

Using long short-term memory networks for river flow prediction

Wei Xu, Yanan Jiang, Xiaoli Zhang, Yi Li, Run Zhang and Guangtao Fu 

ABSTRACT

Deep learning has made significant advances in methodologies and practical applications in recent years. However, there is a lack of understanding on how the long short-term memory (LSTM) networks perform in river flow prediction. This paper assesses the performance of LSTM networks to understand the impact of network structures and parameters on river flow predictions. Two river basins with different characteristics, i.e., Hun river and Upper Yangtze river basins, are used as case studies for the 10-day average flow predictions and the daily flow predictions, respectively. The use of the fully connected layer with the activation function before the LSTM cell layer can substantially reduce learning efficiency. On the contrary, non-linear transformation following the LSTM cells is required to improve learning efficiency due to the different magnitudes of precipitation and flow. The batch size and the number of LSTM cells are sensitive parameters and should be carefully tuned to achieve a balance between learning efficiency and stability. Compared with several hydrological models, the LSTM network achieves good performance in terms of three evaluation criteria, i.e., coefficient of determination, Nash–Sutcliffe Efficiency and relative error, which demonstrates its powerful capacity in learning non-linear and complex processes in hydrological modelling.

Key words | hydrological modelling, LSTM, machine learning, river flow prediction

HIGHLIGHTS

- Long short-term memory (LSTM) networks are assessed for river flow prediction.
- The impacts of network structures and parameters on learning efficiency are analysed.
- The batch size and the number of LSTM cells are sensitive parameters for learning.
- LSTM has good predictive accuracy compared to hydrological and data-driven models tested.

INTRODUCTION

The rainfall–runoff process of a river basin is normally characterized by a high degree of nonlinearity. The process is one of the most important components in the hydrological cycle and its accurate modelling is crucial for water resources and flood management (Clarke 1994; Nourani 2017). Rainfall–

This is an Open Access article distributed under the terms of the Creative Commons Attribution Licence (CC BY 4.0), which permits copying, adaptation and redistribution, provided the original work is properly cited (<http://creativecommons.org/licenses/by/4.0/>).

doi: 10.2166/nh.2020.026

Wei Xu


Yi Li

Run Zhang

College of River and Ocean Engineering, National Engineering Research Center for Inland Waterway Regulation, Chongqing Jiaotong University, Chongqing, China

Wei Xu

Yanan Jiang

Guangtao Fu  (corresponding author)
Center for Water Systems, College of Engineering, Mathematics and Physical Sciences, University of Exeter, Exeter EX4 4QF, UK
E-mail: g.fu@exeter.ac.uk

Yanan Jiang

College of Water Resources and Architectural Engineering, Northwest Agriculture and Forestry University, Yanglin, Shaanxi 712100, China

Xiaoli Zhang

School of Water Conservancy, North China University of Water Resources and Electric Power, Zhengzhou, China

Guangtao Fu

The Alan Turing Institute, 96 Euston Road, London NW1 2DB, UK

runoff models are usually classified into three main classes: distributed, conceptual and black box models (Clarke 1994). Distributed and conceptual models are based on various hydrological processes; however, they are limited by our understanding and ability to accurately represent these processes and computational resources. By contrast, black box models are normally data-driven but can provide an accurate prediction in many situations (Tanty & Desmukh 2015; Nourani 2017). Artificial neural network (ANN) models are

one of the typical black box models. Since Daniell (1991) applied ANNs to streamflow modelling, ANNs have been widely applied in hydrological modelling because of its strong non-linear fitting ability (ASCE Task Committee 2000a, 2000b). Currently, various ANN models have been employed to study the rainfall–runoff process, such as fuzzy neural networks (Nayak *et al.* 2004), wavelet neural networks (Wang & Ding 2003; Alexander & Thampi 2018) and Bayesian neural networks (Bateni *et al.* 2007; Kayabasi *et al.* 2017). Traditionally, the ANN learns the relationships between input and output variables from historical data provided and does not have the ability to automatically select the input variables or factors. ANNs with multiple hidden layers have excellent learning ability (Hinton & Salakhutdinov 2006), thus, deep neural networks are increasingly used in hydrology to simulate the rainfall–runoff relationships building on big data which have become available in recent years (Hu *et al.* 2018, 2019; Kratzert *et al.* 2019; Le *et al.* 2019).

In recent years, deep learning has made significant advances in the field of machine learning and data science (Negnevitsky & Pavlovsky 2005). Many deep learning algorithms have shown great potential in solving real-world problems (Khan & Yairi 2018), for example, recurrent neural networks (RNN) (Shin *et al.* 2017) and convolutional neural networks (Zhou *et al.* 2019). In particular, the RNN network has a strong learning ability for time series data (Bengio *et al.* 1994; Hochreiter & Schmidhuber 1997; Saon & Picheny 2017) as it includes loops to allow the information from previous time steps to be passed to the next time step. However, the gradient disappearance or explosion problem makes the RNN gradually lose the ability to learn long-distance information (Bengio *et al.* 1994). To overcome the deficiency, the long short-term memory (LSTM) network, a special type of RNN, was developed for learning with long sequence data (Hochreiter & Schmidhuber 1997), as it is capable of learning long-term dependencies in the data series. Based on the concept of LSTM networks, many similar networks have been constructed to improve the learning ability for different tasks (Sutskever *et al.* 2013; Bellegarda & Monz 2016). At present, the LSTM has been successfully used in speech recognition and text translation (Bellegarda & Monz 2016; Rocha *et al.* 2019).

In the last few years, LSTM networks have been tested and studied in watershed hydrological modelling, and their

potential has been demonstrated in many applications, such as river flow and flood predictions (Shen 2018). Kratzert *et al.* (2018) applied the LSTM network to simulate the daily flows of 241 basins and found that it greatly outperforms hydrological models that are calibrated both at the regional level and at the individual basin level. Lee *et al.* (2018) developed an LSTM for daily runoff simulations based on the water level data of 10 stations at the upper Mekong River and showed that the LSTM performs better than the Soil and Water Assessment Tool model (SWAT). Zhang *et al.* (2018) compared four different neural networks, namely multilayer perceptron, wavelet neural network (WNN), LSTM and gated recurrent unit (GRU), in predicting the daily discharges of combined sewer overflow structures, and showed that LSTM and GRU are highly capable of multi-step-ahead time series prediction. Sahoo *et al.* (2019) applied LSTM to forecast daily flows during low-flow periods in the Mahanadi River basin, India. Kratzert *et al.* (2019) proposed an Entity-Aware-LSTM (EA-LSTM), which performed substantially better at the regional level with 531 basins than several hydrological models calibrated individually for each basin. Hu *et al.* (2018) tested an LSTM model on 98 flood events and indicated that the LSTM model outperformed conceptual and physical models. Yan *et al.* (2019) constructed an LSTM with historical flow and weather data and weather forecasts and indicated that the LSTM outperforms support vector machines in flood predictions, especially for flood peak flow forecasts. Karimi *et al.* (2019) compared three data-driven methods, i.e., ANN, LSTM and Least Absolute Shrinkage and Selection Operator (LASSO) for flood flow predictions in sewer systems, and concluded that all three models provide acceptable prediction performance, but LSTM outperforms ANN due to the inherent memory integrated with a feedback structure. Muhammad *et al.* (2019) proposed a hybrid model by combining LSTM and GRU for river flow simulations, which was used for early flood warning. Hu *et al.* (2019) integrated an LSTM and reduced order model to represent the spatial–temporal distribution of floods. Le *et al.* (2019) used the LSTM in modelling 1-, 2- and 3-day flood events in Vietnam's Da River basin. In summary, previous research has shown the ability of LSTM in river flow predictions. However, there is a lack of understanding on how LSTM structures and parameters are linked to predictive accuracy in hydrological modelling.

The main aim of this paper is to assess the performances of LSTM networks in river flow predictions in terms of LSTM structures and parameters. In this study, the LSTM networks with different network structures, i.e., fully connected layers and LSTM cells are trained and their performances compared using two case studies of different characteristics – the Hun river basin and the upper river basin of Yangtze River, China. The trained LSTM networks are used to predict the river flows in the two case study river basins. Finally, the LSTM networks are compared with four models, i.e., the SWAT, Xinanjiang model (XAJ), multiple linear regression model (MLR) and back-propagation neural networks (BP).

METHODOLOGY

In this section, the LSTM network for flow simulation and predication is first presented and the key components including the network structure, LSTM cells and loss function are explained. Then the data pre-processing and evaluation criteria used in this study are introduced. Finally, different simulation scenarios designed to study the performance of the LSTM network are explained.

LSTM network

Network structure

In the LSTM network, the key components are fully connected layers and LSTM cells. As shown in Figure 1, the LSTM network contains four types of layers: (a) the input layer which receives the input sequence data; (b) the fully connected layer *a* which transfers the dimensions of the input data into the dimensions of LSTM cells and

establishes a bridge between the input layer and the LSTM cell layer; (c) the LSTM cell layer of *n* cells (i.e., simple networks) which provides different memory abilities; (d) output layers including fully connected layers *b1*, *b2* and the output flow vector, which transfer the outputs of LSTM cells to flows.

