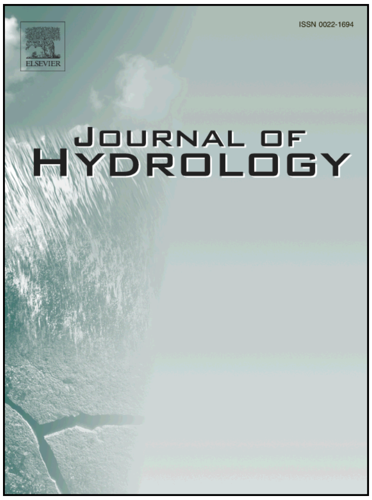

Research papers

Streamflow and rainfall forecasting by two long short-term memory-based models

Lingling Ni, Dong Wang, Vijay P. Singh, Jianfeng Wu, Yuankun Wang, Yuwei Tao, Jianyun Zhang

PII: S0022-1694(19)31031-5
DOI: https://doi.org/10.1016/j.jhydrol.2019.124296
Reference: HYDROL 124296

To appear in: *Journal of Hydrology*

Received Date: 8 June 2019
Revised Date: 27 October 2019
Accepted Date: 28 October 2019

Please cite this article as: Ni, L., Wang, D., Singh, V.P., Wu, J., Wang, Y., Tao, Y., Zhang, J., Streamflow and rainfall forecasting by two long short-term memory-based models, *Journal of Hydrology* (2019), doi: https://doi.org/10.1016/j.jhydrol.2019.124296

Streamflow and rainfall forecasting by two long short-term memory-based models

Lingling Ni[1], Dong Wang*[1], Vijay P. Singh[2], Jianfeng Wu*[1], Yuankun Wang[1], Yuwei Tao[1], Jianyun Zhang[3]

[1] Key Laboratory of Surficial Geochemistry, Ministry of Education, Department of Hydrosciences, School of Earth Sciences and Engineering, Nanjing University, Nanjing 210023, P.R. China

[2] Department of Biological and Agricultural Engineering, and, Zachry Department of Civil Engineering, Texas A & M University, College Station, TX77843, USA

[3]Nanjing Hydraulic Research Institute, Nanjing, P.R. China

Abstract: Prediction of streamflow and rainfall is important for water resources planning and management. In this study, we developed two hybrid models, based on long short-term memory network (LSTM), for monthly streamflow and rainfall forecasting. One model, WLSTM, applied a trous algorithm of wavelet transform to do series decomposition, and the other, CLSTM, coupled convolutional neural network to extract temporal features. Two streamflow datasets and two rainfall datasets are used to evaluate the proposed models. The prediction accuracy of WLSTM and CLSTM was compared with that of multi-layer perceptron (MLP) and LSTM. Results indicated that LSTM was applicable for time series prediction, but WLSTM and CLSTM were superior alternatives.

* Corresponding author: Dong Wang; Jianfeng Wu

E-mail: wangdong@nju.edu.cn; jfwu@nju.edu.cn

Key words: Long short-term memory, Wavelet transform, Convolutional layers, Hydrometeorological variables prediction

1 Introduction

Forecasting of streamflow and rainfall hydrometeorological variables plays a significant role in reservoir management, risk evaluation, irrigation, flood prevention, disaster management, and water planning and management (Liu et al., 2015). A multitude of forecast models have been developed which can be divided into two categories: process-based models and data-driven models (He et al., 2015; Zhang et al., 2015).

Process-based models have an advantage that they are based on physical principles, provide insights into physical processes (Zhang et al., 2015), are subject to many simplifying assumptions, and have large data requirements (Mehr et al., 2013). One the other hand, data-driven models are empirical, based on historical observations, are simple and easy to apply, and do not require information on physical processes (Liu et al., 2015). Currently, statistical and machine learning methods are often employed to develop data-driven models.

Statistical data-driven methods, such as multiple linear regression and autoregressive moving average (AMRA) and its variants have been applied for hydrometeorological forecasting since the 1970s. Studies have shown that statistical models produce satisfactory prediction when time series are linear or near-linear but do not capture nonlinear and nonstationary patterns hidden in time series (Zhang et al., 2016). However, hydrometeorological time series are characterized by dynamic complexity and non-stationarity (Nourani et al., 2014; Unnikrishnan and Jothiprakash, 2018). In recent years, machine learning techniques have received considerable attention for their strong learning ability and suitability for modeling complex and nonlinear processes. Hence, a variety of machine learning models, such as artificial neural networks (ANN), support vector regression (SVR), genetic programming (GP), and adaptive neuro-fuzzy inference system (ANFIS), have been developed and they have been found to produce satisfactory forecasts of nonlinear hydrological and meteorological processes (Chen et al., 2018; Aksoy and Dahamsheh, 2018; Liang et al., 2018; Nourani et al., 2014; Shi et al., 2015; Yaseen et al., 2015).

Recurrent neural network (RNN) is a kind of advanced ANN that has been especially designed for understanding temporal dynamics by involving feedback connections in the architecture to “remember” previous information (Elman, 1990). Advances in the RNN architecture and their training have enables them to handle tasks that involve sequential inputs in various problem domains, such as natural language processing, machine translation, and time series modeling (LeCun et al., 2015; Shen, 2018). Hence, RNNs are considered suitable for modeling complex hydrological and meteorological time series (Coulibaly and Baldwin, 2005; Kumar et al., 2004). However, learning with vanilla recurrent networks can be challenging due to the difficulty of learning long-range dependencies. The problem of vanishing and exploding gradients occurs when back-propagating errors flow across many time steps (Bengio et al., 1994; Hochreiter et al., 2001). Many solutions have been advanced to overcome this weakness of the vanilla RNN, and one of the solutions is long short-term memory (LSTM) (Chung et al., 2014).

The central idea of the LSTM architecture is a memory cell, which can maintain its state over time, and nonlinear gating units, which regulate the information flow into and out of the cell (Greff et al., 2017). LSTM networks have subsequently been shown to be more effective than conventional RNNs. Stimulated by the success of LSTM in many domains related to sequential data, a few studies have investigated the power of LSTM for hydrological and meteorological problems and have reported promising results. Zhang et al. (2018) employed LSTM to predict water table in agricultural areas compared it with feed-forward neural network (FFNN), and showed that LSTM outperformed FFNN. Zhang et al. (2018) built four different neural network models, namely multilayer perceptron (MLP), wavelet neural network (WNN), LSTM, and gated recurrent unit (GRU, a variant of LSTM) for predicting the water level of a combined sewage outflow structure. Comparison showed that LSTM and GRU were superior for multi-step-ahead time series prediction. Kratzert et al. (2018) investigated the potential of LSTM for modeling runoff from a number of catchments with meteorological observations. Results showed that LSTM was able to predict runoff with accuracy comparable to the well-established benchmark model. Shi et al. (2015) proposed a convolutional LSTM (ConvLSTM) to do precipitation nowcasting using observed radar maps.

Suitable data preprocessing improves the performance of data-driven models (Wu et al., 2009). Coupling data preprocessing techniques with machine learning, a number of studies have recently shown superior accuracy to corresponding single models for water resources and hydrology (Nourani et al., 2014; Ravansalar et al., 2017; Zhang et al., 2016). The wavelet-based model is one of the most popular hybrid models, as it often leads to significant improvement in forecast accuracy. Based on wavelet filters and convolution operation, wavelet transform (WT) can decompose a time series into separate sub-series and provide a more coherent structure of the series to a data-driven model, and for this reason it has become a popular tool for hydrological and meteorological forecasting (He et al., 2015; Quilty and Adamowski, 2018; Quilty et al., 2019).

Similar to WT, convolutional neural networks (CNNs) detect local conjunctions of features using discrete convolution operation based on a filter bank, which refers to convolutional layer (LeCun et al., 2015; Liu et al., 2017). CNNs are designed to process data that come in the form of arrays (such as signals, sequences and images) and local connections, and shared weights enable CNNs to make strong and mostly correct assumptions about the nature of signals (stationarity of statistic and locality of dependencies) (Krizhevsky et al., 2012; LeCun et al., 2015). CNNs can effectively extract invariant structures and hidden features in data, which is the reason that it has been applied in various fields (Gu et al., 2018; Krizhevsky et al., 2012; Liu et al., 2017). Studies have mathematically analyzed wavelet transform-based CNN for feature extraction, exploring the inherent connection between wavelet transform and convolutional layers (Mallat, 2012; Wiatowski and Bölcskei, 2018).

