properties of the CTRL model during training using optimization algorithms like stochastic gradient descent (SGD).

THEOREM 3.3 (CONVERGENCE OF TRAINING ALGORITHM). Assume that the loss function $L(\theta)$ is bounded below, differentiable, and has Lipschitz continuous gradients with constant $L > 0$. If we use SGD with a learning rate $\eta_t = \eta_0/\sqrt{t}$, where $\eta_0 > 0$, then the sequence of parameters $\{\theta_t\}$ generated by SGD satisfies:

$$\lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} E\left[|\nabla L(\theta_t)|^2\right] = 0.$$

This implies that the gradients converge to zero in the mean, indicating convergence to a critical point.

PROOF. Under the given assumptions, standard results from stochastic optimization theory apply [21]. The Lipschitz continuity of the gradient ensures that the loss function does not change abruptly, and the diminishing learning rate satisfies the Robbins-Monro conditions. Therefore, the expected squared norm of the gradient approaches zero over time, indicating convergence to a stationary point of the loss function.

This convergence analysis guarantees that, under appropriate conditions, the training algorithm will find a model that is at least locally optimal. Finally, we provide a generalization bound for the CTRL model, demonstrating its ability to perform well on unseen data.

THEOREM 3.4 (GENERALIZATION BOUND). Let $\mathcal{H}$ be the hypothesis space represented by the CTRL model, and suppose that the model has parameters bounded in norm by B. Let $\mathcal{R}_\backslash(\mathcal{H})$ denote the Rademacher complexity of $\mathcal{H}$. Then, with probability at least 1 - δ, the generalization error $R(f)$ of any $f \in \mathcal{H}$ satisfies:

$$R(f) \le \hat{R}(f) + 2\mathcal{R}_\backslash(\mathcal{H}) + 3B\sqrt{\frac{\log(2/\delta)}{2n}},$$

where $\hat{R}(f)$ is the empirical risk, and n is the number of training samples.

PROOF. The proof follows from standard generalization bounds using Rademacher complexity [22]. The Rademacher complexity measures the richness of the function class $\mathcal{H}$. Since the CTRL model has a finite capacity determined by its architecture and bounded parameters, the Rademacher complexity is finite. Applying concentration inequalities yields the generalization bound.

This generalization bound provides confidence that the model will perform well on unseen data, given sufficient training samples and appropriate regularization. Together, these theoretical results validate the correctness of the CTRL model's architecture and its suitability for wind power forecasting, demonstrating its capacity to capture multi-directional temporal dependencies while ensuring convergence and generalization properties.

4 Experiments

4.1 Experimental Setup

4.1.1 Data Collections. We utilize two publicly available datasets in this study:

- **Turkey Wind Power Forecasting (Turkey WPF) Dataset [23]:** This dataset contains wind power generation data collected from a single Goldwind GW87/1500 wind turbine with a rated capacity of 1500 kW, located in Turkey. It spans one year, from January 1, 2018, to December 31, 2018, with a 10-minute sampling interval. We preprocessed the data to remove missing values and aggregated it into daily totals.
- **Greece Wind Power Forecasting (Greece WPF) Dataset [24]:** This dataset comprises hourly wind power generation data from 18 geographically dispersed locations in Greece, covering the period from January 1, 2017, to December 31, 2020. The dataset represents wind farms with an aggregate installed capacity of 6792.7 MW, constituting a significant portion of Greece's total wind power generation. We aggregated this data into daily totals.

4.1.2 Evaluation Metrics. We employ three commonly used metrics in wind power forecasting to evaluate model performance:

- **Mean Squared Error (MSE):** Measures the average squared difference between predicted and actual values. MSE heavily penalizes large prediction errors, making it particularly relevant to energy dispatch decisions where significant deviations can lead to grid instability or financial losses.
- **Mean Absolute Error (MAE):** Calculates the average absolute difference between predictions and actual values, providing insight into the typical magnitude of forecast errors.
- **Coefficient of Variation of Root Mean Square Error (CV-RMSE):** This metric normalizes the RMSE by the mean, allowing for a scale-invariant comparison across different datasets with varying power generation levels. Lower CV-RMSE values indicate a higher degree of prediction consistency relative to the average power output.

