electronics

Article

Research on Runoff Prediction Based on Time2Vec-TCN-Transformer Driven by Multi-Source Data

Yang Liu [1,*], Yize Wang [2], Xuemei Liu [2], Xingzhi Wang [2], Zehong Ren [2] and Songlin Wu [2]

1 Provincial Collaborative Innovation Center for Efficient Utilization of Water resources in the Yellow River Basin, North China University of Water Resources and Electric Power, Zhengzhou 450046, China
2 School of Information Engineering, North China University of Water Resources and Electric Power, Zhengzhou 450046, China; wangyize@stu.ncwu.edu.cn (Y.W.); liuxuemei@ncwu.edu.cn (X.L.); wangxingzhi@stu.ncwu.edu.cn (X.W.); renzehong@stu.ncwu.edu.cn (Z.R.); x20231090809@stu.ncwu.edu.cn (S.W.)
* Correspondence: yangliu@ncwu.edu.cn; Tel.: +86-130-2775-0150

Abstract: Due to the frequent occurrence of extreme weather in recent years, accurate runoff prediction is crucial for the rational planning and management of water resources. Addressing the high uncertainty and multiple influencing factors in runoff prediction, this paper proposes a runoff prediction method driven by multi-source data. Based on multivariate observed data of runoff, water level, temperature, and precipitation, a Time2Vec-TCN-Transformer model is proposed for runoff prediction research and compared with LSTM, TCN, and TCN-Transformer models. The results show that the Time2Vec-TCN-Transformer model outperforms other models in metrics including MAE, RRMSE, MAPE, and NSE, demonstrating higher prediction accuracy and reliability. By effectively combining Time2Vec, TCN, and Transformer, the proposed model improves the MAPE for forecasting 1–4 days in the future by approximately 7% compared to the traditional LSTM model and 4% compared to the standalone TCN model, while maintaining NSE consistently between 0.9 and 1. This model can better capture the periodicity, long-term scale information, and relationships among multiple variables of runoff data, providing reliable predictive support for flood forecasting and water resources management.

Keywords: multi-source data; runoff prediction; Time2Vec; TCN; Transformer

Citation: Liu, Y.; Wang, Y.; Liu, X.; Wang, X.; Ren, Z.; Wu, S. Research on Runoff Prediction Based on Time2Vec-TCN-Transformer Driven by Multi-Source Data. *Electronics* **2024**, *13*, 2681. https://doi.org/10.3390/electronics13142681

Academic Editors: Hao Xue and Du Huynh

Received: 16 May 2024
Revised: 11 June 2024
Accepted: 19 June 2024
Published: 9 July 2024

1. Introduction

In recent years, with global warming and the frequent occurrence of extreme weather events, we are facing an increasing threat from natural disasters such as floods and droughts [1]. The uncertain meteorological conditions make the prediction of future scenarios more complex [2]. Therefore, improving the quality of runoff prediction, considering issues at different scales, and integrating different features will help to adapt better to today's climate change. This will enable timely adjustments to water resource management and disaster prevention measures, contributing to economic and social development, water resource management, environmental protection, and the safety of people's lives [3].

Runoff prediction models are mainly divided into two categories [4]: process-driven models and data-driven models. Process-driven runoff prediction models are based on physical formation processes, with practical physical significance, describing various processes in hydrological systems through mathematical methods. For example, Manguerra et al. [5] used the SWAT model to address important parameterization issues in watershed hydrological runoff prediction modeling, successfully predicting stream runoff in Indiana with abundant subsurface drainage. Data-driven runoff prediction models [6], on the other hand, mainly rely on statistical and machine learning techniques, learning patterns from historical observational data without the need for predefined physical laws. For instance, Carlson et al. [7] used an autoregressive moving average (ARMA) model to analyze the annual runoff time series of rivers such as the Saint Lawrence, Missouri, Neva, and Niger, successfully predicting the runoff of the Missouri River one year in advance. In recent years, with the continuous development of artificial intelligence technology, deep learning has been applied to the prediction field and has achieved significant results [8]. The main characteristic of deep learning models is their ability to automatically learn feature representations of data and model complex relationships through multi-layer neural network structures. For example, Li et al. [9] tested the Long Short-Term Memory (LSTM) network model in a watershed in Houston, Texas, using 10 years of precipitation and river flow data from 153 rain gauges. They designed numerical experiments to evaluate the performance of the established model in predicting river flow. Xuan-Hien Le et al. [10] used Gated Recurrent Unit (GRU) networks to predict water levels at downstream locations 1–4 time steps ahead in the Anshou irrigation culvert of the Ro River in Vietnam. The study results indicate that when the target forecasting station is significantly affected by tides, the GRU model demonstrates good performance with only a small amount of data. Additionally, Amanambu et al. [11] used Transformer for predicting hydrological data in the Apalachicola River in Florida, showing that Transformer can accurately predict hydrological droughts in the Apalachicola River, aiding in water resource planning and drought mitigation in the region.

Additionally, single models are often limited in capturing nonlinear features, which may result in issues such as low accuracy and poor stability. In contrast, hybrid models demonstrate outstanding performance in terms of prediction accuracy and stability. For example, Huiqi Deng et al. [12] combined CNN and LSTM to study their applicability in runoff simulation and the impact of input parameters on model prediction performance. The study showed that the CNN-LSTM model outperformed the LSTM model in predicting daily runoff, significantly improving prediction accuracy. Guangchao Qiao et al. [13] constructed a PSO-SVR long-term prediction model, using Particle Swarm Optimization (PSO) algorithm to determine the penalty coefficient, insensitive coefficient, and gamma parameter of the Support Vector Regression (SVR) Gaussian radial basis kernel function. Experimental results demonstrated that compared to multiple regression analysis, the PSO-SVR model exhibited higher prediction accuracy, stronger stability, and greater credibility. Wenchuan Wang et al. [14] proposed a hybrid prediction model VF-EMD-SSA-ELM, which combines Time-Varying Filtering (TVF) Empirical Mode Decomposition (EMD), Salp Swarm Algorithm (SSA), and Extreme Learning Machine (ELM), applied to monthly runoff forecasting at Manwan Hydropower Station, Hongjiadu Hydropower Station, and Yingluoxia Hydrological Station. The experiments demonstrate that the prediction accuracy of this model is significantly superior to that of individual models.