The objective of this study is to investigate the potential of LSTM for forecasting streamflow and rainfall, and develop two LSTM-based models to enhance forecast accuracy. One model coupled LSTM with wavelet transform (hereafter termed WLSTM), and the wavelet transform was employed to decompose time series into simpler components. The other model combined LSTM with convolutional layers (hereafter termed CLSTM), applying CNN to extract temporal features. To the best of our knowledge, it is the first time that such models have been constructed for forecasting streamflow and rainfall. Two monthly streamflow volume data from Cuntan and Hankou, and two monthly rainfall data from Jinan and Wenjiang (Chengdu) were employed in this study. Additionally, MLP was employed as benchmark model. The remainder of this paper is organized as follows: A general description of the methods and performance measures is provided in Section 2. The architecture of developed models is illustrated in Section 3. Section 4 presents discussion of results and application. Finally, Section 5 concludes the paper.

2 Methodology

This part gives a brief introduction to long short-term memory (LSTM), wavelet transform (WT), convolutional neural network (CNN), and performance measures.

2.1 Long short-term memory (LSTM)

As LSTM is a special kind of recurrent neural network (RNN), it is appropriate to revisit RNN. Different from traditional ANN, RNN has a recurrent hidden unit, which forms a self-looped cycle to implicitly maintain information about the history of all the past elements of the sequence (see Fig. 1(a)) (Elman, 1990; LeCun et al., 2015; Lipton et al., 2015). At time t, the hidden state h_t receives input from the current element x_t and also from the previous hidden state h_{t-1}. In this way, an RNN can map an input sequence with elements x_t into an output sequence with elements y_t, with each y_t depending on all the previous $x_{t'}$ (for $t' \le t$):

$$h_t = f(U_h h_{t-1} + W_i x_t + b_h) \quad (1)$$

$$y_t = f(W_o h_t + b_o) \quad (2)$$

where f is a non-linear activation function; W_i, U_h, and W_o are the weights of layers; and b_h, and b_o are the bias parameters.

The dynamics of the network depicted in Fig. 1(a) across time steps can be visualized by unfolding it in time as shown in Fig. 1(b). Given this picture, RNN can be interpreted not as cyclic, but rather as a very deep network with one layer per time step, and shares the same weights across time steps (the same parameters are used at each time step).

Insert Figure 1 here.

RNNs are very powerful dynamic systems but training them has proved to be problematic because of the problem of vanishing/exploding gradient. This problem can be solved by the structure of LSTM, proposed by Hochreiter and Schmidhuber (1997). A number of minor modifications to the original LSTM have since been made. We follow the notation of (Graves et al., 2013) (without peephole connections)

LSTM replaces the ordinary recurrent hidden unit with a memory cell which cell acts like a gated leaky neuron: it has a connection to itself at the next step that has a weight of one, but this self-connection is multiplicatively gated by another unit that learns to decide when to clear the content of the memory (LeCun et al., 2015). An LSTM unit can be seen in Fig. 2.

Insert Figure 2 here.

The gates are a distinctive feature of LSTM, which control the information flow within the LSTM cell. The first gate, introduced by (Gers et al., 2000), is the forget gate to decide what information would be thrown away from the cell state. A 1 represents “completely keep the previous cell state,” while 0 represents “completely get rid of it.”

The next step is to decide what new information would be stored in the cell state. This contains two parts. One is input gate, deciding which information is used to update the cell state. The other is a tanh layer that creates a new candidate value that could be added to the state.

Then the cell state c_t is updated. The third gate, output gate, controls the information of cell state that would be flown into the new hidden state. Finally, the new hidden state is calculated.

Forget gate $$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \quad (3)$$

Input gate $$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \quad (4)$$

Potential cell state $$\tilde{c}_t = \tanh(W_{\tilde{c}} x_t + U_{\tilde{c}} h_{t-1} + b_{\tilde{c}}) \quad (5)$$

Cell $$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \quad (6)$$

Output gate $$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \quad (7)$$

Hidden state $$h_t = \tanh(c_t) \odot o_t \quad (8)$$

where $\sigma(\cdot)$ represents the logistic sigmoid function, $\tanh(\cdot)$ is the hyperbolic tangent; $\odot$ denotes element-wise multiplication; and (W_f, U_f, b_f), (W_i, U_i, b_i), ($W_{\tilde{c}}, U_{\tilde{c}}, b_{\tilde{c}}$), ($W_o$, U_o, b_o) define the set of learnable weights.

2.2 Wavelet transform (WT)

The wavelet transform is a tool that cuts data into different frequency components, and then studies each component with a resolution matched to its scale. Its main property is that it provides a time-scale localization of the process under consideration (Daubechies, 1992). Observed hydrological series in nature are usually discrete signals, thus discrete wavelet transform (DWT) is usually preferred.

Discrete wavelet transform can be calculated by a pyramid algorithm (Mallat, 1989), which takes into count its representation at multiple resolutions of time and frequency. Through two sets of linear filters (wavelet and scaling filters), the wavelet and scaling coefficients are defined as (Percival and Walden, 2006):

$$W_{j,t}=\sum_{l=0}^{L-1}h_l V_{j-1,2t+1-l \bmod N_{j-1}} \tag{9}$$

$$V_{j,t}=\sum_{l=0}^{L-1}g_l V_{j-1,2t+1-l \bmod N_{j-1}} \tag{10}$$

where $W_{j,t}$ ($V_{j,t}$) is the j-th level discrete wavelet (scaling) coefficient; $\{h_l\}$ ($\{g_l\}$) is the wavelet (scaling) filter; L is the length of the wavelet (scaling) filter; N is the number of samples; mod refers to the modulo operator; and $N_j=N/2^j$.

A typical DWT is a non-redundant transform which is prone to variable sensitivity and is therefore an undesirable feature (related to boundary conditions) when applied to problems related to forecasting (Maheswaran and Khosa, 2012). This problem, caused by boundary conditions, can be overcame by a trous (AT) wavelet transform (Shensa, 1992), which promotes an understanding of the underlying process properties in terms of observation data.

Corresponding to the original series $X(t)$, smoother versions and detailed component can be derived as follows (Quilty and Adamowski, 2018; Rathinasamy et al., 2013):

$$V_{0,t}=X(t) \tag{11}$$

$$\widetilde{V}_{j,t}=\sum_{l=0}^{L-1}\widetilde{g}_l \widetilde{V}_{j-1,t-2^{j-1}l \bmod N} \tag{12}$$

$$\widetilde{W}_{j,t}=\widetilde{V}_{j-1,t}-\widetilde{V}_{j,t} \tag{13}$$

The original signal $X(t)$ may be reconstructed as

$$X(t)=\widetilde{V}_J+\sum_{j=1}^{J}\widetilde{W}_j \tag{14}$$

2.3 Convolutional neural network (CNN)

There are numerous variants of the CNN architecture in the literature, however, their basis components are very similar, including two types of layers, namely convolutional and pooling layers (Gu et al., 2018). The convolutional layer aims to learn feature representations of inputs, and is composed of several convolution kernels (or filter banks). The outputs of a convolutional layer are passed through a nonlinear activation function before being fed into the next layer. When the input is a 1-D signal, a convolutional layer *h*, is constructed by means of series of k=1,…,N_k, small filters ($L\times 1$) as (Ince et al., 2016; Laloy et al., 2018):

$$h_i^k = f(\sum_{l=1}^{L} w_l^k X_{i+l} + b^k) \tag{15}$$

where a common choice for $f(\cdot)$ is the rectified linear unit (ReLU).

2.4 Performance measures

The following three performance measures were used to qualitatively evaluate the performance of the models developed: root mean squared error (RMSE), Nash-Sutcliffe model efficiency coefficient (NSE), and mean absolute relative error (MARE), expressed, respectively, as

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_{pred} - y_{obs})^2} \tag{13}$$

$$\text{NSE} = 1 - \frac{\sum_{i=1}^{n}(y_{obs} - y_{pred})^2}{\sum_{i=1}^{n}(y_{obs} - \overline{y_{obs}})^2} \tag{14}$$

$$\text{MARE} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_{pred} - y_{obs}}{y_{obs}}\right| \tag{15}$$

where n is the number of observations; y_{pred} represents the predicted flow; y_{obs} is the observed flow; and $\overline{y_{obs}}$ denotes the average of observed flow.

3 Two hybrid forecasting models

The procedures of two long short-term memory (LSTM)-based models for streamflow and rainfall forecasting are described as follows.

