

## Article

# A Hydrological Data Prediction Model Based on LSTM with Attention Mechanism

Zhihui Dai <sup>1</sup>, Ming Zhang <sup>2</sup>, Nadia Nedjah <sup>3</sup> , Dong Xu <sup>4</sup> and Feng Ye <sup>1,\*</sup><sup>1</sup> School of Computer and Information, Hohai University, Nanjing 211100, China<sup>2</sup> Water Resources Department of Jiangsu Province, Nanjing 210029, China<sup>3</sup> Department of Electronics Engineering and Telecommunications of the Engineering Faculty, State University of Rio de Janeiro, Rio de Janeiro 20550-013, Brazil<sup>4</sup> College of Water Conservancy & Hydropower Engineering, Hohai University, Nanjing 211100, China

\* Correspondence: yefeng1022@hhu.edu.cn

**Abstract:** With the rapid development of IoT, big data and artificial intelligence, the research and application of data-driven hydrological models are increasing. However, when conducting time series analysis, many prediction models are often directly based on the following assumptions: hydrologic time series are normal, homogeneous, smooth and non-trending, which are not always all true. To address the related issues, a solution for short-term hydrological forecasting is proposed. Firstly, a feature test is conducted to verify whether the hydrological time series are normal, homogeneous, smooth and non-trending; secondly, a sequence-to-sequence (seq2seq)-based short-term water level prediction model (LSTM-seq2seq) is proposed to improve the accuracy of hydrological prediction. The model uses a long short-term memory neural network (LSTM) as an encoding layer to encode the historical flow sequence into a context vector, and another LSTM as a decoding layer to decode the context vector in order to predict the target runoff, by superimposing on the attention mechanism, aiming at improving the prediction accuracy. Using the experimental data regarding the water level of the Chu River, the model is compared to other models based on the analysis of normality, smoothness, homogeneity and trending of different water level data. The results show that the prediction accuracy of the proposed model is greater than that of the data set without these characteristics for the data set with normality, smoothness, homogeneity and trend. Flow data at Runcheng, Wuzhi, Baima Temple, Longmen Town, Dongwan, Lu's and Tongguan are used as input data sets to train and evaluate the model. Metrics RMSE and NSE are used to evaluate the prediction accuracy and convergence speed of the model. The results show that the prediction accuracy of LSTM-seq2seq and LSTM-BP models is higher than other models. Furthermore, the convergence process of the LSTM-seq2seq model is the fastest among the compared models.

**Keywords:** seq2seq model; attention mechanism; LSTM model; hydrological prediction; feature analysis

**Citation:** Dai, Z.; Zhang, M.; Nedjah, N.; Xu, D.; Ye, F. A Hydrological Data Prediction Model Based on LSTM with Attention Mechanism. *Water* **2023**, *15*, 670. <https://doi.org/10.3390/w15040670>

Academic Editor: Padam Jee Omar

Received: 31 December 2022

Revised: 4 February 2023

Accepted: 6 February 2023

Published: 8 February 2023



**Copyright:** © 2023 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) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

The principles of runoff formation and evolutionary patterns are changing under the influence of climate change and human activities. These alterations challenge the applicability of existing prediction models and methods. At the same time, the continuous evolution of hydrological processes in the basin increases the difficulty of future runoff prediction. Therefore, runoff prediction has become an urgent problem to be solved nowadays.

Runoff prediction models can be divided into two categories: process-driven models and data-driven models [1]. Data-driven models focus on data mining and are able to analyze a large amount of hydrological data from multiple perspectives without involving complex physical processes [2]. Existing work mainly uses recurrent neural network for flow prediction. However, there is still room for improvement in prediction accuracy and speed. For this purpose, some research works perform statistical characterization of hydrological sensor data before making predictions.

In this paper, a solution is proposed for short-term hydrological forecasting. First, a feature test is carried out on the hydrological time series to verify whether it has the characteristics of normality, homogeneity, smoothness and non-trendiness. Then, a sequence-to-sequence (seq2seq)-based short-term water level prediction model (LSTM-seq2seq) is proposed and implemented to improve the accuracy of hydrological prediction. The model uses a long short-term memory neural network (LSTM) to encode the historical flow sequence into a context vector, and another LSTM to decode the context vector, thus predicting the target runoff, by superimposing on the attention mechanism to improve the prediction accuracy. The main innovations of the article are as follows:

1. The model uses an LSTM as an encoding layer to encode the historical water flow sequence into a context vector, and another LSTM as a decoding layer to decode the context vector to predict the target runoff.
2. The model explores the usage of an attention mechanism to improve the prediction accuracy.
3. Based on the analysis of normality, smoothness, homogeneity and trend of different runoff data, we demonstrate the positive prediction effect of the model on runoff data with different characteristic properties.

This paper is organized as follows: Section 2 reviews the related current research and contrasts it with the main contributions of this work. Section 3 describes the main structure of the model proposed in this paper. Specifically, it describes the model principle and parameter setting, as well as introduces the model training method and evaluation metrics. Section 4 applies the proposed model to a real case study. Mainly it introduces the experimental environment, used data set and achieved results. Furthermore, it compares the model with three machine learning models and four deep learning models. Section 5 presents the conclusions of this work and points out some directions for future work.

## 2. Related Works

In the field of hydrology, time series analysis is the main tool used by scholars to study hydrological sensor data [3]. Due to the neglect of statistical analysis, many scholars often carry out time series analysis based on the following assumptions: hydrological time series have characteristics such as normality, homogeneity, smoothness and non-trendiness, expecting that the deviation of these assumptions in real scenarios will not affect the analysis results. In short, existing works ignore the fact about whether the time series have these characteristics. The time aforementioned properties of the time series data are not verified nor calculated. In most cases, the hydrological time series data obtained based on sensors can hardly fully satisfy the above characteristics due to the influence of changes in natural elements, such as climate, land cover and disturbances from human activities. It is clear that time series analysis based on incorrect assumptions will lead to unreliable and incorrect analytical conclusions [4].

Normality, smoothness, trendiness and homogeneity [5] are the key characteristics of hydrological time series. The analysis of these characteristics is mainly achieved by feature testing algorithms. For each feature of hydrological time series, there are different testing algorithms, which are widely used in statistical analysis of static data sets. Many scholars have been currently focusing on two aspects of hydrological feature testing algorithms.