With the increase in data volume and diversification of data types, integrating multi-source data with deep learning algorithms has become one of the key research methods in the field of runoff prediction [15]. At the same time, constructing hybrid models as an effective approach to improving prediction accuracy is widely applied. Faced with time series prediction problems, early solutions typically used recurrent neural networks [16], but traditional recurrent neural networks suffer from problems such as vanishing gradients and exploding gradients, as well as difficulties in capturing long-term temporal dependencies. Moreover, traditional GRU [17], LSTM [18], and various recurrent neural networks are structurally similar to Markov decision processes [19], making it difficult to learn global temporal information. In multivariate problems, traditional time series models also struggle to capture the relationships between multiple variables and learn the feature information of each variable. As convolutional neural networks are increasingly applied to sequence problems [20], they provide more insights into multivariate time series prediction tasks. With the development of artificial intelligence, Generative Artificial Intelligence (Gen-AI) also provides new benchmarks for predictive practices [21]. Meanwhile, employing statistical tests such as Diebold–Mariano (DM) [22], Kolmogorov–Smirnov (KS) [23], etc., assists in effectively assessing the accuracy and stability of models, further enhancing the accuracy and efficiency of predictive tasks.

Against this background, this study proposes a data-driven multivariate runoff prediction model based on Time2Vec-TCN-Transformer, which predicts and evaluates daily runoff. Experiments show that the Time2Vec-TCN-Transformer model significantly improves prediction accuracy and reliability compared to traditional and single models.

2. Related Work

2.1. Time2Vec

Time2Vec, as a method of temporal encoding, employs a functional encoding calculation approach to obtain the relative positions of time series. It retrieves corresponding vectors by indexing the position numbers through a matrix and then trains them [24].

Time2Vec has a periodic characteristic, where the periodic pattern primarily accounts for seasonal variations in runoff prediction, while the non-periodic pattern captures extreme natural events or anomalous situations. Time2Vec is capable of simultaneously handling both periodic and non-periodic patterns, thereby comprehensively representing the features of time series data, enabling the model to make more accurate predictions and analyses. Moreover, Time2Vec exhibits stability, implying that the model maintains good performance across different time ranges and is not prone to failure or performance degradation due to changes in time scale. This stability enables Time2Vec to demonstrate good generality and applicability when dealing with data of different time spans. Additionally, compared to other encoding methods, Time2Vec is more easily embeddable into different types of models. Its formula expression is as follows:

$$t2v(\tau)[i] = \begin{cases} \omega_i \tau + \varphi_i, & if\ i = 0 \\ \mathcal{F}(\omega_i \tau + \varphi_i), & if\ 1 \leq i \leq k \end{cases} \tag{1}$$

In the equation, the time vector is denoted by τ, and its embedded time vector representation is $t2v(\tau)$, with a size of $k+1$; ω is the frequency parameter used to introduce periodic components; φ is the phase offset used to adjust the initial phase of time encoding; and F is an activation function for the period.

2.2. TCN

TCNs (Temporal Convolutional Networks) are a specialized type of convolutional neural network. It is an improved version based on convolutional neural networks, introducing dilated convolutions and residual modules for time series processing while adhering to causality constraints [25]. TCNs feature gradient stability and higher efficiency. Additionally, they avoids the risk of data leakage by not introducing future information.

2.2.1. Causal Convolution

Causal convolution is one of the key components in the TCN (Temporal Convolutional Network) model, designed to simulate the sequential characteristics of time series data. In tasks such as time series prediction, it is typically undesirable for the model to use future information to predict current outputs, as this can lead to information leakage and model instability. Causal convolution ensures that each output time step depends only on past time steps of the input sequence and not on future time steps. When the causal convolution kernel slides, it only moves forward, avoiding the acquisition of future information. In other words, yt is determined by $x0$, $x1$, …, xt, rather than using future inputs, $xt+1$, $xt+2, \ldots$ [26]. Additionally, the first TCN layer is a one-dimensional fully convolutional network, where each intermediate layer has the same size as the input layer and uses zero-padding of the same size to obtain subsequent layers of the same size [27]. This characteristic enables causal convolution to simulate the temporal order of sequence data, preventing information leakage and the utilization of future information, making it better suited for tasks such as time series prediction.

2.2.2. Dilated Convolution

Dilated convolution is an important technique used in the TCN (Temporal Convolutional Network) model to capture long-term dependencies. In traditional convolution operations, the receptive field of the convolution kernel is usually limited by the kernel size, resulting in the model only capturing short-term time dependencies. Dilated convolution is a convolution operation that increases the receptive field by introducing holes in the convolution kernel. This ensures that the receptive field of the convolution kernel is expanded while maintaining the same input and output sizes, thereby capturing larger-scale temporal sequence information [28]. The principle behind dilated convolution is to increase the receptive field of convolution layers by introducing blank spaces based on dilation factors in the convolution network, enabling the network to capture long-term dependencies in sequences and improve the model's generalization capability. The specific formula for dilated convolution is as follows:

$$F(X_t) = (X_d^* F)(X_t) = \sum_{k=1}^{K} f_k \cdot X_{(t-d)(K-k)} \tag{2}$$

In the equation, X_t is the value of the input time series at time step t, $*$ is the dilation convolution operation, f_k is the kth weight of the convolution kernel, K is the size of the filter, d is the dilation factor, and the receptive field is denoted as $F = (K-1)d + 1$.

2.2.3. Residual Connection

Residual connection is a method of transmitting information by adding the input to the output [29]. The residual connection in the TCN consists of a one-dimensional fully convolutional network and residual blocks. It connects the input data with the output of the previous layer to train the model and avoid the vanishing gradient problem. The model adds the input sequence to the sequence output of the convolutional layer. Unlike traditional ResNet models that directly add the input sequence to the output sequence of the residual block, which may fail to learn all useful information from residual blocks, TCN adds an additional 1×1 convolution to ensure the input and output sequences have the same size. If the dimensions of the input and output do not match, this convolutional layer adjusts the input to return to the original number of channels. If the dimensions match, the input is directly passed as the residual for backward propagation. Finally, the residual is added to the output to obtain the final output.

2.3. Transformer

Transformer is a deep learning model architecture proposed by Vaswani et al. in 2017 [30] and widely used in NLP (natural language processing) tasks. The core of Transformer is the Self-Attention mechanism, which eliminates the sequential dependency problem in traditional recurrent neural networks and convolutional neural networks by introducing attention mechanism. This enables Transformer to process different positional information in input sequences in parallel, playing an excellent role in various NLP tasks. In recent years, Transformer has also been applied in the field of time series prediction.