The first one is wavelet-LSTM model (namely, WLSTM). In hydrological and water resources, wavelet-based forecasting commonly adopted the Mallat discrete wavelet transform algorithm, where performing WT would send some amount of future information into forecasting. This issue was discussed by (Karthikeyan and Kumar, 2013), further explored by Zhang et al. (2015), Quilty and Adamowski (2018), and Fang et al. (2019) . This problem is usually addressed by iteratively applying an appending-decomposing-sampling (ADS) operation to the testing dataset, which is time-consuming. To overcome the “future data” issue, and the high computational cost of ADS operation, in this study, we applied a trous (AT) algorithm to perform time series decomposition, which does not require data from the future time (>t) to calculate wavelet (scaling) coefficients at the present (at t). Due to the property of AT, WT can be performed directly on the whole dataset, instead of iteratively applying the ADS operation.

The WLSTM model first selects explanatory variables needed to be decomposed, then employs AT to divide the time series of selected explanatory and target variables into several more stable sub-series. Then, it removes the wavelet coefficients affected by the boundary condition at the beginning of series. After that, it divide the whole dataset into training set and testing set. A network containing two layer LSTM with a dense layer atop is trained to compute p step ahead forecast using q lagged records.

The steps of the WLSTM forecasting approach are as follows

1. Let Y, t= 1, 2, …, n, denote the time series of target variable; and X_1, X_2,…, X_N, t= 1, 2, …, n, denote the time series of explanatory variables.

2. Decide the lag-time (q, the number of previous time steps to use as inputs) and the lead-time (p, the number of the next time periods we want to predict) to create a combination of inputs and outputs. That is, $(Y^{t+1}, Y^{t+2}, \ldots, Y^{t+p}) = f\left((X_i^t, X_i^{t-1}, \ldots, X_i^{t-q+1}), (Y^t, Y^{t-1}, \ldots, Y^{t-q+1})\right)$, i=1,…N.

3. Select the wavelet function and decomposition level J and the explanatory variables needed to be decomposed.

4. Decompose and reconstruct the selected explanatory and target variables into J+1 sub-series (D_1, D_2,…, D_J, S_J), and remove boundary affected coefficients at the beginning.

5. Divide the time series into training and testing sets, (Y_{train}, X_{train}) and (Y_{test}, Y_{test}), respectively.

6. Use the training set to train the network (containing two-layer LSTM and a dense layer atop) for p-step ahead forecast.

7. Use the testing set to obtain p-step ahead forecast.

The second model coupled LSTM with convolutional neural network (CNN), namely CLSTM. In CLSTM, a stack of convolutional layers is applied to capture temporal features of variables. The features extracted by CNN are fed into a network containing two layer LSTM with a dense layer atop. Then, the whole CLSTM model is trained to compute p step ahead forecast using q lagged records. The steps of the CLSTM forecasting model are similar to those of WLSTM model:

1. Let Y, t= 1, 2, …, n, denote the time series of target variable; and X_1, $X_{2,\ldots}$, X_N, t= 1, 2, …, n, denote the time series of explanatory variables.

2. Decide the lag-time (q, the number of previous time steps to use as inputs) and the lead-time.

3. Select the number and size of convolutional filters.

4. Divide the time series into training and testing sets, (Y_{train}, X_{train}) and (Y_{test} , Y_{test}), respectively.

5. Use the training set to train the CLSTM network for p-step ahead forecast.

6. Use the testing set to obtain p-step ahead forecast.

Figs. 3 and 4 show the flowcharts of WLSTM and CLSTM forecasting models.

Insert Figures 3 and 4 here.

4 Applications

In order to explore the potential of WLSTM and CLSTM models, two kinds of hydrological and meteorological series from four representative stations were used. Case 1 was to forecast streamflow at Cuntan and Hankou stations on Yangtze River basin covering the period of 1959-2008. Case 2 was to forecast monthly rainfall data at Jinan (1951-2017) and Wenjiang (1961-2017, located in Chengdu) stations. Fig. 5 presents four stations in China.

Inset Figure 5 here

In this study, we investigated the potential use of WLSTM and CLSTM models for 1-, 3-, 6-, step ahead forecasting with inputs of 12- lagged records. In the WLSTM model, The “db4” was used as mother wavelet, and the decomposition level was set 3 and 2 for streamflow and rainfall forecasting, respectively. In the CLSTM model, the two-layer convolutional layer was followed by the two-layer LSTM with a dense layer atop. A three-layer MLP was applied to compare the performance with LSTM and WLSTM. All the hyper-parameters of neural networks (NNs) were set though a trial-and-error procedure. A dropout method (Srivastava et al., 2014) was applied to prevent overfitting. The last 10 years of record were used as the testing data.

4.1 Case 1: Monthly streamflow series from Cuntan and Hankou stations on Yangtze River basin

Monthly streamflow volume data from Cuntan and Hankou stations on Yangtze River basin, China, were selected for the application of the WLSTM and CLSTM models. The Yangtze River, with a length of 6280 km, is the longer river in China and the third longest river in the world. Cuntan station is the inflow gauge point of the upper Yangtze River, while Hankou station is located at the midstream of the Yangtze River. To improve the performance of NNs, the meteorological forcing data, including precipitation, temperature, and humidity, were used.

The RMSE, NSE, MARE statistics of the WLSTM, CLSTM, LSTM, MLP are given in Table 1. The best error measures are highlighted in red. The MARE values were not calculated for rainfall forecasting, as the rainfall contains zero. The table points that the WLSTM and CLSTM performed comparatively well.

Insert Table 1 here.

In the case of Cuntan, the WLSTM model had the best accuracy for 1- and 3-month ahead forecast in terms of RMSE=81.6E+8m^3, 85.22 E+8m^3, NSE=0.84, 0.82, MARE=0.17, 0.18, respectively, while the CLSTM model outperformed all NNs in the case of 6-step ahead forecast with RMSE=93.35E+8m^3. As shown in Table 1, the model forecast accuracy reduced as the forecast steps increased. For 1-step ahead forecast, all the NNs showed satisfactory results, with low RMSE, high NSE, and low MARE. When the forecast steps tuned into 3 and 6, the performance of MLP decreased rapidly, compared to other LSTM-based models. LSTM performed better than did MLP, as LSTM improved the RMSE by 4.6, 10.0, and 13.3%, improved the MARE by 37.9, 26.6, and 48.9% in comparison to MLP across different time steps. In the case of Cuntan, although CLSTM got lower RMSE, MARE, and higher NSE than did LSTM, it did not provide significantly improved forecast accuracy compared to WLSTM. WLSTM outperformed LSTM, as it reduced the forecast RMSE by nearly 10% relative to LSTM.

In the case of Hankou, the results analyzed were similar to Cuntan. The MLP had the worst performance amongst other models. The difference was that CLSTM had the best performance amongst all the NNs, and it outperformed LSTM, and MLP in terms of all the performance measures, gaining the best RMSE of 135.73 E+8m^3, 150.97 E+8m^3, 155.59 E+8m^3, NSE of 0.82, 0.77, 0.77, and MARE of 0.17, 0.20, 0.29, respectively. CLSTM reduced the RMSE by 10.0, 14.1, and 18.4% and reduced MARE by 26.1, 28.6, and 35.5 % across different time steps compared to MLP.

To further illustrate the performance of MLP, LSTM, WLSTM and CLSTM in a more intuitive way, hydrographs of the observed versus 1-, 3-, 6-, step ahead predicted streamflow in the testing sets were drawn, as shown in Fig. 6.

Insert Figure 6 here.

Generally speaking, it can be observed from the hydrographs that all four models had a good performance for forecasting monthly streamflow. In the case of Cuntan, it showed that WLSTM better forecasted the peak values, while MLP showed the worst performance among all the NNs, especially, MLP performed worst in dry seasons. Similar to Cuntan, in the case of Hankou, MLP overestimated streamflow in most of the dry seasons, and the LSTM overestimated peak values in 6-step ahead forecating. WLSTM and CLSTM performed better than did MLP and LSTM, as they better forecasted the sudden change of streamflow.

