Daily runoff forecasting based on data-augmented neural network model

Xiao-ying Bi IWA, Bo Li, Wen-long Lu and Xin-zhi Zhou IWA

ABSTRACT

Accurate daily runoff prediction plays an important role in the management and utilization of water resources. In order to improve the accuracy of prediction, this paper proposes a deep neural network (CAGANet) composed of a convolutional layer, an attention mechanism, a gated recurrent unit (GRU) neural network, and an autoregressive (AR) model. Given that the daily runoff sequence is abrupt and unstable, it is difficult for a single model and combined model to obtain high-precision daily runoff predictions directly. Therefore, this paper uses a linear interpolation method to enhance the stability of hydrological data and apply the augmented data to the CAGANet model, the support vector machine (SVM) model, the long short-term memory (LSTM) neural network and the attention-mechanism-based LSTM model (AM-LSTM). The comparison results show that among the four models based on data augmentation, the CAGANet model proposed in this paper has the best prediction accuracy. Its Nash–Sutcliffe efficiency can reach 0.993. Therefore, the CAGANet model based on data augmentation is a feasible daily runoff forecasting scheme.

Key words | attention mechanism, combined neural network, daily flow prediction, data augmentation, GRU

Xiao-ying Bi IWA
Xin-zhi Zhou IWA (corresponding author)
College of Electronics and Information
    Engineering,
Sichuan University,
Chengdu, Sichuan,
China
E-mail: xz.zhou@scu.edu.cn

Bo Li
Joint Laboratory of Water Cconservancy
    Information,
Sichuan University,
Chengdu, Sichuan,
China

Wen-long Lu
Wan jiang Gang-li technology Joint Stock Company,
Chengdu, Sichuan,
China

HIGHLIGHTS

- Our research proposes a combined neural network model, which shows good prediction performance and high robustness in the prediction of data with strong variability such as daily runoff.
- Our research proposes a simple but useful data processing method, which can effectively improve the prediction performance.
- Compared with other models that predict daily runoff, the model and method proposed in our study have better prediction performance and can provide model basis for daily runoff prediction in other watersheds.

INTRODUCTION

Accurate and timely river flow prediction plays an important role in water resources planning and management, risk assessment, and flood prevention (Wang *et al.* 2009). A large number of predictive models have been studied over the decades and can be divided into two categories: models based on physical processes and data-driven models. Process-based models have the advantage of describing complex hydrological processes through functions that can provide observations of physical processes, but these models are subject to many empirical assumptions and require a large amount of data (Mehr *et al.* 2013). The data-driven model is a model based on historical observations. It is an end-to-end model that directly explores the relationship between various historical hydrological

features and targets without detailed physical process explanations.

Since the 1970s, statistical data-driven methods such as multiple linear regression and autoregressive moving average (ARMA) have been applied to hydrometeorological forecasting. Studies have shown that when the time series are linear or nearly linear, the statistical model can produce satisfactory predictions, but it cannot capture the nonlinear and non-stationary modes hidden in the time series (Toth *et al.* 2000). However, the hydrometeorological time series are complex and non-stationary (Nourani *et al.* 2014). In recent years, machine learning technology has attracted extensive attention because of its strong learning ability and adaptability to modeling the complex nonlinear process. Therefore, various machine learning models, such as artificial neural networks (ANN), adaptive neuro-fuzzy inference systems (ANFIS), genetic programming (GP), and support vector machines (SVM), have been developed and found to be able to produce satisfactory predictions of hydrological processes (Sivapragasam *et al.* 2001; Moghaddamnia *et al.* 2009; Citakoglu *et al.* 2014; Taormina & Chau 2015).

Despite the success of the ANN model, these prediction methods do not take into account the trend, periodicity, and stochastic characteristics of actual hydrological data (Bai *et al.* 2014). Recurrent neural networks (RNN) are a kind of neural network with memory, specifically designed to understand the dynamics of time. By adding feedback connections to the structure to memorize previous information, RNN architecture can handle tasks involving time series (Shen 2018). Therefore, RNN are suitable for modeling complex hydrological time series predictions (Kumar *et al.* 2004; Coulibaly & Baldwin 2005). However, it is difficult for traditional RNN to learn the long-term dependence of time series. When back-propagating errors flow through multiple time steps, gradient disappearance and gradient explosion problems will occur (Bengio *et al.* 1994). In order to solve this problem, two variants of RNN, long short-term memory networks (LSTM) and gated recurrent networks (GRU), have been developed. GRU simplifies the structure of LSTM neural network and improves the learning speed. Some researchers (Kratzert *et al.* 2018) have studied the potential of LSTM in simulating runoff from multiple river basins through meteorological observations, and found that LSTM has better prediction and generalization capabilities. Multi-layer

perceptron (MLP), wavelet neural network (WNN), LSTM network, and GRU network were used to predict the groundwater level in agricultural areas (Zhang *et al.* 2018), and the results show that LSTM and GRU prediction are better.

Proper data preprocessing can improve the performance of data-driven models (Wu *et al.* 2009). A large number of studies have shown that, compared with the corresponding single model, the combination of data preprocessing technology and machine learning has higher accuracy in hydrological prediction (Nourani *et al.* 2014; Ravansalar *et al.* 2017). In the data preprocessing technology, a wavelet transform (WT)-based processing method can decompose sequence features, which makes WT very popular in hydrological prediction models (Quilty *et al.* 2019). Similar to WT, convolution neural networks (CNN) use discrete convolution operations based on filter banks to detect and extract invariant structures and hidden features in the data, so it is widely used in various fields. In hydrological prediction, CNN also gradually became popular, and researchers have combined CNN and RNN neural networks for hydrological forecasting (Miao *et al.* 2019).

Since not all factors in flood forecasting are informative, and irrelevant factors often produce a lot of noise, we need to pay more attention to these useful information factors. However, the original GRU did not have a strong attention ability. To solve this problem, the attention mechanism (AM) was introduced. With the widespread application of AM, researchers began to apply AM to time series prediction (Zhao *et al.* 2016). The attention mechanism is combined with the LSTM network for flood forecasting (Wu *et al.* 2018), and the results show that compared with the traditional LSTM and SVM models, its prediction accuracy is higher.

The daily rainfall and daily runoff time series are data series collected in units of days. During the rainy season, rainfall surges and sudden heavy rains are frequent and sharp. This will lead to the decrease of the smoothness of the data collected, and loss of intermediate information in the rainfall–runoff process. This will add difficulty to daily runoff time series prediction, and is a common problem in hydrological time series. In order to solve this problem, this paper adopts the method of linear interpolation (LI) to improve the relative stability of daily collected data. The linear interpolation method is to insert the linear mean value of two adjacent data into the data, aiming to increase the stability of data

without changing characteristics such as trend and period. In the study, the liner interpolation is performed twice to augment the data, denoted as First-LI and Second-LI in this paper.

The aim of this study is to combine linear interpolation with combined neural networks to establish a daily runoff prediction framework. Taking a small basin with large daily runoff changes in southwest China as an example, the influence of the combined neural network model on the accuracy of daily runoff prediction after data augmentation was studied. To evaluate the predictive performance of the model based on data augmentation, the model was compared with SVM, LSTM, and AM-LSTM.

METHODS

Studied basin and data

The basin studied in this paper is Qingxi river basin located in Xuanhan County, Sichuan Province, Southwest China. Qingxi river is a mountain stream with a length of 46 km and a drainage area of 297 km². The relative height difference is more than 300–500 m, and the river width is about 15–30 m. The Qingxi river basin is frequently flooded by heavy rain, with the annual maximum flow occurring from May to September, and the maximum in July.

As shown in Figure 1, in Qingxi river basin, there is a basic hydrological station and three rain gauge stations with historical hydrological data, which span from 1 January 1986 to 31 December 2005, including daily rainfall, daily evaporation, and daily runoff data. In Table 1, the data are

divided into a training set and test set. The first 18 years are used for model training and validation (the first 16 years of data are used for training and the next two years of data are used for verification), and the last two years are used for model testing. The data came from the Hydrology and Water Resources Bureau of Sichuan Province.