The Transformer model consists of multiple encoder layers and decoder layers. However, in time series prediction, the decoder part can be omitted. On the one hand, this can reduce computational costs, and on the other hand, it can reduce the risk of overfitting [31]. Transformer includes the following components:

1. Embedding Layer: Similar to the embedding layer in NLP, each time step in the time series data is encoded through the embedding layer to transform it into a vector representation. These vector representations contain information about the time step as well as other relevant features.
2. Positional Encoding: Since Transformer lacks built-in capabilities to handle temporal information, positional encoding is added to inform the model about the relative

position of each time step. This can be achieved by adding positional encoding vectors to the embedding vectors. The specific formula is as follows:

$$PE(pos, 2i) = \sin(\frac{pos}{10000^{\frac{2i}{d_{\text{model}}}}}) \tag{3}$$

$$PE(pos, 2i+1) = \cos(\frac{pos}{10000^{\frac{2i}{d_{\text{model}}}}}) \tag{4}$$

In the equation, pos denotes the position information.

3. Self-Attention Mechanism: Self-attention mechanism aids the model in capturing dependencies between different time steps in a sequence. It allows the model to selectively attend to other time steps for each time step to determine their importance to the current time step. This enables the model to capture long-term dependencies in the time series. Self-attention mechanism is a type of attention mechanism that reduces reliance on external information and is better at capturing internal correlations within data or features. Compared to traditional recurrent neural networks, it exhibits superior parallel computing capabilities.
This module has three input vectors—query vector Q, key vector K, and value vector V—all of which are computed from input vectors. By computing the dot product of a single query vector and all key vectors, dividing it by $\sqrt{d_k}$, and then applying a $softmax$ function to obtain corresponding weights, the model weights the value vectors. The specific formula is as follows:

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{5}$$

In the equation, d_k represents the dimensionality of the query vector Q and the key vector K.
4. Residual Connection: This allows the network to focus only on the current differences, preventing the problem of vanishing gradients caused by deepening network layers.
5. Feedforward Neural Network: It is a two-layer fully connected layer that further processes the output of the self-attention mechanism.
6. Output Layer: The output of the encoder layer is fed into a fully connected output layer to generate predictions for the time series. Meanwhile, the dimensionality of the output layer matches the dimensionality of the time series.

2.4. Time2Vec-TCN-Transformer Prediction Model

Due to the influence of multiple factors on runoff changes, this paper adopts a multivariate input single-variable output mode for design. Combining the mechanisms of Time2Vec, TCN, and Transformer, a Time2Vec-TCN-Transformer model is proposed. Figure 1 shows the structure of the Time2Vec-TCN-Transformer model. The model reads the data using a sliding window approach, with a stride of 7 and a batch size of 128. MSE (Mean Squared Error) is used as the loss function, Adam is used as the optimizer, and the learning rate is set to 0.001. The specific workflow is as follows:

1. First, select four feature variables including daily runoff, water level, temperature, and precipitation as input data, with daily runoff as the output data. Then, perform normalization to uniformly adjust the original data to the interval [0, 1], to accelerate the convergence speed of the model and improve the prediction accuracy.
2. After preprocessing, Time2Vec is used as the positional embedding. The data are mapped from the original 4-dimensional feature space to a 64-dimensional hidden space using Time2Vec, which introduces periodic features of time using sinusoidal functions. Then, the data are mapped back to the original dimensionality through linear transformation and fed into the TCN layer. Time2Vec is employed to address the lack of a learnable encoding mechanism in Transformer.

3. In the TCN layer, multiple convolutional layers are utilized to extract features from the input time series data. Dilated convolution is employed to capture longer-term temporal dependencies. Weight normalization and ReLU activation functions are used as residual connections between layers. The output is then fed into the Transformer layer. Specifically, a hidden layer with units [1,4,16,64] is defined. The convolutional kernel size is set to 1×3, and the dilation factors are 1, 2, 4, and 8, respectively.
4. In the Transformer layer, positional embedding and the data processed through the TCN layer are further processed through an attention mechanism to introduce correlation information between different time steps and different feature variables. The output is then passed through residual connections and layer normalization before being fed into a feedforward network. Since only a single variable, runoff volume, needs to be output, no decoder parallel computation is used. Instead, a single feedforward neural network is employed to map the data to one dimension for output.
5. Finally, after training is completed, the predicted results are inverse-normalized to more accurately assess the gap between the predicted values and the actual values.