4.2 Case 2: Monthly rainfall series from Jinan and Wenjiang Stations

The other application is using WLSTM and CLSTM to predict monthly rainfall at Jinan (1951-2017) and Wenjiang (1961-2017, located in Chengdu) station. Jinan, the capital of Shandong Province, is located in eastern China, adjacent to the south of Tai Mountain and the north of Yellow River. The climate of Jinan is a warm temperate continental monsoon climate, with cold dry winters and hot wet summers. Wenjiang station is located in Chengdu, the capital of Sichuan Province located in western China. Chengdu is located on the eastern edge of the Qinghan-Tibet Plateau, in the western part of the Sichuan basin. Chengdu has a monsoon-influenced humid subtropical climate, with hot and humid summers, and mild winters.

Besides meteorological forcing variables, including temperature and humidity, based on the previous studies (Ouyang et al., 2014; Peng et al., 2014; Yang et al., 2017), climate indices, including PDO (Pacific decadal oscillation), SOI (southern oscillation index), and Nino3.4 (East central tropical Pacific sea surface temperature

(5N-5S)(170-120W)), were chosen as explanatory variables for rainfall forecasting. The performances of NNs are given in Table 2.

Insert Table 2 here.

In the case of rainfall forecasting, results produced similar conclusions as for streamflow forecasting. CLSTM got the best accuracy among all of the NNs at both Jinan and Wenjiang stations. With longer time steps, the performance of MLP deteriorated rapidly, while WLSTM and CLSTM showed better stability for long-term rainfall forecasting. In the case of Jinan, CLSTM outperformed LSTM, and MLP in terms of all the performance measures, gaining the best RMSE of 43.29 mm, 47.18 mm, 49.40 mm; and NSE of 0.71, 0.66, 0.62, respectively. CLSTM reduced the RMSE by 23.5, 21.6, and 25.1% and improved NSE by 39.2, 55.0, and 82.4 % across different time steps compared to MLP. In the case of Wenjiang, CLSTM showed the best accuracy among all of the models. Compared to MLP, CLSTM reduced the RMSE by 22.1, 13.6, and 26.4% and improved NSE by 18.9, 26.0, and 59.0 % across different time steps. Fig. 7 presented the hydrographs of observed rainfall and forecasted rainfall in the testing sets.

Insert Figure 7 here.

Fig. 7 showed the 1-, 3-, and 6-step ahead prediction of rainfall at Jinan and Wenjiang. In the case of Jinan, CLSTM and WLSTM outperformed MLP in predicting peak values. MLP either underestimated the peak value or overestimated rainfall in dry seasons. Similar observations can also be found at Wenjiang. Modeling and forecasting longer time steps might make the forecasts of MLP smoother, not being able to capture the extremes of the data.

As can be seen from Tables 1 and 2, and Figs. 6 and 7, LSTM outperformed MLP, while CLSTM and WLSTM were superior to LSTM in terms of all performance measures. CLSTM got the best forecast accuracy in most cases. With longer time steps, MLP deteriorated continually, while the LSTM-based models showed more stable performance. Especially, CLSTM and WLSTM better captured the peak values, while MLP performed worst in forecasting peak value or rainfall in dry seasons. It can be inferred from the results that wavelet transform and convolutional layers-based feature extractor improved the prediction accuracy of LSTM.

4.3 Discussion

In this part, further discussion is given.

The first thing is the difference between MLP and LSTM for multi-step ahead forecasting. For MLP, the lagged record was treated as a fixed-sized input, and the multi-step ahead forecast was a fixed-sized output. This mode of processing was a one-to-one case, in which all the temporal records were directly connected with each other though neuron weights, while LSTM processed differently compared to MLP, as it was regarded as a sequence-to-sequence problem. LSTM processed an input sequence one element at a time, and generated a sequence output, in which the past temporal information was implicitly maintained by the memory cell. Similar to LSTM, CLSTM and WLSTM processed the forecasting as a sequence-to-sequence problem. Fig. 8 illustrates the processing of MLP and LSTM. The sequence-to-sequence processing (Fig.8 (b)) looks more concise compared to inputting and outputting vectors altogether, and it was more consistent with our understanding of multi-step ahead forecasting by processing one element of sequence input at a time, then generating output one by one. Combined with the results obtained, one can see that LSTM showed that an implicit processing was superior to MLP, also indicating that LSTM was applicable for multi-step ahead forecasting.

Insert Figure 8 here.

The second is about the difference between CLSTM and WLSTM. Both CLSTM and WLSTM can be seen as data pre-processing methods, as they all apply the convolution operation to extract temporally local information of data. Wavelet transform extracts high- and low-frequency into different sub-time series (Quilty and Adamowski, 2018). CNN encodes temporally local information at the level of one convolutional layer (Laloy et al., 2018). However, the filters of WLSTM are pre-specified structured filters, while the filters of CLSTM are trained by data, which are learnable. Moreover, the outputs of convolutional layers are activated by non-linear functions. From the results of application, it can be seen that the learnable and non-linear CLSTM performed better than did WLSTM in most cases, though WLSTM showed its superiority to LSTM. It is suggested that when the explanatory variables were small and the accurate forecast of peak values was more expected, WLSTM seemed a better alternative. If no prior information was there about modeling, CLSTM seemed a better choice, as it performed well in most cases. In this study, we only applied two convolutional layers to construct CLSTM as the limit of insufficient data. The more data are available to train CLSTM, the more convolutional layers can be stacked. Also, the larger the stacked convolutional layers, the richer the representation of the input data and better forecast results can be expected.

5 Conclusions

Due to nonlinear and non-stationary nature of streamflow, traditional data-driven models for streamflow forecasting require data pre-processing techniques integrated with novel models to forecast streamflow. Long short-term memory (LSTM) is a popular neural network (NN) suitable for sequential data. This paper investigated the potential use of LSTM in streamflow and rainfall forecasting, and proposed two LSTM-based models to perform multi-step ahead forecasting. One model, WLSTM, applied “a trous” wavelet transform algorithm to do series decomposition, the other model, CLSTM, coupled convolutional layers to extract temporal features. The models were applied to predict monthly streamflow at Cuntan and Houkou stations on Yangtze River, and performed monthly rainfall forecasting at Jinan and Wenjiang stations.

Results were compared with those of MLP and LSTM. The results obtained indicated that LSTM was applicable for rainfall forecasting, and the wavelet transform and convolutional layers improved the forecast accuracy of LSTM, especially for longer time step ahead forecasting. CLSTM and WLSTM were superior alternatives when longer time steps ahead forecasting is expected.

Acknowledgments

This study was jointly supported by the National Key Research and Development Program of China (2017YFC1502704, 2016YFC0401501), and the National Natural Science Foundation of China (No. 41571017, 51679118, 91647203) and Jiangsu Province"333 Project" (BRA2018060)..

The data of meteorological variables including rainfall, temperature, humidity,

can be download from National Meteorological Information Center, China (http://data.cma.cn/). 3 climate indices data are retrieved from the National Oceanic and Atmospheric Administration Earth System Research Laboratory (NOAA-ESRL) (http://www.esrl.noaa.gov/psd/data/climateindices/list/). Streamflow data from Cuntan and Hankou station to support this study are from the Bureau of Hydrology, Ministry of Water resource, China. The 649th order issued in 2000 of Ministry of Water Resource---Regulations on the national secret and the scope of their secret level in hydraulic engineering work, states that hydrological data can not be provided and spread unauthorized, and readers can apply for access, following by the 601th order issued in 2009 of Ministry of Water Resource, China.

References:

Aksoy, H. and Dahamsheh, A., 2018. Markov chain-incorporated and synthetic data-supported conditional artificial neural network models for forecasting monthly precipitation in arid regions. Journal of hydrology, 562: 758-779.

Bengio, Y., Simard, P. and Frasconi, P., 1994. Learning long-term dependencies with gradient descent is difficult. IEEE transactions on neural networks, 5(2): 157-166.

Chen, L. et al., 2018. Flood Forecasting Based on an Improved Extreme Learning Machine Model Combined with the Backtracking Search Optimization Algorithm. WATER, 10(136210).

Chung, J., Gulcehre, C., Cho, K. and Bengio, Y., 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv:1412.3555.

Coulibaly, P. and Baldwin, C.K., 2005. Nonstationary hydrological time series forecasting using nonlinear dynamic methods. Journal of Hydrology, 307(1-4): 164-174.

Daubechies, I., 1992. Ten lectures on wavelets, 61. Siam.

Elman, J.L., 1990. Finding structure in time. Cognitive science, 14(2): 179-211.

Fang, W. et al., 2019. Examining the applicability of different sampling techniques in the development of decomposition-based streamflow forecasting models. Journal of Hydrology, 568: 534-550.