The trend of runoff data can be intuitively seen in Figure 2. The maximum flow of runoff is 453 m³/s, the minimum flow is 0.057 m³/s, and the average flow is 5.627 m³/s.

CAGANet model

Figure 3 shows the structural diagram of CAGANet. The combined neural network structure proposed in this study is composed of a convolutional layer, an AM, and a GRU. In order to improve the robustness of the model, the results of predicting the linear part of the AR model are added. We divide the data into long-term historical data $\{X_i\} = X_1, \ldots, X_n$ and short-term data S, where the short-term data S is used for linear prediction. In this article, $n = 6$ was selected through testing. The short-term data reflect a linear response, and the choice is 1 day. The workflow of this model is:

(1) divide the data into long-term data X_i and short-term data S;
(2) input long-term data into the convolutional layer to extract the temporal distribution characteristics of hydrological data variables and the local dependencies between the variables;
(3) input the extracted characteristics into the attention mechanism layer, and assign the attention weights to the input;

Figure 1 │ Map of Qingxi river basin.

Table 1 | Data set

Station	Data type	Training data				Testing data	
		Min.	Max.	Mean,	Min.	Max.	Mean.
Nanping	Rainfall (mm/d)	0	136.4	3.69	0	203.8	5.14
Fengcheng	Rainfall (mm/d)	0	170.6	3.70	0	183.0	4.81
Laojun	Rainfall (mm/d)	0	152.6	3.54	0	229.4	4.62
Qingxi	Rainfall (mm/d)	0	167.9	3.16	0	200.5	4.07
Qingxi	Evaporation (mm/d)	0	9.0	1.71	0	7.7	1.69
Qingxi	Runoff (m^3/s)	0.057	307.0	5.28	0.027	453.0	7.60
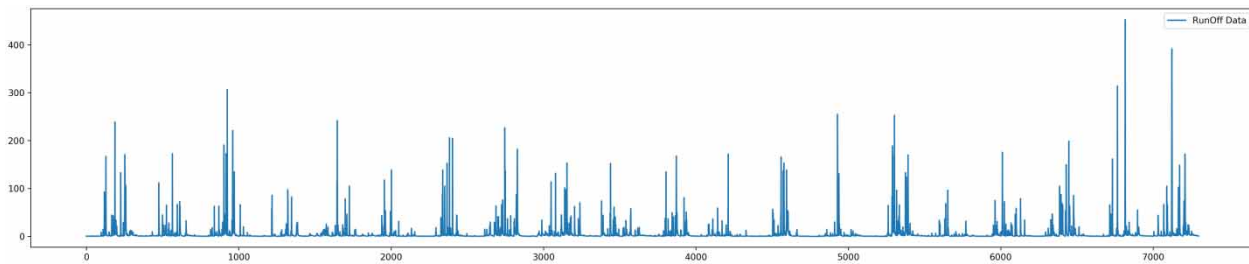
Figure 2 | Daily runoff data of Qingxi station.
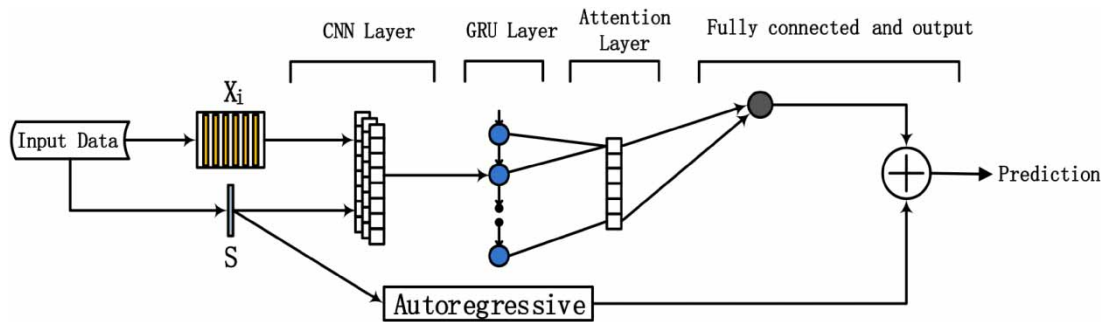
Figure 3 | Model structure overview of CAGANet.

(4) input the data assigned with the attention weights into the GRU network layer for nonlinear part prediction;

(5) input short-term data S into the AR model to obtain the prediction results of the linear part;

(6) integrate the prediction results of nonlinear and linear parts into the final prediction.

The detailed work of the CAGANet model is described below.

First, the input data are divided into long-term data and short-term data, and then they are input into the convolutional layer without pooling to extract the local dependencies between the input variables and the characteristics of time distribution. Both long-term and short-term data need to enter the nonlinear network. Taking the long-term data as an example, the input variable matrix is $X, X \in R^{D \times T}$, the convolution kernel size of the convolutional layer is $\omega \times D$, in which ω is the time dimension, and D is the variable dimension. The expression of the k-th filter sweeps through the input matrix X, which can be formulated as:

$$h_k = RELU(W_k * X + b_k) \tag{1}$$

where * denotes the convolution operation, the output h_k would be a vector, and the RELU function is $RELU\,(x) = max\,(0, x)$. We use the zero-padding mode, and the output

matrix vector of the convolutional layer is $d_c \times T_c$, in which d_c represents the number of filters, and $T_c = T - \omega - 1$.

We apply an attention layer (Zhao *et al.* 2016) to the convolutional layer's output matrix over the time dimension. That is, we can view the matrix as a sequence of d_c dimensional vectors and the sequence length is T_c. We apply attention over the time dimension so that our model can select relative time across all time steps adaptively.

The attention layer is applied to the recurrent network layer, acting on the hidden state of the recurrent network layer. The recurrent component is a recurrent layer with the GRU and uses the RELU function as a hidden activation function. The hidden state of the recurrent unit at time t can be formulated as:

$$r_t = \sigma(x_t W_{xr} + h_{t-1} W_{hr} + b_r) \tag{2}$$

$$u_t = \sigma(x_t W_{xu} + h_{t-1} W_{hu} + b_u) \tag{3}$$

$$c_t = RELU(x_t W_{xc} + r_t \odot (h_{t-1} W_{hc}) + b_c) \tag{4}$$

$$h_t = (1 - u_t) \odot h_{t-1} + u_t \odot c_t \tag{5}$$

where $\odot$ is the element-wise product, σ is the sigmoid function, x_t is the input of this layer at time t and the output is the hidden state of each time stamp of this layer. It is then output to a fully connected layer (FCL). The FCL combines long-term data output and short-term data output to obtain the results of the nonlinear part of the data prediction. The output of the dense layer is computed as:

$$y_t^D = W^D [o_L; o_S] + b \tag{6}$$

in which $o_L = \{o_1; o_2; \ldots; o_T\}$ is the output of long-term historical data prediction, while o_S is the output of short-term historical data prediction, and y_t^D is the nonlinear output of the neural network.

Due to the nonlinearity of the convolutional layer and the recurrent layer, the linear components of daily runoff prediction are not fully predicted. To solve this problem, the traditional AR model is used for linear prediction, and the output of the AR model is $y_t^L, y_t^L \in R^D$. The coefficients of the AR model are $W^{ar} \in R^{p^{ar}}$ and $b^{ar} \in R$, where p^{ar} is the window size of the input matrix, and S is the input variable matrix. In this model, all dimensions share the same

linear parameters. The AR model is expressed as follows:

$$y_{t,i}^L = \sum_{k=0}^{p^{ar}-1} w_k^{ar} s_{t-k,i} + b^{ar} \tag{7}$$

The final prediction of the CAGANet model is $\hat{y}$, which is obtained by integrating the outputs of the nonlinear part of neural network y_t^D and the linear output of AR model y_t^L:

$$\hat{y} = y_t^D + y_t^L \tag{8}$$

In the training process, the mean absolute error (L1-loss) is adopted as the loss function of the prediction task, and the optimization objective is:

$$O(y_T, \hat{y}_T) = \frac{1}{N} \sum_{j=1}^{N} \sum_{i=i}^{D} |(\hat{y}_{T,i}^j - y_{T,i}^j)| \tag{9}$$

where N is the number of training samples and D is the dimension of target data. All neural models were trained using the Adam optimizer.