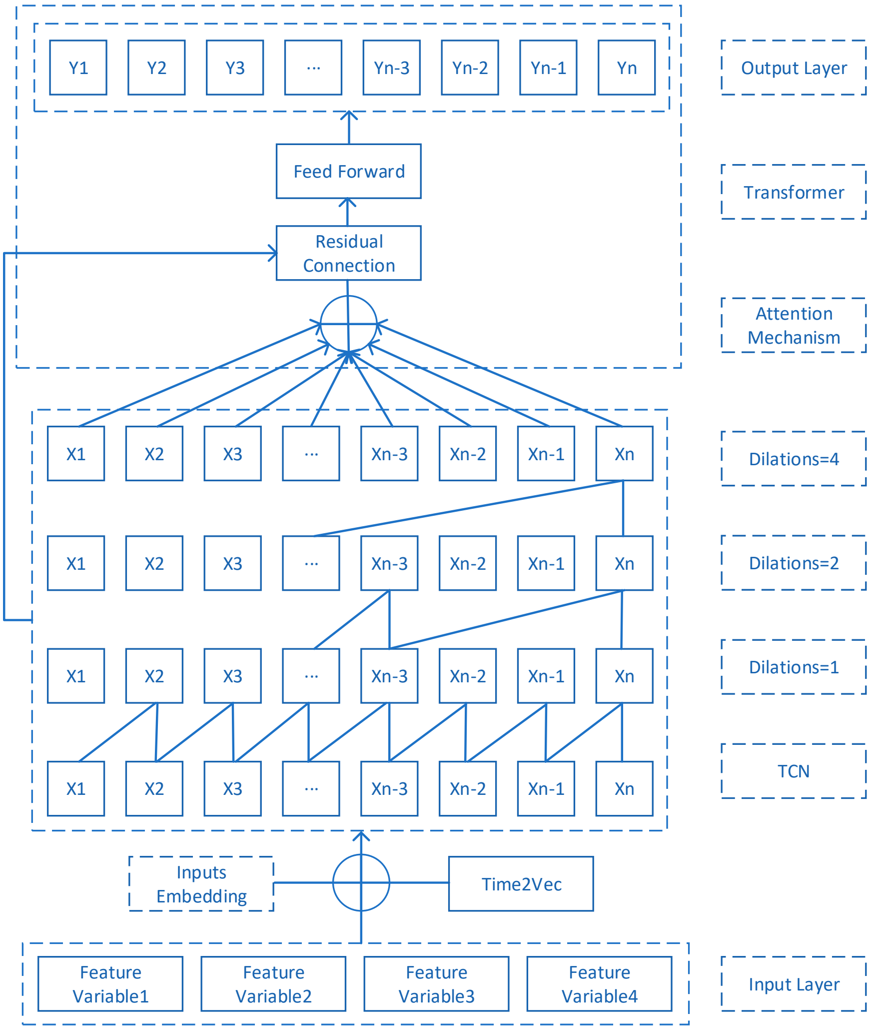


Figure 1. Time2Vec-TCN-Transformer model flowchart.

3. Experiment and Results

3.1. Data Description

The Huayuankou area in the Yellow River Basin marks the beginning of the "hanging river above the ground" in the lower reaches of the Yellow River. Due to its unique geographical location, it is often heavily affected by flood disasters. The hydrological station at Huayuankou is located 4696 km from the source of the river and 768 km from its mouth, controlling a drainage area of 730,000 km^2, which accounts for 92% of the total drainage area of the Yellow River Basin. Therefore, data from the Huayuankou hydrological station have always been crucial for flood control, water resource management, and development in the Yellow River Basin [32]. In recent years, extreme weather events have become more frequent, and disasters such as the "7·20 super flood" have become increasingly unpredictable [33]. Therefore, this study aims to provide more data-driven support for runoff prediction in order to address these challenges.

We know that runoff changes are influenced by various factors, among which the trends in factors such as rainfall, water level, and temperature are similar to those of runoff changes [34]. Therefore, this study selected the observed data of runoff, water level, temperature, and precipitation from the Huayuankou hydrological station in the Yellow River Basin and the meteorological station in Zhengzhou City from 1 January, 2010, to 1 December, 2020, totaling 3988 × 4 days, as the research objects (the data presented in this study are available on request from the corresponding author). These data were divided into 60% for the training set and 40% for the testing set. MAE, RMSE, MAPE, and NSE were used as evaluation criteria for the model, and the performance of the Time2Vec-TCN-Transformer model was compared with LSTM, TCN, and TCN-Transformer models. The hyperparameter settings for LSTM, TCN, and TCN-Transformer are the same as those for the Time2Vec-TCN-Transformer model. Figure 2 shows the division of runoff data into training and testing sets.

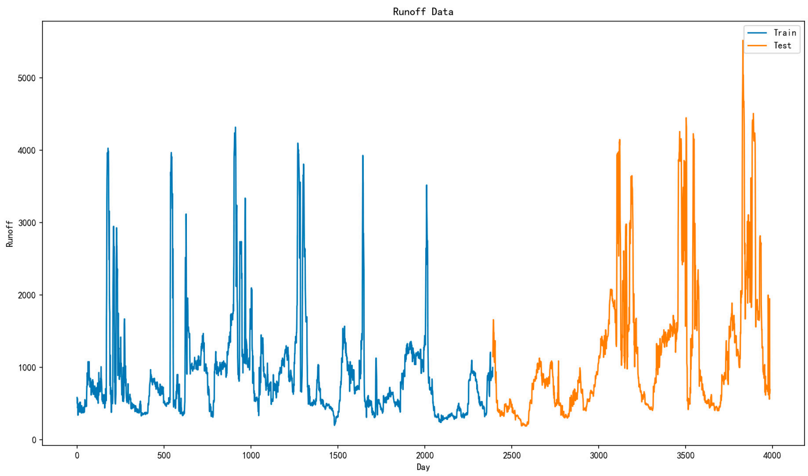


Figure 2. The runoff data divided into the training set and the testing set.

3.2. Performance Evaluation

This study adopted MAE (Mean Absolute Error), RRMSE (Relative Root Mean Square Error), MAPE (Mean Absolute Percentage Error), and NSE (Nash–Sutcliffe Efficiency) as evaluation criteria for the model. When MAE, RMSE, and MAPE are smaller, the model's predictions are more accurate. When NSE is closer to 1, it indicates better prediction performance, while when NSE is closer to 0, it indicates lower reliability of the model. Additionally, since the differences in prediction errors between different models may not be significant, this study not only used the values of MAE, RRMSE, MAPE, and NSE to evaluate model performance, but also employed the Diebold–Mariano (DM) test for statistical significance testing. The formulas are as follows:

$$MAE = \frac{1}{n}\sum_{t=1}^{n}\left|y_t - y_t^{'}\right| \tag{6}$$

$$MAPE = \frac{100\%}{n}\sum_{t=1}^{n}\left|\frac{y_t - y_t^{'}}{y_t}\right| \tag{7}$$

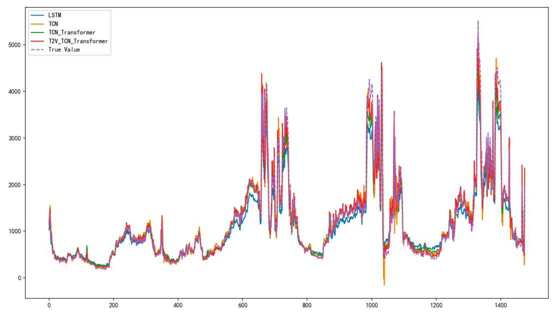
$$RRMSE = \frac{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}}{\frac{1}{n}\sum_{i=1}^{n} y_i} \tag{8}$$

$$NSE = 1 - \frac{\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2}{\sum_{i=1}^{n}\left(y_i - \bar{y}_i\right)^2} \tag{9}$$

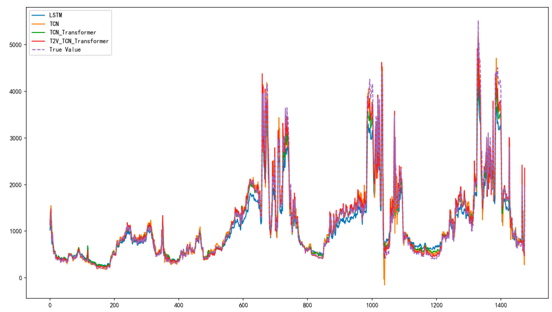
where n represents the length of the runoff sequence, y_i denotes the true runoff values, $\hat{y}_i$ represents the model predictions, and $\overline{y}$ stands for the mean of the predictions.