Gers, F.A., Schmidhuber, J. and Cummins, F., 2000. Learning to forget: Continual prediction with LSTM. Neural Computation, 12: 2451-2471.

Graves, A., Mohamed, A. and Hinton, G., 2013. Speech recognition with deep recurrent neural networks, IEEE international conference on acoustics, speech and signal processing, pp. 6645-6649.

Greff, K., Srivastava, R.K., Koutník, J., Steunebrink, B.R. and Schmidhuber, J., 2017. LSTM: A search space odyssey. IEEE transactions on neural networks and learning systems, 28(10): 2222-2232.

Gu, J. et al., 2018. Recent advances in convolutional neural networks. Pattern Recognition, 77: 354-377.

He, X., Guan, H. and Qin, J., 2015. A hybrid wavelet neural network model with mutual information and particle swarm optimization for forecasting monthly rainfall. Journal of Hydrology, 527: 88-100.

Hochreiter, S. and Schmidhuber, J., 1997. Long short-term memory. Neural computation, 9(8): 1735-1780.

Hochreiter, S., Bengio, Y., Frasconi, P. and Schmidhuber, J., 2001. Gradient flow in recurrent nets: the difficulty of learning long-term dependencies. A field guide to dynamical recurrent neural networks. IEEE Press.

Ince, T., Kiranyaz, S., Eren, L., Askar, M. and Gabbouj, M., 2016. Real-time motor fault detection by 1-D convolutional neural networks. IEEE Transactions on Industrial Electronics, 63(11): 7067-7075.

Karthikeyan, L. and Kumar, D.N., 2013. Predictability of nonstationary time series using wavelet and EMD based ARMA models. Journal of Hydrology, 502: 103-119.

Kratzert, F., Klotz, D., Brenner, C., Schulz, K. and Herrnegger, M., 2018. Rainfall-Runoff modelling using Long-Short-Term-Memory (LSTM) networks. Hydrology and Earth System Sciences, 22(11): 6006-6022.

Krizhevsky, A., Sutskever, I. and Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks, Advances in neural information processing systems, pp. 1097-1105.

Kumar, D.N., Raju, K.S. and Sathish, T., 2004. River flow forecasting using recurrent neural networks. Water resources management, 18(2): 143-161.

Laloy, E., Hérault, R., Jacques, D. and Linde, N., 2018. Training-image based geostatistical inversion using a spatial generative adversarial neural network. Water Resources Research, 54(1): 381-406.

LeCun, Y., Bengio, Y. and Hinton, G., 2015. Deep learning. nature, 521(7553): 436.

Liang, Z., Li, Y., Hu, Y., Li, B. and Wang, J., 2018. A data-driven SVR model for long-term runoff prediction and uncertainty analysis based on the Bayesian framework. THEORETICAL AND APPLIED CLIMATOLOGY, 133(1-2): 137-149.

Lipton, Z.C., Berkowitz, J. and Elkan, C., 2015. A critical review of recurrent neural networks for sequence learning. arXiv:1506.00019.

Liu, W. et al., 2017. A survey of deep neural network architectures and their applications. Neurocomputing, 234: 11-26.

Liu, Z., Zhou, P., Chen, X. and Guan, Y., 2015. A multivariate conditional model for streamflow prediction and spatial precipitation refinement. Journal of Geophysical Research: Atmospheres, 120(19).

Maheswaran, R. and Khosa, R., 2012. Comparative study of different wavelets for hydrologic forecasting. Computers & Geosciences, 46: 284-295.

Mallat, S., 2012. Group invariant scattering. Communications on Pure and Applied Mathematics, 65(10): 1331-1398.

Mallat, S.G., 1989. A theory for multiresolution signal decomposition: the wavelet representation. IEEE Transactions on Pattern Analysis & Machine Intelligence(7): 674-693.

Mehr, A.D., Kahya, E. and Olyaie, E., 2013. Streamflow prediction using linear genetic programming in comparison with a neuro-wavelet technique. Journal of Hydrology, 505: 240-249.

Nourani, V., Baghanam, A.H., Adamowski, J. and Kisi, O., 2014. Applications of hybrid wavelet–artificial intelligence models in hydrology: a review. Journal of Hydrology, 514: 358-377.

Ouyang, R. et al., 2014. Linkages between ENSO/PDO signals and precipitation, streamflow in China during the last 100 years. Hydrology and Earth System Sciences, 18(9): 3651-3661.

Peng, Z., Wang, Q.J., Bennett, J.C., Pokhrel, P. and Wang, Z., 2014. Seasonal precipitation forecasts over China using monthly large-scale oceanic-atmospheric indices. Journal of hydrology, 519: 792-802.

Percival, D.B. and Walden, A.T., 2006. Wavelet methods for time series analysis, 4. Cambridge university press.

Quilty, J. and Adamowski, J., 2018. Addressing the incorrect usage of wavelet-based hydrological and water resources forecasting models for real-world applications with best practices and a new forecasting framework. Journal of hydrology, 563: 336-353.

Quilty, J., Adamowski, J. and 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.

Rathinasamy, M., Adamowski, J. and Khosa, R., 2013. Multiscale streamflow forecasting using a new Bayesian model average based ensemble multi-wavelet Volterra nonlinear method. Journal of Hydrology, 507: 186-200.

Ravansalar, M., Rajaee, T. and Kisi, O., 2017. Wavelet-linear genetic programming:

A new approach for modeling monthly streamflow. Journal of hydrology, 549: 461-475.

Shen, C., 2018. A transdisciplinary review of deep learning research and its relevance for water resources scientists. Water Resources Research, 54(11): 8558-8593.

Shensa, M.J., 1992. The discrete wavelet transform: wedding the a trous and Mallat algorithms. IEEE Transactions on signal processing, 40(10): 2464-2482.

Shi, X. et al., 2015. Convolutional LSTM network: A machine learning approach for precipitation nowcasting, Advances in neural information processing systems, pp. 802-810.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. and Salakhutdinov, R., 2014. Dropout: a simple way to prevent neural networks from overfitting. The Journal of Machine Learning Research, 15(1): 1929-1958.

Unnikrishnan, P. and Jothiprakash, V., 2018. Daily rainfall forecasting for one year in a single run using Singular Spectrum Analysis. Journal of hydrology, 561: 609-621.

Wiatowski, T. and Bölcskei, H., 2018. A mathematical theory of deep convolutional neural networks for feature extraction. IEEE Transactions on Information Theory, 64(3): 1845-1866.

Wu, C.L., Chau, K.W. and Li, Y.S., 2009. Predicting monthly streamflow using data-driven models coupled with data-preprocessing techniques. Water Resources Research, 45(8).

Yang, T. et al., 2017. Developing reservoir monthly inflow forecasts using artificial intelligence and climate phenomenon information. Water Resources Research, 53(4): 2786-2812.

Yaseen, Z.M., El-Shafie, A., Jaafar, O., Afan, H.A. and Sayl, K.N., 2015. Artificial intelligence based models for stream-flow forecasting: 2000–2015. Journal of Hydrology, 530: 829-844.

Zhang, D., Lindholm, G. and Ratnaweera, H., 2018. Use long short-term memory to enhance Internet of Things for combined sewer overflow monitoring. Journal of Hydrology, 556: 409-418.

Zhang, H., Singh, V.P., Wang, B. and Yu, Y., 2016. CEREF: A hybrid data-driven model for forecasting annual streamflow from a socio-hydrological system. Journal of Hydrology, 540: 246-256.

Zhang, J., Zhu, Y., Zhang, X., Ye, M. and 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.

Zhang, X., Peng, Y., Zhang, C. and Wang, B., 2015. Are hybrid models integrated with data preprocessing techniques suitable for monthly streamflow forecasting? Some experiment evidences. Journal of Hydrology, 530: 137-152.

