https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

An explainable deep learning model based on hydrological principles for flood simulation and forecasting

Xin Xiang, Shenglian Guo, Chenglong Li, Yun Wang

State Key Laboratory of Water Resources Engineering and Management, Wuhan University, Wuhan, 430072, China

Correspondence to: Shenglian Guo (slguo@whu.edu.cn)

Abstract: Deep learning (DL) models always perform well in hydrological simulation but lack physical-based principles. To address this gap, we integrate the runoff generation and flow routing principals of Xinanjiang (XAJ) model into the architecture of recurrent neural network (RNN) units and establish a physical-based XAJRNN neural network layer. Subsequently, this layer is fused with LSTM layers to construct an explainable deep learning (EDL) model, which underwent testing at the Lushui River and Qingjiang River basins in China. Compared to benchmark models, the proposed EDL model performs very well, the average Nash-Sutcliffe efficiency (*NSE*)values for these two basins are 0.98 and 0.94, respectively. The small flood peak relative errors (*PRE*) and peak timing difference (ΔT) close to zero demonstrate that the EDL model can accuracy simulate flood events. Notably, the EDL model not only enhances simulation accuracy over ordinary DL models but also enhances interpretability by incorporating physical principles, thereby offering fresh insights for the fusion of DL and hydrological models for flood simulation and forecasting.

1 Introduction

In the modern era, flood disasters present substantial threats to both human societies and the natural environment (Guido et al., 2023). With the intensification of global climate change and rapid urbanization, the accuracy and timeliness of flood forecasting have become increasingly important. Flood forecasting typically relies on hydrological models (Thaisiam et al., 2024). By analyzing the rainfall-runoff relationships from historical periods, hydrological models simulate hydrological processes within a watershed. Combining these with forecasted rainfall data, such models can forecast flow discharges,

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

river water levels, and probabilities of flood occurrence. In recent years, advancements in computational power and artificial intelligence (AI) technologies have significantly improved the accuracy and real-time responsiveness of hydrological models (Hirabayashi et al., 2013), offering more scientific and efficient support for disaster prevention and mitigation efforts.

Traditional hydrological models often rely on statistical methods and empirical formulas; however, these approaches exhibit limitations when tackling complex and nonlinear hydrological processes (Roy et al., 2023). In recent years, deep learning (DL) technologies have made significant advancements in various fields, particularly in time series prediction, where they display strong potential. DL, a domain dedicated to uncovering patterns and extracting knowledge from large datasets, enables computers to autonomously learn algorithms, analyze extensive sample data, and identify patterns, facilitating predictions on unfamiliar data. This process closely aligns with hydrological modeling, which discerns patterns by analyzing historical hydrometeorological data, generalizing, and simulating hydrological processes. Consequently, DL has attracted widespread attention in hydrology (Nearing et al., 2021).

DL commonly refers to deep neural networks, a form of representational learning technique that links simple nonlinear computational units through multi-layer network architectures to understand intricate relationships. It falls within the realm of machine learning (ML) (Yann et al., 2015). The term "deep" in DL signifies network structures with multiple layers and neurons, although there is no precise definition of "deep." Generally, it denotes models that necessitate substantial data and encompass numerous layers and neurons. These layers convert their inputs into higher-level features, magnifying crucial factors for output variability while reducing irrelevant variations. This facilitates automatic feature extraction, contrasting with "shallow" networks or conventional ML algorithms that rely on expert knowledge and engineering skills for designing feature extractors. This is a key rationale behind the increasing application of DL models over shallow networks in recent years (Frank et al., 2020). Structurally, the standard recurrent neural network (RNN), exemplified by Long Short-Term Memory (LSTM), remains the foundational model architecture for deep learning-driven hydrological forecasting. As an RNN subset in DL, LSTM has gained prominence for its efficacy in managing sequential data and capturing long-term dependencies. LSTM tackles the challenges of vanishing and exploding gradients in traditional RNNs when handling lengthy sequences through gated mechanisms, resulting in superior

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

performance in time-dependent prediction tasks (Hochreiter and Schmidhuber, 1997).

DL has been extensively utilized in various fields. In hydrology, where processes are not yet fully understood, it exhibits promise in identifying physical processes through a data-mining lens. However, achieving accurate predictions is not the sole aim. Hydrologists are interested in whether models are in line with fundamental physical principles, are interpretable, and contribute to scientific knowledge advancement. Traditional physics-based hydrological models generally provide better interpretability and physical consistency, relying less on data and complementing DL models. As a result, the fusion of physics-based mechanisms and data-driven models has garnered significant attention in recent years, showcasing potential in advancing scientific inquiry (Nearing et al., 2021; Shen, 2018). Currently, the coupling of DL and physics-based models focuses on four main aspects.