3.3. Results and Analysis

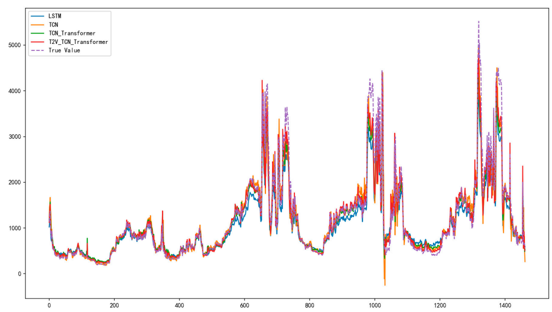
The paper utilizes continuous observations spanning 7 days as input for the model and forecasts future observations spanning 1 to 4 days as output for training, followed by predictions on the test set. Figure 3 presents a comparison of the runoff prediction results of the Time2Vec-TCN-Transformer model with LSTM, TCN, and TCN-Transformer models for the next 4 days (where blue, orange, green, red, and purple represent the predicted values of LSTM, TCN, TCN-Transformer, and the true values of the sequence, respectively). Among them, LSTM has a larger overall error compared to other models, while TCN, although its prediction error is smaller than LSTM's, has a larger error in predicting peak flow than other models. However, the Time2Vec-TCN-Transformer model, through the mechanism of Transformer, better learns the overall trend in feature changes, thereby improving the nonlinear fitting ability of TCN. The results indicate that the Time2Vec-TCN-Transformer model has a better fitting effect compared to the other three models, with higher prediction accuracy.



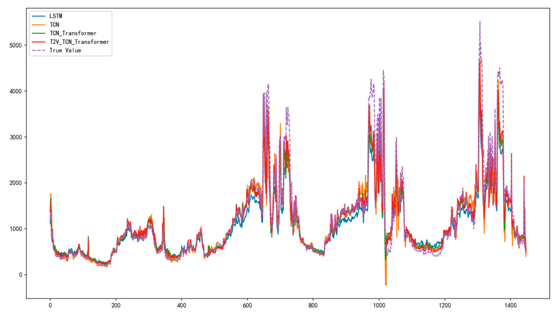
(**a**) Next day.



(**b**) Next 2 days.



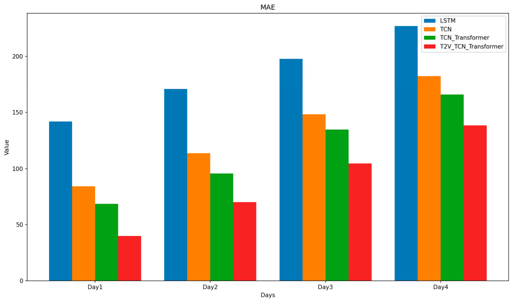
(**c**) Next 3 days.



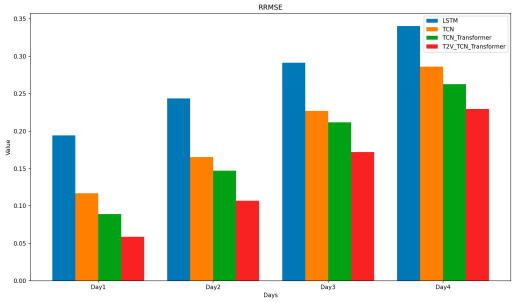
(**d**) Next 4 days.

Figure 3. Comparison of results for different forecast times.

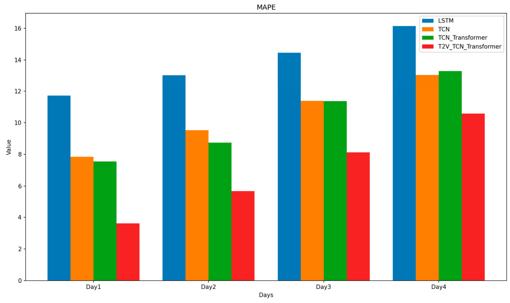
Figure 4 presents bar charts showing the MAE, RRMSE, MAPE, and NSE for the predictions of the four models, indicating that the Time2Vec-TCN-Transformer model exhibits higher reliability compared to the other three models.



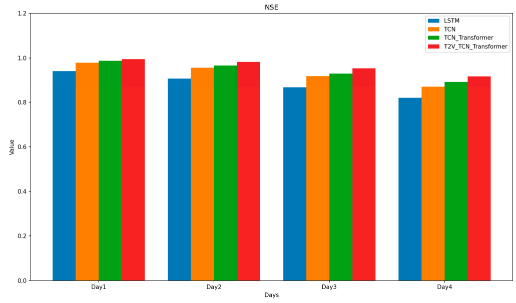
(**a**) MAE.



(**b**) RRMSE.



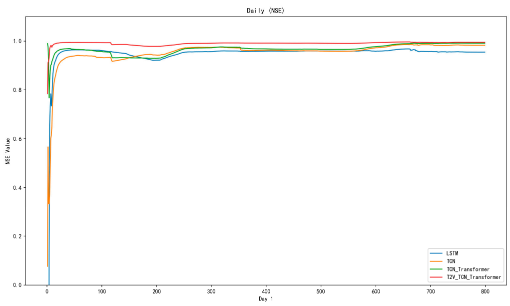
(**c**) MAPE.



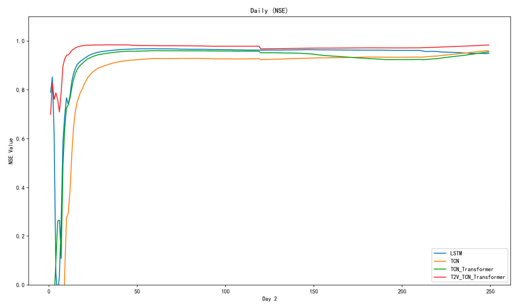
(**d**) NSE.

Figure 4. The error metric charts for different forecast times.