These metrics are commonly employed in wind power forecasting as they assess different aspects of prediction accuracy. Minimizing these metrics translates to more accurate predictions, which in turn contribute to enhanced grid stability and energy market efficiency.

4.1.3 Model Configurations and Training Settings. To ensure fair comparisons, all models undergo the same preprocessing steps [25]. We divide the data into training and testing sets using an 80/20 split. An input window length of 10 days is used for all models, representing a history of 10 days of daily power generation data. The models predict the wind power output one day ahead (day-ahead forecasting).

For CTRL and all baselines, we preprocessed data using min-max normalization applied to the entire time series. We implemented

Table 1: Hyperparameters for CTRL and Baseline Models

Model	Hyperparameter	Values	
		Search Space	Optimized
Dlinear [29]	Decomposition kernel size	3-9 (2 per step)	7
CNN1D	CNN out channels	16-128 (16 per step)	32
	CNN kernel size	3-9 (2 per step)	9
TCN [30]	TCN Layers	[8], [8, 16], [8, 16, 32]	[8, 16, 32]
LSTM [31]	Hidden size	16-128 (16 per step)	32
GRU [32]	Layers	1-3 (1 per step)	2
EncoderDecoder [33]	Bidirectional	True, False	TRUE
Transformer [34]	The label length	1-9 (1 per step)	5
	Dropout Rate	0.00-0.30 (0.05 per step)	0.1
Informer [35]	Encoder layers	1-3 (1 per step)	2
	Decoder layers	1-3 (1 per step)	11
Autoformer [36]	The numbers of heads	1, 2, 4, 8, 16	4
	The dimension of the model	16-128 (16 per step)	48
PatchTST [37]	The patch length	1–10 (1 per step)	5
	The patch stride	1–10 (1 per step)	1
LSTNet [38]	CNN out channels	16-128 (16 per step)	16
	CNN kernel size	3-9 (2 per step)	9
	Skip window size	1-5 (1 per step)	5
	Skip GRU hidden size	16-128 (16 per step)	32
	Skip GRU hidden size	1-3 (1 per step)	1
TPA-LSTM [39]	CNN out channels	16-128 (16 per step)	64
	CNN kernel size	3-9 (2 per step)	9
	GRU hidden size	16-128 (16 per step)	64
	GRU layers	1-3 (1 per step)	2
	Residual window size	1-10 (1 per step)	5
NHiTS [40]	NHiTS hidden size	16-128 (16 per step)	128
	NHiTS pooling size	1, 2, 4, 8, 16	8
GAIN [25]	GAT hidden size	16-128 (16 per step)	64
	The number of heads of GAT	1, 2, 4, 8, 16	2
AGCRN [41]	AGCRN hidden size	16-128 (16 per step)	32
	AGCRN embedding Dimension	1-5 (1 per step)	3
MSL [17]	shapelet size	2-10 (2 per step)	2
TCOAT [42]	GRU hidden size	16-128 (16 per step)	32
	GRU layers	1-3 (1 per step)	2
	GRU bidirectional	True, False	TRUE
	Residual window size	1-10 (1 per step)	3
CoDR [43]	Hidden size	16-128 (16 per step)	32
CTRL	GRU hidden size	16-128 (16 per step)	32
	GRU layers	1, 2, 3	2

all models in PyTorch v2.3.1. We used the Adam optimizer [26] with Mean Squared Error (MSE) as the loss function to train all models. We optimized hyperparameters using a grid search. Table 1 summarizes the optimal hyperparameters used for CTRL and the most relevant baseline models, along with the search space considered during grid search.

We conducted training using a server with an Intel®Xeon®Gold 5218R CPU (2.10 GHz), 256 GB of memory, and four Tesla V100-PCIE-16 GB GPUs.