First, there is a comparison of different testing algorithms for a particular characteristic. In the normality test, Bin Yang [6] used random simulation to compare the advantages and disadvantages of five algorithms: the Anderson–Darling test (AD test), Pearson chi-square test (Pearson test), Kolmogorov–Smirnov test (KS test), Shapiro–Francia test (SF test) and Cramer-von Mises test (CVM test); the experiments proved that the AD test, CVM test and KS test outperformed the remaining two algorithms regardless of the size of the sample. Li Yizhen et al. [7] used the Pettitt mutation test combined with Mann–Kendall trend test, turn test, inverse ordinal number test and unit root test to analyze the stability of the annual runoff polar series at Bajiadu hydrological station for the past 60 years, and concluded that the turn test and inverse ordinal number test are applicable to the stability of the runoff polar

series, while the unit root test is prone to misclassification. Regarding time series trendiness, Jiang Yao et al. [8] comprehensively compared and analyzed the accuracy and reliability of five commonly used trend detection algorithms for hydrological analysis, including the cumulative distance level method, linear regression method, Mann–Kendall rank order test method, discrete wavelet transform method and empirical modal decomposition method, evaluated their performance on hydrological time series trend detection using real-world data and finally provided the application scenarios for each algorithm. Liu Jia et al. [9] selected more than one hundred stations with continuous observation records in Sichuan province for the last 50 years of temperature data, and integrated six homogeneity testing algorithms, such as the SNHT (standard normal homogeneity test) and von Neumann ratio method to test the homogeneity of the annual average temperature of the province, so as to evaluate the applicability and performance of the six algorithms in this scenario. The results of the above-mentioned literature show that different algorithms have their own advantages and disadvantages in different scenarios and on different data volumes when testing for normality, smoothness, trend, or homogeneity.

Recurrent neural networks (RNNs) are gradually being applied to runoff forecasting research because their structure is particularly suitable for relational simulation of time series data [10]. However, a simple RNN suffers from the gradient disappearance problem when learning longer time series [11], which makes it difficult to transmit information that is far apart. For this reason, different researchers have proposed improved RNN models, such as LSTM [12] and gated recurrent unit (GRU) neural network [13]. Kratzert et al. [14] applied LSTM to 241 watersheds affected by snowmelt and experimentally verified that it could obtain better model performance for rainfall runoff forecasting at the daily scale; Zhaokai Yin et al. [15] established a watershed rainfall runoff model based on LSTM for different forecasting periods to explore the application of LSTM in hydrological forecasting. The results show that the accuracy of LSTM forecasting is high when the forecasting period is about 0–2 days and poor when the forecasting period is 3 days. Zuo et al. [16] propose a medium and long-term runoff forecasting model based on LSTM together with the corresponding discussion and analysis. The results show that SF-VMD-LSTM is robust and efficient in predicting highly non-smooth and non-linear streamflow. LSTM models are able to solve the time series and long-term dependence issues, and are widely used [17,18]. Hu et al. [19] explored LSTM models to forecast Fen River basin runoff; Feng Jun et al. [20] present an LSTM-BP, which is a multi-model combination based on LSTM and BP neural network for hydrological forecasting; Fang et al. [21] applied an LSTM model to soil water content; Zhang et al. [22] discuss the importance of LSTM model in urban flood control and overflow control.

Although the LSTM model solves the problem of long-term dependency, the traditional LSTM model is not well suited for many problems with unequal length sequences, thus giving rise to the seq2seq model. Yuan Meixue et al. [23] propose a two-layer bidirectional seq2seq hybrid model based on wavelet decomposition denoising and LSTM to predict water quality changes; Sun Yingjun et al. [24] constructed a hybrid deep learning interval prediction model based on one-dimensional convolution and long-short-term memory network convolutional-sequence-to-sequence network (CNN-seq2seq) to predict river water level intervals; Liu Xulin et al. [25] propose a deep learning forecasting model based on convolutional neural network and sequence-to-sequence to improve the accuracy of PM<sub>2.5</sub> future one-hour concentration prediction; Zheng Zongsheng et al. [26] present a new typhoon rank prediction model using time-series typhoon satellite cloud images by introducing the attention mechanism and sequence-to-sequence into the model to predict future typhoon images, and then using a convolutional neural network to predict the behavior of the predicted typhoon; Liu Yan et al. [27] propose a short-term water level prediction model based on sequence-to-sequence (seq2seq) by using the Liu Xi River data to build water level prediction models for different predictions. A performance comparison of the obtained model with LSTM models and artificial neural network (ANN) models show that seq2seq models have higher prediction accuracy.

Most researchers focus on the model prediction accuracy, but there is no statistical analysis of the data prior to prediction. For this reason, the prediction accuracy of the model always differs from that obtained using sensor data with different characteristic properties. Based on this observation, a short-term water level prediction model based on sequence-to-sequence (seq2seq) is proposed in this work, in an attempt to improve the quality of the predictions, thus improving its applicability in saving lives and resources.

### 3. Model Construction

This section deals with three issues: the proposed model structure in Section 3.1; data preprocessing and feature analysis in Section 3.2; parameter setting and evaluation metrics in Section 3.3.

#### 3.1. Model Structure

The proposed seq2seq-based short-term water level prediction model uses a LSTM as an encoding layer to encode the historical flow sequence into a context vector, and another LSTM as a decoding layer to decode the context vector to predict the target runoff. Furthermore, the model includes an additional attention mechanism to improve the prediction accuracy. The model prediction process is shown in Figure 1.

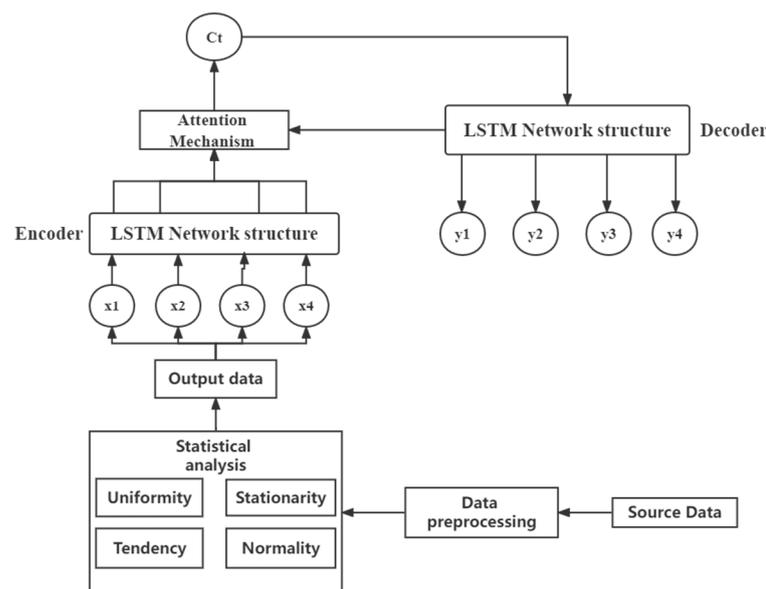


Figure 1. Proposed model structure.

Seq2seq is an encoder–decoder network whose input and output are sequences. An encoder converts a variable length signal sequence into a fixed length vector, while a decoder converts the fixed length vector into a variable length target signal sequence.

In this paper, encoder and decoder are completed with the help of special LSTM-like neural networks. The cyclic neural network accepts input at each location (time point), and at the same time processes information fusion, and may also perform output operation at some location.