In the LSTM network training, a batch of training samples is used for each learning process (Kratzert et al. 2018). The sequence precipitation data of the meteorological stations in the watershed are formulated as an input matrix

$$[\bar{x}_t, \bar{x}_{t+1}, \dots, \bar{x}_{t+T}] = \begin{bmatrix} x_t^1 & x_{t+1}^1 & \dots & x_{t+T}^1 \\ x_t^2 & x_{t+1}^2 & \dots & x_{t+T}^2 \\ \dots & \dots & \dots & \dots \\ x_t^m & x_{t+1}^m & \dots & x_{t+T}^m \end{bmatrix} \quad (1)$$

where *m* is the number of meteorological stations; *T* is the batch size of precipitation data; *t* is the start time step; \bar{x}_t is the precipitation vector at time step *t* and represented as $\bar{x}_t = [x_t^1, x_t^2, \dots, x_t^m]$.

The observed flow data at hydrological stations are used as targets for training, i.e., to compare with the simulated flows from the LSTM network

$$[\bar{q}_t, \bar{q}_{t+1}, \dots, \bar{q}_{t+T}] = \begin{bmatrix} q_t^1 & q_{t+1}^1 & \dots & q_{t+T}^1 \\ q_t^2 & q_{t+1}^2 & \dots & q_{t+T}^2 \\ \dots & \dots & \dots & \dots \\ q_t^g & q_{t+1}^g & \dots & q_{t+T}^g \end{bmatrix} \quad (2)$$

where *g* is the number of hydrological stations; q_t^g is the observed flow of the *g*th hydrological station at time step *t*; \bar{q}_t is the flow vector at time step *t*.

During the training processes of the LSTM network at a time step, as shown in Figure 1, the fully connected layer *a*

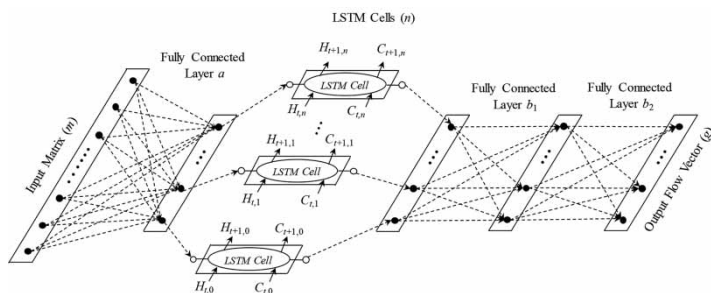


Figure 1 | The structure of the LSTM network.

transfers the precipitation vector (e.g., \bar{x}_t) with m dimensions into n dimensions (i.e., n is the number of LSTM cells) as

$$y_{out}[n] = \bar{x}[m] \times W[m, n] + b[n] \tag{3}$$

where y_{out} is the output vector of layer a ; W and b are the weight matrix and bias, respectively; n is the total number of the LSTM cells.

After recurrent learning of the LSTM cells, an output of n dimensions is generated and sent to the fully connected layer $b1$. The fully connected layers $b1$ and $b2$ are neural layers with activation functions and used to transfer the LSTM cell output to flow as:

$$y_{out}^f[k] = f_{ReLU}(\bar{x}_{in}[h] \times W[h, k] + b[k]) \tag{4}$$

where \bar{x}_{in} is the input vector of fully connected layer with h dimensions; in the fully connected layers $b1$, the input vector should be the output of the LSTM cells with n dimensions (i.e., $h = n$). y_{out}^f is the output vector of layer $b1$, which is transferred by activation function f_{ReLU} . k is the dimension of the output vector. The output flow vector is the output of the final layer, and k should be the number of hydrological stations (i.e., $k = g$). f_{ReLU} represents the ReLU activation function which was chosen according to preliminary analysis and suggestions from the literature (Khan et al. 2019).

LSTM cell structure

Figure 2 shows the structure of the LSTM cell. There are two key states in LSTM cell calculation, i.e., cell state and hidden state. In Figure 2, C_{t-1} and H_{t-1} represent the cell state and hidden state at time step $t - 1$, respectively. The cell state is the main chain of the data stream, which allows the data to flow forward substantially unchanged. However, the data in the hidden state can be added or removed from the cell state (Le et al. 2019), which is carefully controlled by ‘forget gate’, ‘input gate’ and ‘output gate’, represented by the dashed boxes in Figure 2. The gates are neural network layers with a series of matrix operations, which contain different individual weights and biases. The LSTM cell uses gates to control the memory process to avoid the long-term dependency problem (Hochreiter & Schmidhuber 1997). The cell

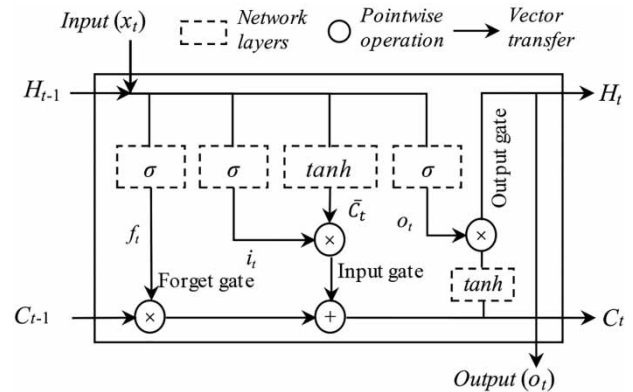


Figure 2 | The structure of the LSTM cell.

learns from time series data using five simple network layers, including three sigmoid layers and two tanh network layers (Le et al. 2019).

The ‘forget gate’ determines what information in the hidden state is forgotten, shown as the f_t process in Figure 2. By the forget gate, the meteorological precipitation information of the past time steps can be recalled at the current time step as:

$$f_t = \sigma(W_f \cdot [H_{t-1}, x_t] + b_f) \tag{5}$$

where σ represents the sigmoid network layer where the sigmoid function is used as the activation function; x_t is the input data; W_f and b_f are weight matrix and bias in the sigmoid network layer, respectively; f_t is the forget vector with values in the range $[0, 1]$, where 1 means ‘completely reserved’ and 0 means ‘completely forgotten’.

The ‘input gate’ determines what information in the cell state C_t is to be updated by x_t and H_{t-1} . There are a sigmoid layer and a tanh layer at this gate. The tanh layer is expressed by the output weights as a one-dimensional matrix, which determines how the information of the cell state is to be updated according to x_t and H_{t-1} as:

$$\bar{C}_t = \tanh(W_C \cdot [H_{t-1}, x_t] + b_C) \tag{6}$$

where \bar{C}_t is a one-dimensional matrix with values in the range $[0,1]$; W_C and b_C are the weight matrix and bias in the tanh network layer in the ‘input gate’.

The sigmoid layer in the ‘input gate’ determines the information in the hidden state to participate in the

update, which operates as:

$$i_t = \sigma(W_i \cdot [H_{t-1}, x_t] + b_i) \quad (7)$$

where i_t is a one-dimensional matrix with values in the range [0,1]; W_i and b_i are the weight matrix and bias in the sigmoid network layer in the 'input gate'.

Combining the outputs from 'forget gate' and 'input gate', the information in the cell state C_t can now be updated by:

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

where the first component represents the passthrough information from the forget gate and the second component represents the update information from the input gate. In this way, the impact of precipitation from previous times on the runoff at the current time step can be learned.

The 'output gate' uses the sigmoid layer to determine which information of the hidden state is taken as the output.

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

where W_o and b_o are the weight matrix and bias in the output gate; o_t is the output of the LSTM cell; the hidden state H_t can be determined based on the output of cell and

hidden state C_t .

$$H_t = o_t \times \tanh(C_t) \quad (10)$$

Loss function

In this study, observed flow data of the hydrological station are used as targets to evaluate the loss values of the simulated flows by LSTM network, and the Adam algorithm which was proposed by Kingma & Ba (2014) is applied to optimize and update the network weights. The loss values are estimated below using the difference between network outputs and target values

$$\text{Loss}(y, z) = \sum_{t=1}^T \text{MSE}(y_{\text{out},t}^f, \bar{q}_t) \quad (11)$$

where T is the batch size of training samples for each training; \bar{q}_t is the target value at time step t ; $y_{\text{out},t}^f$ is the LSTM network output; MSE is the mean square error.

Pseudo code of LSTM network

The pseudo code of the LSTM network training is shown in Algorithm 1. In the pseudo code, the parameters include the

Algorithm 1 | The pseudo code of the LSTM neural network.

Input:

Input data matrix: $[T, m]$.

Target data matrix: $[T, g]$.

Initial Parameters:

(1) Fully connected layers: Weights and Bias.

(2) The number of LSTM cells is n ; Set initial states of $C_{k,0}$ and $S_{k,0}$ ($k \in n$) to zero matrix.

Fully Connected Layer:

Transforms input matrix $[T, m]$ to $[T, n]$ using Equation (3);

LSTM Cells:

For $t = 0$ in length (T):

For $k = 0$ in length (n):

Update states for LSTM cell: C_t and H_t using Equations (8) and (10), separately;

Generate cell output: $o_{k,t}$ using Equation (9).

Get the outputs of LSTM cells after the iteration of the loop: $[T, n]$.

Fully Connected Layers:

Get the outputs $y_{\text{out},t}^f$: Transform matrix $[T, n]$ to $[T, g]$ using fully connected layers b_1 and b_2 .

Loss Function:

Comparing the simulated flows ($y_{\text{out},t}^f$) and observed flows (\bar{q}_t) and, the loss value is evaluated using Equation (11).

Weights Updating:

Based on the loss value, the weights of the networks are updated using the Adam algorithm.

batch size of training input data (T), the dimensions of the input data (m), the dimensions of the output flow vector (g), LSTM cell size (n) and the input and output dimensions of the fully connected layers (i.e., h and k). In the case study, the LSTM network training ends after 1,500 epochs.