Long short-term memory neural network (LSTM)

Long short-term memory networks are a special class of RNN that contain a feedback connection that allows past information to affect the current output, making them very effective for tasks involving sequential input (Lecun *et al.* 2015). There are many studies on flood forecasting based on LSTM neural networks (Qi *et al.* 2019), and the results show that the LSTM model has a stronger time-lag prediction capability compared with the traditional BP, RBNN and LSSVM models.

LSTM neural network based on attention mechanism (AM-LSTM)

Human vision can quickly find important target areas while other areas are only roughly analyzed or even ignored. This active, selective mental activity is called the visual attention mechanism (AM). AM originates from the simulation of the human brain's attention characteristics. Its core idea is to give more attention to important information and less attention to other information, thereby greatly improving the

information receiving sensitivity and processing speed of the concerned area. Strictly speaking, AM is an idea, not a model. Assuming that each element of the input data set consists of a key and a value, the given target is T and the final result is the attention weight, then the core idea of AM can be represented by Figure 4.

Following the development of AM, it began to be applied in various fields, first in image processing and machine translation, and later in time prediction (Zhao *et al.* 2016). Some researchers have combined AM and LSTM models to predict floods (Wu *et al.* 2018).

Evaluation indicators

In this paper, three statistical methods, including mean absolute error (MAE), root mean square error (RMSE) and Nash–Sutcliffe efficiency (NSE) (Nash & Sutcliffe 1970), are used to measure the performance of the model. These three metrics are defined as follows.

Mean absolute error:

$$MAE = \frac{1}{N} \sum_{i=1}^{N} |y_i - \hat{y}_i| \tag{10}$$

Root mean square error:

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2} \tag{11}$$

Nash–Sutcliffe efficiency:

$$NSE = 1 - \frac{\sum_{i=1}^{N} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{N} (y_i - \bar{y}_i)^2} \tag{12}$$
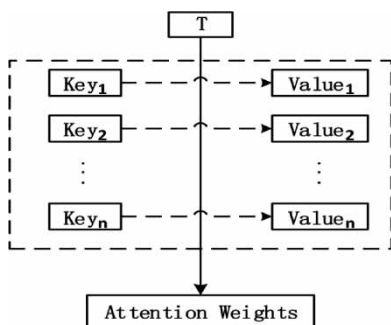

Figure 4 | Core idea of AM.

where N is the total number of observations, y_i and $\hat{y}_i$ are the observed and predicted flow, respectively, and $\bar{y}_i$ is the average value of the observed flow. The RMSE value is a good indicator of the goodness of fit at high flow rates, and NSE provides a measure of the model's predictive power. Generally, higher NSE values and lower MAE and RMSE values indicate a good model. These three statistical methods can well evaluate the performance of hydrological models (Legates & McCabe 1999).

RESULTS AND DISCUSSION

Selection of input variables

In order to have a better and more effective flood prediction, the thermal map of the linear Pearson correlation coefficients of historical flow, rainfall, evaporation, and flow during the forecast period (F) at each station is drawn in Figure 5.

From the thermal map, we can see the linear correlation. Combined with the actual situation, the input variables are the rainfall of the four stations in the past 2 days $R(t-2)$, $R(t-1)$, evaporation in the past day $E(t-1)$, as well as the flow in the past 2 days $F(t-2)$ and $F(t-1)$, a total of 16 characteristics. These 16 features are enhancements to the data, and we used all possible inputs (i.e., X_t, X_{t-1}, X_{t-2}, ... X_{t-6}) for these 16 features, that is, a set of six time-lagged inputs is used to forecast the current value, as shown in Table 2, where X represents 16 input features.

Selection of parameters

A grid search was performed on all the adjustable hyperparameters of the LSTM, AM-LSTM, and CAGANet models, and appropriate values were selected from $\{2^2, 2^3, \ldots, 2^8\}$, and then the optimal parameter setting determined by decreasing or increasing the value. In addition, both the AM-LSTM model and the CAGANet model need to set a suitable time step.

The results in Table 3 show that the number of convolutional layer and GRU neurons suitable for CAGANet model is 16. In addition, both the AM-LSTM model and the CAGA-Net model need to set a suitable time step. Time step refers to how many input variables the attention mechanism will
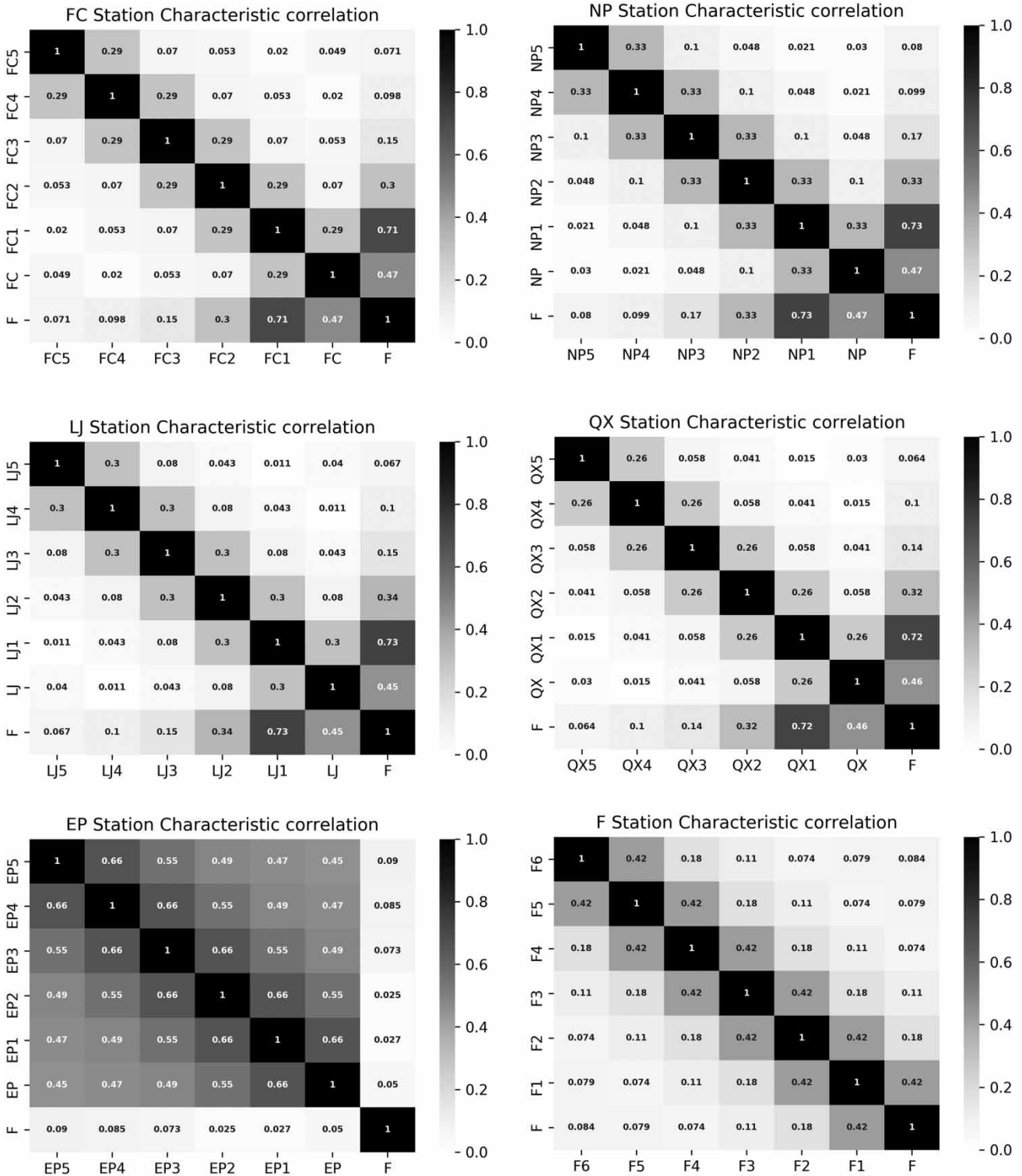
Figure 5 │ Thermal diagram of Pearson correlation coefficient between input variables and runoff.