The encoder processes each element in the input sequence, compiling the captured information into a vector. After processing the entire input sequence, the encoder sends the context to the decoder, in which, item by item, begins producing the output sequence. Each element in the input data is usually encoded into a dense vector that passes through the LSTM with the hidden output of the last layer as the context vector.

The LSTM model is a special type of RNN. It has been widely used to solve all kinds of problems achieving good results. Specifically, LSTM is designed to avoid the problem of long-term dependency. Its essence is to retain information over a long period of time.

All RNN structures are copied from modules of the exact same structure. In an RNN, this module structure is very simple, such as a single tanh layer.

Compared with the common RNN cyclic neural network, LSTM has three more threshold controllers, input gate, forget gate and output gate [12], as depicted in Figure 2.

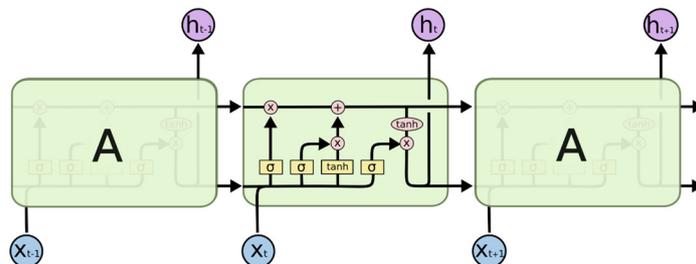


Figure 2. LSTM network structure.

The input gate determines, based on the current input  $x_t$ , the information to be stored in the unit state  $C_{t-1}$ , which is composed of three parts. The first part is a  $\tanh$  layer and is used to create new information vector of the current node  $\tilde{C}_t$ . The second part is a sigmoid layer  $i_t$  equivalent to produce a sieve, used for screening and the current node information  $\tilde{C}_t$ , which should join state  $C_{t-1}$ . The third part adds the information of the current node filtered by the output gate to the state  $C_{t-1}$  to generate a new state  $C_t$ .

$$i_t = \sigma(w_i[h_{t-1}, x_t] + b_i) \tag{1}$$

$$\tilde{C}_t = \tanh(w_c \cdot [h_{t-1}, x_t] + bc) \tag{2}$$

$$C_t = f_t \times C_{t-1} + i_t \times \tilde{C}_t \tag{3}$$

The oblivion gate determines the information to forget from  $C_t$ . The gate reads  $h_{t-1}$  and  $x_t$ , outputs a vector  $f_t$  of the same length as  $C_{t-1}$ , with values between 0 and 1, and multiplies  $f_t$  with  $C_{t-1}$  point by point. In this way, we can see that the 1 in  $f_t$  means “keep” the corresponding element in  $C_{t-1}$ , and the 0 means “discard” the corresponding element in  $C_{t-1}$ . This is equivalent to generating a sieve  $f_t$  of the same size as the state  $C_{t-1}$  to filter the information in  $C_{t-1}$ .

$$f_t = \sigma(w_f \cdot [h_{t-1}, x_t] + b_f) \tag{4}$$

The output gate is used to produce the output of the current node. It consists of two parts: first, the total output of  $C_t$  of the new state is generated by  $C_t$  of the new state through a  $\tanh$  layer; then, the information in the overall output to produce the output of the current node.

$$O_t = \sigma(w_o[h_{t-1}, x_t] + b_o) \tag{5}$$

$$h_t = O_t \times \tanh(C_t) \tag{6}$$

The proposed LSTM-seq2seq model incorporates an attention mechanism. Each time the decoder updates a state, it will look at all the states of the encoder again and informs the encoder which part to focus on more. The main idea of the attention mechanism handles the input as a key–value pair. The information to be output is Query, as shown in Figure 3.

When the attention value is calculated, the similarity between Query and each key is calculated to obtain the associated weights. Then, a SoftMax function is used to normalize these weights. Finally, the weight and the corresponding key–value are weighted and summed to get the final attention value. Figure 4 illustrates the aforementioned process to compute the attention value. In Figure 4, the \* represents the process of weighted summation.

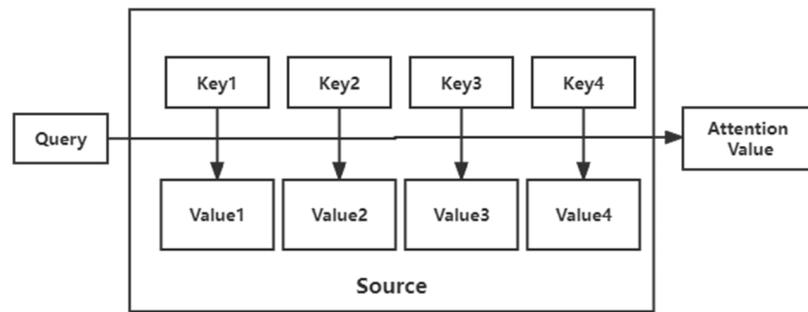


Figure 3. The structure of attention mechanism.

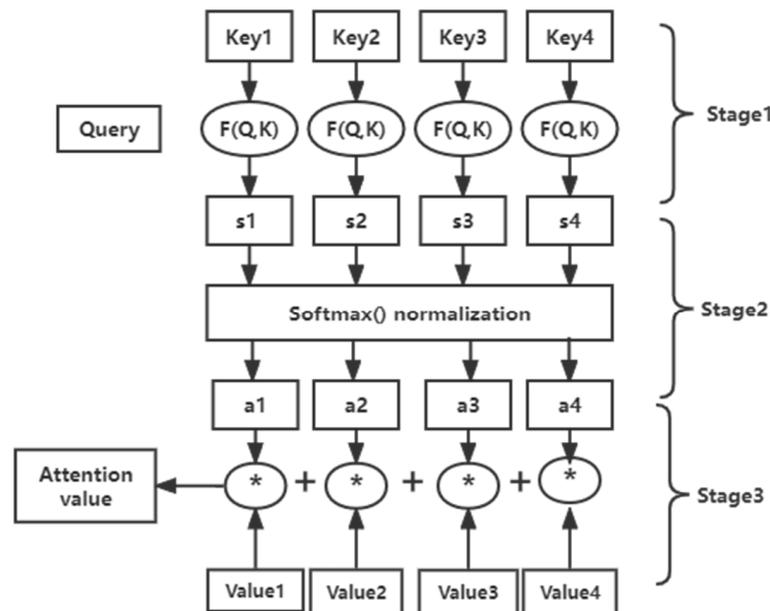


Figure 4. Illustration of the process used to compute attention values.

### 3.2. Data Preprocessing and Feature Analysis

In the sequel, we first give, in Section 3.2.1, details about the applied data preprocessing, then we explain the feature analysis process.

