

Article

An STL-TCN-LSTM Hybrid Model for Dissolved Oxygen Forecasting in River Systems

Hongmei Li , Haodong Guo *, Luxia Yang and Hongrui Zhang

College of Computer Science and Technology, Taiyuan Normal University, Jinzhong 030619, China; hongmeili@tynu.edu.cn (H.L.); ylxxyz328@163.com (L.Y.); zhanghongrui@tynu.edu.cn (H.Z.)

* Correspondence: xguohaodongx@163.com; Tel.: +86-157-2168-8189

Abstract

River water quality prediction is a crucial aspect of water environment management and ecological conservation, holding significant importance for ensuring the sustainable utilization of water resources. As a key indicator for assessing river self-purification capacity and aquatic ecosystem health, the accurate prediction of dissolved oxygen (DO) is particularly vital for water quality early warning. To address the challenges that single deep learning models face in collaboratively modeling long- and short-term dependencies, and that most hybrid methods fail to adequately consider the characteristic differences in various components within a time series, this paper proposes an STL-TCN-LSTM model for predicting DO concentration in river water. The proposed model first employs seasonal-trend decomposition using Loess (STL) to decompose the original time series into three components: trend, seasonality, and residual, aiming to separate features at different time scales. Then, three parallel Temporal Convolutional Networks (TCNs) are utilized to extract temporal features from each component and reconstruct the sequence. Finally, the reconstructed results are fed into a Long Short-Term Memory (LSTM) network to further model their dynamic temporal dependencies, thereby enhancing prediction accuracy. The performance of the proposed model is validated on three river water quality datasets from different river basins with varying sampling frequencies. The experimental results on the three river datasets show that the STL-TCN-LSTM model consistently outperforms all baseline models, including LSTM, TCN, BiLSTM, GRU, CNN-LSTM, and XGBoost. Specifically, the Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and Mean Absolute Percentage Error (MAPE) are reduced by an average of 14.47%, 14.51%, and 14.27%, respectively, while the coefficient of determination (R^2) improves by an average of 0.79%. The Wilcoxon signed-rank test confirms that all performance improvements are statistically significant ($p < 0.05$). These results demonstrate that the proposed model achieves higher prediction accuracy and exhibits stronger generalization capability in DO forecasting, thereby offering a reliable tool for water quality early warning and aquatic environmental management.



Academic Editors: Ataur Rahman and Christos S. Akratos

Received: 23 April 2026

Revised: 20 May 2026

Accepted: 1 June 2026

Published: 3 June 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

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

Keywords: dissolved oxygen prediction; time series decomposition; seasonal-trend decomposition using Loess; Temporal Convolutional Network; Long Short-Term Memory

1. Introduction

Rivers, as the primary source of freshwater resources, are not only vital for sustaining human production and daily life but also play a critical role in maintaining watershed ecological balance. Their health status directly impacts regional sustainable development [1,2].

Dissolved oxygen (DO) serves as a key indicator for assessing the self-purification capacity and aquatic ecological health of rivers. Its deficiency can easily lead to ecological issues such as water eutrophication and fish mortality [3,4]. Therefore, achieving accurate prediction of DO concentration is of significant practical importance for water quality early warning and ecological conservation.

The dynamic variation in DO is subject to the superimposed influences of multiple factors such as water temperature, hydrological conditions, and human activities. Its time series data exhibit complex nonlinearity and periodicity, which poses significant challenges for accurate prediction [5–7]. Traditional time series forecasting methods, such as the Autoregressive Integrated Moving Average (ARIMA) model [8,9] and the Vector Autoregression (VAR) model [10], can effectively capture the linear relationships within data, but they struggle to handle the coupling problem between the prevalent nonlinear relationships and the short- and long-term fluctuations in DO concentration [11]. For instance, the ARIMA model assumes that the data satisfies the stationarity assumption, and its prediction accuracy drops significantly when faced with trend shifts in DO due to seasonal changes. Although the VAR model can handle multivariate correlations, it is prone to overfitting, and computational complexity increases dramatically as the variable dimensions and time window increase [12]. These limitations make it difficult for traditional methods to meet the practical requirements for prediction accuracy and stability in water environment management.

In recent years, deep learning has become a mainstream tool in the field of water quality prediction, owing to its strong capacity for nonlinear fitting and automatic feature extraction. The Recurrent Neural Network (RNN) first demonstrated its advantage in capturing long-term dependencies in sequences within Natural Language Processing. This advantage led to its subsequent application in water quality prediction, making it one of the earlier deep learning models adopted in this field. The Long Short-Term Memory (LSTM) network, a variant of RNN, addresses the vanishing gradient problem of standard RNNs via its gating mechanism [13], which has led to its extensive application in water quality prediction [14,15]. For example, Ye et al. [16] applied an LSTM model to predict the permanganate index in Shanghai rivers, demonstrating its superior accuracy and generalization capability over the traditional RNN. Furthermore, to enhance the cross-dataset generalization of LSTM, Dheda et al. [17] introduced a genetic algorithm to optimize its hyperparameters and constructed a weighted ensemble model, which significantly improved prediction stability. Khullar et al. [18] employed a Bidirectional LSTM (Bi-LSTM) for water quality prediction in India's Mun River, demonstrating its advantage over models like Support Vector Regression (SVR) and Random Forest.

While RNN and their variants excel at temporal modeling, they struggle to capture spatial dependencies within data. In contrast, spatial feature extraction methods, represented by Convolutional Neural Networks (CNNs), can effectively learn local and spatial patterns, thereby compensating for the limited local feature extraction capability of RNN-based models. For instance, Gopi et al. [19] utilized a CNN for water quality data classification, and their comparative results demonstrated the model's effectiveness. Separately, Fu et al. [20] applied a Temporal Convolutional Network (TCN) to predict dissolved oxygen in mariculture environments, achieving 91.91% prediction accuracy with higher training efficiency than an LSTM model.

River water quality variation is a complex dynamic process influenced by the interplay of multiple factors, with its spatiotemporal correlation evolving over time. Single models often struggle to adequately capture these dynamic correlations [21]. To address this, hybrid deep learning models have been developed to enhance the prediction accuracy and robustness. For instance, Chen et al. [22] proposed an Attention-based LSTM (AT-LSTM)

model that focuses on key features, significantly improving DO prediction accuracy for the Burnett River. Similarly, Bi et al. [23] incorporated an attention mechanism into LSTM for water quality prediction, reporting superior accuracy compared to standard LSTM and other models. Furthermore, Barzegar et al. [24] constructed a CNN-LSTM hybrid model that combines the spatial feature extraction capability of CNNs with the temporal modeling ability of LSTM, demonstrating its advantage in precisely modeling both high and low concentration dynamics. Yang et al. [25] also combined a CNN-LSTM network with an attention mechanism for water quality prediction in the Beilun Estuary, showing that the composite network outperformed other models. Beyond these, Yuan et al. [26] proposed a CA-TCN-BiGRU model, which utilized a TCN to capture temporal dependencies and was enhanced with a channel attention mechanism, demonstrating high accuracy and strong robustness in multi-indicator water quality prediction.

Despite the notable performance of hybrid models, the complex influencing factors behind water quality changes mean that their deep spatiotemporal dependencies can be obscured by dominant temporal patterns like trends and seasonality [27]. To more reliably discern these patterns, researchers have adopted a “decompose-then-predict” strategy. This involves first decoupling the original non-stationary series into components at different scales using signal decomposition techniques, before applying deep learning models for targeted prediction. For instance, Ke et al. [28] combined the STL method with an LSTM model for runoff forecasting. Chen et al. [29] proposed an STL-TCN model that employs parallel TCNs to predict the individual components derived from STL, demonstrating its advantages in reducing series noise and enhancing multi-step prediction accuracy. Furthermore, Chen et al. [30] developed an STL-MIMO-LSTM model, which exhibited superior prediction accuracy and stability compared to baseline models on the Beijing–Hangzhou Grand Canal dataset.

Apart from STL, various adaptive signal decomposition methods have also been integrated with deep learning models. Zhang et al. [31] combined Empirical Mode Decomposition (EMD) with LSTM for water quality prediction, with results indicating that the EMD-LSTM hybrid achieved higher forecasting accuracy than methods incorporating Short-Time Fourier Transform. Yan et al. [32] integrated Variational Mode Decomposition (VMD) with a Gated Recurrent Unit (GRU) for ammonia nitrogen concentration prediction, verifying the superiority of VMD-GRU over a standalone GRU. It should be noted that the EMD method suffers from mode mixing issues, and its iterative decomposition process has high computational complexity. Wavelet transform has also been applied in this context. Wu et al. [33] combined Discrete Wavelet Transform (DWT) with LSTM, achieving effective water quality prediction for the Jinjiang River. Similarly, Guan et al. [34] employed DWT to perform multi-scale decomposition and denoising on original spectral signals, subsequently combining it with CNN to improve the prediction accuracy for Chemical Oxygen Demand (COD).

In summary, existing research improved water quality prediction performance mainly through two strategies: model hybridization and the integration of series decomposition methods. Nevertheless, some limitations remain: (1) River water quality data typically exhibit distinct multi-scale coupling characteristics, such as seasonality, trend, and randomness. When applied directly, hybrid deep learning models often fail to effectively separate and utilize these implicit patterns at different temporal scales. (2) In methods based on series decomposition, existing approaches typically use the decomposed individual components as separate inputs, often failing to adequately account for the interrelationships among these components. Additionally, these methods often face challenges like high computational complexity and a strong dependency on the selection of wavelet bases.