4.2 Comparative Results and Analysis

Table 2 presents the performance comparison between CTRL and 20 existing forecasting methods on the Turkey WPF and Greece WPF datasets, measured using MSE, MAE, and CV-RMSE. CTRL consistently outperforms all baseline models across all three metrics and on both datasets, achieving a maximum performance improvement of 10.42% in MSE compared to the best-performing baseline model (LSTM) on the Greece WPF dataset. On the Turkey WPF dataset, CTRL achieves a 9.67% improvement in MSE over the best baseline (MSL). These results strongly indicate the effectiveness of CTRL's

Table 2: Comparison of CTRL with 20 Forecasting Methods

Model	Turkey WPF			Greece WPF		
	MSE	MAE	CV-RMSE	MSE	MAE	CV-RMSE
GAR [27]	162.797150	10.604651	0.617701	107.678520	8.160718	0.512205
AR [28]	162.691498	10.607114	0.617500	108.077423	8.184706	0.513153
DLinear [29]	162.961243	10.611975	0.618012	106.377663	8.098560	0.509102
CNN1D	163.851562	10.713211	0.619698	105.228844	8.065003	0.506345
TCN [30]	169.090134	10.752358	0.629526	108.374596	8.311706	0.513858
LSTM [31]	157.044418	10.103687	0.606689	97.204147	7.754950	0.486655
GRU [32]	160.838882	10.339690	0.613975	101.228256	8.102424	0.496627
EncoderDecoder [33]	161.547363	10.676886	0.615325	101.541145	8.146605	0.497394
Transformer [34]	163.935318	10.651772	0.619856	106.000389	8.329231	0.508198
Informer [35]	158.295029	10.678200	0.609100	100.119843	8.157362	0.493900
Autoformer [36]	179.489853	11.100290	0.648597	108.551010	8.205036	0.514276
PatchTST [37]	166.279922	10.716121	0.624273	107.465233	8.039746	0.511697
LSTNet [38]	160.043625	10.409985	0.612455	101.634254	7.945269	0.497622
TPA [39]	159.243408	10.562973	0.610922	104.613098	8.245862	0.504861
NHiTS [40]	158.400986	10.380313	0.609304	100.570366	7.870064	0.495010
GAIN [25]	163.103210	10.446630	0.618281	102.940331	7.971294	0.500809
AGCRN [41]	162.386444	10.648910	0.616921	102.682831	8.152772	0.500182
MSL [17]	154.822159	10.305735	0.602381	104.388702	8.156321	0.504320
TCOAT [42]	164.905319	10.782366	0.621688	100.375717	7.872735	0.494531
CoDR [43]	162.285614	10.620929	0.616730	99.564186	7.886658	0.492528
CTRL (Ours)	**139.854446**	**9.175537**	**0.572523**	**87.077072**	**6.723298**	**0.460608**

novel collaborative representation learning approach for capturing the complex dynamics of wind power generation.

Breaking down the results by model category, we observe that:

- **Linear Models (AR, GAR, DLinear):** CTRL significantly outperforms all linear models on both datasets, demonstrating its superior capability in capturing nonlinear dependencies in wind power data.
- **Deep Learning Models (LSTM, GRU, Transformer, etc.):** Even against advanced deep learning models, CTRL exhibits a consistent performance advantage. This highlights the benefits of the multi-directional temporal learning mechanism incorporated in CTRL's CRU module.
- **Hybrid Models:** CTRL also outperforms hybrid models that combine different architectural elements, such as MSL and NHiTS. This indicates that the unique benefits of collaborative representation learning in CTRL lead to more accurate and robust forecasts, even when compared to sophisticated hybrid architectures.
- **Recent State-of-the-Art Models:** CTRL demonstrates substantial improvements over recent state-of-the-art models like CoDR and TCOAT, further validating its effectiveness in wind power forecasting.

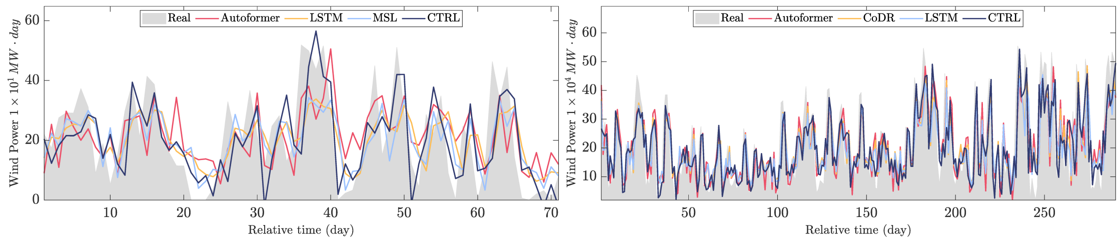
The consistent improvement across different metrics (MSE, MAE, and CV-RMSE) and datasets underscores CTRL's robustness and potential to advance wind power forecasting. Paired t-tests confirm that these improvements are statistically significant (p-value < 0.05) compared to all baseline models, further validating CTRL's collaborative representation learning approach as a promising direction for enhancing forecasting accuracy in renewable energy applications.