#### 3.2.1. Data Preprocessing

In the field of hydrology, sensors are used to observe various hydrological indicators instead of carrying out this observation manually. The monitored data are saved and sent to a background process for further processing. Different from industrial sensors placed in confined spaces, hydrology sensors are usually placed in the water environment with high humidity and frequent weather changes. Direct contact with fresh water or seawater accelerates the corrosion and aging of the sensors, which shortens the working cycle of the sensors and makes it difficult to maintain in a stable working state for a long time. Therefore, in the field of hydrology, the data quality read from a sensor needs to be handled with care. Thus, the preprocessing of hydrological sensor data has become very important.

Traditional data preprocessing is carried out after all the data arrives, so it is not suitable for unbounded sensor data. In view of this, this paper implements the data preprocessing method based on time window, which uses the rolling time window to deal with redundancy, missing data, duplication and outliers that usually hinder the quality of raw sensor data. Figure 5 describes the overall preprocessing process of hydrologic sensor data, as used in this paper.

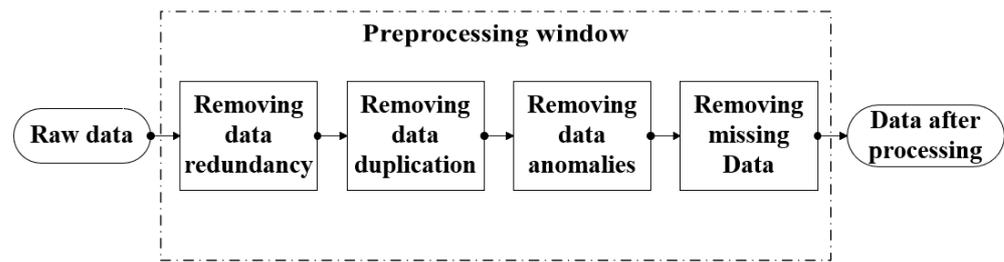


Figure 5. Hydrologic sensor data preprocessing process.

In this paper, the rolling window mechanism is adopted to establish a preprocessing window for  $L_i$ , in which the preprocessing of the hydrological time series is completed. The main steps of this process are as follows:

Step 1: Set the window *Size* according to the sampling frequency of the sensor.

Step 2: Open the preprocessing window and determine the start time  $T_{start}$  and end time  $T_{end}$  of the preprocessing window according to the time stamp  $t$  of the first arriving data in the stream data.

$$T_{start} = t - (t + Size) \% Size \tag{7}$$

Step 3: If the time stamp of the subsequent data arrives at  $t \geq T_{end}$ , the rolling window with the time range of  $[T_{start} + Size, T_{end} + Size)$  is opened synchronously. In this case, all the data of the previous window are considered to be completed.

Step 4: In the previous window, handle data redundancy, duplication, exceptions, and missing, output the processed data, and close the window.

Step 5: Repeat steps 2 to 4 as long as data continues to arrive.

### 3.2.2. Analysis of Features

In hydrology-related studies, the analysis of time series is mostly based on a set of basic assumptions, declaring the hydrological time series as normal, uniform, stable, with no trend or change, non-periodic and having no persistence [28]. In practice, most natural time series, including hydrological, climatic and environmental ones, do not usually satisfy all the characteristic aforementioned assumptions. Therefore, for all the hydrological research involving the use of hydrological time series data, the feature testing algorithm must be used in advance to conduct a preliminary analysis of each feature of the manipulated data. Otherwise, incorrect and/or unreliable results may be obtained. This paper analyzes the normality, uniformity, stationarity and tendency of hydrological sensor data, which allows us to deal with reliable data to reach the accurate prediction of runoff data with different characteristics. The feature testing algorithm adopted in this paper is shown in Table 1.

Table 1. Feature verification algorithm [5].

Stability	Students T test Simple T test Mann–Whitney test
Normality	Kolmogorov–Smirnov test Jarque Bera test Geary’s test
Tendency	Kendall’s Rank Correlation test Mann–Kendall test SROC test
Uniformity	Bayesian test Dunnnett test Von Neumann test

### 3.3. Model Setting and Evaluation Index

In the sequel, we present the model setting in Section 3.3.1 followed by Section 3.3.2, where we describe the used evaluation metrics.

#### 3.3.1. Model Setting

In this paper, two LSTM neural network structures are used as the encoder and decoder as part of the seq2seq proposed model to predict runoff data. The encoder codifies the length sequence data into an intermediate vector of fixed length, while the decoder transforms the intermediate vector of fixed length as the predictive output, realizing the purpose that the input sequence of any length to be mapped to the output sequence of any length. The length of the historical time series data is denoted as  $a$  while the length of the predictive output time series is denoted as  $b$ . The input vector is denoted as  $x = (x_1, x_2, x_3, \dots, x_n)$  and the output vector is denoted as  $y = (y_1, y_2, y_3, \dots, y_n)$ , where  $n$  is the total number of samples. Note that  $x = (x_1, x_2, x_3, \dots, x_n)$  represents the historical input runoff,  $y = (y_1, y_2, y_3, \dots, y_n)$  indicates the predicted runoff. The encoder consists of LSTM that reads the input sequence in chronological order. The hidden layer state  $h_t$  at each moment is determined by the input data  $x_t$  at the current moment combined with the hidden layer state  $h_{t-1}$  and cell state  $C_{t-1}$  at the previous moment. The encoder is updated every  $n$  time steps, and through the attention mechanism used during the training process, the input sequence is encoded as the hidden state  $h_n$  of the final time step. Since the recurrent neural network operates as a long-term memory, and given the attention mechanism, the output sequence can be updated more accurately each time. Thus,  $h_n$  contains the complete information of the input sequence.

The decoder also consists of an LSTM and accepts the final state of the encoder  $h_n$  as its initial input value. The hidden layer state  $h_t$  of the decoder at every moment is updated using the hidden layer state  $h_{t-1}$  and cell state  $C_{t-1}$  of the previous moment through the input  $h_n$  of the current moment. The decoder-specific operation used to update  $h_{t-1}$  is the same as that of the encoder when updating  $h_t$ . After decoding step by step, the final output sequence is obtained.

In this paper, the input length of the model is set as 24 h, and the samples in the historical data set are used to train the model. The mean square difference is used as the loss function during the training process, which is implemented by the optimized ADAM algorithm with a give a citation for ADAM algorithm learning rate of 0.001. The maximum number of training epochs is set to 100.

#### 3.3.2. Model Evaluation Metrics

The mean square deviation (MSE) [29], root mean square error (RMSE) [30] and Nash efficiency coefficient (NSE) [31] are selected as the evaluation metrics of the model training and prediction accuracy.

$$MSE = \frac{1}{N} \sum_{t=1}^N (y_t - \bar{y}_t)^2 \quad (8)$$

$$RMSE = \sqrt{\frac{1}{N} \sum_{t=1}^N (y_t - \bar{y}_t)^2} \quad (9)$$

$$NSE = 1 - \frac{\sum_{t=1}^N (Q_0^t - Q_m^t)^2}{\sum_{t=1}^N (Q_0^t - \bar{Q}_0)^2} \quad (10)$$

among them,  $N$  represents the data length,  $y_t$  the actual data,  $\bar{y}_t$  the forecast data. Variable  $Q_0^t$  represents the time  $t$  observations while  $Q_m^t$  represents the value of  $\bar{Q}_0$ , which is the overall average of the observations.