To address the aforementioned issues, this paper proposes a hybrid STL-TCN-LSTM model that integrates STL, TCN, and LSTM networks. It is specifically designed to model the non-stationary characteristics in DO prediction. The main contributions of this work are as follows:

- (1) In the data preprocessing stage, STL is applied to decompose the original water quality series, yielding the trend-cycle, seasonal, and remainder components. These components respectively reflect the long-term tendency, seasonality, and random disturbances inherent in the water quality sequence.
- (2) In the feature extraction and prediction stage, a hybrid TCN-LSTM model is constructed. First, parallel TCN sub-networks are established. Each branch employs a cascade of four convolutional kernels with different respective fields to learn spatial features from the long-term trend, seasonal, and remainder components, respectively, thereby enhancing the model's ability to focus on scale-specific key features. Subsequently, the heterogeneous features extracted from the three branches are integrated and fed into an LSTM network. The LSTM effectively captures dependencies from local to global contexts to perform short-term forecasting of dissolved oxygen concentration.
- (3) Through ablation and comparative experiments conducted on three real-world river datasets with distinct characteristics, the prediction accuracy and generalization capability of the proposed model were validated. The experiments also confirmed the critical role played by each constituent module in the overall model performance.

2. Methods and Model

2.1. Basic Theory

2.1.1. Seasonal-Trend Decomposition Using Loess

Seasonal-trend decomposition using Loess (STL) is a time series filtering decomposition method based on Loess proposed by R. B. Cleveland et al. [35]. Its core objective is to accurately decompose a time series with seasonal characteristics into three additive components: trend, seasonality, and residual, thereby providing a foundation for subsequent feature extraction and predictive modeling. In this study, STL is applied to decompose the river water quality parameters, aiming to separate feature patterns at different temporal scales and reduce the interference caused by the coupling of multiple factors in dissolved oxygen prediction. The core decomposition relationship is expressed as follows:

$$Y_v = T_v + S_v + R_v \quad (1)$$

where Y_v represents the original time series; T_v is the trend component, reflecting the low-frequency fluctuations and long-term level changes in the data; S_v denotes the seasonal component, capturing the variations within a fixed period; and R_v is the residual component, representing the unexplained random noise after removing the trend and seasonality, which approximates white noise or a random walk.

2.1.2. Temporal Convolutional Network

Temporal Convolutional Network (TCN) [36] is a variant of Convolutional Neural Networks (CNNs) specifically designed for modeling temporal data. Its core design aims to overcome the limitations of CNNs in temporal tasks, such as limited receptive field and difficulty in capturing long-term dependencies, through two key mechanisms: dilated convolution and residual connections. At the same time, it solves the problem of low parallel computing efficiency of Recurrent Neural Networks (RNNs) based models during

training. In DO prediction, TCN can efficiently extract local temporal features and capture short-term fluctuation patterns.

(1) Causal Convolution

Causal convolution is the foundation for ensuring strict causality in temporal forecasting. Its core constraint is that the model's output at prediction time t must depend solely on historical observations from the current time t and earlier, without relying on any future information. This property fundamentally prevents information leakage in water quality data prediction, ensuring logical rigor in practical applications. Its mathematical definition is as follows:

$$P(x_t) = \prod_{t=1}^T P(x_t | x_1, x_2, \dots, x_{t-1}) \quad (2)$$

where $P(x_t)$ denotes the prediction probability at time step t , and T represents the total length of the sequence.

(2) Dilated Convolution

Dilated convolution is key to effectively expanding the receptive field and capturing long-range dependencies, which aids in modeling distant correlations within DO sequences. By introducing a dilation factor d into the standard convolution kernel, the kernel samples the input sequence at intervals that grow exponentially with the network depth. For a convolutional kernel of size k , the effective receptive field with a dilation factor d expands to $(k-1) \times d + 1$. This mechanism enables the TCN to expand its receptive field exponentially without significantly increasing the network depth or the number of parameters, thereby efficiently modeling long-range dependency features in temporal sequences. The structure of the dilated causal convolution is shown in Figure 1, and its operational formula is defined as:

$$F(x_s) = (x \times f_d)(s) = \sum_{i=0}^{k-1} f(i) \cdot x_{(s-d \cdot i)} \quad (3)$$

where $F(x_s)$ denotes the output feature at time step s , $x_{(s-d \cdot i)}$ represents the input sequence, f_d is the convolution kernel function, k is the kernel size, and d is the dilation factor.

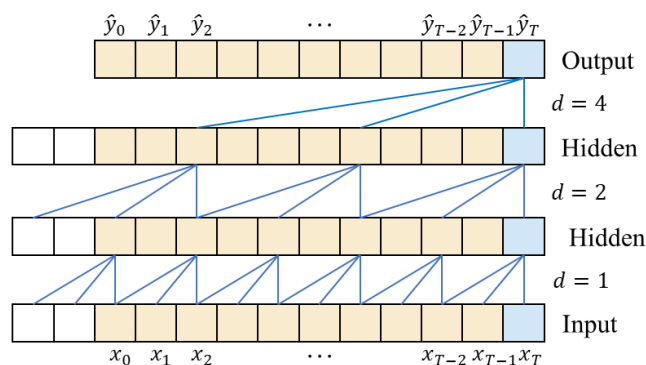


Figure 1. Schematic diagram of the dilated causal convolution [36].

(3) Residual Connection

The residual connection is a core component for constructing deep TCNs and mitigating the vanishing gradient problem. Its fundamental idea is to sum the module input x with the nonlinearly transformed residual function $\mathcal{F}(x)$, and then pass the result through an activation function to obtain the final output. This is expressed as:

$$o = \text{Activation}(x + \mathcal{F}(x)) \quad (4)$$

As illustrated in Figure 2, a typical TCN module with residual connections usually comprises two layers of dilated causal convolution, supplemented by ReLU activation functions, Weight Normalization, and dropout regularization. When a mismatch exists between

the input and output feature dimensions, a 1×1 convolution is introduced for dimension alignment. This design enables the TCN to be stably trained even with depths exceeding 12 layers, providing a solid structural foundation for modeling complex long-term temporal dependencies, thus making it suitable for river water quality prediction tasks.

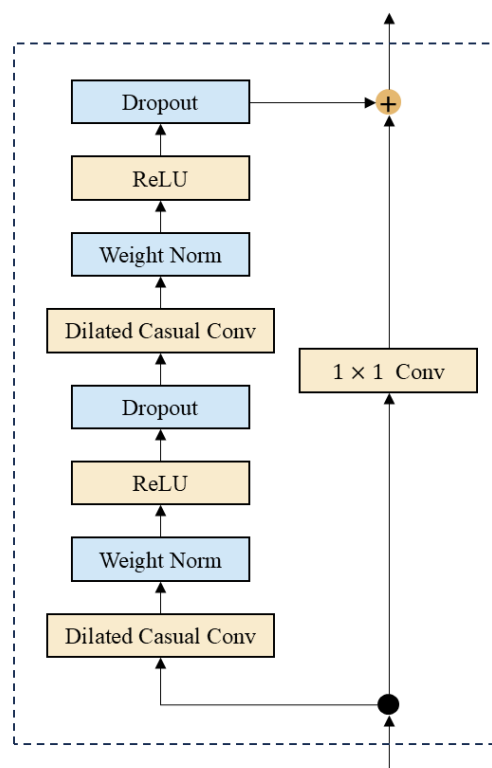


Figure 2. TCN module structure diagram with residual connection [36].

2.1.3. Long Short-Term Memory Network

Long Short-Term Memory (LSTM) network [37] is a special type of Recurrent Neural Network (RNN) proposed by Hochreiter and Schmidhuber in 1997; its structure is shown in Figure 3. LSTM introduces gating mechanism and a cell state, enabling the model to dynamically control the writing, forgetting, and output of information to the cell state through these gates. This allows for stable long-distance information transmission and addresses the issues of gradient vanishing or exploding encountered in traditional RNNs. In DO prediction, LSTM can capture long-term dependencies, enhancing the model's capability to model dynamic temporal features.

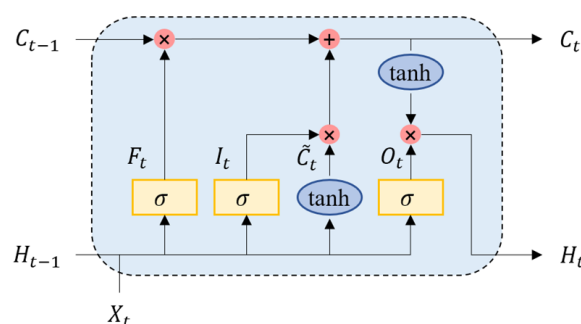


Figure 3. Structure of the LSTM network [37].

(1) The forget gate is responsible for determining which information from the previous cell state C_{t-1} should be retained. Based on the current input X_t and the previous hidden

state h_{t-1} , a vector between 0 and 1 is generated via a sigmoid activation function. This vector is then multiplied element-wise with the previous cell state C_{t-1} to decide which historical information should be preserved or discarded. This process can be expressed as:

$$f_t = \sigma(W_f \cdot [h_{t-1}, X_t] + b_f) \quad (5)$$

where σ is the sigmoid activation function, X_t is the input vector, h_{t-1} denotes the hidden state from the previous time step, and b_f is the bias vector.