4.3 Visualized Predictions and Interpretation

Figure 2 presents a visual comparison of the actual and predicted wind power outputs on segments of the Turkey WPF and Greece WPF datasets for CTRL, MSL, LSTM, CoDR, and Autoformer. On both datasets, the predicted wind power output by CTRL closely aligns with the actual values, particularly in capturing sharp peaks and dips in power generation.

In contrast, MSL, LSTM and CoDR struggle to capture extreme variations, indicating limitations in their ability to adapt to rapid changes in wind patterns. Specifically, on the Turkey WPF dataset, MSL and LSTM underestimate peaks observed around day 38, 48, and 64, while overestimating power generation around day 52, and 70. These deviations highlight the challenges of relying solely on single-directional temporal dynamics for capturing rapid changes in wind power output. Furthermore, Autoformer consistently exhibits poor performance on both datasets, reflecting its inability to effectively capture the inherent complexity of wind power generation data. Its predictions are largely insensitive to fluctuations in the actual power output, resulting in consistently higher error rates compared to other models.

The experimental results unequivocally demonstrate CTRL's superior capabilities in day-ahead wind power forecasting. By consistently outperforming a diverse array of baseline models across multiple datasets and evaluation metrics, CTRL establishes itself as a robust and versatile forecasting tool. The model's collaborative temporal representation learning approach enables it to effectively capture multi-directional dependencies in wind power data, resulting in more precise predictions across diverse wind conditions. This enhanced forecasting accuracy has significant implications for grid



(a) Predictions on Turkey WPF.

(b) Predictions on Greece WPF.

Figure 2: Comparison of Actual and Predicted Wind Power Output. The shaded gray areas represent the real wind power generation, while the colored lines denote the predictions from different models.

stability, energy market efficiency, and the broader integration of wind power into sustainable energy systems.

5 Conclusion and Future Work

This paper presents Collaborative Temporal Representation Learning (CTRL), a novel deep learning model for enhancing the accuracy and robustness of wind power forecasting. CTRL leverages the power of collaborative representation learning to capture multi-directional temporal dependencies in wind power data, effectively addressing limitations of existing forecasting methods. The model's unique integration of Reversible Instance Normalization (RevIN), RNN-based hidden state learning, and a specialized collaborative representation unit enables it to effectively learn complex temporal patterns and correlations. Experimental evaluations on two real-world wind power datasets demonstrate that CTRL significantly outperforms 20 state-of-the-art forecasting methods, including physical models, machine learning techniques, and other deep learning approaches, achieving substantial improvements in prediction accuracy.

Future work will focus on three key directions. First, a more detailed ablation study will analyze the individual contributions of CTRL's components. Second, we will investigate CTRL's transferability to other renewable energy sources, such as solar and hydropower, broadening its potential impact on sustainable energy management. Finally, we plan to integrate additional data sources, like numerical weather predictions, to explore further performance gains.

Acknowledgments

This work is partially supported by the Education Department of Fujian Province (Grant no: JAT231193), and the Undergraduate Education and Teaching Research Project in Fujian Province (Grant no: FBJY20230126).

References

[1] Chengfu Wang, Qijun Teng, Xiaoyi Liu, Feng Zhang, Suoying He, Zhengtang Liang, and Xiaoming Dong. Optimal sizing of energy storage considering the spatial-temporal correlation of wind power forecast errors. IET Renewable Power Generation, 13(4):530–538, 2019.

[2] KekeWang, Dongxiao Niu, Lijie Sun, Hao Zhen, Jian Liu, Gejirifu De, and Xiaomin Xu. Wind power short-term forecasting hybrid model based on ceemd-se method. Processes, 7(11):843, 2019.

[3] Kaleb Phipps, Sebastian Lerch, Maria Andersson, Ralf Mikut, Veit Hagenmeyer, and Nicole Ludwig. Evaluating ensemble post-processing for wind power forecasts. Wind Energy, 25(8):1379–1405, 2022.