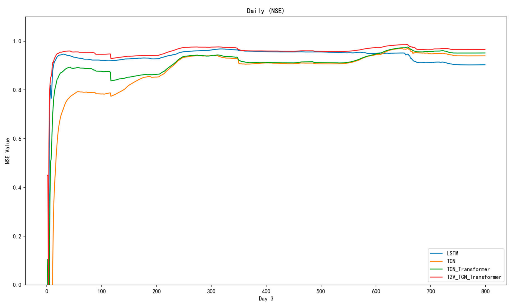
Figure 5 shows the real-time NSE results of the four models for predicting runoff volume for the next 4 days. It can be observed that the curve trend in the Time2Vec-TCN-Transformer model is more favorable, with higher stability across different time periods compared to other models, indicating better generalization ability.



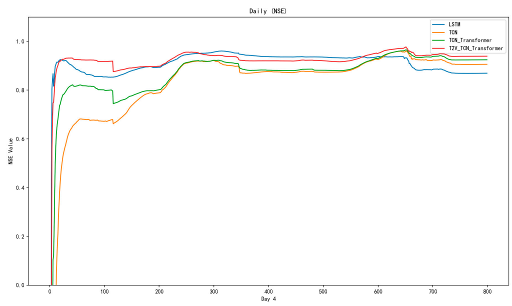
(**a**) Next day.



(**b**) Next 2 days.



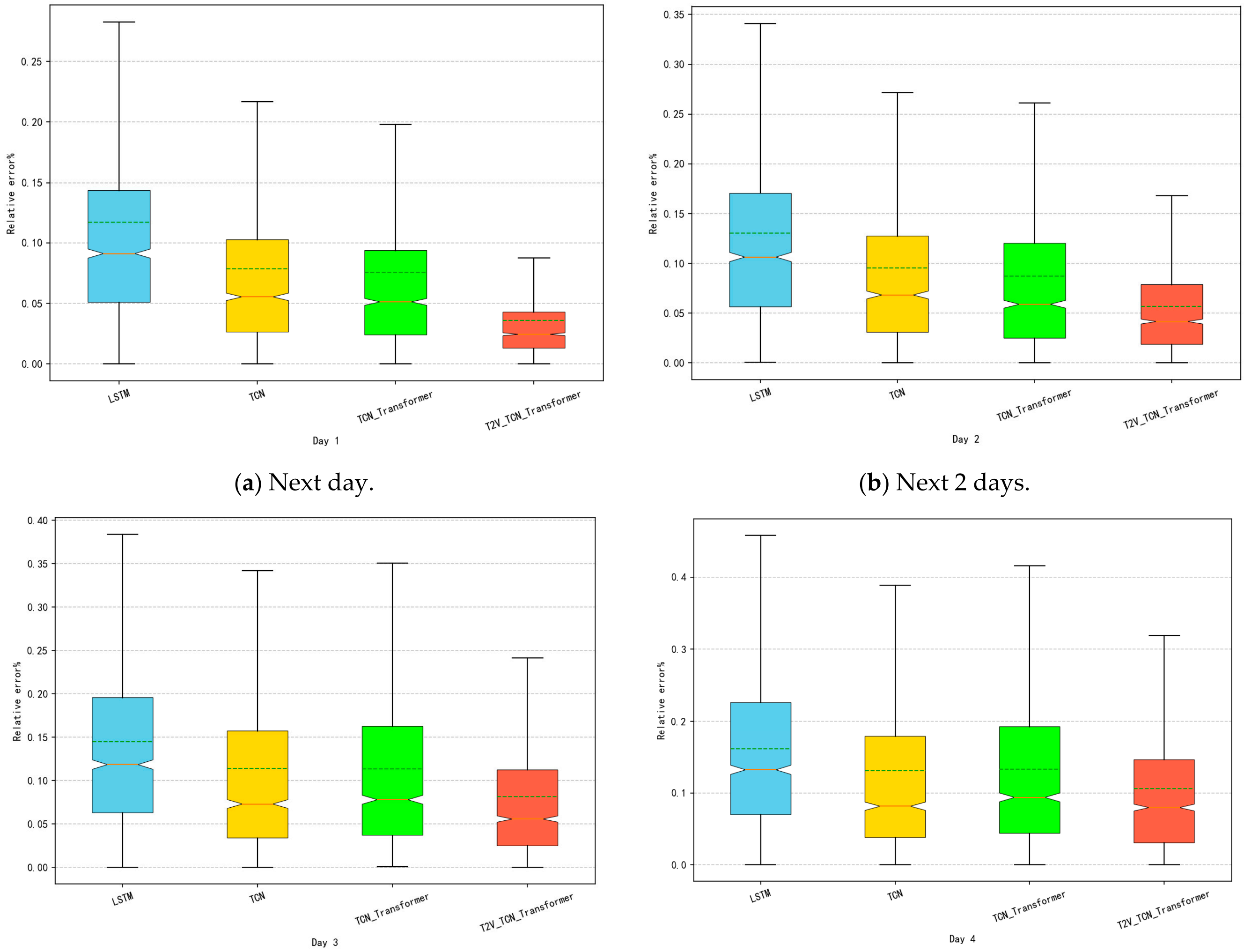
(**c**) Next 3 days.



(**d**) Next 4 days.

Figure 5. NSE for different forecast times.

Figure 6 depicts the box plots of relative errors for the predictions of runoff for the next 4 days by the four models (where the dashed line represents the mean of the errors and the solid line represents the median of the errors). It can be observed that the Time2Vec-TCN-Transformer model has smaller error intervals, lower means, and lower medians compared to the other three models when predicting for the next 1–4 days.



(**a**) Next day. (**b**) Next 2 days.

(**c**) Next 3 days. (**d**) Next 4 days.

Figure 6. Box plot of relative errors for different forecast times.

Table 1 presents the DM statistics comparing the Time2Vec-TCN-Transformer model with the other three models. The DM statistic indicates the difference in prediction error sequences between two models. A larger DM value signifies a greater difference between the two models, while a DM value less than 0 indicates that the Time2Vec-TCN-Transformer model outperforms the compared model. When the significance level is less than 5%, the null hypothesis is rejected, suggesting that the two models perform differently, implying a significant difference. Conversely, a p-value greater than 5% does not reject the null hypothesis. The results show that, in the test set, the DM test results reject the null hypothesis at a 1% significance level for all comparisons. This indicates that the Time2Vec-TCN-Transformer model has a significant difference compared to the other three models, with superior prediction accuracy and stability.