(2) The input gate is responsible for determining which information from the current input X_t should be written into the cell state C_{t-1} . It achieves dynamic information update through two parallel network layers. First, a sigmoid layer generates a gating vector $i_t \in (0, 1)$, which determines the retention weight of the candidate information; simultaneously, a tanh layer generates a candidate memory vector $\tilde{C}_t$, which extracts potential information from the current input and the previous hidden state. Finally, both are combined to generate the incremental information actually written to the cell state. This design enables the model to selectively memorize information, effectively distinguishing the long-term value of information. The above process is expressed as follows:

$$i_t = \sigma(W_i \cdot [h_{t-1}, X_t] + b_i) \quad (6)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, X_t] + b_C) \quad (7)$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \quad (8)$$

where i_t is the gating vector of the input gate, $\tilde{C}_t$ is the candidate memory vector, W_i and W_C are the weight matrices, C_t is the cell state at the current time step, and C_{t-1} is the cell state from the previous time step.

(3) The output gate regulates which information from the cell state is transmitted to the final output at the current time step (i.e., the hidden state), based on the current input and the previous hidden state. A gating vector is generated via a sigmoid activation layer. This vector then undergoes element-wise multiplication with the normalized cell state to selectively determine the memory content for the final output. This process is mathematically expressed as:

$$o_t = \sigma(W_o \cdot [h_{t-1}, X_t] + b_o) \quad (9)$$

$$h_t = o_t \odot \tanh(C_t) \quad (10)$$

where o_t is the gating vector of the output gate, and h_t is the hidden state at the current time step. In this study, the LSTM is employed to further model the dynamic dependencies within the sequences decomposed by STL and to integrate the local features extracted by the TCN, thereby enhancing the accuracy and stability of dissolved oxygen prediction.

2.2. Architecture of the STL-TCN-LSTM Model

To address the challenges of significant non-stationarity, multi-scale feature coupling, and complex dynamic dependencies in DO time series, this paper proposes an STL-TCN-LSTM model; the overall structure is shown in Figure 4. STL is first performed to adaptively separate the original multi-feature time series into three stationary components: trend, seasonality, and residual. This process separates feature information across different temporal scales. The trend component reflects the long-term evolution direction of the sequence. The seasonal component extracts periodic fluctuation patterns, and the residual component contains random noise and short-term perturbations, thereby alleviating the

modeling interference caused by multi-scale coupling. Then, to address the temporal feature differences in each component, three parallel TCN branches are employed to capture their local long-term dependencies, and feature reconstruction is performed to enhance cross-scale representation capability. Finally, the fused features are fed into an LSTM to further model the dynamic long-term dependencies of the sequence, thereby achieving high-precision prediction of DO concentration. The above process is expressed as follows:

$$y_s, y_t, y_c = stl_decomp(x_{input}) \quad (11)$$

$$y_{predict} = FC(Dropout(LSTM(TCN(y_t) + TCN(y_s) + TCN(y_c)))) \quad (12)$$

where y_s , y_t and y_c represent the seasonal, trend, and residual components of the input historical time window information x decomposed by the STL module, respectively. $y_{predict}$ denotes the final output value after the fully connected linear layer.

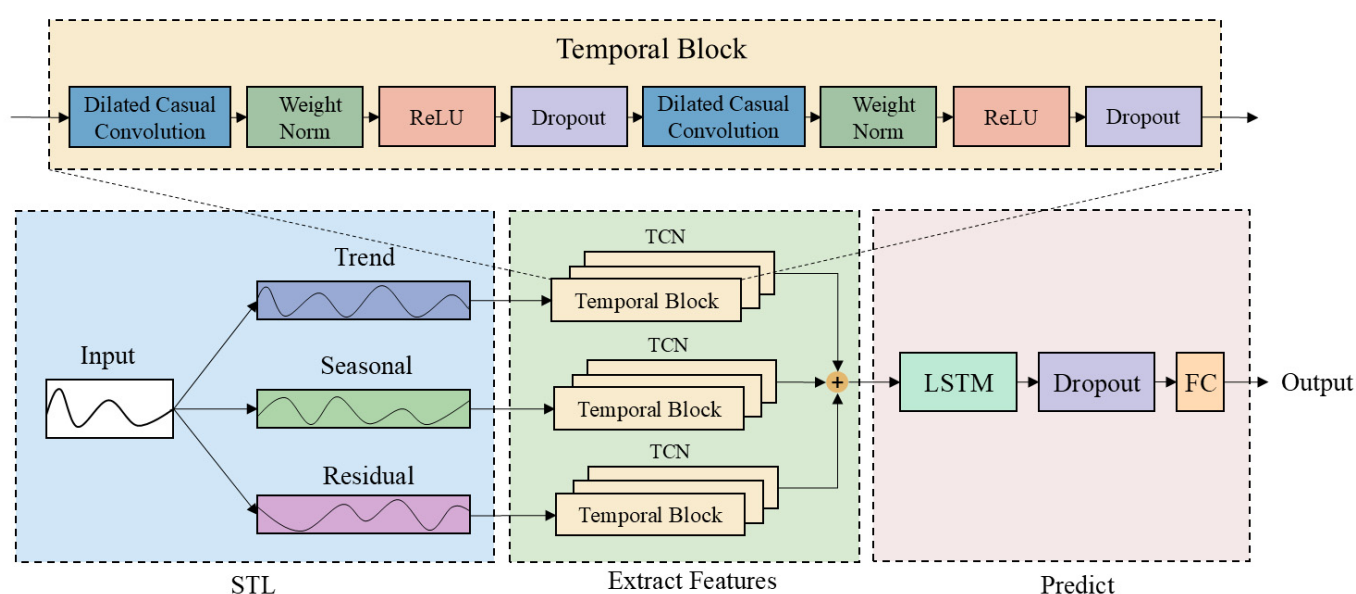


Figure 4. Structure of the STL-TCN-LSTM model.

The overall workflow for water quality prediction based on the STL-TCN-LSTM model is illustrated in Figure 5, and the core steps are outlined as follows:

Step 1. Data Preprocessing. The original water quality data is first preprocessed, primarily involving missing value imputation and data normalization, to eliminate scale effects and enhance data quality.

Step 2. Dataset Division. The preprocessed data is partitioned into training, validation, and test sets according to a specific ratio to ensure the independence of model training, hyperparameter tuning, and final performance evaluation.

Step 3. Model Training and Validation. The STL-TCN-LSTM model is trained using the training set data. The training process is monitored on the validation set to prevent overfitting, while model hyperparameters are optimized simultaneously.

Step 4. Model Testing and Performance Evaluation. The optimally trained model is applied to the independent test set. Metrics such as MAE, RMSE, MAPE, and R^2 are calculated to comprehensively assess the model's generalization capability and prediction accuracy on unseen data.

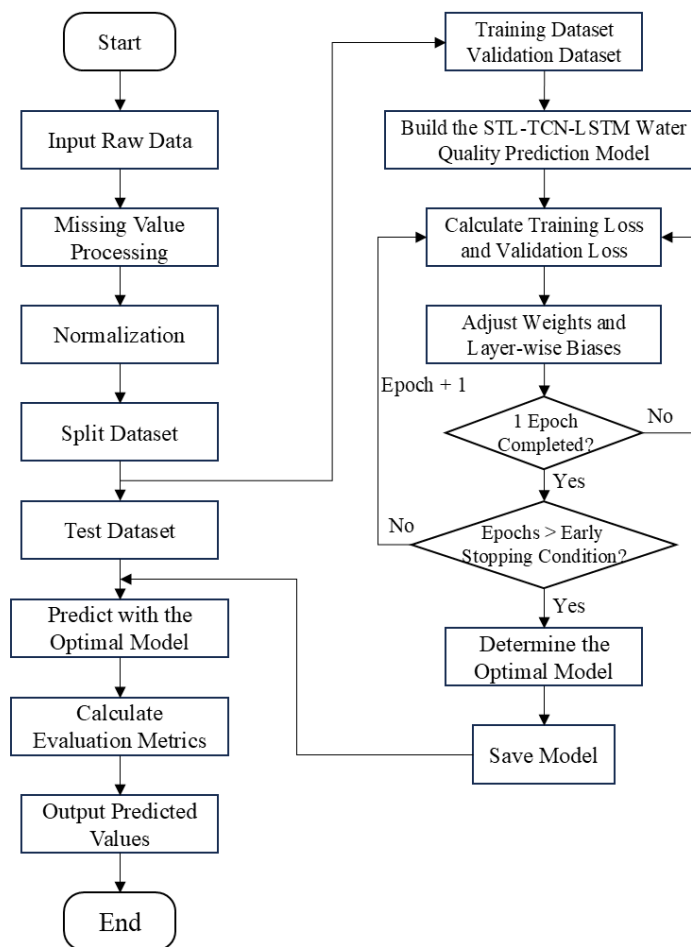


Figure 5. Flowchart of the STL-TCN-LSTM water quality prediction algorithm.