[4] Feng Huang, Zhixiong Li, Shuchen Xiang, and Rui Wang. A new wind power forecasting algorithm based on long short-term memory neural network. International Transactions on Electrical Energy Systems, 31(12):e13233, 2021.

[5] K. Mushtaq, R. Zou, A. Waris, K. Yang, J. Wang, and J. Iqbal. Multivariate wind power curve modeling using multivariate adaptive regression splines and regression trees. PLOS ONE, 18(8):e0290316, 2023.

[6] Yun Wang, Runmin Zou, Fang Liu, Lingjun Zhang, and Qianyi Liu. A review of wind speed and wind power forecasting with deep neural networks. Applied Energy, 304:117766, 2021.

[7] Wen-Hui Lin, PingWang, Kuo-Ming Chao, Hsiao-Chung Lin, Zong-Yu Yang, and Yu-Huang Lai. Wind power forecasting with deep learning networks: Time-series forecasting. Applied Sciences, 11(21):10335, 2021.

[8] Hong Liu and Zijun Zhang. Development and trending of deep learning methods for wind power predictions. Artificial Intelligence Review, 57:112, 2024.

[9] Ciaran Gilbert, Jakob W Messner, Pierre Pinson, Pierre-Julien Trombe, Remco Verzijlbergh, Pim van Dorp, and Harmen Jonker. Statistical post-processing of turbulence-resolving weather forecasts for offshore wind power forecasting. Wind Energy, 23(4):884–897, 2020.

[10] Jianzhou Wang, Tong Niu, Haiyan Lu, Zhenhai Guo, Wendong Yang, and Pei Du. An analysis-forecast system for uncertainty modeling of wind speed: A case study of large-scale wind farms. Applied Energy, 211:492–512, 2018.

[11] Miguel Godinho and Rui Castro. Comparative performance of ai methods for wind power forecast in portugal. Wind Energy, 24(1):39–53, 2021.

[12] Kunpeng Shi, Ying Qiao, Wei Zhao, Qing Wang, Menghua Liu, and Zongxiang Lu. An improved random forest model of short-term wind-power forecasting to enhance accuracy, efficiency, and robustness. Wind energy, 21(12):1383–1394, 2018.

[13] Jinhua Zhang, Jie Yan, David Infield, Yongqian Liu, and Fue-sang Lien. Short-term forecasting and uncertainty analysis of wind turbine power based on long short-term memory network and gaussian mixture model. Applied Energy, 241:229–244, 2019.

[14] Yulong Ma, Cristina L Archer, and Ahmad Vasel-Be-Hagh. Comparison of individual versus ensemble wind farm parameterizations inclusive of sub-grid wakes for the wrf model. Wind Energy, 25(9):1573–1595, 2022.

[15] Zi Lin and Xiaolei Liu. Wind power forecasting of an offshore wind turbine based on high-frequency scada data and deep learning neural network. Energy, 201:117693, 2020.

[16] Ignacio Salfate, Julio C Marin, Omar Cuevas, and Sonia Montecinos. Improving wind speed forecasts from the weather research and forecasting model at a wind farm in the semiarid coquimbo region in central chile. Wind Energy, 23(10):1939–1954, 2020.

[17] Zhijin Wang and Bing Cai. COVID-19 cases prediction in multiple areas via shapelet learning. Applied Intelligence, 52(1):595 – 606, 2022.

[18] Taesung Kim, Jinhee Kim, Yunwon Tae, Cheonbok Park, Jang-Ho Choi, and Jaegul Choo. Reversible instance normalization for accurate time-series forecasting against distribution shift. In ICLR'22, Virtual Conference, Apr. 2022. OpenReview.net.

[19] Zhijin Wang, Yaohui Huang, and Bingyan He. Dual-grained representation for hand, foot, and mouth disease prediction within public health cyber-physical systems. Software: Practice and Experience, 51(11):2290 – 2305, 10 2021.

[20] Zichao Yang, Zhiting Hu, Yuntian Deng, Chris Dyer, and Alex Smola. Neural machine translation with recurrent attention modeling. arXiv preprint arXiv:1607.05108, 2016.