Structure scenarios

In this section, the scenarios of different LSTM structures and parameters are presented to test the network performances as shown in Table 1. The LSTM network structure in Figure 1 is taken as scenario A1 and used as a benchmark for comparison of the other network structures. Building on scenario A1, Scenario A2 has a fully connected neural layer with an activation function added between the input layer and the fully connected layer a . Scenario A3 is developed by removing the activation functions in the fully connected layers $b1$ and $b2$ based on A1.

Comparing the performances between scenario A1 and A3 allows us to analyse the impact of activation functions which establish a non-linear transformation between the LSTM cells and the output runoff vector. To evaluate the impacts of the number of layers on the learning efficiency of LSTM, scenarios B1 and B2 are constructed by adding one and two fully connected layers to the benchmark A1, respectively.

The batch sizes of training samples (T) and number of LSTM cell (n) are important network parameters, which determine the learning efficiency of the LSTM network. Table 1 shows their values tested in this study.

Streamflow data pre-process

In a large river basin with multiple flow stations, the flow rates at different stations may vary in a wide range due to different sizes of drainage areas. The difference may cause the network to ignore small flows, leading to learning inefficiency or failure. Thus, the flow processes for each hydrological station are pre-processed as:

$$q'_{t,i} = \frac{q_{t,i}}{\bar{q}_i} \quad i \in [1, g]; t \in [1, N], \tag{12}$$

where $q_{t,i}$ represents the observed flow of the hydrological station i at time step t . \bar{q}_i represents the mean value of the observed flow process of the hydrological station i . $q'_{t,i}$ represents the pre-processed flow; g is the number of hydrological stations; N represents the length of the flow data.

Model evaluation criteria

In this study, the simulation performances of the models are evaluated by the following three criteria. The coefficient of determination (R^2) provides a statistical measure that assesses how well a hydrological model explains and predicts future flows, and it indicates the level of explained variability in the data set. The Nash–Sutcliffe Efficiency (NSE) is used to quantitatively describe the accuracy of the hydrological model (Nash & Sutcliffe 1970). The NSE value is between 1 and negative infinity. An NSE value of 1 corresponds to a perfect match of simulated flows to observed data. The

Table 1 | The four scenarios of the CNN convolutional layers

Scenarios	Structures		
A1	$P[T,m] \rightarrow FC[m,n] \rightarrow Cell[n] \rightarrow FCa[n,50] \rightarrow FCa[50,30] \rightarrow O[T,g]$		
A2	$P[T,m] \rightarrow FCa[m,n] \rightarrow FC[m,m] \rightarrow Cell[n] \rightarrow FCa[n,50] \rightarrow FCa[50,30] \rightarrow O[T,g]$		
A3	$P[T,m] \rightarrow FC[m,n] \rightarrow Cell[n] \rightarrow FC[n,50] \rightarrow FC[50,30] \rightarrow O[T,g]$		
B1	$P[T,m] \rightarrow FC[m,n] \rightarrow Cell[n] \rightarrow FCa[n,50] \rightarrow FCa[50,100] \rightarrow FCa[100,30] \rightarrow O[T,g]$		
B2	$P[T,m] \rightarrow FC[m,n] \rightarrow Cell[n] \rightarrow FCa[n,50] \rightarrow FCa[50,100] \rightarrow FCa[100,50] \rightarrow FCa[50,30] \rightarrow O[T,g]$		
P	Input matrix	T	Batch size of precipitation data
O	Output flow vector	M	Number of meteorological stations
FC	Fully connected layer without activation function	N	Number of LSTM cells
FCa	Fully connected layer with ReLU activation function	g	Number of hydrological stations
$Cell$	LSTM cells		

relative error (RE) represents the degree of difference between the observed and simulated values (Kan et al. 2015).

$$R^2 = \left(\frac{\sum_{t=1}^N [(q_{o,t} - \bar{q}_o) \times (q_{s,t} - \bar{q}_s)]}{\sqrt{\sum_{t=1}^N (q_{o,t} - \bar{q}_o)^2} \times \sqrt{\sum_{t=1}^N (q_{s,t} - \bar{q}_s)^2}} \right)^2 \times 100\% \quad (13)$$

$$NSE = \left(1 - \frac{\sum_{t=1}^N (q_{o,t} - q_{s,t})^2}{\sum_{t=1}^N (q_{o,t} - \bar{q}_o)^2} \right) \times 100\% \quad (14)$$

$$RE = \left(\frac{\sum_{t=1}^N (q_{o,t} - q_{s,t})}{\sum_{t=1}^N q_{o,t}} \right) \times 100\% \quad (15)$$

where $q_{s,t}$ and $q_{o,t}$ represent the simulated and observed flow at time step t , respectively. \bar{q}_s and \bar{q}_o represent the means of the simulated and observed flows, respectively.

CASE STUDY

Study area

In this study, two basins are taken as case studies, the Hun river basin and the upper Yangtze river basin, which are located in the northeast and southwest of China, respectively (Figure 3). The basic characteristics of the two basins are shown in Table 2.

The Hun river originates from the Changbai Mountain. The precipitation is affected by temperate monsoon continental climate. The vegetation in this basin is well maintained and is rarely interrupted by human activities. 70% of the annual precipitation occurs from June to September.

The Upper Yangtze river originates from the Qinghai-Tibet Plateau. The drainage area above the Three Gorges dam in the Yangtze River is the upper reach. The precipitation of the basin is affected by the topography and the monsoon, which makes the annual precipitation unevenly distributed both spatially and temporally. The precipitation

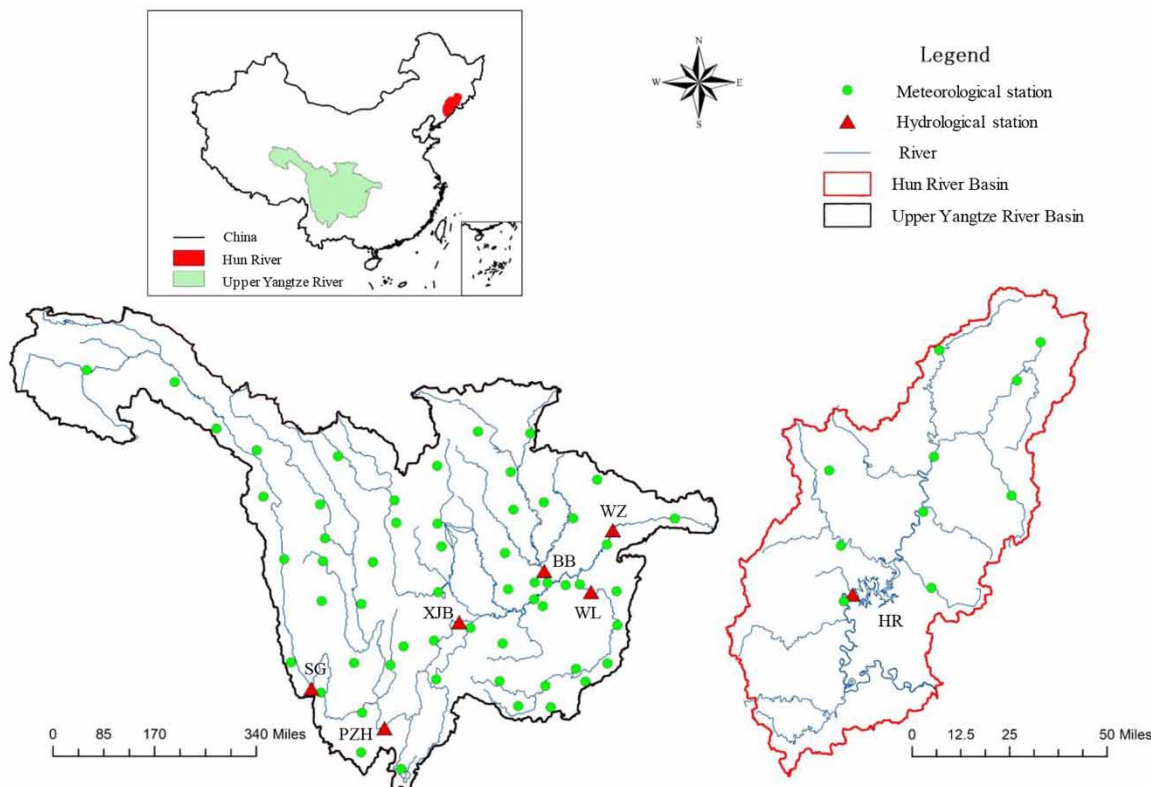


Figure 3 | Location map for the Hun river and Upper Yangtze river.

Table 2 | The basic characteristics of the two study basins

Characteristic	Upper Yangtze river	Hun river
Location	106°14'~111°28' E, 28°16'~31°44' N	124°43'~126°50' E, 40°40'~42°15' N
Area (10 ⁴ km ²)	100.23	1.48
Channel length (km)	4,000	435
Mean annual rainfall (mm)	1,250	860
Mean annual streamflow (m ³ /s)	14,300	227

increases from northwest to southeast, and about 70% of the precipitation occurs from May to September.

Data source

In the Hun river, the 10-day average meteorological data from 10 stations (i.e., $m = 10$) and river flow data from the Huanren station (HR) (i.e., $g = 1$) from 1970 to 2010 were obtained. The 30-year data from 1970 to 1999 are used to train LSTM networks and calibrate the comparison models which are introduced in Section Comparison models. The data from 2000 to 2010 are used to verify the performances of the models.