3. Experiment Setup

3.1. Dataset Description

The experimental data used in this paper came from two institutions: (1) data collected from the Burnett River and Baffle Creek by relevant departments of the Queensland Government and environmental protection organizations; and (2) the Fenhe River dataset released daily by the China National Environmental Monitoring Center. The relevant data are available from the corresponding official websites [38,39]. All three datasets include six water quality features: dissolved oxygen (DO), Chlorophyll-a (Chl-a), pH, Electrical Conductivity (EC), water temperature (Temp), and Turbidity (Turb). The data were partitioned into training, validation, and test sets at an 8:1:1 ratio.

The Burnett River (Dataset A) is located in the southeast of Queensland, Australia, with a total length of 270 miles (435 km) and a watershed area of 12,440 square miles (32,220 km²). This river is an important source of agricultural irrigation and ecological water for the region, where small crops and sugarcane are primarily grown. Water quality is significantly affected by climate change and agricultural activities. To assess the impact of surrounding activities on the water quality of the Burnett River, an automatic monitoring station at the river's source conducts monitoring long-term water quality monitoring, as shown in Figure 6. The monitoring station collects data every 30 min. This paper selects a total of 50,682 data points from January 2017 to December 2019 for the experiment. The original data for the six water quality features are shown in Figure 7.

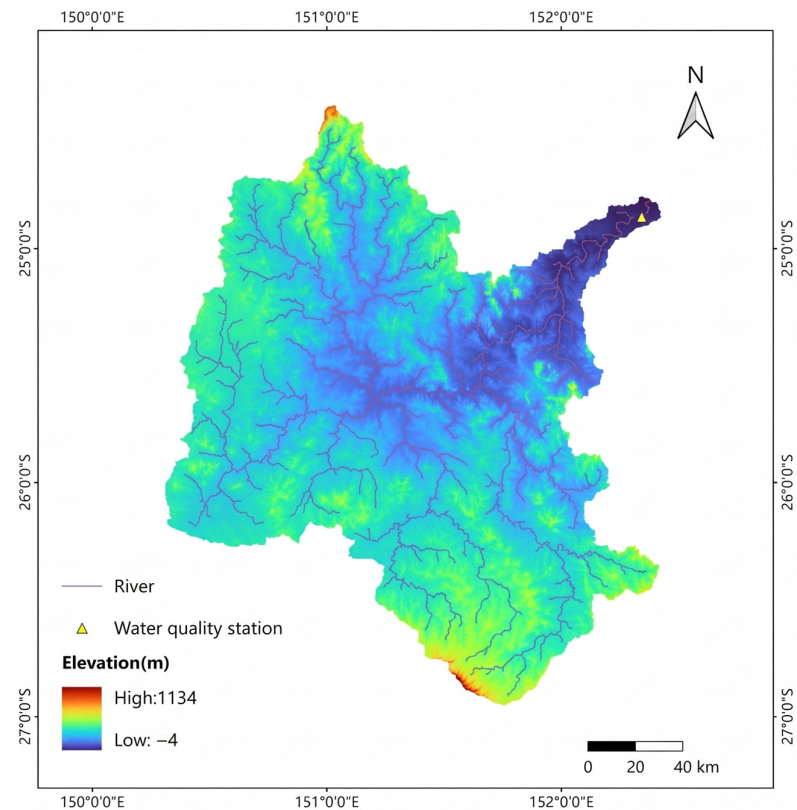


Figure 6. Overview map of the study area.

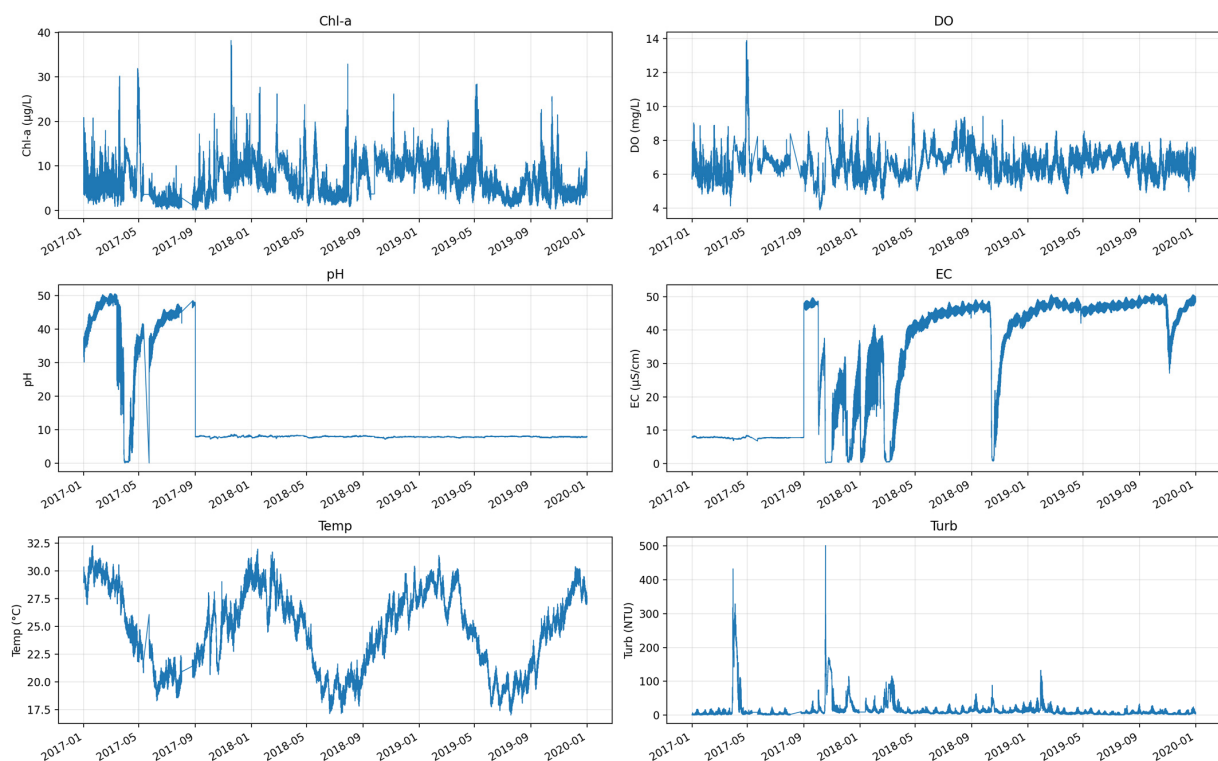


Figure 7. Line plots of the original data for the six water quality parameters from the Burnett River.

The Baffle Creek (Dataset B) is located in southern Queensland, Australia, with a sampling frequency of every 10 min. For the experiment, a total of 51,651 records from the year 2020 were selected.

The Fenhe River outlet section of China (Dataset C) has a sampling frequency of every 4 h. For the experiment, a total of 4031 records from the period 2021 to 2022 were utilized.

3.2. Data Preprocessing

During the long-term data collection process, data duplication or missing values occurred due to environmental fluctuations and instrument anomalies. To ensure the integrity and accuracy of the dataset, data preprocessing was performed by deleting duplicate records and using the forward filling method to improve data quality. The basic statistical information of the preprocessed data for the three datasets is presented in Table 1, Table 2, and Table 3.

Table 1. Basic statistical information of Dataset A after preprocessing.

Indicator	Unit	Min	Max	Average	Median	Standard Deviation
Chl-a	µg/L	0.1	38.2	6.91	6.3	3.94
DO	mg/L	3.9	13.9	6.55	6.52	0.83
pH	-	0.01	13.99	7.62	7.81	0.88
EC	µS/cm	1	750.5	54.02	43.93	121.24
Temp	°C	17	32.29	24.57	24.95	3.57
Turb	NTU	0.1	492.9	15.51	7.6	31.13

Table 2. Basic statistical information of Dataset B after preprocessing.

Indicator	Unit	Min	Max	Average	Median	Standard Deviation
Chl-a	µg/L	0.75	64.92	5.89	3.28	6.69
DO	mg/L	2.74	10.3	6.64	6.6	1.19
pH	-	7.13	8.3	7.86	7.9	0.21
EC	µS/cm	1.17	58.24	36.07	45.26	20
Temp	°C	16.78	33.59	25.27	26.23	4.11
Turb	NTU	0.05	394.34	4.48	2.48	7.49

Table 3. Basic statistical information of Dataset C after preprocessing.

Indicator	Unit	Min	Max	Average	Median	Standard Deviation
Chl-a	µg/L	0.01	32.51	25.78	32.51	13.15
DO	mg/L	1.55	17	9.91	9.5	2.34
pH	-	7.32	9.05	8.46	8.49	0.18
EC	µS/cm	1.11	887.33	712.38	717.86	81.81
Temp	°C	1.11	32.19	13.45	12.64	8.13
Turb	NTU	0.18	98	3.15	2.69	3.14

Due to significant differences in the units and orders of magnitude among multi-source water quality indicators, using them directly for model training may lead to features with larger numerical values dominating the learning process, thereby affecting the model's convergence and prediction accuracy. To eliminate the scale effects, this study employs Min-Max normalization. Min-Max normalization scales data based on the proportional difference between the maximum and minimum values, constraining the data range of all water quality features within the interval [0, 1]. The normalization formula is as follows:

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (13)$$

where x is the sample data, x' is the normalized value, $\min(x)$ and $\max(x)$ representing the minimum and maximum values in the dataset, respectively.