Table 2 │ Selection of input variables

Station	Feature	Inputs
Nanping, Fengcheng, Laojun, Qingxi	(Rainfall) R_t, R_{t-1}, R_{t-2}	X_t, X_{t-1}, X_{t-2}, … X_{t-6}
Qingxi	(Evaporation) E_t, E_{t-1}	
Qingxi	(Runoff) F_t, F_{t-1}	

Table 3 │ Selection of convolutional layer and GRU structure

Convolutional structure	GRU structure	NSE	RMSE	MAE
8	8 − 8	0.669	19.75	4.40
16	8 − 8	0.718	17.99	4.36
8	16 − 16	0.805	14.97	4.04
16	**16 − 16**	**0.854**	**12.23**	**3.11**
32	16 − 16	0.787	15.66	3.79
32	32 − 32	0.768	16.02	3.89

notice. These input data will be assigned appropriate weights according to the time step. This process is part of model training. The time step chosen after testing is 6.

Prediction results

The four models were applied to Qingxi river basin of Sichuan province. We have a total of 20 years of data, of which the first 16 years are used to train the model, then two years are used to verify the model, and the last two years are used to test the model. The forecast results of the four models on the last two years of the 20 years data are as follows.

In Figures 6–9, the prediction results of the four models are consistent in the predicted and observed values. The SVM prediction value fluctuates more seriously and the prediction effect is the worst. The peak prediction of CAGANet is closer to the observed values than that of SVM, LSTM, and AM-LSTM, and the overall prediction performance of CAGANet is higher than that of the other three models. In Table 4, CAGANet's prediction indicator without data enhancement is NSE = 0.854, RMSE = 12.23, MAE = 3.11.

Each of the four models has a number of points where the predicted value is higher than the observed value on the data set without interpolation. This does not affect the
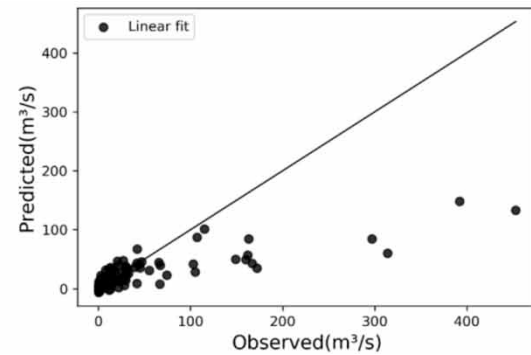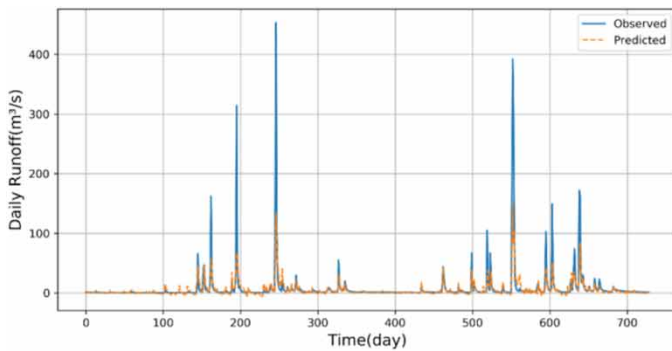

Figure 6 │ Final prediction result and scatter diagram of the SVM original data set.
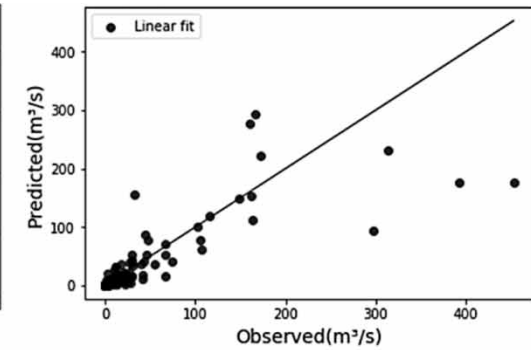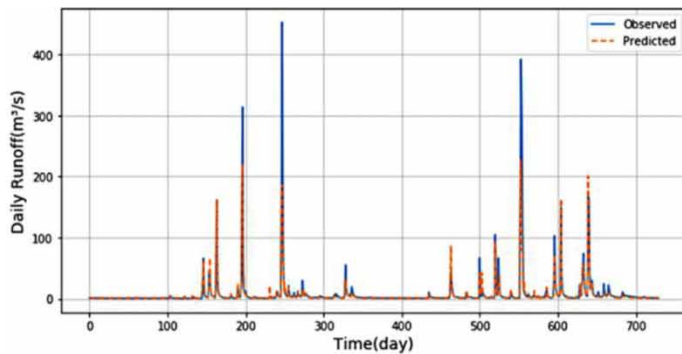

Figure 7 │ Final prediction result and scatter diagram of the LSTM original data set.
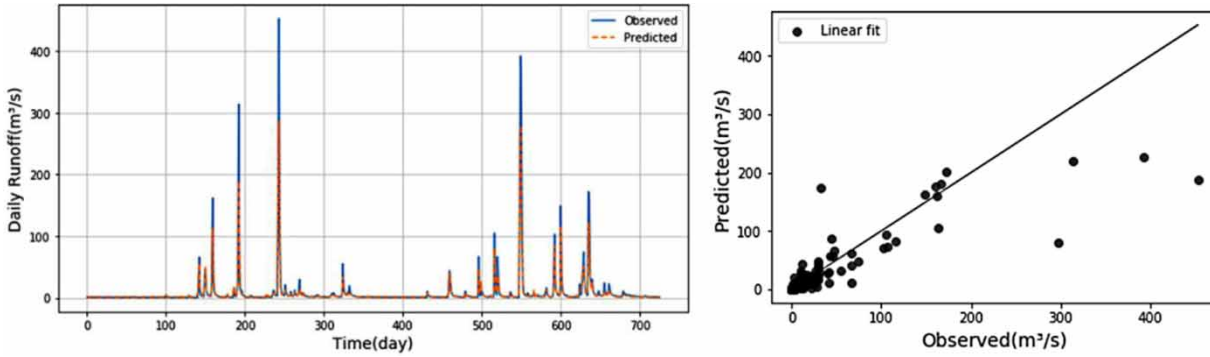
Figure 8 │ Final prediction result and scatter diagram of the AM-LSTM original data set.
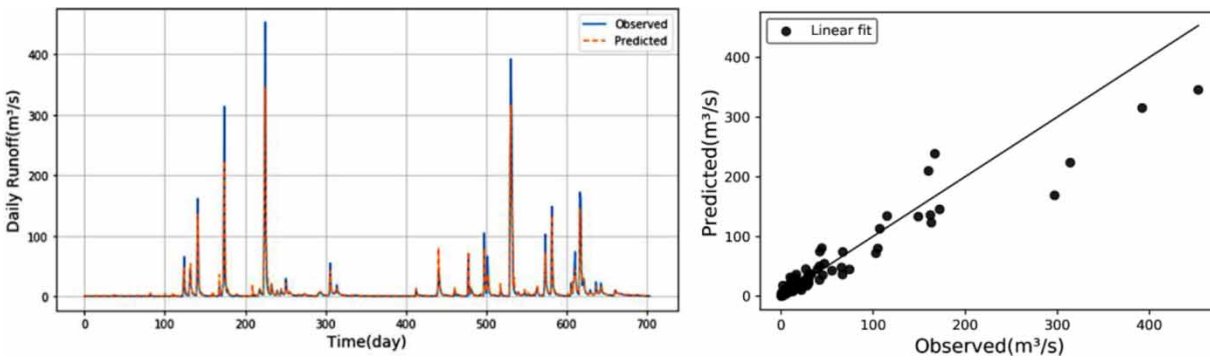

Figure 9 │ Final prediction result and scatter diagram of the CAGANet original data set.