In the Upper Yangtze river, the daily meteorological data were obtained from 1991 to 1998, including 57 meteorological stations (i.e., $m = 57$). The observed daily streamflow from six hydrological stations (i.e., $g = 6$) as shown in Figure 3 was obtained from 1991 to 1998, i.e., Shi gu (SG), Pan zhi hua (PZH), Bei bei (BB), Wu long (WL), Xiang jia ba (XJB) and Wang zhou (WZ). The data from 1991 to 1995 are used for model training and calibration, and the data from 1996 to 1998 are used for verification.

Comparison models

In the Hun river basin case study, four models are constructed as comparison models to evaluate the performance of the proposed LSTM network, i.e., SWAT, XAJ, MLR and BP. In the upper Yangtze river basin, SWAT is used only.

SWAT model

The SWAT model is a distributed hydrological model developed by the U.S. Department of Agriculture and the

Agricultural Research Service (Guzman et al. 2015). The SWAT model has strong physical mechanisms. In this study, the DEM raster data with a resolution of 90 m and the land use raster data with a resolution of 3 km were obtained in Hun river basin. The DEM and land use data of 3 km × 3 km in the Upper Yangtze river basin were obtained. In this study, the SWAT model is calibrated using the Calibration Uncertainty Procedure (SWAT-CUP) program, which is an auto-calibration tool that allows for sensitivity analysis, calibration, validation and uncertainty analysis of the SWAT model (Abbaspour 2011; Abbaspour et al. 2015). Finally, the key parameters for the two basins are listed in Table 3.

XAJ model

The XAJ model is a conceptual rainfall-runoff model, which was developed by Zhao (1992). This model has been widely used in China, particularly in humid and semi-humid regions (Xu et al. 2014). The XAJ model assumes that runoff is not produced until the soil water content of the aeration zone reaches its field capacity. The actual evapotranspiration is

Table 3 | Key parameters of SWAT for Hun and Upper Yangtze river basins

Parameters	Definition	Hun river	Upper Yangtze river
CN2	SCS runoff curve number for moisture condition II	72	35
ALPHA_BF	Base flow recession constant (days)	0.8	0.61
GW_DELAY	Delay time for aquifer recharge (days)	31	15
CH_N2	Manning's n value for the main and tributary channels	0.1	0.3
CH_K2	Effective hydraulic conductivity of channel (mm/hr)	5	20
SOL_AWC	Available water capacity (mm/mm)	0.005	0.004
SMTMP	Snow melt minimum temperature (°C)	-1	-2
CANMX	Canopy maximum storage (mm)	4	9
ESCO	Soil evaporation compensation factor (mm)	0.95	0.6
REVAPMN	Threshold depth for evaporation to occur (mm)	71	120

computed from potential evapotranspiration while soil storage deficit is calculated in three layers, i.e., upper, lower and deep soil layers. The XAJ model parameters of the Hun

Table 4 | Parameters of XAJ model

Parameters	Physical meaning (unit)	Value
U_m	Tension water capacity of upper layer (mm)	5
L_m	Tension water capacity of lower layer (mm)	50
D_m	Tension water capacity of deep layer (mm)	10
IMP	Ratio of impervious area to the total catchment area (%)	0.05
C	Evapotranspiration coefficient of the deeper layer (-)	0.1
B	Exponential of the distribution of tension water capacity (-)	0.36
S_m	Free water capacity (mm)	5
E_x	Exponent of the distribution of free water capacity (-)	1.4
kg	Outflow coefficient of the free water storage to groundwater (-)	0.2
k_i	Outflow coefficient of the free water storage to interflow (-)	0.5

river basin have been calibrated using a Genetic Algorithm (Xu & Peng, 2016). In calibration, NSE and RE are used to evaluate the performance of the parameters as shown in Equations (14) and (15), respectively.

The calibrated parameters of the XAJ model are shown in Table 4. In this model, the surface runoff, interflow and groundwater are routed using instantaneous unit lines, and the parameters of the three lines, i.e., (n and k) are set to (3, 4), (4, 5) and (4.5, 7), respectively.

MLR model

In the MLR model, one regression model was developed for each time step to identify the factors of physical and statistical significance in the Hun river basin. There are 36 time steps in a year, i.e., 10 days for each time step. In the MLR models, the least square method is used to estimate the parameters of the regression equations based on the observed runoffs and factors, and the equations are shown in Table 5.

Table 5 | Equations of different time steps in the MLR model

Time step	Equation	Time step	Equation
1	$F_t = 3.5 + 0.55Q_{t-1}$	19	$F_t = -250 + 0.05Q_{t-1} + 0.31P_{t-1} + 0.48P_t$
2	$F_t = 2.75 + 0.65Q_{t-1}$	20	$F_t = -50 + 0.33Q_{t-1} - 0.04P_{t-1} + 0.43P_t$
3	$F_t = 1.94 + 0.33(Q_{t-1} + Q_{t-2})$	21	$F_t = -70 + 0.29Q_{t-1} + 0.06P_{t-1} + 0.39P_t$
4	$F_t = 2.97 + 0.66Q_{t-1}$	22	$F_t = -25 + 0.25Q_{t-1} + 0.09P_{t-1} + 0.26P_t$
5	$F_t = 1.06 + 0.45(Q_{t-1} + Q_{t-2})$	23	$F_t = 13 + 0.49Q_{t-1} + 0.01P_{t-1} + 0.09P_t$
6	$F_t = 4.47 + 0.68Q_{t-1}$	24	$F_t = -5.0 + 0.7Q_{t-1} + 0.01P_{t-1} + 0.08P_t$
7	$F_t = 10.50 + 0.49Q_{t-1}$	25	$F_t = -7 + 0.51Q_{t-1} + 0.05P_{t-1} + 0.17P_t$
8	$F_t = 17.10 + 1.18Q_{t-1}$	26	$F_t = -3 + 0.67Q_{t-1} + 0.05P_{t-1} + 0.04P_t$
9	$F_t = 35.0 + 1.0Q_{t-1}$	27	$F_t = -12 + 0.73Q_{t-1} + 0.03P_{t-1} + 0.13P_t$
10	$F_t = -20.0 + 0.6Q_{t-1} + 0.1P_{t-1} + 0.1P_t$	28	$F_t = 60.0 + 0.75Q_{t-1}$
11	$F_t = 3.0 + 0.44Q_{t-1} + 0.1P_{t-1} + 0.15P_t$	29	$F_t = 90.0 + 0.05(Q_{t-1} + Q_{t-2}) + 0.4P_{t-1}$
12	$F_t = -10 + 0.37Q_{t-1} + 0.07P_{t-1} + 0.19P_t$	30	$F_t = 50.0 + 0.35(Q_{t-1} + Q_{t-2}) + 0.2P_{t-1}$
13	$F_t = -10 + 0.4Q_{t-1} + 0.16P_{t-1} + 0.04P_t$	31	$F_t = 2.50 + 0.48Q_{t-1} + 0.13P_{t-1}$
14	$F_t = -40 + 0.56Q_{t-1} + 0.08P_{t-1} + 0.14P_t$	32	$F_t = -1.0 + 0.90Q_{t-1}$
15	$F_t = -90 + 0.32Q_{t-1} + 0.11P_{t-1} + 0.28P_t$	33	$F_t = 11.0 + 0.48Q_{t-1}$
16	$F_t = -45 + 0.28Q_{t-1} + 0.14P_{t-1} + 0.19P_t$	34	$F_t = 10.0 + 0.44Q_{t-1}$
17	$F_t = -40 + 0.35Q_{t-1} + 0.06P_{t-1} + 0.25P_t$	35	$F_t = 3.5 + 0.65Q_{t-1}$
18	$F_t = -270 + 0.54Q_{t-1} + 0.07P_{t-1} + 0.55P_t$	36	$F_t = 3.5 + 0.65Q_{t-1}$

F_t is the simulated runoff at the time step t ; Q_{t-1} is the observed runoff at time step $t - 1$; P_t is the average precipitation at time step t .

BP neural network

The BP model takes the flows at time step $t - 2$ and $t - 1$ and the average precipitation of the watershed at time step $t - 2$, $t - 1$ and t as inputs and takes the flow at time step t as output. The BP model constructed using four-layer neural network, and the network nodes for each layer are 5 (input layer), 50 (hidden layer), 50 (hidden layer) and 1 (output layer), respectively. The outputs of the hidden layer are transformed by the sigmoid activation function. The BP network was trained for 900 epochs as it was already converged.

RESULTS

The network structure and parameters have a great influence on the learning efficiency. In this study, the LSTM networks are tested on the Hun and Upper Yangtze river basins, respectively. First, the effects of LSTM network structure are analysed and the performances of the network parameters are evaluated in terms of the number of cells (n) and the batch size (T). Then, the structure and parameters with the best performance are selected to predict the river flows of the two study cases. Finally, the performances of the LSTMs are compared with the results from the comparison models.

Learning efficiency with different structures

Effects of activation function

Scenarios A1, A2 and A3 are trained for the Hun river and Upper Yangtze river case studies, and the variations of the

loss function values are shown in Figure 4. For each network structure, the network weights are trained for 1,500 epochs. The training loss values of each scenario in Figure 4 are the average values of 10 independent training runs. Note that the model parameter values used for the results in Figure 4 are those optimal values identified in Table 6. The results from the two case studies in Figure 4 show the following key points:

- (1) The loss value of A1 is rapidly decreased, demonstrating a rapid learning.
- (2) The inputs in A2 are nonlinearly transferred by activation functions before being passed to the LSTM cells; as a result, the LSTM cells cannot effectively capture the long-term time dependencies in the time series data. Thus, the loss values cannot be reduced rapidly during the training.
- (3) In A3, there is only a simple linear transformation between the LSTM cells and output layer. This makes learning difficult with a slow reduction in loss values before they start to increase after 500 epochs.