After normalization, the data were segmented into fixed length samples according to the predefined time windows τ . Considering the different sampling frequencies of Datasets A, B, and C, this study uniformly determined the time window length based on the weekly data volume: Dataset A had a sampling interval of 30 min (48 samples per day), so $\tau = 336$ (corresponding to 7 days); the time windows τ for Datasets B and C were set to 1008 and 42, respectively, both covering a full week of observation data. The sample set was then partitioned into training, validation, and test sets according to a predetermined ratio. The validation set was used exclusively for monitoring the training process, calculating the validation loss, and triggering early stopping; it did not participate in model parameter updates or the final testing evaluation. The sample counts of the three datasets obtained from the split are summarized in Table 4.

Table 4. Sample sizes of the training, validation, and test sets for the three datasets.

Dataset	Train Set	Validation Set	Test Set	τ
A	40,546	5068	5068	336
B	41,321	5165	5165	1008
C	3225	403	403	42

Subsequently, to separate the key temporal features, the STL method was applied to decompose the dataset into trend, seasonal, and residual components. Figure 8 shows the DO decomposition results of STL on Dataset A.



Figure 8. The DO decomposition results of STL on Dataset A.

3.3. Evaluation Metrics

To validate and evaluate the predictive performance of the proposed STL-TCN-LSTM model, four commonly used evaluation metrics in water quality forecasting were employed to analyze the results, namely the Mean Absolute Error (MAE), Root Mean Square Error (RMSE), Mean Absolute Percentage Error (MAPE), and the coefficient of determination (R^2). The calculation formulas for each metric are as follows:

$$MAE = \frac{1}{n} \sum_{i=1}^n |\hat{y}_i - y_i| \quad (14)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (\hat{y}_i - y_i)^2} \quad (15)$$

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right| \times 100\% \quad (16)$$

$$R^2 = 1 - \frac{\sum_i (\hat{y}_i - y_i)^2}{\sum_i (\bar{y} - y_i)^2} \quad (17)$$

where $\hat{y}$ represents the true value, y denotes the predicted value, $\bar{y}$ is the mean of the true values, i is the sample index, and n is the total number of samples. Smaller values of MAE, RMSE, and MAPE indicate smaller prediction errors and higher model accuracy. Conversely, an R^2 value closer to 1 signifies a better fit, meaning the predicted values are closer to the true values.

In addition, the Wilcoxon signed-rank test was employed in this study to analyze the statistical significance of the differences in performance evaluation metrics between the proposed model and all baseline models. A p -value less than 0.05 indicates that the performance difference is statistically significant.

3.4. Parameter Optimization and Experimental Configuration

To determine the optimal model configuration, a systematic grid search was conducted. The candidate ranges for all hyperparameters were defined based on domain knowledge and common practices in deep learning. The specific search process and results are as follows:

(1) STL Seasonal Period: tested at values of 24, 48, and 96 to examine cyclical assumptions. A period of 48 yielded the best performance.

(2) Input Sequence Length: evaluated lengths of 144, 336, and 672. A length of 336 was optimal for capturing the necessary contextual information.

(3) TCN Configuration: the number of channels in the TCN layers was explored with configurations of [16, 16, 16], [32, 32, 32], and [64, 64, 64]. The [32, 32, 32] configuration offered the best balance. The kernel size was optimized from {2, 3, 5}, with a size of three selected.

(4) LSTM Hidden Units: the number of units was chosen from {32, 64, 128}, with 64 performing best.

(5) Training Parameters: the dropout rate was tuned in {0.1, 0.2, 0.3}, with 0.2 being most effective. The Adam optimizer's initial learning rate was searched in $\{1 \times 10^{-4}, 5 \times 10^{-4}, 1 \times 10^{-3}\}$, and 1×10^{-3} was optimal. Batch sizes of 16, 32, and 64 were tested, with 32 selected. Early stopping patience was chosen from {5, 10}, and a value of 10 was used.

This optimal configuration, determined on Dataset A, provided the most stable and superior validation performance. It was subsequently fixed and used for the final training and testing on all datasets.

All experiments were conducted on a Windows 11 Pro 64-bit operating system. The hardware platform included an AMD Ryzen 9 7845HX with Radeon Graphics processor (Advanced Micro Devices, Inc., Santa Clara, CA, USA), 16 GB of DDR5 memory, and an NVIDIA GeForce RTX 4070 GPU (NVIDIA Corporation, Santa Clara, CA, USA). Model training and inference were performed within a Python 3.10 virtual environment created using Anaconda, relying primarily on PyTorch 2.8, statsmodels 0.14, and scikit-learn 1.7.1. To ensure the reproducibility of the results, all experiments were run in an identical environment with a fixed random seed.

4. Results Analysis

4.1. Comparative Analysis of Models

To evaluate the predictive performance of the STL-TCN-LSTM model, it was compared with mainstream forecasting models LSTM, TCN, BiLSTM, GRU, CNN-LSTM, and XGBoost on Dataset A. All performance evaluation metrics are listed in Table 5, where MAE, RMSE, MAPE, and R^2 are presented in the form of “mean $\pm$ standard deviation”.

Table 5. Comparative analysis of the predictive performance of various models for DO on Dataset A. (mean $\pm$ SD; p -values).

Model	MAE	p _Value	RMSE	p _Value	MAPE(%)	p _Value	R^2	p _Value
BiLSTM	0.00696 ± 0.00006	$p < 0.05$	0.00981 ± 0.00007	$p < 0.05$	2.7769 ± 0.03053	$p < 0.05$	0.97268 ± 0.00038	$p < 0.05$
GRU	0.00692 ± 0.00018	$p < 0.05$	0.00978 ± 0.00028	$p < 0.05$	2.76821 ± 0.08037	$p < 0.05$	0.97281 ± 0.00153	$p < 0.05$
TCN	0.00719 ± 0.00004	$p < 0.05$	0.0102 ± 0.00007	$p < 0.05$	2.88592 ± 0.02298	$p < 0.05$	0.97046 ± 0.00041	$p < 0.05$
LSTM	0.00743 ± 0.00011	$p < 0.05$	0.01061 ± 0.00007	$p < 0.05$	2.97437 ± 0.04418	$p < 0.05$	0.968 ± 0.00045	$p < 0.05$
CNN-LSTM	0.00711 ± 0.00019	$p < 0.05$	0.01011 ± 0.00027	$p < 0.05$	2.84744 ± 0.08337	$p < 0.05$	0.97093 ± 0.00156	$p < 0.05$
XGBoost	0.00668 ± 0.00016	$p < 0.05$	0.00941 ± 0.00019	$p < 0.05$	2.65397 ± 0.07781	$p < 0.05$	0.97477 ± 0.00104	$p < 0.05$
STL-TCN-LSTM	0.00603 ± 0.00012	-	0.00854 ± 0.00014	-	2.41583 ± 0.04989	-	0.97929 ± 0.00069	-

As can be seen from Table 5, the STL-TCN-LSTM model achieved optimal performance across all evaluation metrics, and all these improvements over the baseline models are statistically significant ($p < 0.05$). Specifically, it attained an average MAE of 0.00603, which represents reductions of 18.84%, 16.13%, 13.36%, 12.86%, 15.19%, and 9.73% compared to the LSTM, TCN, BiLSTM, GRU, CNN-LSTM, and XGBoost models, respectively. The average RMSE was 0.00854, corresponding to reductions of 19.51%, 16.27%, 12.95%, 12.68%, 15.53%, and 9.25% against the benchmark models. The average MAPE was 2.41583%, showing improvements of 18.76%, 16.29%, 13.01%, 12.73%, 15.15%, and 8.98%. Furthermore, the average R^2 value reached 0.97929, indicating increases of 1.17%, 0.91%, 0.68%, 0.67%, 0.86%, and 0.46% over the comparative models.

A comparison of the predicted versus true dissolved oxygen values for each model is illustrated in Figure 9. Although the overall prediction trends of all models are relatively consistent, it can be observed from the magnified local plots that the STL-TCN-LSTM model has the highest goodness of fit to the true values, and the smallest deviations in peak and valley positions and amplitudes.