Table 4 │ Evaluation indicators of the original data set

Indicators	NSE	RMSE	MAE
CAGANet	**0.854**	**12.23**	**3.11**
AM-LSTM	0.816	16.22	3.93
LSTM	0.753	17.28	4.36
SVM	0.505	22.76	5.41

forecast for larger floods, but it is unfriendly for smaller ones. The phenomenon is caused by the instability of the data, so we adopt the method of linear interpolation to make the data relatively stable.

Figures 10–13 present the prediction results on the First-LI data set (data set gained after performing linear interpolation the first time). After the data augmentation, the
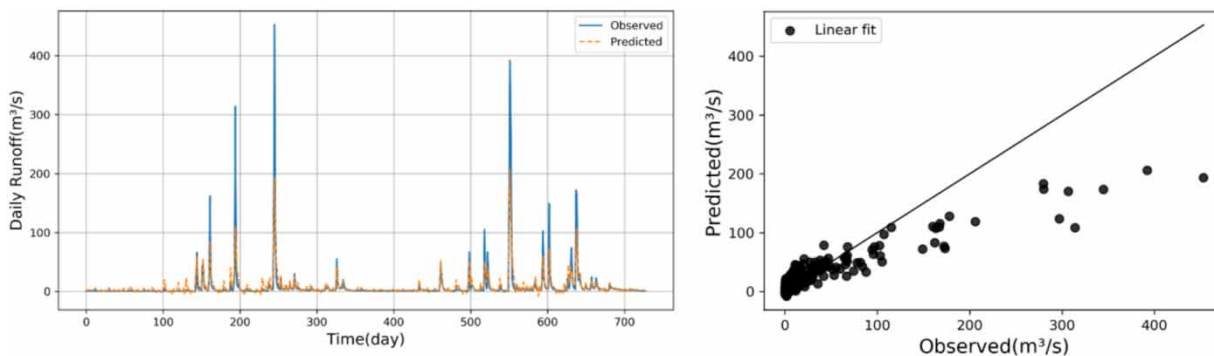

Figure 10 │ Final prediction result and scatter diagram of the SVM on First-LI original data set.

prediction performance of CAGANet was significantly improved, and its peak value and overall prediction accuracy are far higher than that of the SVM, LSTM, and AM-LSTM. In particular, the predicted value of CAGANet was close to the observed value in terms of peak flow prediction, which was the most concerned by flood forecasting. The predictive performance of the other three models has been

improved compared with that before data augmentation, but they still need to improve their peak prediction.

In Table 5, CAGANet's indicator NSE reached 0.979, which is 12% higher than the original value, and 24%, 12%, and 6% higher than that of SVM, LSTM, and AM-LSTM, respectively. The results show that the data augmentation can effectively improve the accuracy of daily runoff
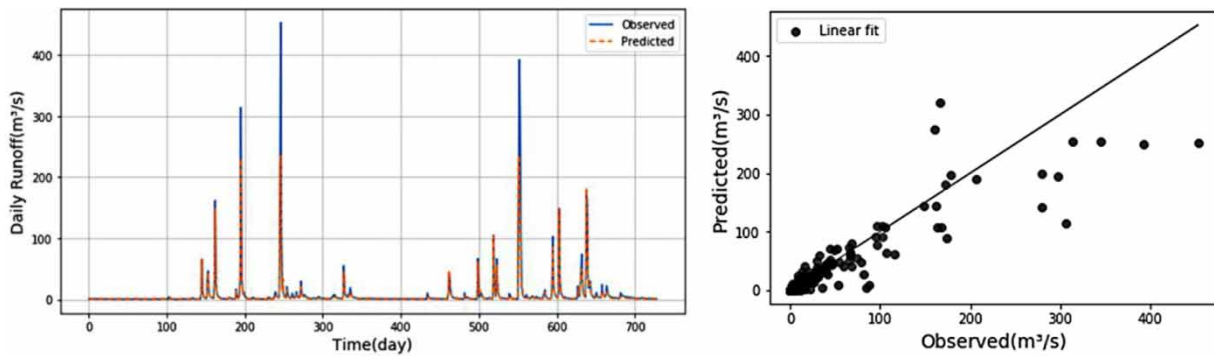

Figure 11 | Final prediction result and scatter diagram of the LSTM on First-LI original data set.
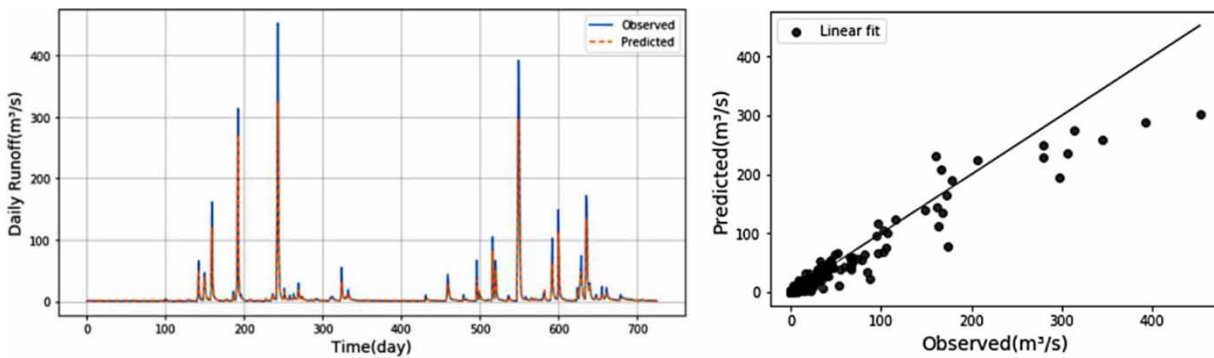

Figure 12 | Final prediction result and scatter diagram of the LAM-LSTM on First-LI original data set.
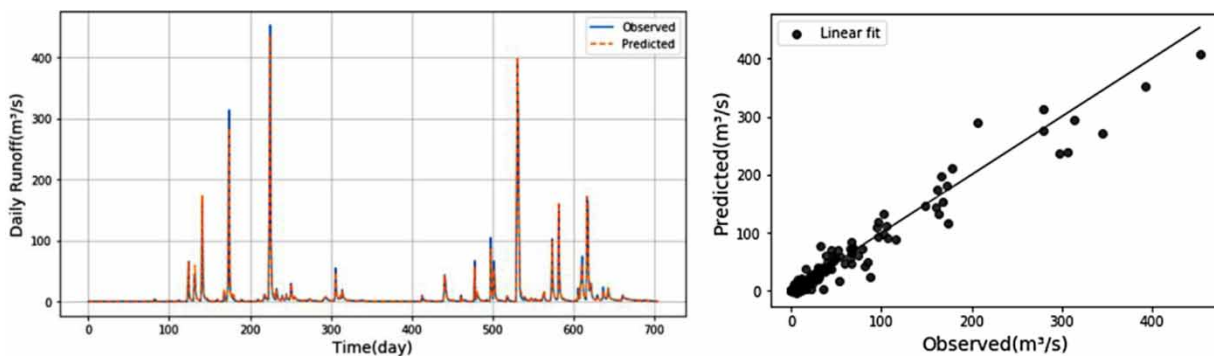

Figure 13 | Final prediction result and scatter diagram of the CAGANet on First-LI original data set.

Table 5 │ Evaluation indicators on First-LI data set

| Indicators	NSE	RMSE	MAE
CAGANet	**0.979**	**6.22**	**1.62**
AM-LSTM	0.911	10.54	2.05
LSTM	0.859	12.58	2.42
SVM	0.738	15.58	4.80

prediction, and also indicate that the generalization ability and robustness of the CAGANet model are better than the other three models.