It is worth noting that NSE varies from  $-\infty$  to 1. The closer it is to 1, the more perfect the fitting and the higher the model credibility. Metrics MSE and RMSE are used to measure the deviation between simulated and measured values. Their value may vary from 0 to  $+\infty$ . The closer they are to 0, the better the overall simulation effect of the mode.

#### 4. Experiment and Result Analysis

In this section, we analyze the performance of the proposed model. For this, first, in Section 4.1, we give define the computational system setting used during the evaluation as well as the data set used. Then, in Section 4.2, we contrast experiment of different characteristics and properties. After that, in Sections 4.3 and 4.4, we compare the prediction accuracy and the convergence rate of the proposed model to that obtained when using existing different models. Subsequently, in Section 4.5, we evaluate the robustness of the proposed model.

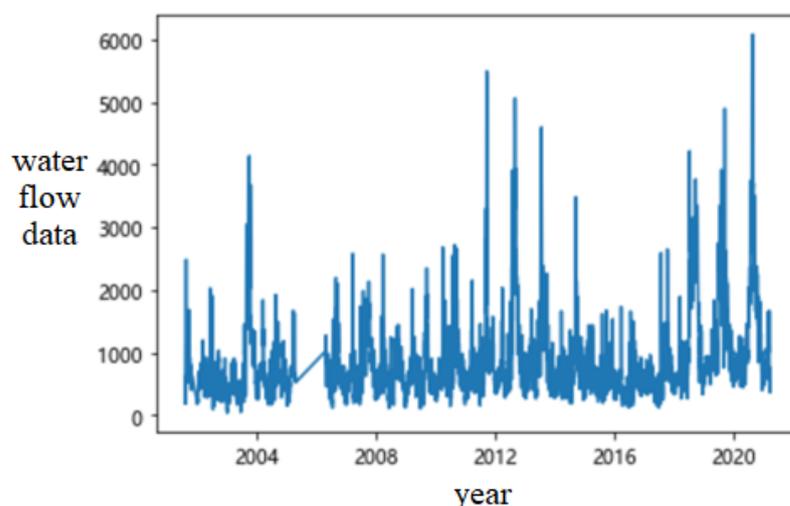
##### 4.1. Experimental Environment and Data Set

The experimental environment of this paper is shown in Table 2.

**Table 2.** Experimental environment.

Processor	11th Gen Intel(R) Core(TM) i5-1155G7 @ 2.50GHz 2.50 GHz
Operating system	Windows10
Deep learning Framework	Pytorch
Integrated development environment	Pycharm2021
Programming language	Python3.6

The data set utilized for model training is the Tongguan water flow data registered during the period of time between 9 August 2001 and 31 March 2021, with a total of 6719 records. As shown in Figure 6, although the flow at Tongguan is random, it is still periodic to a certain extent. We used the data from 9 August 2001 to 9 July 2016 as the training set, and the data from 10 July 2016 to 31 March 2021 as the test set. The flow data from the main hydrological observatories in the Qinhe River Basin, including Runcheng and Wuzhi, and in the Yiluo River basin, including Baima Si, Longmen Town, Dongwan and Lu Shi, are also used.



**Figure 6.** Water flow data at Tongguan.

Here, we use the Chuhe River water level data from 2015 to 2017 as the data set to predict the effect of LSTM-seq2seq model on the water level data with different characteristics. This data set contains more than 18 million data records read from 69 different sensors. Note that each sensor is numbered in the form of 8-digit, such as “12,910,280”. The sampling time interval of this data set is 5 min, and its format is shown in Table 3. Table 4 presents the data sets used in the experiment and the purpose of each data set.

**Table 3.** Data format of Chuhe water level sensor.

Time Stamp	Sensor Number	Water Level Value/m
1 January 2015 00:00:00	12,911,060	60.510
11 March 2015 13:00:00	12,910,420	55.576
22 April 2015 12:55:00	62,916,750	5.580
26 June 2015 14:45:00	60,403,200	7.210

**Table 4.** Experimental data sets and uses.

Data Set	Purpose of Experiment
Chuhe River water level data	The impact of the statistical analysis on the prediction effect of the model on hydrological data with different characteristics.
Runcheng, Wuzhi, White Horse Temple, Longmen town, East Bay and Lu’s water flow data	The prediction accuracy of LSTM-seq2seq model comparison to 3 machine learning models and 4 deep learning models.
Water flow data at Tongguan	The algorithm robustness and convergence speed comparison.

4.2. Impact of Statistical Characteristics

In this comparison, the models LSTM-seq2seq, LSTM-BP [20], LSTM and BP are trained using data sets with different characteristics, and NSE is used to evaluate the performance of these models. Table 5 shows the four models’ NSE values with respect to the data provided by the four sensors of Table 3. Note that we use the data read from a given sensor to analyze only one of the statistical properties, as shown in Table 5. The NSE values of the four sensors regarding stationarity are all greater than 0.65, indicating that the model has a good prediction effect. The NSE values of the LSTM-BP and the LSTM-seq2seq models are much larger than those of the LSTM and the BP models. Moreover, the NSE value of the LSTM-seq2seq model is 0.02 larger than that of the LSTM-BP model. This indicates that the prediction effect of LSTM-seq2seq and LSTM-BP models is much greater than that of the LSTM and BP models. Moreover, the prediction effect of the LSTM-seq2seq model is better than that of LSTM-BP model. The data sets with normality of site 12,910,420 are all greater than 0.75, indicating that the four models have good prediction effect on the data set with trend.

**Table 5.** NSE values of different models in different feature properties.

Model	NSE			
	12,911,060	12,910,420	60,403,200	62,916,750
	Stationarity	Normality	Tendency	Uniformity
LSTM-BP	0.81	0.79	0.75	0.78
LSTM-seq2seq	0.83	0.81	0.79	0.79
LSTM	0.71	0.77	0.69	0.76
BP	0.69	0.76	0.68	0.74

In the trend data set of site 60,403,200, the prediction effect of the LSTM-seq2seq model and LSTM-BP model is better than that of LSTM model and BP model, and the NSE value of LSTM-seq2seq model is 0.04 larger than that of LSTM-BP model, indicating that the prediction effect of the LSTM-seq2seq model is better than that of the LSTM-BP model.

The NSE values are all greater than 0.7 in the data set of site 62,916,750. These results show that the four models have good prediction effect on the homogeneous data set, and the difference is not significant.