[21] Léon Bottou, Frank E Curtis, and Jorge Nocedal. Optimization methods for large-scale machine learning. SIAM review, 60(2):223–311, 2018.

[22] Peter L Bartlett and Shahar Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. Journal of Machine Learning Research, 3(Nov):463–482, 2002.

[23] Berker Isen. Turkey wind power forecasting (turkey wpf) data collection, 2018.

[24] Argyrios Vartholomaios, Stamatis Karlos, Eleftherios Kouloumpris, and Grigorios Tsoumakas. Short-term renewable energy forecasting in greece using prophet decomposition and tree-based ensembles. In DEXA'21, volume 1479, pages 227 – 238, Linz, Austria, Sep. 2021. Springer.
[25] Zhijin Wang, Xiufeng Liu, Yaohui Huang, Peisong Zhang, and Yonggang Fu. A multivariate time series graph neural network for district heat load forecasting. Energy, 278:127911, 2023.
[26] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In ICLR'15, pages 1 – 15, San Diego, CA, USA, May 2015. OpenReview.net.
[27] George EP Box, Gwilym M Jenkins, Gregory C Reinsel, and Greta M Ljung. Time series analysis: forecasting and control. John Wiley & Sons, 2015.
[28] José F. Torres, Dalil Hadjout, Abderrazak Sebaa, Francisco Martínez-álvarez, and Alicia Troncoso. Deep learning for time series forecasting: a survey. Big Data, 9(1):3 – 21, 2021.
[29] Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In AAAI'23, volume 37, pages 11121 – 11128, 2023.
[30] Jin Fan, Ke Zhang, Yipan Huang, and Baiping Chen. Parallel spatio-temporal attention-based tcn for multivariate time series prediction. Neural Computing and Applications, 35(18):13109 – 13118, 2023.
[31] Sima Siami-Namini, Neda Tavakoli, and Akbar Siami Namin. The performance of lstm and bilstm in forecasting time series. In Big Data'19, pages 3285 – 3292, 2019.
[32] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv, abs/1412.3555, 2014.
[33] Kyunghyun Cho, Bart van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder-decoder approaches. In SSST'14, pages 103 – 111, Doha, Qatar, Oct. 2014. Association for Computational Linguistics.
[34] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In NeurIPS'17, volume 30, pages 5998 – 6008, Long Beach, CA, USA, December 2017. Curran Associates, Inc.
[35] Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In AAAI'21, pages 11106 – 11115, Virtual Event, Feb 2021. AAAI Press.
[36] Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. In NeurIPS'21, pages 22419 – 22430, Virtual, December 2021.
[37] Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In ICLR'23, pages 1 – 12, Kigali, Rwanda, May 2023.
[38] Guokun Lai, Wei-Cheng Chang, Yiming Yang, and Hanxiao Liu. Modeling longand short-term temporal patterns with deep neural networks. In SIGIR'18, pages 95 – 104, Ann Arbor, MI, USA, Jul. 2018. ACM.
[39] Shun-Yao Shih, Fan-Keng Sun, and Hung-yi Lee. Temporal pattern attention for multivariate time series forecasting. Machine Learning, 108(8-9):1421 – 1441, 2019.
[40] Carlos Challu, Kevin G Olivares, Boris N Oreshkin, Frederic Garza Ramirez, Manuel Mergenthaler Canseco, and Artur Dubrawski. Nhits: Neural hierarchical interpolation for time series forecasting. In AAAI'23, volume 37, pages 6989 – 6997, 2023.
[41] Yaohui Huang, Yuan Zhao, ZhijinWang, Xiufeng Liu, Hanjing Liu, and Yonggang Fu. Explainable district heat load forecasting with active deep learning. Applied Energy, 350:121753, 2023.
[42] Yue Hu, Hanjing Liu, Senzhen Wu, Yuan Zhao, Zhijin Wang, and Xiufeng Liu. Temporal collaborative attention for wind power forecasting. Applied Energy, 357:122502, 2024.
[43] Zhijin Wang, Hanjing Liu, Senzhen Wu, Niansheng Liu, Xiufeng Liu, Yue Hu, and Yonggang Fu. Explainable time-varying directional representations for photovoltaic power generation forecasting. Journal of Cleaner Production, 468:143056, 2024.