The results from Figure 4 show the LSTM structures in Scenarios A2 and A3 could not provide efficient learning and the model outputs cannot match the target values well.

Effects of fully connected layers

The test results from Scenarios B1 and B2 are shown in Figure 5. The results indicate that all the scenarios can learn effectively in the two case studies. However, the detailed loss value variations between epochs 1,250 and

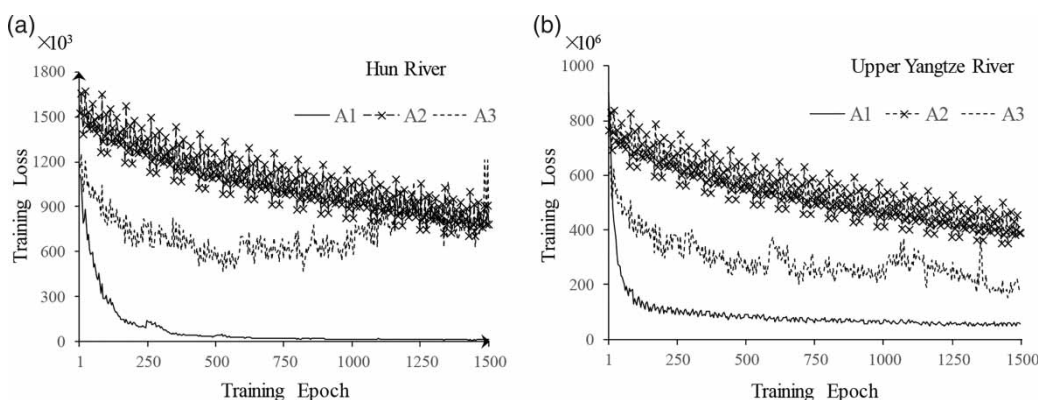


Figure 4 | The training loss variations of different LSTM network structures.

Table 6 | The structure and parameters of the LSTM network for the study cases

Watershed	Structures and parameters
Hun river	$P[T=90, m=10] \rightarrow FC[m, n] \rightarrow Cell[n=20] \rightarrow FCa[n, 50] \rightarrow FCa[50, 30] \rightarrow O[T, g=1]$
Upper Yangtze river	$P[T=360, m=57] \rightarrow FC[m, n] \rightarrow Cell[n=50] \rightarrow FCa[n, 50] \rightarrow FCa[50, 30] \rightarrow O[T, g=6]$

1,500 as shown in Figure 5(a2,b2) reveals that scenario A1 converges faster and is more stable than the other two scenarios. This implies that the network does not need a large number of fully connected layers between the LSTM cells and the output layer to improve the learning efficiency. Therefore, the LSTM structure in A1 is selected to predict the river flows in the two case study basins.

Learning efficiency with different parameters

Effects of T

In the LSTM network training, the batch size (T) of training samples has a significant effect on the learning efficiency. In this study, a number of T values are tested for Hun river (i.e., 10, 30, 50, 70 and 90) and Upper Yangtze river (i.e., 30, 60, 120, 180 and 360). Figure 6 shows the training loss variations of the Hun river and Upper Yangtze river. The loss values fluctuate dramatically when the batch size is small, i.e., $T=10$ or 30 in the case of Hun river and $T=30, 60$ or 120 in the case of Upper Yangtze river. This indicates that the LSTM cells cannot capture the periodicity in the time series using small batch sizes, when the network weights are updated frequently. With T increasing, the amount of training samples used for

learning can gradually reflect the periodicity. Therefore, the fluctuation of the loss values is gradually reduced. Figure 6(a2,b2) shows the magnified training loss curves during epochs from 1,250 to 1,500. The results indicate that the learning curves are stable after $T=50$ and 180 for the Hun river and Upper Yangtze river basins, respectively.

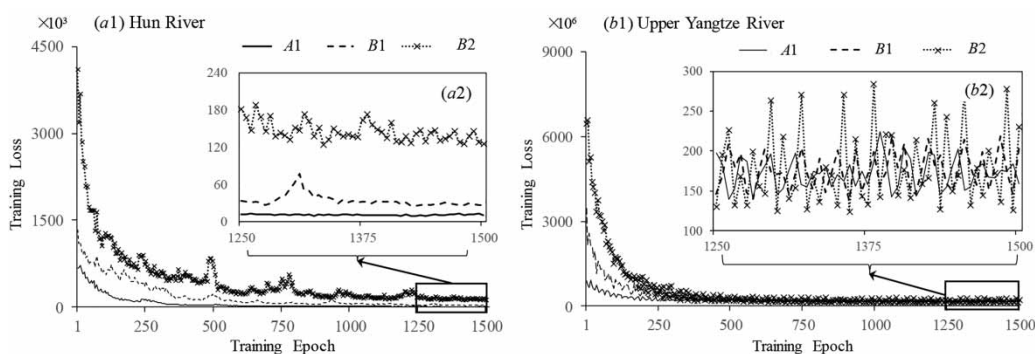
Figure 7 shows the means of the loss values from epochs from 1,250 to 1,500 in Figure 6. The results show that the loss means are gradually decreasing with T increasing, which represent the learning efficiencies of the LSTM network are gradually improved. In the case of Upper Yangtze river, when $T > 180$, the loss value cannot be reduced.

Effects of cell number

The LSTM cell is the core concept in the LSTM network. The number of the cells (n) determines the performances of the network. Figure 8 shows the means of the loss values from epochs 1,250 to 1,500 using different cell numbers. With the Hun river basin, when the number of cells reaches 20, LSTM achieves a good learning efficiency as shown in Figure 8(a). When the cell number exceeds 20, the efficiency shows very slow improvement. Figure 8(b) reflects the network performances in the Upper Yangtze river. When the number of cells reaches 40, the learning efficiency begins to stabilize. In this study, 20 and 50 cells are used in the LSTM networks for the Hun river and the Upper Yangtze river, respectively.

Performance evaluation

Based on the analysis of the structure and parameters of the LSTM network as above, the structure and parameters with

**Figure 5** | The training loss variations for the different number of fully connected layers.

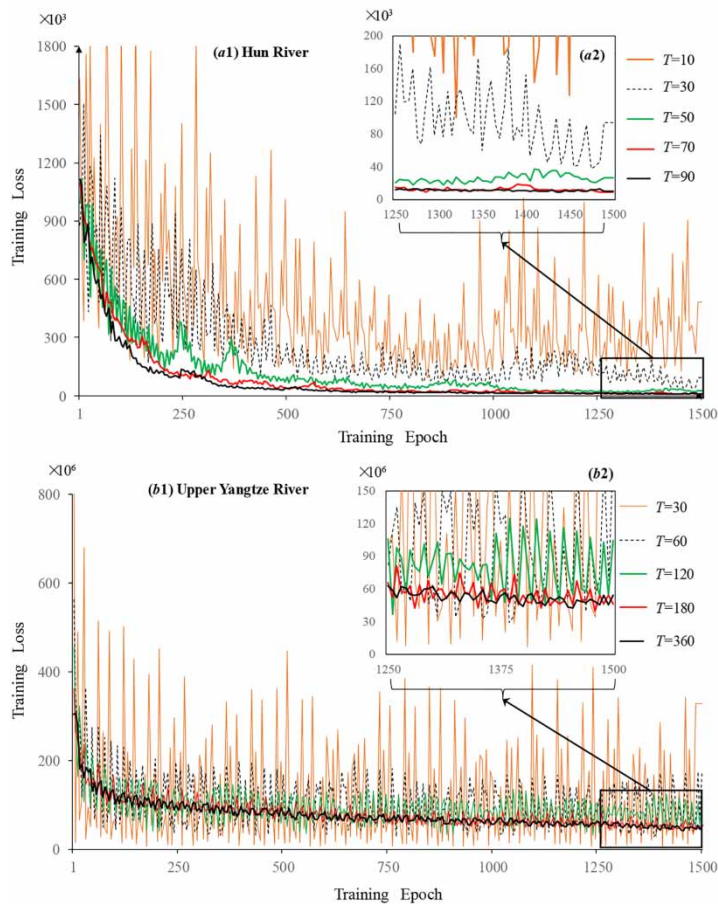


Figure 6 | The training loss variations of the two study basins with different T scenarios.

the best performance as shown in Table 6 are selected to learn and predict the flows of the Hun river and Upper Yangtze river, respectively.

Hun river basin

Figure 9(a) shows the observed and simulated flows from 1970 to 1999, and the performance evaluation criteria are shown in Table 7. The simulated flows in Figure 9(a) fit well with the observed flows. In training, the LSTM achieves an NSE value of 0.98. The results show that LSTM has a strong non-linear learning ability and it outperforms the other models in the Hun river basin. During the models, the MLR model performs worst.

In the verifying process, the predictive ability of the LSTM is shown in Table 7 to be slightly worse than that of the SWAT. However, the LSTM slightly outperforms the

XAJ model in terms of NSE but is much better than other models. Figure 9(b) shows the predicted and observed flows from 2000 to 2010. Though the predicted flows from the LSTM did not match well with the observed in some periods, most peak flows are predicted well. This is clearly shown in the scatter plots of the observed and simulated flows from the training and verification periods in Figure 10. Though the overall performances of SWAT and XAJ in terms of the three criteria are better than those of LSTM, LSTM performs much better for peak flows, which are of particular concern in flood predictions.