In order to verify the prediction performance of daily runoff under different data augmentation plans, the prediction of the Second-LI data set was also performed, and the results are shown in Figures 14–17.

Figures 14–17 show the prediction results on the Second-LI data set (data set gained after performing linear interpolation the second time). The CAGANet prediction results are almost coincident with the observed values. In contrast, the peak prediction performance of the LSTM was not significantly improved, and although SVM and AM-LSTM had a better prediction performance compared with their prediction on the First-LI data set, such improvement is relatively slow. In Table 6, the evaluation indicators of the CAGANet model on Second-LI data set NSE = 0.993, RMSE = 2.58, and MAE = 0.6 were obtained, which are 12%, 10%, and 3% higher than that of the SVM, LSTM, and AM-LSTM. In addition, it can be seen that many predicted values of SVM on the Second-LI data set are higher than the observed values, indicating that enhancing data stability cannot improve the prediction accuracy of the SVM model on small values. The results show that CAGANet has good generalization ability and prediction ability.

Comparing the total of 12 prediction results of the four models on the original data set, the First-LI data set, and the Second-LI data set, it can be concluded that the CAGANet prediction results on the data-augmented Second-LI data set are better than the other 11 forecast methods. With the
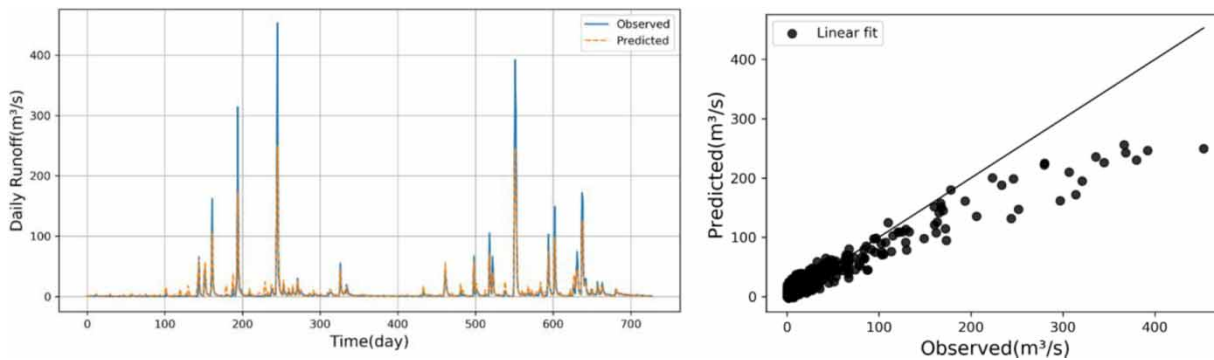
Figure 14 │ Final prediction result and scatter diagram of the SVM on Second-LI original data set.
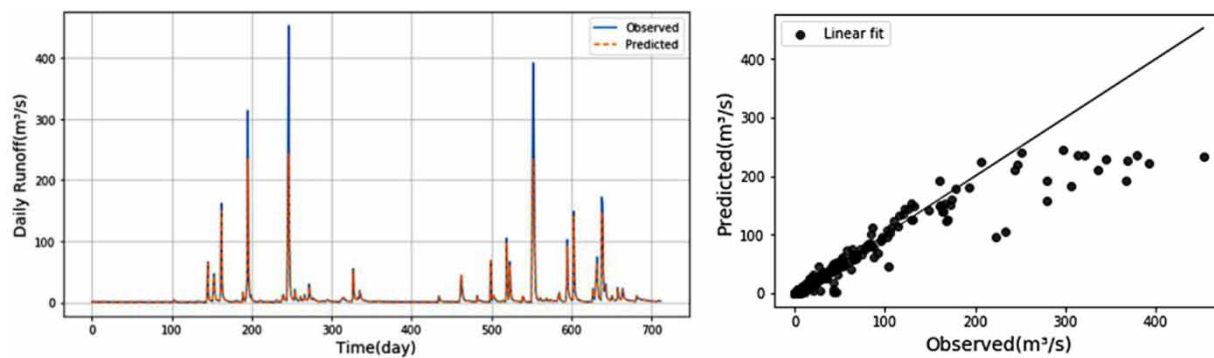
Figure 15 │ Final prediction result and scatter diagram of the LSTM on Second-LI original data set.
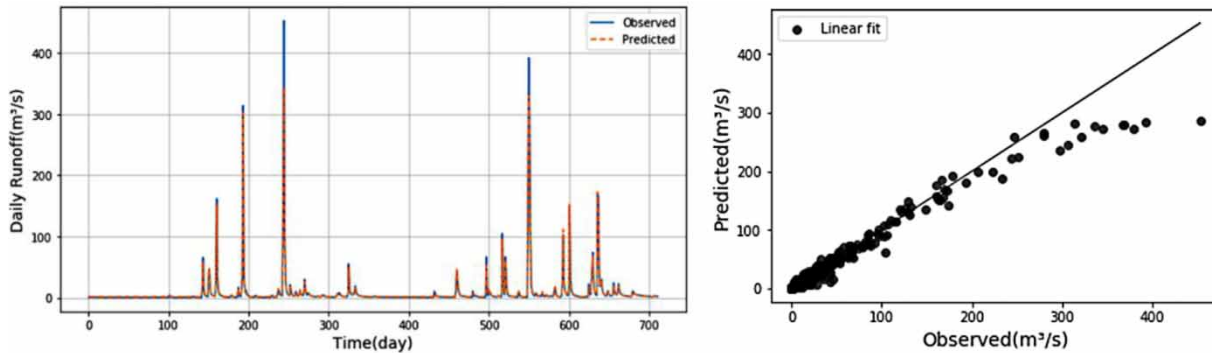
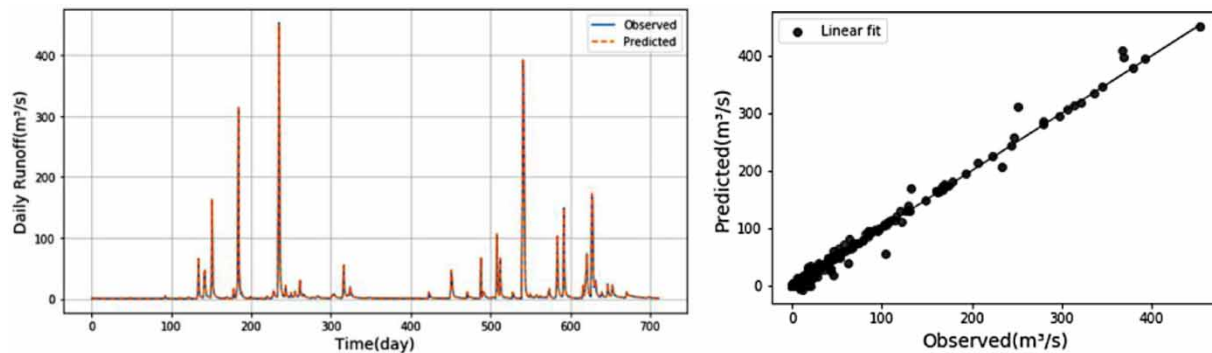**Figure 16** │ Final prediction result and scatter diagram of the LAM-LSTM on Second-LI original data set.

Figure 17 │ Final prediction result and scatter diagram of the CAGANet on Second-LI original data set.

Table 6 │ Evaluation indicator on Second-LI data set

Indicators	NSE	RMSE	MAE
CAGANet	**0.993**	**2.58**	**0.60**
AM-LSTM	0.958	6.14	1.04
LSTM	0.893	10.37	1.34
SVM	0.872	10.70	3.39

increase of the stability of the data, the predictive accuracy of results on all four models has improved, among which the most obvious one is the CAGANet model proposed in this paper, which shows that the CAGANet model has good prediction performance.