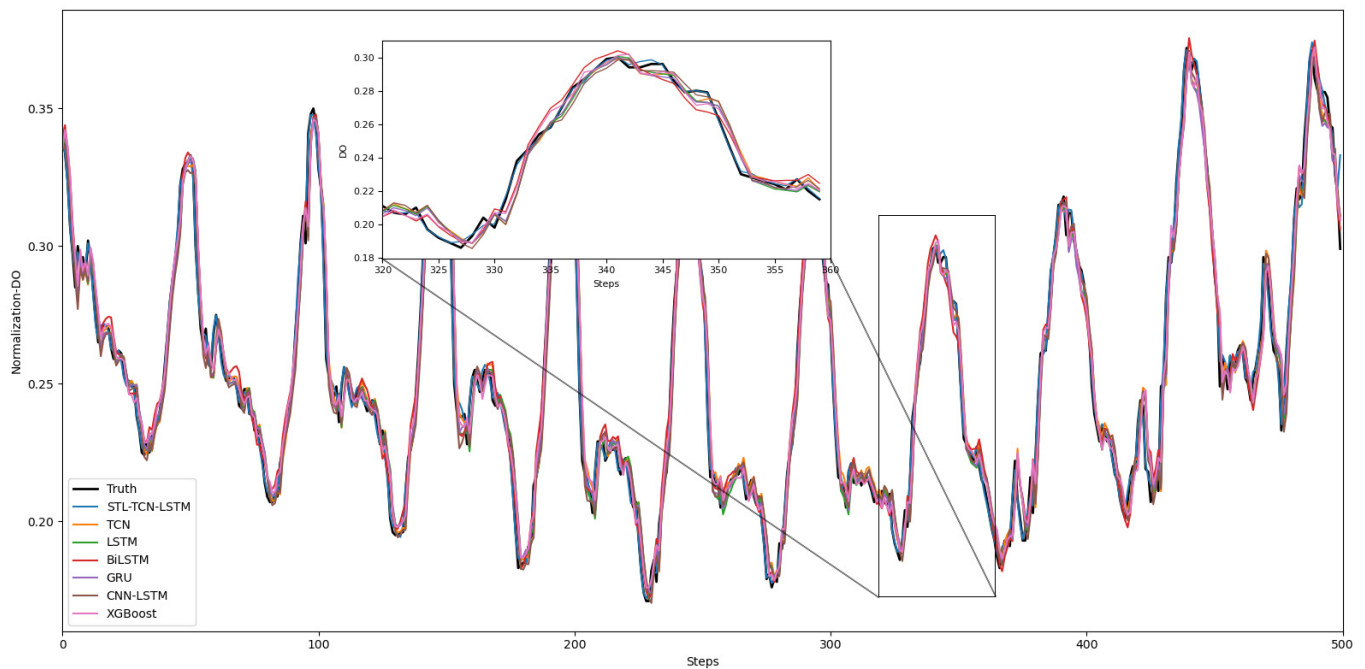


Figure 9. Comparison of predicted values and true values from different models.

In summary, under the same input features and training configuration, the STL-TCN-LSTM model consistently yields superior performance over the comparative methods in both prediction error and goodness of fit ($p < 0.05$), demonstrating its superior capability in characterizing the nonlinear and multi-scale characteristics of water quality DO time series.

4.2. Ablation Study

To effectively evaluate the contribution of each module to the model's performance, an ablation study was conducted on the STL-TCN-LSTM model using Dataset A. Table 6 lists the effectiveness of different models in DO predicting task.

Table 6. Performance comparison of ablation models for predicting DO.

Model	MAE	RMSE	R ²
STL-LSTM	0.00699	0.00999	0.97162
STL-TCN	0.00659	0.00917	0.97611
TCN-LSTM	0.00703	0.00996	0.97179
TCN	0.00719	0.01020	0.97046
LSTM	0.00743	0.01061	0.96800
STL-TCN-LSTM	0.00603	0.00854	0.97929

In the comparison of predictive performance between the baseline models, TCN and LSTM, the LSTM model exhibited relatively weaker performance. For instance, in terms of MAE, the TCN model (0.00719) was 3.23% lower than LSTM (0.00743). Regarding RMSE, the TCN model (0.01020) was 3.86% lower than LSTM (0.01061). Additionally, in terms of R², the TCN model (0.97046) was 0.25% higher than LSTM (0.96800). These results indicate that the LSTM model has a limited capability in capturing long-term dependency features within the water quality prediction task data. Furthermore, TCN and LSTM were integrated to construct a hybrid model TCN-LSTM. Experimental results indicate that this fused model achieves significant improvements across all evaluation metrics. MAE decreased from 0.00743 to 0.00703, a reduction of 5.38%; RMSE decreased from 0.01061 to 0.00996, a reduction of 6.13%; and the R² increased from 0.96800 to 0.97179, an improvement

of 0.39%. These results fully demonstrate that the TCN-LSTM fusion model effectively overcomes the limitations of a single model by synergistically utilizing the local feature extraction capabilities of TCN and the temporal dynamic modeling advantages of LSTM.

Subsequently, based on the baseline models TCN and LSTM, STL was further introduced to construct two hybrid models: STL-LSTM and STL-TCN. The ablation experiment results show that compared to the single deep learning models, both hybrid models exhibit significant improvements in key evaluation metrics. Specifically, compared to the baseline TCN model, the STL-TCN model reduced MAE from 0.00719 to 0.00659 (a reduction of 8.3%), RMSE from 0.01020 to 0.00917 (a reduction of 2.1%), and R^2 from 0.97046 to 0.97611 (an improvement of 0.12%). Similarly, compared to the baseline LSTM model, the STL-LSTM model reduced MAE from 0.00743 to 0.00699 (a reduction of 5.9%), RMSE from 0.01061 to 0.00917 (a reduction of 13.6%), and R^2 from 0.96800 to 0.97611 (an improvement of 0.84%). These results consistently demonstrate that STL effectively separates the trend and seasonal components within the time series, thereby significantly enhancing the ability of deep learning models to extract potential temporal features and improving prediction accuracy and stability.

It is worth noting that the results exhibit a certain complexity when compared horizontally with the joint deep learning model TCN-LSTM. In terms of the MAE metric, both decomposition hybrid models performed better. Compared to TCN-LSTM, STL-LSTM reduced MAE by 0.57%, and STL-TCN reduced it by 6.68%, which further confirms the effectiveness of introducing the decomposition method. However, the results diverged on the RMSE and R^2 metrics. Compared with the TCN-LSTM model, the RMSE of STL-TCN (0.00996) was significantly lower than that of TCN-LSTM (0.00917) by 8.62%, but its R^2 (0.97611) was slightly lower than that of TCN-LSTM (0.97179), with a difference of -0.44% . Conversely, the RMSE of the STL-LSTM model (0.00996) was slightly higher than that of TCN-LSTM (0.00999) by 0.3%, but its R^2 (0.97162) was slightly improved by 0.02%. This divergence may stem from the different statistical properties of the metrics, where RMSE is sensitive to outliers while R^2 focuses on the overall goodness of fit. Nevertheless, considering the significant improvements in the core error metrics MAE and RMSE overall, the introduction of the STL strategy generally optimizes the predictive performance.

Finally, the STL-TCN-LSTM model constructed in this paper achieves the best predictive performance among all compared models by synergistically integrating the advantages of sequence decomposition (STL), temporal feature extraction (TCN), and long-term dependency modeling (LSTM), with an MAE of 0.00603, RMSE of 0.00854, and R^2 of 0.97929. Benefiting from the cascaded architecture of decomposition-feature extraction-temporal modeling, the model can more comprehensively describe the driving mechanisms of water quality variations, thereby achieving a breakthrough in predictive performance.

4.3. Generalization Experiment

To evaluate the generalization capability of the STL-TCN-LSTM model in water quality prediction tasks, we conducted generalization experiments on Dataset B and Dataset C. To ensure a fair comparison, all baseline models TCN, GRU, LSTM, and XGBoost used identical training and testing sample sizes, with their parameter settings kept consistent. The test results of each model on the two datasets are listed in Table 7.

On Dataset B, STL-TCN-LSTM achieved the best performance, and the improvements were statistically significant compared to all baseline models ($p < 0.05$). Compared with TCN model, it reduced MAE and RMSE by 12.71% and 13.78%, respectively, and improved R^2 by 0.38%. Compared with LSTM, it lowered MAE and RMSE by 10.49% and 11.13%, respectively, and increased R^2 by 0.29%. Compared with GRU, it decreased MAE and RMSE by 2.26% and 8.42%, respectively, and raised R^2 by 0.21%. Finally, compared with

XGBoost, it reduced MAE and RMSE by 1.75% and 2.90%, respectively, and improved R^2 by 0.11%.

Table 7. Performance comparison of ablation models for predicting DO. (mean $\pm$ SD; p -values).

Model	Dataset B			Dataset C		
	MAE	RMSE	R^2	MAE	RMSE	R^2
TCN	0.00645	0.01009	0.98543	0.01958	0.0256	0.82427
	± 0.00048	± 0.00038	± 0.00107	± 0.00103	± 0.00118	± 0.01645
	($p < 0.05$)	($p < 0.05$)	($p < 0.05$)	($p < 0.05$)	($p < 0.05$)	($p < 0.05$)
LSTM	0.00629	0.00979	0.98629	0.02403	0.03076	0.74624
	± 0.00035	± 0.00029	± 0.00086	± 0.00052	± 0.00056	± 0.00932
	($p < 0.05$)	($p < 0.05$)	($p < 0.05$)	($p < 0.05$)	($p < 0.05$)	($p < 0.05$)
GRU	0.00576	0.00950	0.98710	0.02335	0.02947	0.76707
	± 0.00018	± 0.00013	± 0.00035	± 0.00305	± 0.00394	± 0.07111
	($p < 0.05$)	($p < 0.05$)	($p < 0.05$)	($p < 0.05$)	($p < 0.05$)	($p < 0.05$)
XGBoost	0.00573	0.00896	0.98813	0.02476	0.03881	0.82124
	± 0.00007	± 0.00007	± 0.00019	± 0.00068	± 0.00108	± 0.00968
	($p < 0.05$)	($p < 0.05$)	($p < 0.05$)	($p < 0.05$)	($p < 0.05$)	($p < 0.05$)
STL-TCN-LSTM	0.00563	0.00870	0.98918	0.01479	0.0202	0.89057
	± 0.00012	± 0.00011	± 0.00028	± 0.00046	± 0.00043	± 0.0047
	-	-	-	-	-	-

On Dataset C, the STL-TCN-LSTM model demonstrated even more significant advantages, and all performance gains were statistically significant ($p < 0.05$). Compared with TCN, it reduced MAE and RMSE by 24.46% and 21.09%, respectively, and improved R^2 by 8.04%. Relative to LSTM, it lowered MAE and RMSE by 38.45% and 34.33%, respectively, and increased R^2 by 19.34%. In comparison with GRU, it decreased MAE and RMSE by 36.66% and 31.46%, respectively, and raised R^2 by 16.10%. When compared with XGBoost, it reduced MAE and RMSE by 40.27% and 47.95%, respectively, and enhanced R^2 by 8.44%.