Upper Yangtze river

In the Upper Yangtze river basin, the daily data of 57 meteorological stations are used as inputs, and the daily flow of six hydrological stations is used as target values.

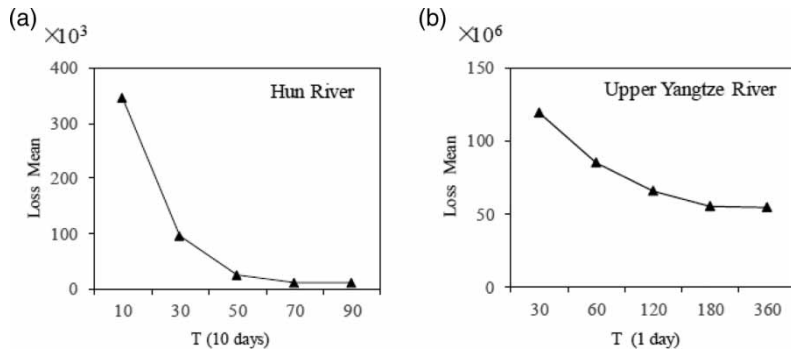


Figure 7 | The mean loss values of epochs 1,250–1,500 for the five T scenarios.

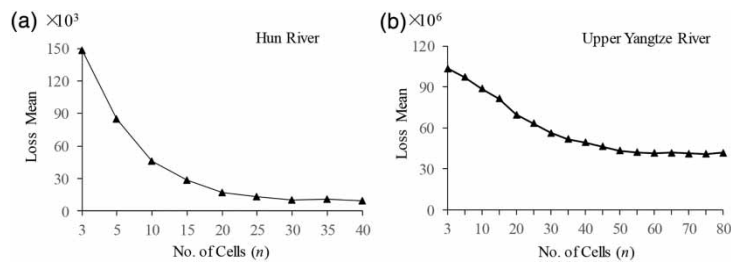


Figure 8 | The means of the loss values from epochs 1,250 to 1,500 for different cell numbers.

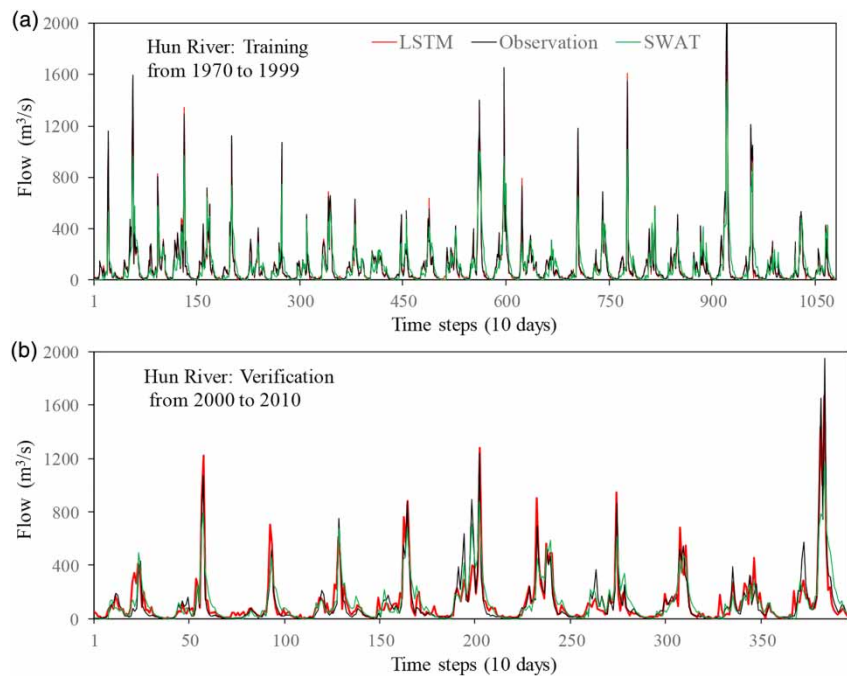


Figure 9 | The observed and estimated flows from LSTM network and SWAT model.

Table 7 | Performances evaluation criteria of the five models in Hun river

Models	Training			Verification		
	NSE (%)	R ² (%)	RE (%)	NSE (%)	R ² (%)	RE (%)
SWAT	84.55	85.96	3.74	79.43	80.87	-4.79
XAJ	78.56	78.56	0.62	73.63	75.82	3.38
MLR	54.53	54.56	-3.08	34.98	36.31	-0.44
BP	85.47	85.48	0.30	68.93	69.26	6.42
LSTM	98.21	98.31	1.62	74.76	75.14	-4.06

The flows of the six stations are simulated using LSTM and SWAT and their performances are shown in Table 8.

In the training, LSTM has a very high performance for the flows of six stations. The NSE and R² values indicate that the LSTM outperforms the SWAT during training. Figure 11 shows the simulated flows at a station (WZ).

The scatter plots of simulated and predicted flows for six hydrological stations are shown in Figure 12. The results indicate that the performance of LSTM in the verifying period is worse than that in the training period. Predicted peak flows are likely to be lower than those observed.

Note that in Upper Yangtze river, the LSTM network is constructed to predict flows at multiple stations at the same time. The training results show that the LSTM network has a strong fitting ability to learn the flow data of multiple hydrological stations.

DISCUSSION

LSTM network training is significantly affected by the training dataset size. It is generally understood that the network training requires a large amount of training sample data. However, the dataset size depends on the characteristics of the catchment and flows of concern, which determines the complexity of the input-output relationships represented by the LSTM. The LSTM has been shown to perform well with a smaller dataset than 30 years used in the Hun river basin. For example, Kratzert et al. (2018) used the daily meteorological data and observed flow data from 15 years in 241 catchments to train LSTM networks, whose performances are comparable to process-based models. Lee et al.

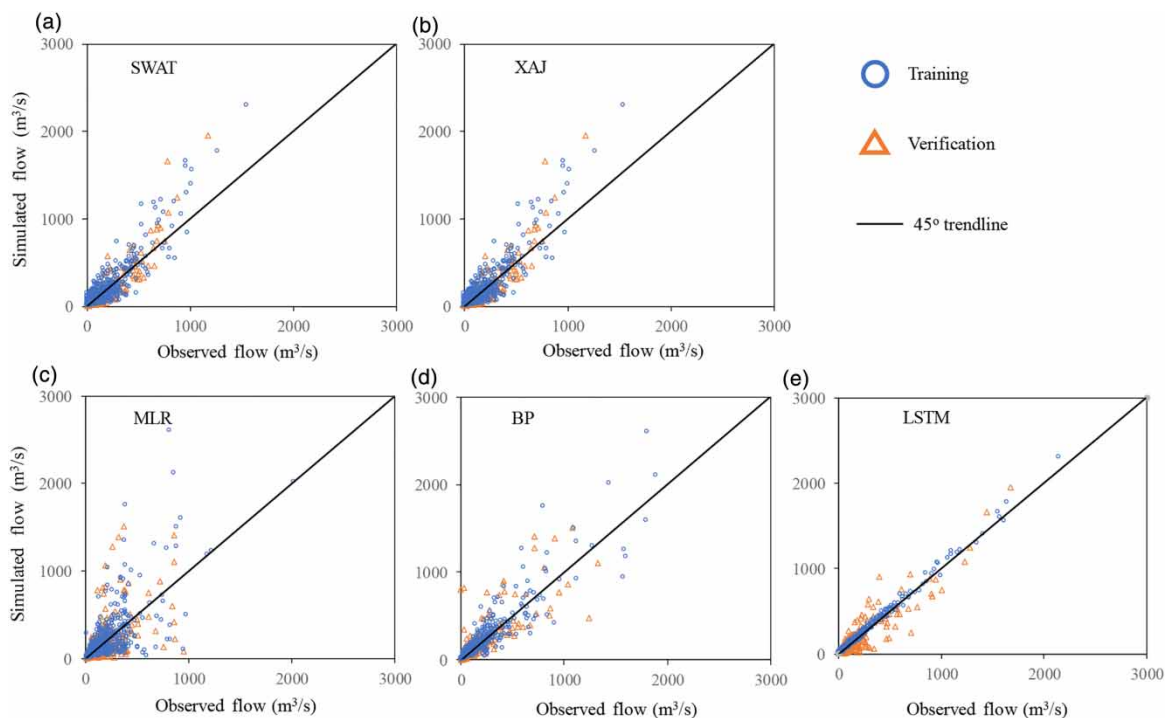
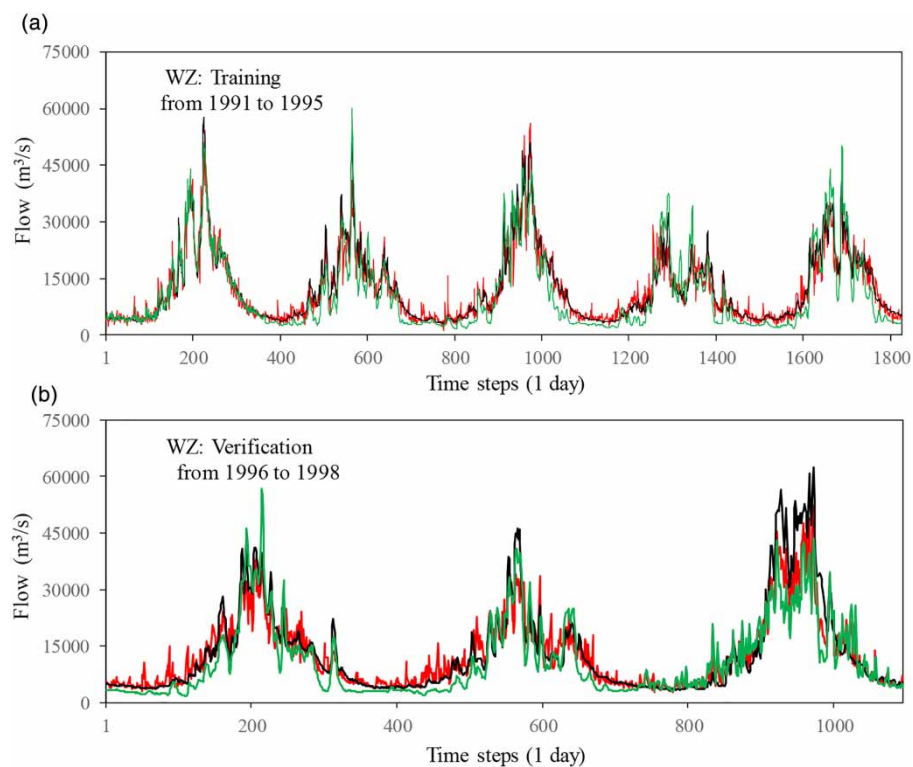
**Figure 10** | The scatter plots of the observed and simulated flows of the five models.