The NSE values of the LSTM-seq2seq and the LSTM-BP models are larger than those of the LSTM and BP models regarding stationarity and tendency. This indicates that LSTM-seq2seq and LSTM-BP models have a better prediction effect than LSTM and BP models on the data set with stationarity and trend. With respect to normality and uniformity, the NSE

values of the LSTM-seq2seq and the LSTM-BP models are slightly larger than those of the LSTM and the BP models, indicating that the prediction effects of the four models have little difference in the data sets with normality and uniformity.

Overall, the NSE value of the LSTM-seq2seq model is slightly higher than that of the LSTM-BP model, indicating that the prediction effect of the LSTM-seq2seq model is slightly better than that of the LSTM-BP model and significantly better than the LSTM and the BP models on the data sets with different characteristics.

#### 4.3. Prediction Accuracy Comparison

In order to evaluate the prediction accuracy of the proposed LSTM-seq2seq model compared with existing ones, three machine learning models and four deep learning models are selected. (1) Linear regression (LR) [32], which is a traditional data analysis method that uses mathematical knowledge to analyze data, finds out correlated problem variables and determines the correlation between these variables. This model is selected by many scholars in their study of similar problems. By determining the relationship between multiple variables, linear regression establishes a correlation equation and loss function, and attempts to find the relevant parameters of the equation when the loss function reaches the optimal value. (2) Autoregressive integrated moving average (ARIMA) [33] is also a traditional statistical analysis method. Compared with LR, ARIMA has a better processing capability for time series. The ARIMA model requires three parameters, which are the number of autoregressive terms, the sliding mean terms and the difference order. Moreover, ARIMA can transform non-stationary series into stationary ones. (3) Random forest [34] (RF) is a machine learning algorithm with stronger ability to process time series. It can be used to deal with two common problems, classification and regression. RF explores multiple decision trees, which make prediction results independently. All prediction results are counted, and the average value is considered the final prediction result obtained by RF through regression calculation. (4) DeepAR [35] is based on an RNN structure, which can effectively perform probability prediction. It is one of the more widely used deep learning prediction models. (5) BP neural networks [36] consist of supervised learning and are a kind of multiple feedforward neural network.

RMSE and NSE are used to evaluate and analyze the compared models regarding the accuracy of water flow prediction for Tongguan River. Figure 7 shows the NSE values of models on different data sets. As the trend change of runoff is not a simple linear relationship, the overall performance of LR is relatively poor. As can be seen in Figure 7, the deep learning model is generally superior to machine learning model when processing time series data. The NSE values of the LSTM-seq2seq model are all greater than 0.72, indicating that its overall prediction accuracy is relatively high and the prediction reliability is also high.

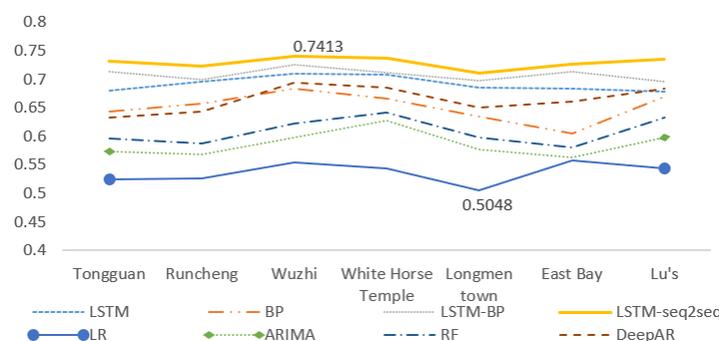


Figure 7. NSE values of compared models considering different data sets.

Figure 8 shows the RMSE values for the compared models on different data sets. The highest value achieved by the LSTM-seq2seq model is 0.98, indicating that the overall simulation effect of the model is good. Considering the metric values of Table 5 and the

comparison in Figure 7, it is possible to conclude that for the data sets with normality, uniformity, stationarity and trendiness, the prediction accuracy of the proposed model is greater than that of the data sets that do not show these statistical characteristics. This indicates that it is very important to analyze the characteristics of the data sets before performing the prediction on hydrological data.

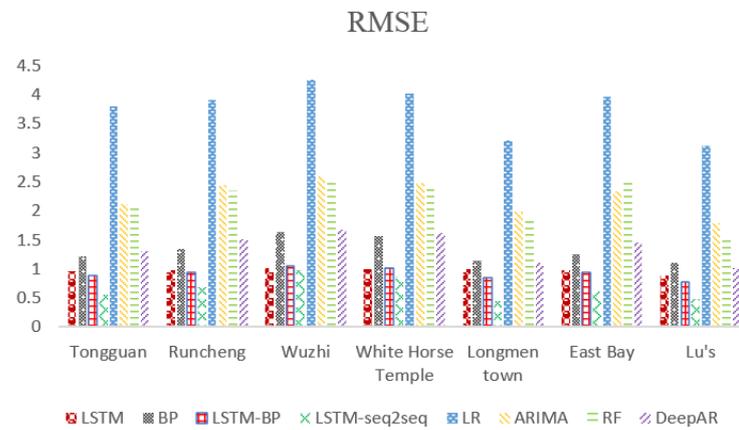


Figure 8. RMSE values for compared models considering different data sets.

Figure 9 shows the comparison between the predicted values, as obtained by the proposed LSTM-seq2seq model, and those effectively observed at Tongguan River. The “real” data shown in Figure 9 are the test data set. It is easy to note that LSTM-seq2seq model has a good fitting effect on the overall prediction.

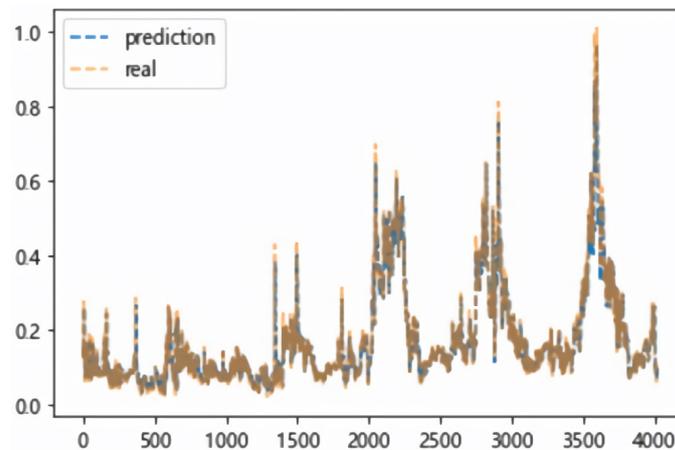


Figure 9. Comparison of the predicted and observed values for Tongguan River.

4.4. Convergence Comparison

Figure 10 allows us to compare the convergence rate of LSTM-seq2seq, LSTM-BP, LSTM and BP models when using the data set obtained at the Tongguan River. It is clear that the LSTM-seq2seq model achieves a faster convergence than the other three models. The RMSE value of LSTM-seq2seq model is smaller than that of the other three models. So, it is safe to conclude that the proposed model has high credibility.