Table 1. The results of the DM test.

Model Comparison	Next Day	Next 2 Days	Next 3 Days	Next 4 Days
Time2Vec-TCN-Transformer: LSTM	−16.277 ***	−12.187 ***	−8.857 ***	−6.911 ***
Time2Vec-TCN-Transformer: TCN	−11.015 ***	−7.103 ***	−4.922 ***	−3.848 ***
Time2Vec-TCN-Transformer: TCN-Transformer	−8.743 ***	−4.525 ***	−3.541 ***	−2.522 ***

Note: *** indicates a 1% significance level.

The results from Figures 3–6 and Table 2 collectively demonstrate that the Time2Vec-TCN-Transformer model outperforms the traditional LSTM model and the standalone TCN model in predicting future 1–4-day MAPE, with improvements of approximately 7% and 4%, respectively. When predicting the runoff volume for the next day, the MAE increases by 72% and 52.9%, while the RRMSE increases by 69.9% and 50% compared to LSTM and TCN models, respectively. When predicting the runoff volume for the next 2 days, the MAE increases by 58.8% and 38.2%, and the RRMSE increases by 55.8% and 35.3% compared to LSTM and TCN models, respectively. When predicting the runoff volume for the next 3 days, the MAE increases by 47% and 29.4%, and the RRMSE increases by 41% and 24.4% compared to LSTM and TCN models, respectively. When predicting the runoff volume for the next 4 days, the MAE increases by 39% and 19%, and the RRMSE increases by 32.6% and 5.3% compared to LSTM and TCN models, respectively. This suggests that the Time2Vec-TCN-Transformer model achieves good performance in short-term prediction. While the improvement decreases slightly in the prediction for days 2–4 in the future, it still remains within an acceptable range, further validating the predictive capability of the model. In addition, NSE performs better at different times compared to the other three models, and NSE remains between 0.9 and 1. This indicates that the Time2Vec-TCN-Transformer model has smaller errors and higher credibility in runoff prediction.

Table 2. Error evaluation metrics for the four models.

Time	Model	MAE	RRMSE	MAPE	NSE
Next day	LSTM	141.726	0.194	11.724	0.941
	TCN	84.451	0.118	7.847	0.978
	TCN-Transformer	68.544	0.089	7.556	0.988
	Time2Vec-TCN-Transformer	39.825	0.059	3.610	0.995
Next 2 days	LSTM	170.587	0.243	13.033	0.908
	TCN	113.791	0.166	9.531	0.953
	TCN-Transformer	95.834	0.147	8.741	0.966
	Time2Vec-TCN-Transformer	70.293	0.108	5.678	0.982
Next 3 days	LSTM	197.799	0.292	14.464	0.868
	TCN	148.350	0.227	11.399	0.919
	TCN-Transformer	134.716	0.212	11.367	0.930
	Time2Vec-TCN-Transformer	104.758	0.172	8.133	0.954
Next 4 days	LSTM	226.870	0.340	16.144	0.820
	TCN	182.477	0.287	13.054	0.872
	TCN-Transformer	165.778	0.263	13.303	0.893
	Time2Vec-TCN-Transformer	138.303	0.229	10.590	0.918

4. Conclusions

This study considers runoff prediction not solely influenced by a single factor, thus introducing multiple relevant variables for experimental research and proposing a runoff prediction model based on Time2Vec-TCN-Transformer. This model enhances the cyclical acquisition ability of runoff data through Time2Vec to overcome the lack of learnable encoding mechanisms in Transformer. It captures long-term scale information of time series through TCN and obtains mutual relationship information among multiple variables through Transformer. By effectively integrating them, it enhances the predictive ability for single models and single runoff information. Compared to single models, the Time2Vec-TCN-Transformer model excels in accuracy, reliability, and learning from time and multi-variable information.

Since this study only utilized runoff, water level, temperature, and precipitation as learning information for the model, while actual runoff variation may be influenced by more factors, further research is needed to address the impact of different factors on runoff and to continue improving the model performance.

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

Funding: This research was funded by The Education Department of Henan Province grant number 24A520021 And The APC was funded by The Education Department of Henan Province.

Data Availability Statement: The data that support the findings of this study can be accessed upon reasonable request from the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

References

1. Jabbari, A.; Bae, D.-H. Application of Artificial Neural Networks for Accuracy Enhancements of Real-Time Flood Forecasting in the Imjin Basin. *Water* **2018**, *10*, 1626. [CrossRef]
2. de la Fuente, A.; Meruane, V.; Meruane, C. Hydrological Early Warning System Based on a Deep Learning Runoff Model Coupled with a Meteorological Forecast. *Water* **2019**, *11*, 1808. [CrossRef]
3. Silvestro, F.; Gabellani, S.; Rudari, R.; Delogu, F.; Laiolo, P.; Boni, G. Uncertainty reduction and parameter estimation of a distributed hydrological model with ground and remote-sensing data. *Hydrol. Earth Syst. Sci.* **2015**, *19*, 1727–1751. [CrossRef]
4. Yang, Q.; Zhang, H.; Wang, G.; Luo, S.; Chen, D.; Peng, W.; Shao, J. Dynamic runoff simulation in a changing environment: A data stream approach. *Environ. Model. Softw.* **2019**, *112*, 157–165. [CrossRef]
5. Manguerra, H.; Engel, B. HYDROLOGIC PARAMETERIZATION OF WATERSHEDS FOR RUNOFF PREDICTION USING SWAT 1. *JAWRA J. Am. Water Resour. Assoc.* **1998**, *34*, 1149–1162. [CrossRef]
6. Carleo, G.; Cirac, I.; Cranmer, K.; Daudet, L.; Schuld, M.; Tishby, N.; Vogt-Maranto, L.; Zdeborová, L. Machine learning and the physical sciences. *Rev. Mod. Phys.* **2019**, *91*, 045002. [CrossRef]
7. Salas, J.D.; Tabios III, G.Q.; Bartolini, P. Approaches to multivariate modeling of water resources time series 1. *JAWRA J. Am. Water Resour. Assoc.* **1985**, *21*, 683–708. [CrossRef]
8. Guo, Y.; Liu, Y.; Oerlemans, A.; Lao, S.; Wu, S.; Lew, M.S. Deep learning for visual understanding: A review. *Neurocomputing* **2016**, *187*, 27–48. [CrossRef]
9. Li, W.; Kiaghadi, A.; Dawson, C. High temporal resolution rainfall–runoff modeling using long-short-term-memory (LSTM) networks. *Neural Comput. Appl.* **2021**, *33*, 1261–1278. [CrossRef]
10. Le, X.-H.; Ho, H.V.; Lee, G. Application of gated recurrent unit (GRU) network for forecasting river water levels affected by tides. In Proceedings of the APAC 2019: The 10th International Conference on Asian and Pacific Coasts, Hanoi, Vietnam, 25–28 September 2019; pp. 673–680.
11. Amanambu, A.C.; Mossa, J.; Chen, Y.-H. Hydrological drought forecasting using a deep transformer model. *Water* **2022**, *14*, 3611. [CrossRef]
12. Deng, H.; Chen, W.; Huang, G. Deep insight into daily runoff forecasting based on a CNN-LSTM model. *Nat. Hazards* **2022**, *113*, 1675–1696. [CrossRef]