Table 8 | Performance evaluation criteria of LSTM and SWAT in Upper Yangtze river

Criteria	Category	Yangtze River					
		SG	PZH	BB	WL	XJB	WZ
		LSTM network					
Training	NSE (%)	89.63	90.77	91.09	90.96	89.84	90.35
	R^2 (%)	89.73	90.89	91.11	91.01	89.96	90.41
	RE (%)	2.15	2.17	1.17	0.05	2.33	1.65
Verification	NSE (%)	54.60	54.13	56.56	61.62	62.48	71.51
	R^2 (%)	64.92	61.6	58.13	69.79	64.11	72.72
	RE (%)	0.94	6.10	-12.71	11.18	8.48	1.85
		SWAT model					
Training	NSE (%)	77.05	72.56	70.67	76.95	79.95	84.31
	R^2 (%)	77.89	80.41	75.65	78.93	81.55	88.67
	RE (%)	13.34	-0.29	4.35	8.07	-2.74	9.66
Verification	NSE (%)	66.03	68.13	62.67	72.78	74.82	74.40
	R^2 (%)	66.04	67.19	63.45	67.18	78.01	79.46
	RE (%)	14.47	8.51	4.92	10.58	5.29	10.94

**Figure 11** | The observed and estimated streamflow from LSTM network and SWAT model.

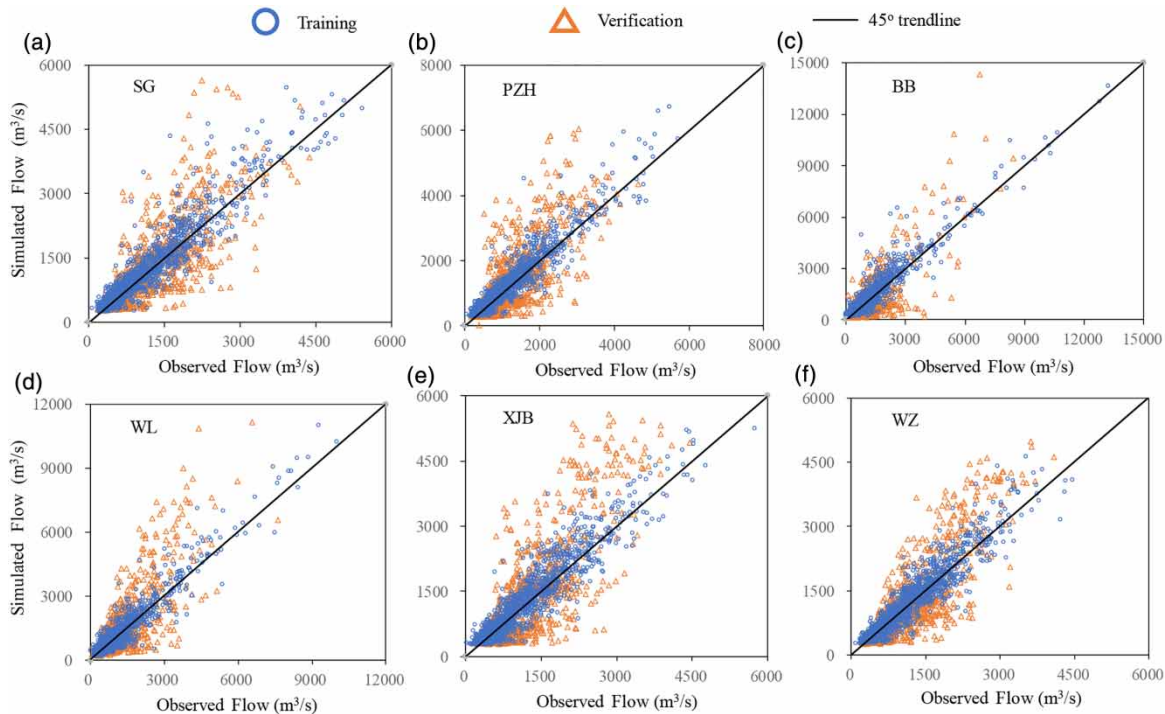


Figure 12 | The scatter plots of the observed and simulated flows from LSTM.

(2018) trained a deep LSTM network using daily rainfall and water level data during two periods: 2000–2002 and 2008–2014 (a total of 9 years) and it outperformed a SWAT model when verified using daily data from 2003 to 2007.

A main disadvantage of using large datasets is the computational time required for LSTM training, particularly when individual networks are trained for a large number of catchments. This issue could be tackled from several aspects: (1) using the advances in computing power such as Graphics Processing Units and cloud computing; (2) pooling the datasets from individual catchments of similar hydrological characteristics to train an LSTM network as a regional model which can predict the discharge for a number of catchments in the region (Kratzert *et al.* 2018); (3) training the network off-line for real-time predictions. The second aspect is also important for flow predictions in ungauged catchments as suggested by Kratzert *et al.* (2018) and provides a new application area in the use of the LSTM network in hydrological predictions.

Transfer learning is powerful and useful in deep learning as it can use the network knowledge gained from solving one problem to help solve another similar problem. Due

to the focus of this work, transfer learning is not investigated here. Future research should explore the potential of transfer learning from two aspects: (1) building on a reference architecture (e.g. Scenario A1 in this study), the network knowledge (e.g. parameters) could be applied to other similar architectures in solving the same problem so training could be continuously improved using prior network knowledge; (2) transferring the LSTM learning knowledge from data-rich catchments to data-scarce catchments, so the flow predictions in data-scarce catchments could be improved.

CONCLUSIONS

In this study, the performance of LSTM networks is assessed for river flow simulations using two river basins: the Hun river and Upper Yangtze river. Different LSTM structures are analysed. The prediction performances are compared against other models including hydrological models and data-driven models. The key research conclusions are summarized below.

In the LSTM network, the activation function in the fully connected layer before the LSTM cell layer can substantially reduce learning efficiency. In the LSTM network, the transformation through activation functions weakens the correlations between precipitation and flow, leading to failure in learning the rainfall–runoff relationships in both study basins. On the contrary, non-linear transformation following the LSTM cells is required to improve learning efficiency due to the different magnitudes of precipitation and streamflow. Further, increasing the number of fully connected layers cannot improve the learning efficiency, instead it needs more epochs due to the fluctuation in the loss values.

In the LSTM network, the batch size and the number of LSTM cells are the sensitive parameters that affect the learning efficiency. The results of this study show that the learning efficiency continues to increase with the batch size and the number of cells increasing. However, when the learning is stable, the number of cells should be kept to the minimum to reduce the complexity of the network.

The LSTM has superior non-linear learning ability for time series data and has a simple structure and few parameters, which has strong application potential in streamflow simulation. The results of this study show that the non-linear learning ability in the training process is very powerful.

The LSTM networks achieve good performance compared to other models considering three criteria, i.e., NSE, R^2 and RE. In the case of Hun river, LSTM outperforms MLR and BP and achieves a similar level of accuracy of XAJ. It is slightly worse than a well-calibrated SWAT but provides more accurate predictions for peak flows. In the case of the Upper Yangtze river, LSTM outperforms SWAT during the training but is worse than SWAT in the verification period. This is mainly because predicted peak flows are likely to be lower than those observed.

ACKNOWLEDGEMENTS

This research is supported by the National Natural Science Foundation of China (Grant No. 51609025 and 51709108) and the UK Royal Society through an industry fellowship to Guangtao Fu (Ref: IF160108) and an international

collaboration project (Ref: IEC\NSFC\170249). Guangtao Fu is also supported by The Alan Turing Institute under the EPSRC (Grant EP/N510129/1). The Open Fund Approval (SKHL1713, 2017), Chongqing technology innovation and application demonstration project (cstc2018jscx-msybX0274, cstc2016shmszx30002). We would like to thank the Hun river cascade hydropower reservoirs development Ltd and Upper Yangtze River Bureau of Hydrological and Water Resources Survey for the case study data.

DATA AVAILABILITY STATEMENT

All relevant data are available from an online repository or repositories (<http://www.hydroshare.org/resource/93f1f580de88403a8c52d2b3238297eb>).