Ablation experiment

The CAGANet model proposed in this paper is a compound neural network model composed of four modules. In order to verify the contribution of each component of the model to the prediction of daily runoff, several sets of ablation experiments were performed in this paper. In the CAGANet model, the convolutional layer module, the AM module, and the GRU module belong to a tandem structure, while the linear AR module and the first three belong to a parallel structure. Based on this structure, the proposed ablation experiment eliminates the three modules of convolutional layer, AM, and AR, respectively. Then the prediction results after the ablation are analyzed and compared with the prediction results without ablation.

First, we ablated the convolutional layer module. The role of convolutional layer in the entire CAGANet model is to extract the invariant structure and hidden features in the data, and it contains multiple convolution kernels. The convolution kernel is smaller than the input data dimension, and when filtering the data, it locally perceives the information and then synthesizes the global information. The

convolutional layer uses parameters in the same convolution kernel, called weight sharing, which can greatly reduce the amount of calculations. We calculated the predicted time after removing the convolutional layer. It takes 33 seconds for one epoch (training all samples once), and it takes only 10 seconds without removing the convolutional layer. In other words, the time it takes after removing the convolutional layer is more than three times that without removing the convolutional layer.

Figure 18 is a prediction result of convolutional layer ablation on a data set without interpolation. It can be seen from the results that after removing the convolutional layer module, the accuracy of the prediction is reduced, and there are some predicted values that are larger than the observed values. After removing the convolutional layer, there is no feature extraction, and the model needs to process all the details of the data, which leads to an increase in the prediction time. At the same time, the excessive attention to some details leads to the phenomenon that the predicted value is greater than the observed value. The evaluation indicators were $MAE = 4.02$, $RMSE = 15.24$, and $NSE = 0.786$.

Then, we ablated the AM module. AM has been briefly introduced in the section introducing the AM-LSTM model. The role AM plays in the CAGANet model is to assign weights to input variables at a given time step. In a separate GRU structure, a lot of data information is integrated into the hidden vector h_{T_x} corresponding to the last time step T. But h_{T_x} is a single vector of a certain length. The representation ability of this single vector and the amount of information it contains are limited, so a lot of information will be lost. AM allows the GRU to consider

the hidden state sequence $(h_1, h_2, \cdots, h_{T_x})$ output by the entire encoder at each time step t, thereby storing more information in all hidden state vectors. The GRU can decide which vectors should be given a higher weight when using these vectors.

Specifically, each output y_t in the target sequence $(y_1, y_2, \cdots, y_{T_x})$ produced by GRU is based on the following conditional distribution:

$$P[y_t | \{y_1, \cdots, y_{t-1}\}, c_t] = softmax(W_s \tilde{h}_t) \tag{13}$$

of which, $\tilde{h}_t$ is the attentional hidden state vector, and it is formulated as:

$$\tilde{h}_t = tanh\,(W_c[c_t; h_t]) \tag{14}$$

h_t is the hidden state of the top level of the GRU, c_t is the context vector, which is calculated from the hidden vector at the current time step h_{T_x}. W_c and W_s are weight parameters. Assume that the sequence length of sample x is T_x, then c_t is calculated as follows:

$$c_t = \sum_{i=1}^{T_x} \alpha_{t,i} h_i \tag{15}$$

where $\alpha_{t,i}$ shows the degree of importance of all vectors in the hidden state sequence, that is, weight allocation, and is
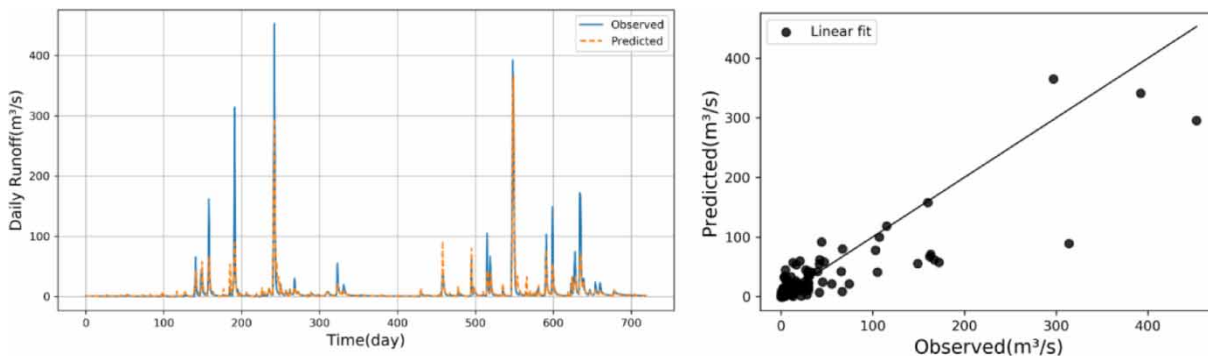

Figure 18 │ Prediction result and scatter diagram of the CAGANet without convolutional layer.

calculated by the *softmax* method:

$$\alpha_{t,i} = \frac{\exp(score(h_t,\ h_i))}{\sum_{j=1}^{T_x} \exp(score(h_t,\ h_j))} \tag{16}$$

And the *score* function is:

$$score(h_t\ ,\ h_i) = h_t^T W_\alpha h_i \tag{17}$$

of which, W_α is a parameter matrix.

Figure 19 is a prediction result of AM ablation on a data set without interpolation. It can be seen from the results that the prediction accuracy is not as good as the result with AM, and the accuracy is reduced. In terms of details, many small-value predictions are not accurate with big fluctuation, which indicates bad attention. Comparing the prediction results of LSTM, we find that the results of LSTM also

have this problem, but the experimental results using AM have no such phenomenon. The most outstanding feature of AM is the concentration of attention weights. The evaluation indicators are MAE = 4.90, RMSE = 17.45, and NSE = 0.712.

Finally, we ablate the AR module. The role of AR in the CAGANet model is to extract short-term linear prediction results. This module is very simple and was introduced earlier. Figure 20 is a prediction result of AR ablation on a data set without interpolation. It can be seen from the results that the prediction accuracy is not as good as the CAGANet model. The AR module will output linear prediction results, while other modules output nonlinear predictions. The predictions without AR module will lose linear results, so the prediction accuracy is reduced. The evaluation indicators are MAE = 4.90, RMSE = 17.45, and NSE = 0.805.

In summary, no matter which module is ablated, the final prediction accuracy will be reduced. It is experimentally
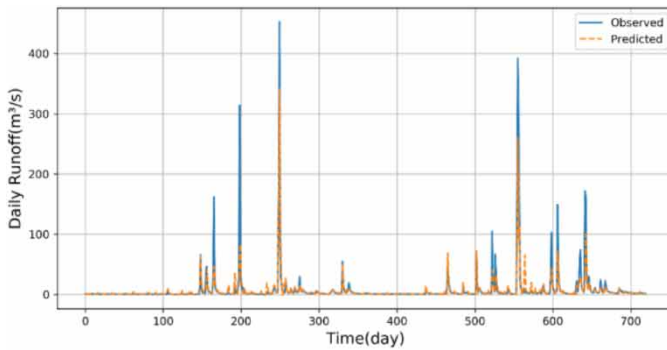

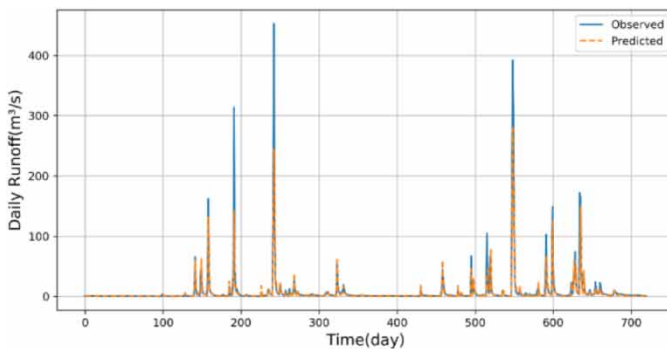**Figure 19** │ Prediction result and scatter diagram of the CAGANet without AM.