List of Tables

List of Figures

Table 1 Performance measures of WLSTM, CLSTM, LSTM, and MLP for streamflow forecasting in the testing period

	Lead time	1			3			6		
Stations	Model	RMSE	NSE	MARE	RMSE	NSE	MARE	RMSE	NSE	MARE
Cuntan	WLSTM	81.60	0.84	0.17	85.22	0.82	0.18	94.83	0.79	0.19
	CLSTM	88.42	0.81	0.20	90.56	0.80	0.19	93.35	0.79	0.19
	LSTM	88.53	0.81	0.18	94.53	0.78	0.22	104.29	0.74	0.24
	MLP	92.84	0.79	0.29	105.09	0.73	0.30	120.33	0.66	0.47
Hankou	WLSTM	140.47	0.80	0.21	153.06	0.77	0.20	161.37	0.75	0.20
	CLSTM	135.73	0.82	0.17	150.97	0.77	0.20	155.59	0.77	0.20
	LSTM	141.90	0.80	0.21	165.90	0.73	0.22	181.58	0.68	0.24
	MLP	150.87	0.77	0.23	175.57	0.69	0.28	190.63	0.65	0.31

Note: LSTM, long short-term memory; WLSTM, wavelet transform coupling with LSTM; CLSTM, convolutional neural network coupling with LSTM; MLP, multi-layer perceptron; RMSE, root mean squared error; NSE, Nash-Sutcliffe model efficiency coefficient; MARE, mean absolute relative error

Table 2 Performance measures of WLSTM, CLSTM, LSTM, and MLP for rainfall forecasting in the testing period

	Lead time	1		3		6	
Stations	Model	RMSE	NSE	RMSE	NSE	RMSE	NSE
Jinan	WLSTM	50.97	0.60	50.11	0.61	49.68	0.62
	CLSTM	43.29	0.71	47.18	0.66	49.40	0.62
	LSTM	52.61	0.57	54.02	0.55	55.03	0.53
	MLP	56.60	0.51	60.23	0.44	65.64	0.34
Wenjiang	WLSTM	60.07	0.62	61.75	0.60	63.01	0.58
	CLSTM	59.44	0.63	59.64	0.63	60.18	0.62
	LSTM	64.44	0.57	66.36	0.54	71.01	0.47
	MLP	67.27	0.53	69.06	0.50	76.06	0.39

Note: LSTM, long short-term memory; WLSTM, wavelet transform coupling with LSTM; CLSTM, convolutional neural network coupling with LSTM; MLP, multi-layer perceptron; RMSE, root mean squared error; NSE, Nash-Sutcliffe model efficiency coefficient;

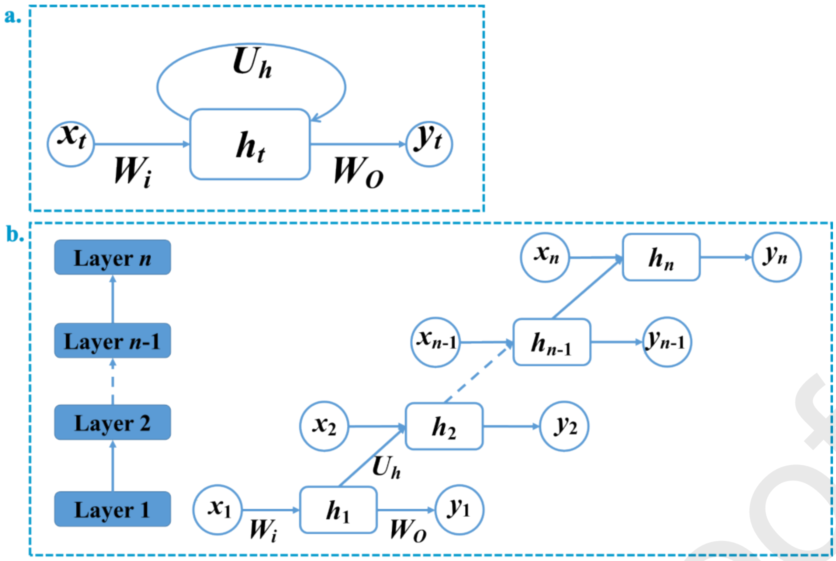

Figure 1 Architecture of recurrent neural network and unfolding in time of RNN

Note: Each rectangle is a neural network layer, arrows represent functions (e.g. matrix multiply), and circles are input and output vectors. A loop allows information to be passed from one step of the network to the next. RNN, unfolded in time, can be seen as deep feedforward networks in which all layers share the same weights (W_i, W_o, U_h).

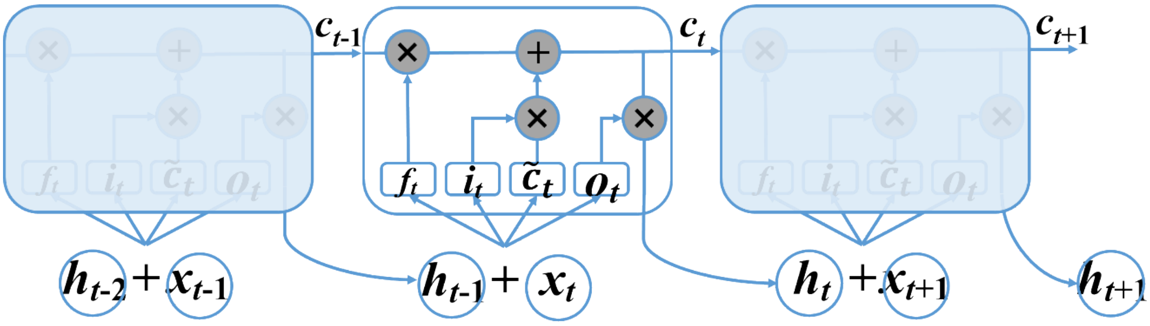

Figure 2 Architecture of LSTM unit

Note: The big rectangles represent a memory cell, the small rectangles are learned neural network layers, a “×” denotes an element-wise multiplication, and a plus means an adding operation. LSTM has two states, cell state c_t and hidden state h_t. c_t changes slowly, while h_t changes faster.

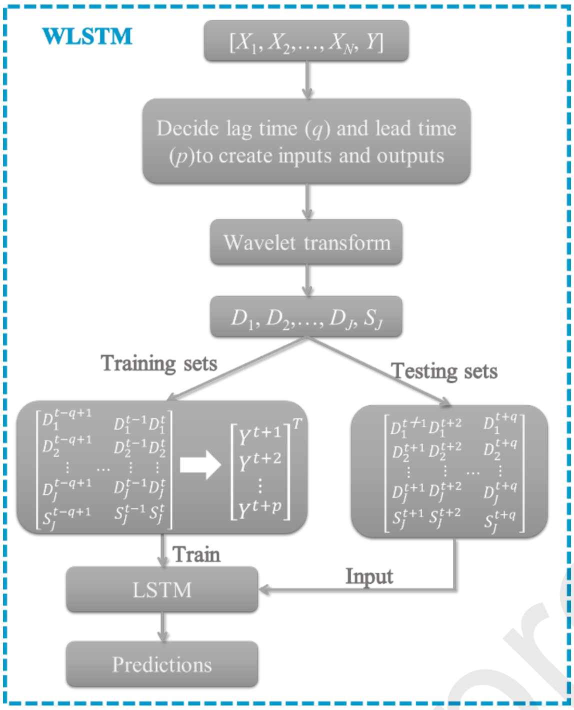
WLSTM
[X_1, X_2,…, X_N, Y]
Decide lag time (q) and lead time (p)to create inputs and outputs
Wavelet transform
D_1, D_2,…, D_J, S_J
Training sets
Testing sets
Train
LSTM
Input
Predictions