The above results show that the STL-TCN-LSTM model exhibits excellent prediction performance on both the Baffle River and Fenhe River datasets, which originate from different watersheds and have distinct sampling frequencies. In terms of MAE, RMSE, and R^2 metrics, this model exhibits substantial improvements over mainstream comparative models such as LSTM, TCN, GRU, and XGBoost with statistical significance, demonstrating its stronger robustness and broader applicability in water quality prediction tasks.

4.4. Training Efficiency and Convergence Performance Analysis

To further assess the practical applicability of the STL-TCN-LSTM model, beyond prediction accuracy, its training efficiency and computational cost during the inference phase are also critical considerations. We conduct a comparative analysis of the training convergence process, training time, and prediction time of the STL-TCN-LSTM, TCN-LSTM, and LSTM models, based on the DO prediction task for Dataset A.

Figure 10 shows the validation loss curves of the three models for the DO prediction task in the Burnett River. The results show that all models converge within approximately 40–50 epochs, indicating stable training processes under the current learning rate and early stopping settings. Among them, STL-TCN-LSTM model exhibited the lowest overall validation loss and a smoother convergence trajectory, demonstrating its superior fitting capability and training stability.

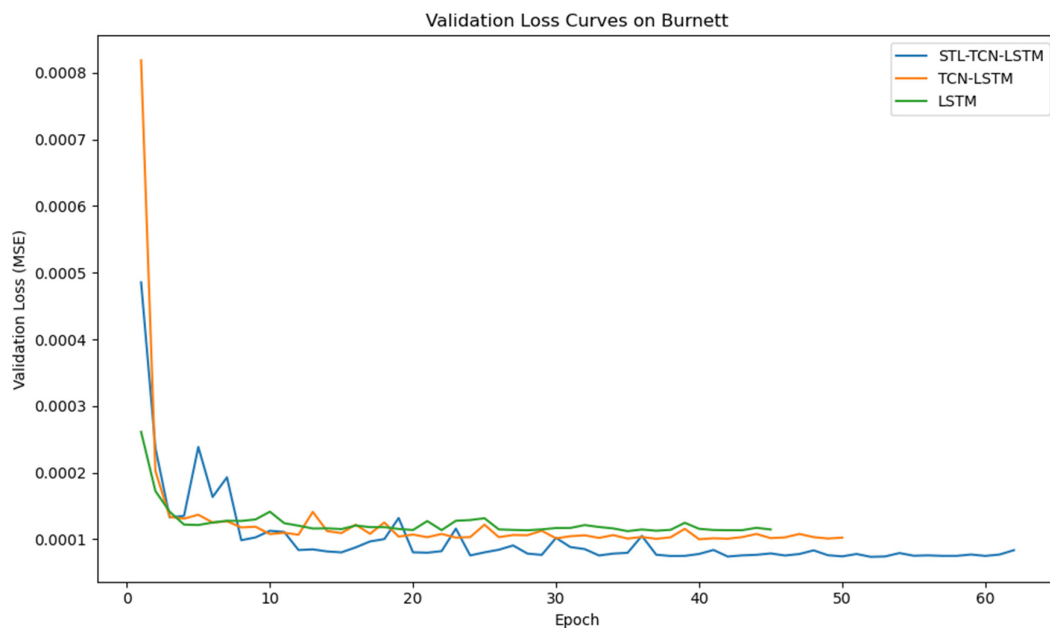


Figure 10. Validation loss curves of the three models on the DO prediction task for Dataset A.

Table 8 compares the total training time of each model under identical training configurations. It can be seen that LSTM has the shortest training time, followed by TCN-LSTM. In contrast, the STL-TCN-LSTM model, which incorporates STL and a multi-branch TCN structure, exhibited a significant increase in training time. This indicates that as the model complexity and parameter scale increase, the proposed model introduces additional computational overhead during the training phase, requiring more computational resources for parameter optimization.

Table 8. Comparison of total training time.

Model	Training Time (s)
LSTM	240.44
TCN-LSTM	596.50
STL-TCN-LSTM	1464.10

Table 9 further compares the inference time of the three models on the test set A. The results show that although the prediction time of STL-TCN-LSTM increases due to its structural complexity, all three models maintain a low computational cost, fully meeting the real-time requirements of practical online forecasting. The additional time overhead is controlled within a reasonable range and does not impose an unbearable computational burden.

Table 9. Prediction time comparison.

Model	Prediction Time (s)
LSTM	0.3631
TCN-LSTM	0.9529
STL-TCN-LSTM	1.8341

In summary, the STL-TCN-LSTM model demonstrates more pronounced advantages in terms of training loss and prediction accuracy, albeit at the cost of increased training and inference time due to its structural complexity. This elevated computational cost represents

a necessary trade-off for achieving superior performance. However, balancing accuracy and efficiency is a critical consideration in practical applications. This study indicates that although the computational overhead of STL-TCN-LSTM is higher, it remains within an acceptable range overall and does not impose an unbearable burden. In real-world scenarios such as river water quality early warning, the accuracy of prediction results is typically prioritized over microsecond-level computational speed. Therefore, for application contexts where prediction accuracy is paramount and corresponding computational resources are available, trading a moderate increase in computation time for a significant enhancement in predictive performance is justifiable and necessary. The STL-TCN-LSTM model achieves a favorable balance in this trade-off, demonstrating strong practical utility.

5. Discussion

The STL-TCN-LSTM model proposed in this study demonstrates robust performance in river water quality. Ablation experiments validated the effectiveness of each module: the STL decouples and mitigates modeling interference caused by multi-scale coupling, enabling the subsequent hybrid TCN-LSTM model to specifically learn the intrinsic patterns within each decomposed component series, thereby significantly enhancing prediction accuracy. Furthermore, comparative experiments conducted on multi-basin datasets with different sampling frequencies indicate that the proposed model is significantly superior to mainstream benchmarks such as LSTM, TCN, and XGBoost, exhibiting strong robustness and generalization abilities.

Although the proposed model demonstrates promising predictive performance, it still presents some limitations. Firstly, the model inputs rely solely on internal water quality indicators, without considering external driving factors such as meteorological and hydrological factors, which constrains its predictive capability in complex environments. Secondly, this study currently focuses on single-step-ahead prediction, while practical water quality early warning requires multi-step-ahead forecasting to provide sufficient response time. Thirdly, the cascade architecture results in high complexity and prolonged training time, hindering its convenient deployment and online updating on edge devices with limited computational resources.

Future research will focus on optimizing the model in the following three aspects: (1) incorporating external influencing factors; (2) extending the prediction task from single-step to multi-step-ahead forecasting to provide a longer early-warning lead time; and (3) adopting lightweight technologies combined with transfer learning to build an edge-adaptive framework, thereby reducing computational and training costs.

6. Conclusions

This paper proposes an STL-TCN-LSTM model that integrates seasonal-trend decomposition with deep learning for predicting DO in rivers. The model first decouples feature interference at different temporal scales by decomposing the original series into trend, seasonal, and residual components using STL. Subsequently, TCN is employed to extract the long-range local features from each component and perform sequence reconstruction. Finally, LSTM captures the dynamic temporal information within the reconstructed sequences, thereby enhancing the overall prediction performance. Experimental results based on data from the Burnett River basin show that the STL-TCN-LSTM model outperforms various baseline models in evaluation metrics such as MAE, RMSE, MAPE, and R^2 . Furthermore, generalization experiments conducted on datasets from the Baffle River and the Fen River (a tributary of the Yellow River) demonstrate that the model maintains stable predictive accuracy under different river conditions and sampling frequencies, and it sustains small errors even during extreme DO variations.

Overall, the STL-TCN-LSTM model effectively combines the advantages of sequence decomposition and deep feature learning, achieving high-precision prediction of DO concentrations. This allows relevant management authorities to anticipate potential hypoxic events or water quality deterioration, thereby enhancing the responsiveness of river ecosystem protection and water resource management. Although the model requires relatively high computational resources during the training phase, the trained model can be deployed on cloud platforms or edge computing devices at monitoring stations, demonstrating immense practical value for practical engineering applications.