4.5. Algorithm Robustness

The robustness test of an algorithm consists of verifying the returned results when adding partial noise to the data and/or introducing partial outliers. Generally speaking, the algorithm with good robustness can still maintain an acceptable prediction accuracy even when the input data are noisy and can also ensure normal operation when there are partial outliers. The robustness test of a proposed algorithm is crucial to deciding whether

the algorithm can be applied in a real-world application. This section designs a robustness test scheme for the algorithm, which mainly adds noise to the input data and evaluates its performance in this situation.

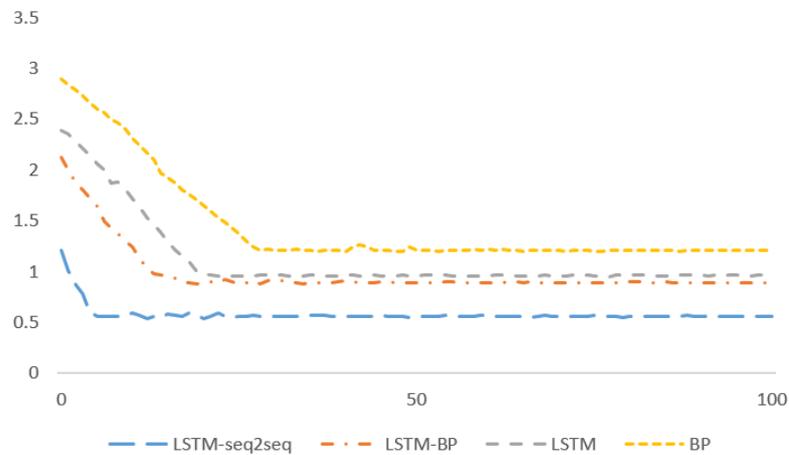


Figure 10. Convergence rates as achieved by the four compared models.

Figure 11 shows the NSE values of different models with no noise, 5% and 10% Gaussian noise. It can be seen that the LSTM-seq2seq model is more stable than other models, and the prediction accuracy of the algorithm is still guaranteed when noise is added.

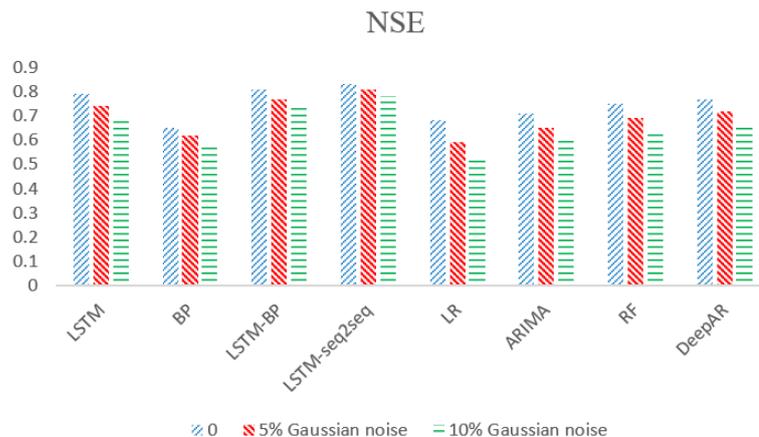


Figure 11. Robustness analysis of different models.

### 5. Conclusions

This work proposes an LSTM-seq2seq model for water flow prediction, which uses two LSTM to process the input and output sequences of data, augmented with an attention mechanism to improve the prediction accuracy. The prediction impact of the proposed model on different water level data sets that present statistical features, such as normality, stationarity, uniformity and tendency is analyzed. We show that the proposed model performs better than the compared models in all cases.

Flow data at Runcheng, Wuzhi, Baima Temple, Longmen Town, Dongwan, Lu’s and Tongguan are used as input data sets to train and evaluate the model. The proposed LSTM-seq2seq model is compared with three machine learning models and four deep learning models. Metrics RMSE and NSE are used to evaluate the prediction accuracy and convergence speed of the model. The results show that the prediction accuracy of LSTM-seq2seq and LSTM-BP models is higher than other models. Furthermore, the convergence process of the LSTM-seq2seq model is the fastest among the compared models. Chuhe River water level data are used to train and evaluate in order to compare the prediction accuracy achieved by the proposed models to that of existing models. Models LSTM-seq2seq,

LSTM-BP, LSTM and BP are compared regarding the correctness of their prediction on the data sets with different characteristics. The results show that the predictions provided by LSTM-seq2seq and LSTM-BP models are greater than those provided by LSTM and BP models on the data set with stationarity and trend. The prediction results of the four models are similar for the data sets with normality and uniformity. Nonetheless, for the data sets with normality, uniformity, stationarity and trend, the prediction accuracy of the proposed model is greater than that obtained when the data set shows none of these statistical characteristics. In terms of algorithm robustness, the LSTM-seq2seq model is more stable than the compared models with noisy input data.

The LSTM-seq2seq model proposed in this work provides a reference for hydrological prediction. Especially, it offers an effective prediction scheme for the data collected by sensors thanks to consideration of the possible influence of specific statistical features inherent to hydrological data.

In future work, more attention will be given to the influence of hydrological characteristics on hydrological prediction, so as to provide more prediction schemes for complex hydrological data.

**Author Contributions:** Methodology, Z.D.; resources, D.X.; data curation, M.Z.; writing—original draft preparation, Z.D.; writing—review, F.Y. and N.N. All authors have read and agreed to the published version of the manuscript.

**Funding:** National Key R&D Program of China (2019YFE0109900); Water Science and Technology Project of Jiangsu Province (2022003,2022057); National Natural Science Foundation of China (52179076 and 51979186).