Figure 3 Flowchart of WLSTM forecasting model

Note: LSTM, long short-term memory; WLSTM, wavelet transform coupling with LSTM

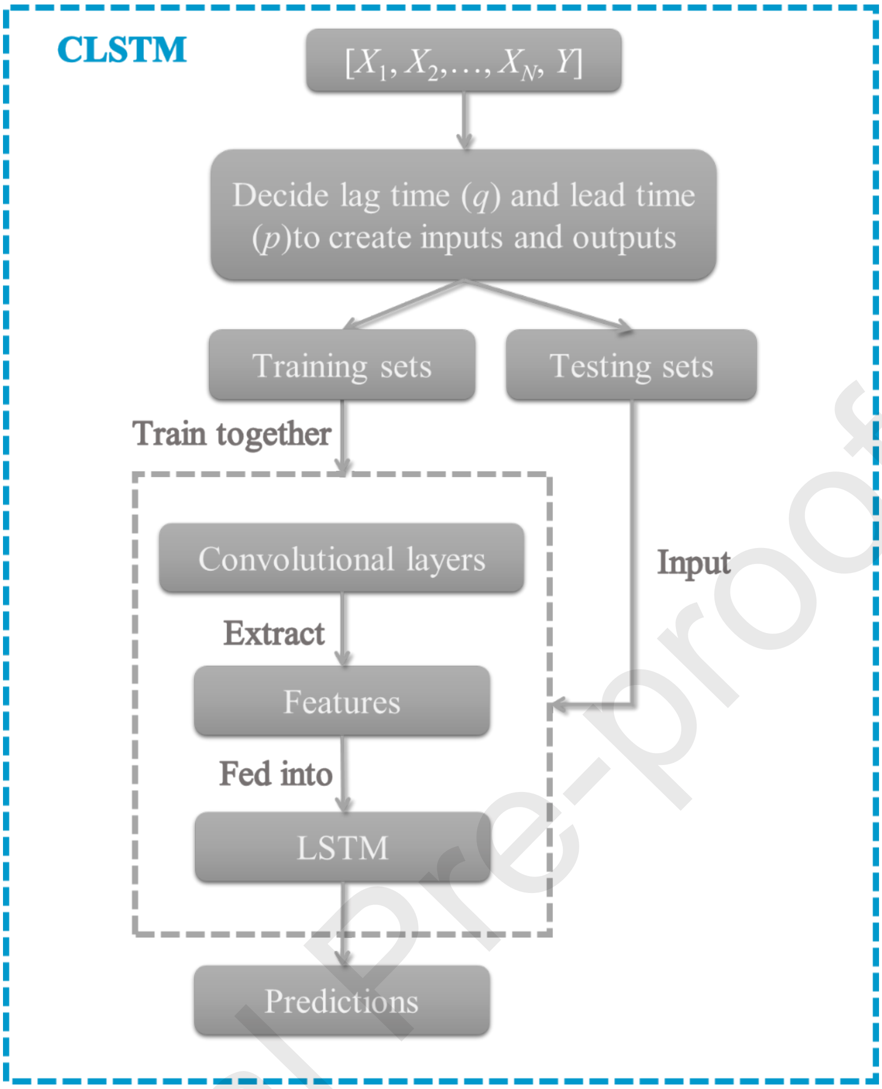

Figure 4 Flowchart of CLSTM forecasting model

Note: LSTM, long short-term memory; CLSTM, convolutional neural network coupling with LSTM;

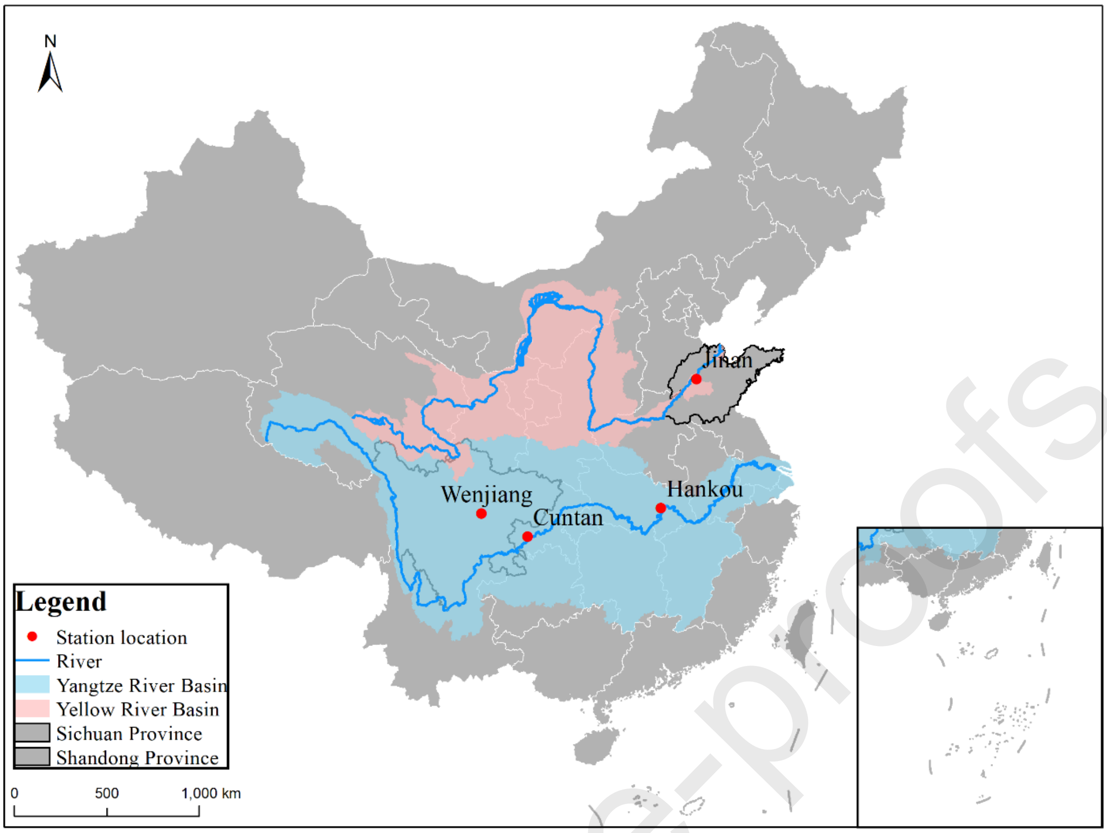

Figure 5. Locations of streamflow and rainfall stations selected

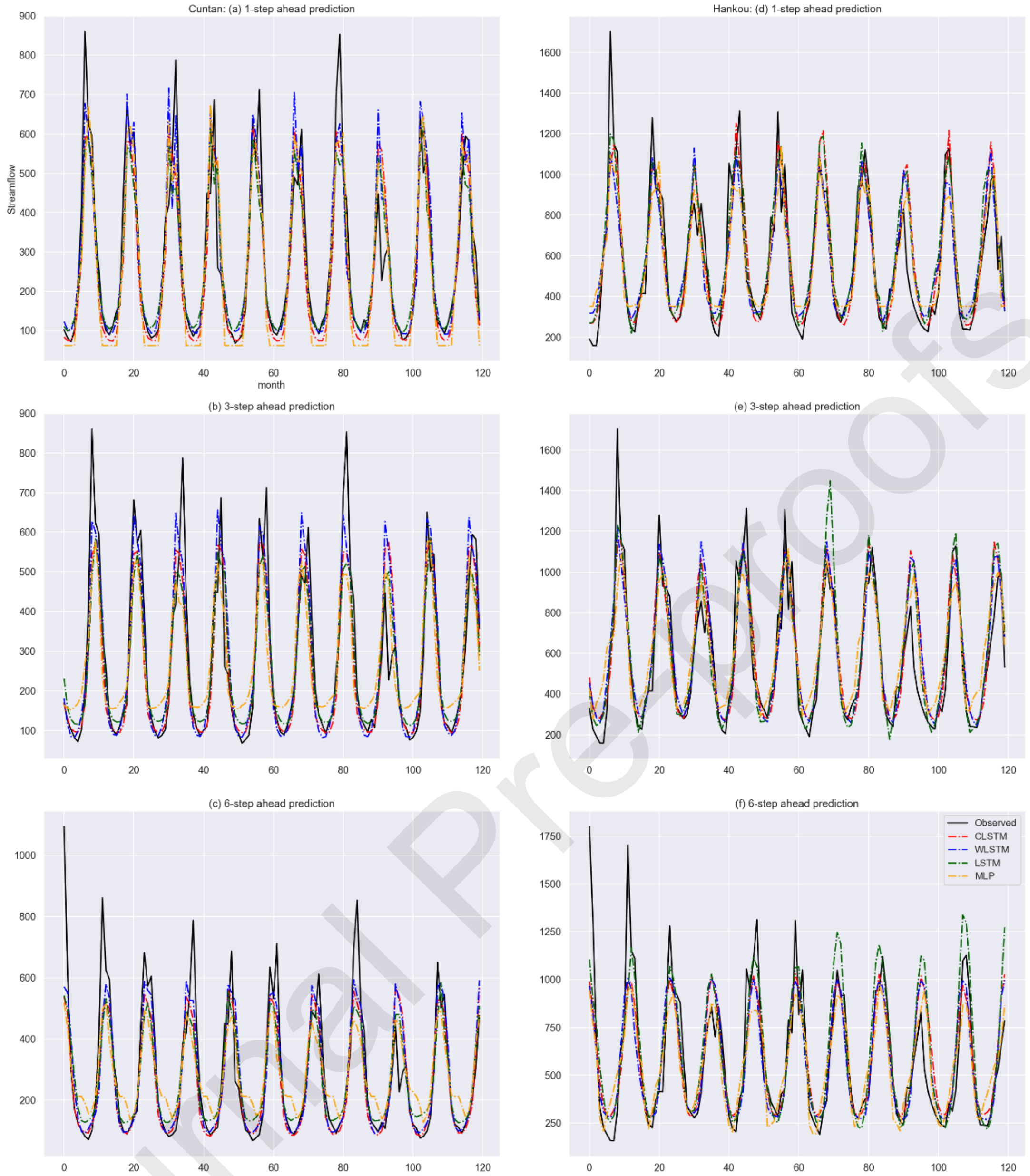

Figure 6 1-, 3-, 6-step ahead forecasts of streamflow by WLSTM, CLSTM, LSTM and MLP