Author Contributions: Conceptualization: H.L., H.G. and L.Y.; methodology: H.L. and H.G.; formal analysis and investigation: H.L., H.G. and H.Z.; writing—original draft preparation: H.L.; writing—review and editing: H.L. and H.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

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

References

- Shi, X.; Mao, D.; Song, K.; Xiang, H.; Li, S.; Wang, Z. Effects of landscape changes on water quality: A global meta-analysis. *Water Res.* **2024**, *260*, 121946. [\[CrossRef\]](#)
- Salerno, F.; Gaetano, V.; Gianni, T. Urbanization and climate change impacts on surface water quality: Enhancing the resilience by reducing impervious surfaces. *Water Res.* **2018**, *144*, 491–502. [\[CrossRef\]](#)
- Olyaie, E.; Abyaneh, H.Z.; Mehr, A.D. A comparative analysis among computational intelligence techniques for dissolved oxygen prediction in Delaware River. *Geosci. Front.* **2017**, *8*, 517–527. [\[CrossRef\]](#)
- Kulkarni, S.J. A review on research and studies on dissolved oxygen and its affecting parameters. *Int. J. Res. Rev.* **2016**, *3*, 18–22.
- Wang, X.; Li, Y.; Qiao, Q.; Tavares, A.; Liang, Y. Water quality prediction based on machine learning and comprehensive weighting methods. *Entropy* **2023**, *25*, 1186. [\[CrossRef\]](#)
- Duan, W.; He, B.; Chen, Y.; Zou, S.; Wang, Y.; Nover, D.; Chen, W.; Yang, G. Identification of long-term trends and seasonality in high-frequency water quality data from the Yangtze River basin, China. *PLoS ONE* **2018**, *13*, e0188889. [\[CrossRef\]](#)
- Li, D.; Zhao, W.; Hu, J.; Zhao, S.; Liu, S. A long-term water quality prediction model for marine ranch based on time-graph convolutional neural network. *Ecol. Indic.* **2023**, *154*, 110782. [\[CrossRef\]](#)
- Faruk, D.Ö. A hybrid neural network and ARIMA model for water quality time series prediction. *Eng. Appl. Artif. Intell.* **2010**, *23*, 586–594. [\[CrossRef\]](#)
- Box, G.E.; Jenkins, G.M.; Reinsel, G.C.; Ljung, G.M. *Time Series Analysis: Forecasting and Control*; John Wiley & Sons: Hoboken, NJ, USA, 2015.
- Sims, C.A. Macroeconomics and reality. *Econometrica* **1980**, *48*, 1–48. [\[CrossRef\]](#)
- Lai, G.; Chang, W.-C.; Yang, Y.; Liu, H. Modeling long-and short-term temporal patterns with deep neural networks. In Proceedings of the 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, Ann Arbor, MI, USA, 8–12 July 2018; pp. 95–104.
- Guo, S.; Lin, Y.; Feng, N.; Song, C.; Wan, H. Attention based spatial-temporal graph convolutional networks for traffic flow forecasting. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; pp. 922–929.
- Pulver, A.; Lyu, S. LSTM with working memory. In Proceedings of the 2017 International Joint Conference on Neural Networks (IJCNN), Anchorage, AK, USA, 14–19 May 2017; pp. 845–851.
- Wang, Y.; Zhou, J.; Chen, K.; Wang, Y.; Liu, L. Water quality prediction method based on LSTM neural network. In Proceedings of the 12th International Conference on Intelligent Systems and Knowledge Engineering (ISKE), Nanjing, China, 24–26 November 2017; pp. 1–5.
- Hu, Z.; Zhang, Y.; Zhao, Y.; Xie, M.; Zhong, J.; Tu, Z.; Liu, J. A water quality prediction method based on the deep LSTM network considering correlation in smart mariculture. *Sensors* **2019**, *19*, 1420. [\[CrossRef\]](#) [\[PubMed\]](#)
- Ye, Q.; Yang, X.; Chen, C.; Wang, J. River water quality parameters prediction method based on LSTM-RNN model. In Proceedings of the 2019 Chinese Control and Decision Conference (CCDC), Nanchang, China, 3–5 June 2019; pp. 3024–3028.

17. Dheda, D.; Cheng, L.; Abu-Mahfouz, A.M. Long short term memory water quality predictive model discrepancy mitigation through genetic algorithm optimisation and ensemble modeling. *IEEE Access* **2022**, *10*, 24638–24658. [\[CrossRef\]](#)
18. Khullar, S.; Singh, N. Water quality assessment of a river using deep learning Bi-LSTM methodology: Forecasting and validation. *Environ. Sci. Pollut. Res.* **2022**, *29*, 12875–12889. [\[CrossRef\]](#)
19. Gopi, A.P.; Naik, K.J. A model for analysis of IoT based aquarium water quality data using CNN model. In Proceedings of the 2021 International Conference on Decision Aid Sciences and Application (DASA), Sakheer, Bahrain, 7–8 December 2021; pp. 976–980.
20. Fu, Y.; Hu, Z.; Zhao, Y.; Huang, M. A long-term water quality prediction method based on the temporal convolutional network in smart mariculture. *Water* **2021**, *13*, 2907. [\[CrossRef\]](#)
21. Singha, S.; Pasupuleti, S.; Singha, S.S.; Singh, R.; Kumar, S. Prediction of groundwater quality using efficient machine learning technique. *Chemosphere* **2021**, *276*, 130265. [\[CrossRef\]](#)
22. Chen, H.; Yang, J.; Fu, X.; Zheng, Q.; Song, X.; Fu, Z.; Wang, J.; Liang, Y.; Yin, H.; Liu, Z. Water quality prediction based on LSTM and attention mechanism: A case study of the Burnett River, Australia. *Sustainability* **2022**, *14*, 13231. [\[CrossRef\]](#)
23. Bi, J.; Lin, Y.; Dong, Q.; Yuan, H.; Zhou, M. An improved attention-based LSTM for multi-step dissolved oxygen prediction in water environment. In Proceedings of the 2020 IEEE International Conference on Networking, Sensing and Control (ICNSC), Nanjing, China, 30 October–2 November 2020; pp. 1–6.
24. Barzegar, R.; Aalami, M.T.; Adamowski, J. Short-term water quality variable prediction using a hybrid CNN–LSTM deep learning model. *Stoch. Environ. Res. Risk Assess.* **2020**, *34*, 415–433. [\[CrossRef\]](#)
25. Yang, Y.; Xiong, Q.; Wu, C.; Zou, Q.; Yu, Y.; Yi, H.; Gao, M. A study on water quality prediction by a hybrid CNN-LSTM model with attention mechanism. *Environ. Sci. Pollut. Res.* **2021**, *28*, 55129–55139. [\[CrossRef\]](#) [\[PubMed\]](#)
26. Yuan, J.; Li, Y. Wastewater quality prediction based on channel attention and TCN-BiGRU model. *Environ. Monit. Assess.* **2025**, *197*, 219. [\[CrossRef\]](#)
27. Wu, H.; Xu, J.; Wang, J.; Long, M. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 22419–22430.
28. Ke, S.-L.; Zhang, J.; Guan, D.-G.; Zeng, D.-J.; Qiu, H.-Y.; Feng, Z.-Z.; Li, S.-F.; Hui, Y. Research on Water Demand Prediction in the Yangtze Valley Based on the STL-LSTM Model. In Proceedings of the 2024 10th International Conference on Architectural, Civil and Hydraulic Engineering (ICACHE 2024), Qingdao, China, 25–27 October 2024; pp. 557–565.
29. Chen, N.; Su, C.; Wu, S.; Wang, Y. El Niño index prediction based on deep learning with STL decomposition. *J. Mar. Sci. Eng.* **2023**, *11*, 1529. [\[CrossRef\]](#)
30. Chen, B.; Cao, T.; Yao, L. Research on Short-Term Multi-Step Prediction of River Dissolved Oxygen based on STL-LSTM. *Front. Comput. Intell. Syst.* **2024**, *9*, 5–13. [\[CrossRef\]](#)
31. Zhang, Y.; Li, C.; Jiang, Y.; Sun, L.; Zhao, R.; Yan, K.; Wang, W. Accurate prediction of water quality in urban drainage network with integrated EMD-LSTM model. *J. Clean. Prod.* **2022**, *354*, 131724. [\[CrossRef\]](#)
32. Yan, K.; Li, C.; Zhao, R.; Zhang, Y.; Duan, H.; Wang, W. Predicting the ammonia nitrogen of wastewater treatment plant influent via integrated model based on rolling decomposition method and deep learning algorithm. *Sustain. Cities Soc.* **2023**, *94*, 104541. [\[CrossRef\]](#)
33. Wu, J.; Wang, Z. A hybrid model for water quality prediction based on an artificial neural network, wavelet transform, and long short-term memory. *Water* **2022**, *14*, 610. [\[CrossRef\]](#)
34. Guan, L.; Zhou, Y.; Yang, S. An improved prediction model for COD measurements using UV-Vis spectroscopy. *RSC Adv.* **2024**, *14*, 193–205. [\[CrossRef\]](#)
35. Cleveland, R.B.; Cleveland, W.S.; McRae, J.E.; Terpenning, I. STL: A seasonal-trend decomposition. *J. Off. Stat.* **1990**, *6*, 3–73.
36. Bai, S. An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling. *arXiv* **2018**, arXiv:1803.01271. [\[CrossRef\]](#)
37. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [\[CrossRef\]](#) [\[PubMed\]](#)
38. Queensland Government. Available online: <https://www.qld.gov.au/> (accessed on 29 July 2025).
39. China National Environmental Monitoring Centre. Available online: <https://www.cnemc.cn/> (accessed on 30 July 2025).

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