Figure 20 │ Prediction result and scatter diagram of the CAGANet without AR.

verified that each module of the CAGANet model proposed in this paper has contributed to daily runoff prediction.

CONCLUSION

In this paper, a combined neural network model CAGANet based on data augmentation is proposed to predict the daily runoff of Qingxi basin in Sichuan province. Compared with single SVM model, neural network models LSTM and AM-LSTM, the proposed CAGANet model has higher prediction accuracy when forecasting on a data set without using data augmentation methods, and its NSE can reach 0.854. In view of the instability of daily hydrological time series, linear interpolation is used to enhance the relative stability of the data in order to further improve the prediction accuracy. The results show that among the four models, the proposed CAGANet model improves its prediction accuracy faster and higher than the other three models with the increase of data stability. These four models all use NSE, RMSE, and MAE evaluation indicators. In the three sets of experimental comparisons, the daily runoff prediction evaluation indicators of the CAGANet model can reach $NSE = 0.993$, $RMSE = 2.58$, and $MAE = 0.60$, and the NSE indicator was 12%, 10%, and 3% higher than that of the SVM, LSTM, and AM-LSTM, respectively. Therefore, the CAGANet model based on data augmentation can effectively improve the accuracy of daily runoff prediction.

REFERENCES

Bai, Y., Wang, P., Xie, J., Li, J. & Li, C. 2014 Additive model for monthly reservoir inflow forecast. *Journal of Hydrologic Engineering* **20** (7), 04014079. https://doi.org/10.1061/(ASCE)HE.1943-5584.0001101.

Bengio, Y., Simard, P. & Frasconi, P. 1994 Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks* **5** (2), 157–166. https://doi.org/10.1109/72.279181.

Citakoglu, H., Cobaner, M., Haktanir, T. & Kisi, O. 2014 Estimation of monthly mean reference evapotranspiration in Turkey. *Water Resources Management* **28** (1), 99–113. https://doi.org/10.1007/s11269-013-0474-1.

Coulibaly, P. & Baldwin, C. K. 2005 Nonstationary hydrological time series forecasting using nonlinear dynamic methods.

Journal of Hydrology **307** (1–4), 164–174. https://doi.org/10.1016/j.jhydrol.2004.10.008.

Kratzert, F., Klotz, D., Brenner, C., Schulz, K. & Herrnegger, M. 2018 Rainfall-runoff modelling using Long-Short-Term-Memory (LSTM) networks. *Hydrology and Earth System Sciences* **22** (11), 6006–6022. https://doi.org/10.5194/hess-22-6005-2018.

Kumar, D. N., Raju, K. S. & Sathish, T. 2004 River flow forecasting using recurrent neural networks. *Water Resources Management* **18** (2), 143–161. https://doi.org/10.1007/BF02704433.

Lecun, Y., Bengio, Y. & Hinton, G. 2015 Deep learning. *Nature* **521**, 436–444. https://xs.scihub.ltd/https://doi.org/10.1038/nature14539.

Legates, D. R. & McCabe, G. J. 1999 Evaluating the use of 'goodness-of-fit' measures in hydrologic and hydroclimatic model validation. *Water Resources Research* **35** (1), 233–241. https://doi.org/10.1029/1998WR900018.

Mehr, A. D., Kahya, E. & Olyaie, E. 2013 Streamflow prediction using linear genetic programming in comparison with a neuro-wavelet technique. *Journal of Hydrology* **505**, 240–249. https://doi.org/10.1016/j.jhydrol.2013.10.003.

Miao, Q., Pan, B., Wang, H., Hsu, K. & Sorooshian, S. 2019 Improving monsoon precipitation prediction using combined convolutional and long short term memory neural network. *Water* **11** (5), 977. https://doi.org/10.3390/w11050977.

Moghaddamnia, A., Ghafari, M., Piri, J., Amin, S. & Han, D. 2009 Evaporation estimation using artificial nctworlcs and adaptive ncuro-fuzzy inference system techniques. *Advances in Water Resources* **32**, 88–97. https://doi.org/10.1016/j.advwatres.2008.10.005.

Nash, J. E. & Sutcliffe, J. V. 1970 River flow forecasting through conceptural models. Part 1: a discussion of principles. *Journal of Hydrology* **10** (3), 282–290. https://doi.org/10.1016/0022-1694(70)90255-6.

Nourani, V., Baghanam, A. H., Adamowski, J. & Kisi, O. 2014 Applications of hybrid wavelet–artificial intelligence models in hydrology: a review. *Journal of Hydrology* **514**, 358–377. https://doi.org/10.1016/j.jhydrol.2014.03.057.

Qi, Y., Zhou, Z., Yang, L., Quan, Y. & Miao, Q. 2019 A decomposition-ensemble learning model based on LSTM neural network for daily reservoir inflow forecasting. *Water Resources Management* **33**, 4123–4139. https://xs.scihub.ltd/https://doi.org/10.1007/s11269-019-02345-1.

Quilty, J., Adamowski, J. & Boucher, M. A. 2019 A stochastic data-driven ensemble forecasting framework for water resources: a case study using ensemble members derived from a database of deterministic wavelet-based models. *Water Resources Research* **55** (1), 175–202. https://doi.org/10.1029/2018WR023205.

Ravansalar, M., Rajaee, T. & Kisi, O. 2017 Wavelet-linear genetic programming: a new approach for modeling monthly streamflow. *Journal of Hydrology* **549**, 461–475. https://doi.org/10.1016/j.jhydrol.2017.04.018.

Shen, C. 2018 A transdisciplinary review of deep learning research and its relevance for water resources scientists. *Water Resources Research* **54** (11), 8558–8593. https://doi.org/10.1029/2018WR022643.

Sivapragasam, C., Shie, Y. L. & Pasha, M. F. K. 2001 Rainfall and runoff forecasting with SSA–SVM approach. *Journal of Hydroinformatics* **3** (3), 141–152. https://doi.org/10.2166/hydro.2001.0014.

Taormina, R. & Chau, K. W. 2015 Neural network river forecasting with multiobjective fully informed particle swarm optimization. *Journal of Hydroinformatics* **17** (1), 99–113. https://doi.org/10.2166/hydro.2014.116.

Toth, E., Brath, A. & Montanari, A. 2000 Comparison of short-term rainfall prediction models for real-time flood forecasting. *Journal of Hydrology* **239** (1), 132–147. https://doi.org/10.1016/S0022-1694(00)00344-9.

Wang, W., Chau, K., Cheng, C. & Qiu, L. 2009 A comparison of performance of several artificial intelligence methods for forecasting monthly discharge time series. *Journal of Hydrology* **374** (3), 294–306. https://doi.org/10.1016/j.jhydrol.2009.06.019.

Wu, C. L., Chau, K. W. & Li, Y. S. 2009 Predicting monthly streamflow using data-driven models coupled with data-preprocessing techniques. *Water Resources Research* **45** (8). https://doi.org/10.1029/2007WR006737.

Wu, Y. R., Liu, Z. Y., Xu, W. G., Feng, J., Shivakumara, P. & Lu, T. 2018 Context-Aware Attention LSTM Network for Flood Prediction. In: *2018 24th International Conference on Pattern Recognition (ICPR)*, Beijing, China, pp. 1301–1306. https://doi.org/10.1109/ICPR.2018.8545385

Zhang, J., Zhu, Y., Zhang, X., Ye, M. & Yang, J. 2018 Developing a long short-term memory (LSTM) based model for predicting water table depth in agricultural areas. *Journal of Hydrology* **561**, 918–929. https://doi.org/10.1016/j.jhydrol.2018.04.065.

Zhao, Y., Ye, L., Li, Z., Song, X., Lang, Y. & Su, J. 2016 A novel bidirectional mechanism based on time series model for wind power forecasting. *Applied Energy* **177**, 793–803. https://doi.org/10.1016/j.apenergy.2016.03.096.