(a)-(c) 1-, 3, 6-step ahead forecasts of streamflow in Cuntan station, respectively, (d)-(f) 1-, 3-, 6-step ahead forecasts of streamflow in Hankou, respectively

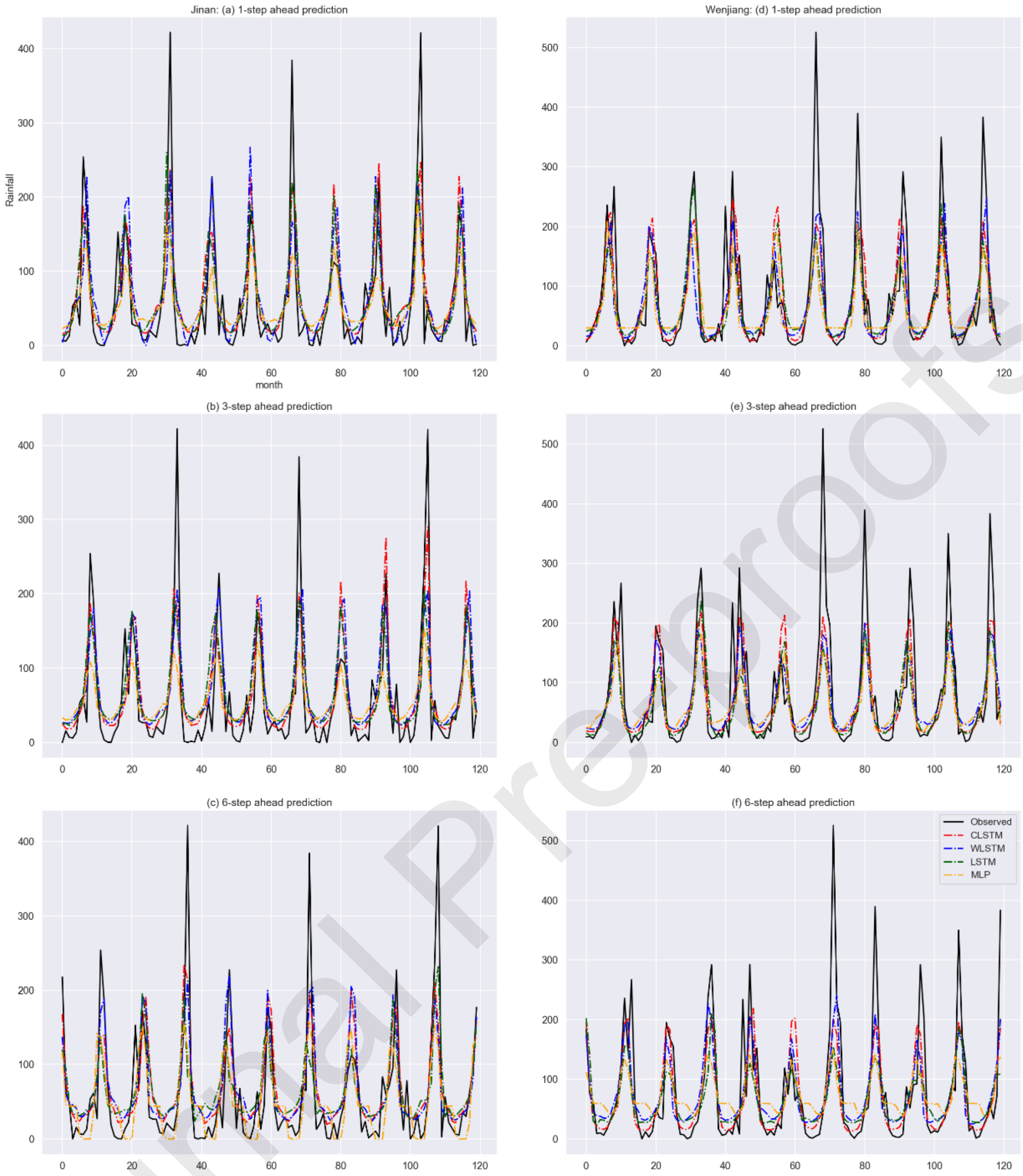

Figure 7 1-, 3-, 6-step ahead forecasts of rainfall by WLSTM, CLSTM, LSTM and MLP

(a)-(c) 1-, 3-, 6-step ahead forecasts of rainfall in Jinan station, respectively, (d)-(f) 1-, 3-, 6-step ahead forecasts of rainfall in Wenjiang station, respectively

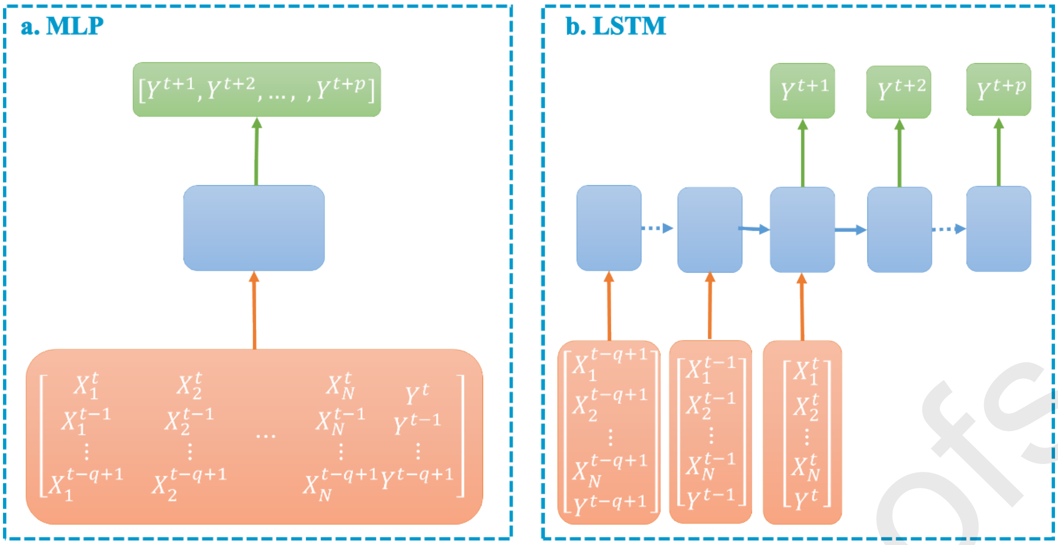
a. MLP
$[Y^{t+1}, Y^{t+2}, \dots, Y^{t+p}]$
$\begin{bmatrix} X_1^t & X_2^t & & X_N^t & Y^t \\ X_1^{t-1} & X_2^{t-1} & \cdots & X_N^{t-1} & Y^{t-1} \\ \vdots & \vdots & & \vdots & \vdots \\ X_1^{t-q+1} & X_2^{t-q+1} & & X_N^{t-q+1} & Y^{t-q+1} \end{bmatrix}$
b. LSTM
Y^{t+1}
Y^{t+2}
Y^{t+p}
$\begin{bmatrix} X_1^{t-q+1} \\ X_2^{t-q+1} \\ \vdots \\ X_N^{t-q+1} \\ Y^{t-q+1} \end{bmatrix}$
$\begin{bmatrix} X_1^{t-1} \\ X_2^{t-1} \\ \vdots \\ X_N^{t-1} \\ Y^{t-1} \end{bmatrix}$
$\begin{bmatrix} X_1^{t} \\ X_2^{t} \\ \vdots \\ X_N^{t} \\ Y^{t} \end{bmatrix}$

Figure 8 Forecast processes of MLP and LSTM

Note: Each rectangle is a vector and arrows represent functions. Input vectors are in orange, output vectors are in green, and blue rectangles hold the neural network's hidden vectors.

Streamflow and rainfall forecasting by two long short-term memory-based models

Highlights:

- Two new models are developed for streamflow and rainfall forecasting
- Long short-term memory (LSTM)-based models produced satisfactory forecasts
- Wavelet transform and convolutional layers were coupled with LSTM, respectively.