**Data Availability Statement:** The data supporting the findings of this study are not publicly available due to privacy.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Yuan, X.; Chen, C.; Lei, X.; Yuan, Y.; Muhammad Adnan, R. Monthly runoff forecasting based on LSTM-ALO model. *Stoch. Environ. Res. Risk Assess.* **2018**, *32*, 2199–2212. [\[CrossRef\]](#)
2. Adamowski, J.; Sun, K. Development of a coupled wavelet transform and neural network method for flow forecasting of non-perennial rivers in semi-arid watersheds. *J. Hydrol.* **2010**, *390*, 85–91. [\[CrossRef\]](#)
3. Sang, Y.F.; Wang, Z.G.; Liu, C.M. Research progress of hydrological time series analysis methods. *Prog. Geogr. Sci.* **2013**, *32*, 20–30.
4. Xiong, L.H.; Jiang, C.; Du, T.; Guo, S.L.; Xu, Z.Y. A review of studies on incongruent hydrological frequency analysis in changing environments. *J. Water Resour. Res.* **2015**, *4*, 310. [\[CrossRef\]](#)
5. Machiwal, D.; Jha, M.K. *Hydrologic Time Series Analysis: Theory and Practice*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2012.
6. Yang, B. Comparison of several methods of normality test. *Stat. Decis. Mak.* **2015**, *4*, 72–74.
7. Li, Y.Z.; Yue, C.F. Stability test of time series of runoff extremum in Jingou River Basin. *Hydropower Energy Sci.* **2019**, *37*, 21–24.
8. Jiang, Y.; Xu, Z.X.; Wang, J. Performance comparison of five trend detection methods based on annual runoff series. *J. Hydraul. Eng.* **2020**, *51*, 845–857.
9. Liu, J.; Ma, Z.F.; Fan, G.Z.; You, Y. A comparative study of various uniformity test methods. *Weather* **2012**, *38*, 1121–1128.
10. Hsu, K.; Gupta, H.V.; Sorooshian, S. Application of a recurrent neural network to rainfall-runoff modeling. In *Aesthetics in the Constructed Environment*; ASCE: Reston, VA, USA, 1997.
11. Hochreiter, S. The vanishing gradient problem during learning recurrent neural nets and problem solutions. *Int. J. Uncertain. Fuzziness Knowl. Based Syst.* **1998**, *6*, 107–116. [\[CrossRef\]](#)
12. Graves, A. Long short-term memory. In *Supervised Sequence Labelling with Recurrent Neural Networks*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 37–45.
13. Chung, J.; Gulcehre, C.; Cho, K.; Bengio, Y. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv Prepr.* **2014**, arXiv:1412.3555.
14. Kratzert, F.; Klotz, D.; Brenner, C.; Schulz, K.; Herrnegger, M. Rainfall-runoff modelling using long short-term memory (LSTM) networks. *Hydrol. Earth Syst. Sci.* **2018**, *22*, 6005–6022. [\[CrossRef\]](#)
15. Yin, Z.K.; Liao, W.H.; Wang, R.J.; Lei, X.H. Rainfall-runoff modelling and forecasting based on long short-term memory (LSTM). *South North Water Transf. Water Sci. Technol.* **2019**, *17*, 1–9.
16. Zuo, G.; Luo, J.; Wang, N.; Lian, Y.; He, X. Decomposition ensemble model based on variational mode decomposition and long short-term memory for streamflow forecasting. *J. Hydrol.* **2020**, *585*, 124776. [\[CrossRef\]](#)

17. Tian, Y.; Xu, Y.P.; Yang, Z.; Wang, G.; Zhu, Q. Integration of a parsimonious hydrological model with recurrent neural networks for improved streamflow forecasting. *Water* **2018**, *10*, 1655. [[CrossRef](#)]
18. Zhang, J.; Zhu, Y.; Zhang, X.; Ye, M.; Yang, J. Developing a Long Short-Term Memory (LSTM) based model for predicting water table depth in agricultural areas. *J. Hydrol.* **2018**, *561*, 918–929. [[CrossRef](#)]
19. Hu, C.; Wu, Q.; Li, H.; Jian, S.; Li, N.; Lou, Z. Deep learning with a long short-term memory networks approach for rainfall-runoff simulation. *Water* **2018**, *10*, 1543. [[CrossRef](#)]
20. Feng, J.; Pan, F. A LSTM-BP multi-model combined hydrological forecasting method. *Comput. Mod.* **2018**, *7*, 82–85+92.
21. Fang, K.; Shen, C.; Kifer, D.; Yang, X. Prolongation of SMAP to spatiotemporally seamless coverage of continental US using a deep learning neural network. *Geophys. Res. Lett.* **2017**, *44*, 11–030. [[CrossRef](#)]
22. Zhang, D.; Lindholm, G.; Ratnaweera, H. Use long short-term memory to enhance Internet of Things for combined sewer overflow monitoring. *J. Hydrol.* **2018**, *556*, 409–418. [[CrossRef](#)]
23. Yuan, M.X.; Wei, S.K.; Sun, M.; Zhao, J.D. Seq2Seq Water quality prediction model based on wavelet denoising and LSTM. *Comput. Bus. Syst.* **2022**, *31*, 38–47.
24. Sun, Y.J.; Tang, W.H.; Wang, C.; Li, Y.D. Prediction method of river level interval based on CNN-Seq2seq. *J. Zhejiang Univ. Technol.* **2022**, *50*, 381–392+405.
25. Liu, X.L.; Zhao, W.F.; Tang, W. The CNN-Seq2seq PM2.5 one-hour concentration prediction model was applied. *Small Microcomput. Syst.* **2020**, *41*, 1000–1006.
26. Zheng, Z.S.; Liu, M.; Hu, C.Y.; Fu, Z.P.; Lu, P.; Jiang, X.Y. Typhoon classification prediction based on Seq2Seq and Attention time series satellite cloud images. *Remote Sens. Inf.* **2020**, *35*, 16–22.
27. Liu, Y.; Zhang, T.; Kang, A.Q.; Li, J.Z.; Lei, X.H. Short-term water level prediction by Seq2Seq model. *Adv. Water Conserv. Hydropower Technol.* **2022**, *42*, 57–63.
28. Adeloje, A.J.; Montaseri, M. Preliminary streamflow data analyses prior to water resources planning study/analyses préliminaires des données de débit en vue d’une étude de planification des ressources en eau. *Hydrol. Sci. J.* **2002**, *47*, 679–692. [[CrossRef](#)]
29. Aklilu, E.G.; Adem, A.; Kasirajan, R.; Ahmed, Y. Artificial neural network and response surface methodology for modeling and optimization of activation of lactoperoxidase system. *S. Afr. J. Chem. Eng.* **2021**, *37*, 12–22. [[CrossRef](#)]
30. Kim, T.; Yang, T.; Gao, S.; Zhang, L.; Ding, Z.; Wen, X.; Gourley, J.J.; Hong, Y. Can artificial intelligence and data-driven machine learning models match or even replace process-driven hydrologic models for streamflow simulation?: A case study of four watersheds with different hydro-climatic regions across the CONUS. *J. Hydrol.* **2021**, *598*, 126423. [[CrossRef](#)]
31. Nash, J.E.; Sutcliffe, J.V. River flow forecasting through conceptual models part I—A discussion of principles. *J. Hydrol.* **1970**, *10*, 282–290. [[CrossRef](#)]
32. Wang, H.W.; Meng, J. Multiple linear regression predictive modeling method. *J. Beijing Univ. Aeronaut. Astronaut.* **2007**, *4*, 500–504.
33. Zhang, S.Q.; Wang, W.; Wang, J. Application of ARIMA model in urban annual electricity consumption forecast. *Power Demand Side Manag.* **2010**, *12*, 31–34.
34. Dong, S.S.; Huang, Z.X. Brief analysis of Random predict theory. *Integr. Technol.* **2013**, *2*, 1–7.
35. He, X.H.; Duan, Q.C.; Yan, L. Probabilistic prediction of short-term wind speed based on DeepAR. *J. Railw.* **2022**.
36. Wang, D.M.; Wang, L.; Zhang, G.M. Short-term wind speed prediction model based on genetic BP neural network. *J. Zhejiang Univ.* **2012**, *46*, 837–841+904.

**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.