13. Qiao, G.; Yang, M.; Zeng, X. Monthly-scale runoff forecast model based on PSO-SVR. *J. Phys. Conf. Ser.* **2022**, *2189*, 012016. [CrossRef]
14. Wang, W.-C.; Cheng, Q.; Chau, K.-W.; Hu, H.; Zang, H.-F.; Xu, D.-M. An enhanced monthly runoff time series prediction using extreme learning machine optimized by salp swarm algorithm based on time varying filtering based empirical mode decomposition. *J. Hydrol.* **2023**, *620*, 129460. [CrossRef]
15. Jiang, Y.; Li, C.; Sun, L.; Guo, D.; Zhang, Y.; Wang, W. A deep learning algorithm for multi-source data fusion to predict water quality of urban sewer networks. *J. Clean. Prod.* **2021**, *318*, 128533. [CrossRef]
16. Zhang, J.W.; Chen, X.H.; Khan, A.; Zhang, Y.K.; Kuang, X.X.; Liang, X.Y.; Taccari, M.L.; Nuttall, J. Daily runoff forecasting by deep recursive neural network. *J. Hydrol.* **2021**, *596*, 126067. [CrossRef]
17. Gao, S.; Huang, Y.F.; Zhang, S.; Han, J.C.; Wang, G.Q.; Zhang, M.X.; Lin, Q.S. Short-term runoff prediction with GRU and LSTM networks without requiring time step optimization during sample generation. *J. Hydrol.* **2020**, *589*, 125188. [CrossRef]
18. Xu, Y.; Hu, C.; Wu, Q.; Jian, S.; Li, Z.; Chen, Y.; Zhang, G.; Zhang, Z.; Wang, S. Research on particle swarm optimization in LSTM neural networks for rainfall-runoff simulation. *J. Hydrol.* **2022**, *608*, 127553. [CrossRef]
19. Rust, J. Structural estimation of Markov decision processes. *Handb. Econom.* **1994**, *4*, 3081–3143.
20. Song, C.M. Data construction methodology for convolution neural network based daily runoff prediction and assessment of its applicability. *J. Hydrol.* **2022**, *605*, 127324. [CrossRef]
21. Hassani, H.; Silva, E.S. Predictions from Generative Artificial Intelligence Models: Towards a New Benchmark in Forecasting Practice. *Information* **2024**, *15*, 291. [CrossRef]
22. Diebold, F.X.; Mariano, R.S. Comparing predictive accuracy. *J. Bus. Econ. Stat.* **2002**, *20*, 134–144. [CrossRef]
23. Hassani, H.; Silva, E.S. A Kolmogorov-Smirnov based test for comparing the predictive accuracy of two sets of forecasts. *Econometrics* **2015**, *3*, 590–609. [CrossRef]
24. Kazemi, S.M.; Goel, R.; Eghbali, S.; Ramanan, J.; Sahota, J.; Thakur, S.; Wu, S.; Smyth, C.; Poupart, P.; Brubaker, M. Time2vec: Learning a vector representation of time. *arXiv* **2019**, arXiv:1907.05321.
25. Hewage, P.; Behera, A.; Trovati, M.; Pereira, E.; Ghahremani, M.; Palmieri, F.; Liu, Y. Temporal convolutional neural (TCN) network for an effective weather forecasting using time-series data from the local weather station. *Soft Comput.* **2020**, *24*, 16453–16482. [CrossRef]
26. Gan, Z.; Li, C.; Zhou, J.; Tang, G. Temporal convolutional networks interval prediction model for wind speed forecasting. *Electr. Power Syst. Res.* **2021**, *191*, 106865. [CrossRef]
27. Xu, Y.; Hu, C.; Wu, Q.; Li, Z.; Jian, S.; Chen, Y. Application of temporal convolutional network for flood forecasting. *Hydrol. Res.* **2021**, *52*, 1455–1468. [CrossRef]
28. Lin, K.; Chen, H.; Zhou, Y.; Sheng, S.; Luo, Y.; Guo, S.; Xu, C.-Y. Exploring a similarity search-based data-driven framework for multi-step-ahead flood forecasting. *Sci. Total Environ.* **2023**, *891*, 164494. [CrossRef]
29. Zhu, R.; Liao, W.; Wang, Y. Short-term prediction for wind power based on temporal convolutional network. *Energy Rep.* **2020**, *6*, 424–429. [CrossRef]
30. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*. [CrossRef]
31. Castangia, M.; Grajales, L.M.M.; Aliberti, A.; Rossi, C.; Macii, A.; Macii, E.; Patti, E. Transformer neural networks for interpretable flood forecasting. *Environ. Model. Softw.* **2023**, *160*, 105581. [CrossRef]
32. Xu, Z.X.; Li, J.Y.; Liu, C. Long-term trend analysis for major climate variables in the Yellow River basin. *Hydrol. Process. Int. J.* **2007**, *21*, 1935–1948. [CrossRef]
33. Hao, S.; Wang, W.; Ma, Q.; Li, C.; Wen, L.; Tian, J.; Liu, C. Model-based mechanism analysis of "7.20" flash flood disaster in Wangzongdian River basin. *Water* **2023**, *15*, 304. [CrossRef]
34. Wang, G.; Zhang, J.; Li, X.; Bao, Z.; Liu, Y.; Liu, C.; He, R.; Luo, J. Investigating causes of changes in runoff using hydrological simulation approach. *Appl. Water Sci.* **2017**, *7*, 2245–2253. [CrossRef]