REFERENCES

- Abbaspour, K. C. 2011 *SWAT-CUP4: SWAT Calibration and Uncertainty Programs: A User Manual*. Eawag, Swiss Federal Institute of Aquatic Sciences and Technology, Dübendorf, Switzerland.
- Abbaspour, K. C., Rouholahnejad, E., Vaghefi, S., Srinivasan, R., Yang, H. & Kløve, B. 2015 *A continental-scale hydrology and water quality model for Europe: calibration and uncertainty of a high-resolution large-scale SWAT model*. *J. Hydrol.* **524**, 733–752.
- Alexander, A. A. & Thampi, S. G. 2018 *Development of hybrid wavelet-ANN model for hourly flood stage forecasting*. *ISH J. Hydraul. Eng.* **24** (2), 266–274.
- ASCE Task Committee on Application of Artificial Neural Networks in Hydrology 2000a *Artificial neural networks in hydrology. I: preliminary concepts*. *J. Hydrol. Eng.* **5** (2), 115–123.
- ASCE Task Committee on Application of Artificial Neural Networks in Hydrology. 2000b *Artificial neural networks in hydrology. II: hydrologic applications*. *J. Hydrol. Eng.* **5** (2), 124–137.
- Bateni, S. M., Jeng, D. S. & Melville, B. W. 2007 *Bayesian neural networks for prediction of equilibrium and time-dependent scour depth around bridge piers*. *Adv. Eng. Software* **38** (2), 102–111.
- Bellegarda, J. R. & Monz, C. 2016 *State of the art in statistical methods for language and speech processing*. *Comput. Speech Lang.* **35**, 163–184.
- Bengio, Y., Simard, P. & Frasconi, P. 1994 *Learning long-term dependencies with gradient descent is difficult*. *IEEE. Trans. Neural. Netw.* **5** (2), 157–166.

- Clarke, R. T. 1994 *Statistical Modelling in Hydrology*. John Wiley & Sons.
- Daniell, T. M. 1991 Neural networks. Applications in hydrology and water resources engineering. In *National Conference Publication- Institute of Engineers*, Australia.
- Guzman, J. A., Moriasi, D. N., Gowda, P. H., Steiner, J. L., Starks, P. J., Arnold, J. G. & Srinivasan, R. 2015 A model integration framework for linking SWAT and MODFLOW. *Environ. Modell. Software* **73**, 103–116.
- Hinton, G. E. & Salakhutdinov, R. R. 2006 Reducing the dimensionality of data with neural networks. *Science* **313** (5786), 504–507.
- Hochreiter, S. & Schmidhuber, J. 1997 Long short-term memory. *Neural Comput.* **9** (8), 1735–1780.
- Hu, C., Wu, Q., Li, H., Jian, S., Li, N. & Lou, Z. 2018 Deep learning with a long short-term memory networks approach for rainfall-runoff simulation. *Water* **10** (11), 1543.
- Hu, R., Fang, F., Pain, C. C. & Navon, I. M. 2019 Rapid spatio-temporal flood prediction and uncertainty quantification using a deep learning method. *J. Hydrol.* **575**, 911–920.
- Kan, G., Yao, C., Li, Q., Li, Z., Yu, Z., Liu, Z., Ding, L., He, X. & Liang, K. 2015 Improving event-based rainfall-runoff simulation using an ensemble artificial neural network based hybrid data-driven model. *Stochastic Environ. Res. Risk Assess.* **29** (5), 1345–1370.
- Karimi, H. S., Natarajan, B., Ramsey, C. L., Henson, J., Tedder, J. L. & Kemper, E. 2019 Comparison of learning-based wastewater flow prediction methodologies for smart sewer management. *J. Hydrol.* **577**, 123977.
- Kayabasi, A., Yildiz, B., Sabanci, K., Yigit, E., Toktas, A. & Tekbas, M. 2017 Colour feature-based classification of wheat grain using ANN with bayesian regularization learning algorithm. In: *3rd International Conference on Science, Ecology and Technology (ICONSETE)*. pp. 14–16.
- Khan, S. & Yairi, T. 2018 A review on the application of deep learning in system health management. *Mech. Syst. Sig. Process.* **107**, 241–265.
- Khan, A., Sung, J. E. & Kang, J. W. 2019 Multi-channel fusion convolutional neural network to classify syntactic anomaly from language-related ERP components. *Inform. Fusion.* **52**, 53–61.
- Kingma, D. P. & Ba, J. 2014 *Adam: A Method for Stochastic Optimization*. Computer Science.
- Kratzert, F., Klotz, D., Brenner, C., Schulz, K. & Herrnegger, M. 2018 Rainfall-runoff modelling using long short-term memory (LSTM) networks. *Hydrol. Earth Syst. Sci.* **22** (11), 6005–6022.
- Kratzert, F., Klotz, D., Shalev, G., Günter, K. & Nearing, G. 2019 Benchmarking a catchment-aware long short-term memory network (LSTM) for large-scale hydrological modeling. *Hydrol. Earth Syst. Sci.* **23** (12), 1–32.
- Le, X. H., Ho, H. V., Lee, G. & Jung, S. 2019 Application of long short-term memory (LSTM) neural network for flood forecasting. *Water* **11** (7), 1387.
- Lee, G. H., Jung, S. H. & Lee, D. E. 2018 Comparison of physics-based and data-driven models for streamflow simulation of the Mekong river. *J. Korea Water Resour. Assoc.* **51** (6), 503–514.
- Muhammad, A. U., Li, X. & Feng, J. 2019 Using LSTM GRU and hybrid models for streamflow forecasting. In *International Conference on Machine Learning and Intelligent Communications*. Springer, Cham, pp. 510–524.
- Nash, J. E. & Sutcliffe, J. V. 1970 River flow forecasting through conceptual models part I – a discussion of principles. *J. Hydrol.* **10** (3), 282–290.
- Nayak, P. C., Sudheer, K. P., Rangan, D. M. & Ramasastri, K. S. 2004 A neuro-fuzzy computing technique for modeling hydrological time series. *J. Hydrol.* **291** (1–2), 52–66.
- Negnevitsky, M. & Pavlovsky, V. 2005 Neural networks approach to online identification of multiple failures of protection systems. *IEEE Transactions on Power Delivery* **20**, 588–594.
- Nourani, V. 2017 An emotional ANN (EANN) approach to modeling rainfall-runoff process. *J. Hydrol.* **544**, 267–277.
- Rocha, P., Silva, W. & Barros, A. 2019 Hierarchical expert neural network system for speech recognition. *J. Control Autom. Electr. Syst.* **30** (3), 347–359.
- Sahoo, B. B., Jha, R., Singh, A. & Kumar, D. 2019 Long short-term memory (LSTM) recurrent neural network for low-flow hydrological time series forecasting. *Acta Geophys.* **67** (5), 1471–1481.
- Saon, G. & Picheny, M. 2017 Recent advances in conversational speech recognition using convolutional and recurrent neural networks. *IBM J. Res. Dev.* **61** (4), 1:1–1:10.
- Shen, C. 2018 A transdisciplinary review of deep learning research and its relevance for water resources scientists. *Water Resour. Res.* **54** (11), 8558–8593.
- Shin, D., Lee, J., Lee, J. & Yoo, H.-J. 2017 14.2 DNPU: An 8.1 TOPS/W reconfigurable CNN-RNN processor for general-purpose deep neural networks. 2017 IEEE International Solid-State Circuits Conference (ISSCC), 5–7 February 2017, San Francisco, CA, IEEE, pp. 240–241.
- Sutskever, I., Martens, J., Dahl, G. & Hinton, G. 2013 On the importance of initialization and momentum in deep learning. In *International Conference on Machine Learning*. pp. 1139–1147.
- Tanty, R. & Desmukh, T. S. 2015 Application of artificial neural network in hydrology – a review. *Int. J. Eng. Technol. Res.* **4**, 184–188.
- Wang, W. & Ding, J. 2003 Wavelet network model and its application to the prediction of hydrology. *Nat. Sci.* **1** (1), 67–71.
- Xu, W. & Peng, Y. 2016 Medium-term river runoff forecast model using data of quantitative precipitation forecasts. *J. Hydroelectric Eng.* **35** (6), 11–19. Chinese.
- Xu, W., Zhang, C., Peng, Y., Fu, G. & Zhou, H. 2014 A two stage Bayesian stochastic optimization model for cascaded hydropower systems considering varying uncertainty of flow forecasts. *Water Resour. Res.* **50** (12), 9267–9286.
- Yan, L., Feng, J. & Hang, T. 2019 Small watershed stream-flow forecasting based on LSTM. In *International Conference on Ubiquitous Information Management and Communication*. Springer, Cham, pp. 1006–1014.

- Zhang, D., Lin, J., Peng, Q., Wang, D., Yang, T., Sorooshian, S. & Zhuang, J. 2018 [Modeling and simulating of reservoir operation using the artificial neural network, support vector regression, deep learning algorithm](#). *J. Hydrol.* **565**, 720–736.
- Zhao, R. J. 1992 [The Xinanjiang model applied in China](#). *J. Hydrol.* **135**, 371–381.
- Zhou, X., Tang, Z., Xu, W., Meng, F., Chu, X., Xin, K. & Fu, G. 2019 [Deep learning identifies accurate burst locations in water distribution networks](#). *Water Res.* **166**, 115058.

First received 26 February 2020; accepted in revised form 14 August 2020. Available online 5 October 2020