(1) Introducing physical mechanisms into DL models' loss functions

Worland et al. (2019) developed a multi-output multilayer perceptron (MLP) model to forecast flow duration curve (FDC) quantiles, incorporating FDC monotonicity constraints into the loss function. This approach resulted in forecasts that adhered to monotonicity and closely matched the FDC derived from observations. Wang et al. (2020) not only incorporated physical laws into the loss function but also integrated expert knowledge in the form of inequalities, constructing Theory-guided Neural Networks (TgNN). TgNN demonstrated superior predictive performance compared to standard DL models. Xie et al. (2021) encoded three physical conditions in rainfall-runoff forecasting into the loss function. Experiments across 531 Catchment Attributes and Meteorology for Large-sample Studies (CAMELS) basins showed improvements in the average Nash-Sutcliffe efficiency (*NSE*) from 0.52 to 0.61, enhanced peak flow simulating, and reduced unreasonable negative values. Pokharel et al. (2023) tested the effects of incorporating mass balance, energy balance, and storage-discharge relationships into the loss function of DL models across 34 basins, finding performance improvements in some basins, particularly with mass and energy balance constraints, which were effective in 38% and 32% of basins, respectively. Frame et al. (2023) concluded that strict adherence to the water balance principle might degrade forecasting performance due to data errors. DL models, which do not necessarily enforce the water balance principle, can accommodate data biases and outperform traditional hydrological models in runoff forecasting.

(2) Using DL models as post-processors

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

Correcting errors in forecasting from physics-based models can significantly improve forecasting accuracy. Cho and Kim (2022) employed LSTM to learn correlations between meteorological data and WRF-Hydro forecast errors, applying this approach to calibrate WRF-Hydro forecasting. Experiments in South Korean basins showed *NSE* values reaching 0.95, compared to 0.72 before calibration. Similarly, Han and Morrison (2022) and Frame et al. (2021) applied LSTM to post-process multi-period forecasting from the National Water Model in the United States. Boucher et al. (2020) utilized the simulated runoff of the GR4J hydrological model and observed water temperature as inputs to construct an MLP model, demonstrating notable improvements compared to models without assimilation. Cui et al. (2021) proposed a novel EDL model combining the Xinanjiang (XAJ) hydrological model with LSTM for multi-step flood forecasting. This model used XAJ outputs as inputs to the LSTM, enhancing the physical mechanism of DL models.

(3) Using DL models to calibrate parameters in traditional hydrological models

Tsai et al. (2021) proposed a parameter learning method to calibrate HBV model parameters. The DL model generated parameters instead of directly outputting runoff, which were then combined with inputs to produce runoff through the physical model. Applying this method across 1,802 basins showed median Kling-Gupta Efficiency (*KGE*) values improving from 0.48 to 0.59 compared to calibration via evolutionary algorithms followed by parameter regionalization. Similarly, Feng et al. (2023, 2022), Shen et al. (2023) and Song et al. (2024) used DL models to calibrate HBV model parameters based on meteorological data and basin attributes, driving hydrological models to simulate runoff. In addition to the above calibration of lumped model parameters, a similar method is also used to calibrate the confluence model parameters. Zhong et al. (2024a, 2024b) used DL model to calibrate parameters in the Muskingum-Cunge method and construct a distributed physics-driven DL hydrological model. Bindas et al. (2024) introduced a novel differentiable routing method (δMC-JuniatahydroDL2) combining the Muskingum-Cunge routing model with neural networks to infer Manning's roughness and channel geometry parameters. The method provided more accurate long-term routing forecasts, especially in untrained sub-basins.

(4) Designing DL models based on physical mechanisms

Encoding rules directly into neural networks represents a direct fusion of physics-based and data-

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

driven models. Hoedt et al. (2021) modified LSTM structures to enforce water balance over specific periods. Experiments across 531 CAMELS basins showed improved peak flow performance despite no overall improvement in *NSE*. Jiang et al. (2020) modified RNN structures to incorporate state variables (e.g., soil moisture) from EXP-HYDRO model as recurrent unit states, combining these with other neural network layers to construct a physics-guided RNN. Experiments in 671 CAMELS basins demonstrated median NSE improvements from 0.60 to 0.71, with reductions in peak flow bias and improved baseflow simulations. De la Fuente et al. (2024) proposed HydroLSTM, which models hydrological principles to enhance interpretability, achieving comparable performance to LSTM models while requiring fewer unit states. Similarly, Li et al. (2024) embedded EXP-HYDRO processes into RNN units, developing a process-driven DL model that enhanced process understanding of rainfall-runoff relationships. Experiments across 531 CAMELS basins demonstrated improvements over LSTM model. Wang et al. (2024) introduced a novel distributed hydrological modeling framework combining HydroPy distributed hydrological model principles encoded into RNN units and DL models to calibrate physical parameters, improving runoff and water volume simulations.

The current integration of DL models with physical mechanisms mainly involves loosely coupled approaches, such as modifying loss functions or calibrating parameters. Further exploration is needed to integrate physical hydrological mechanisms into neural network architectures, constructing models with strong physical consistency and superior forecasting performance to facilitate new hydrological insights. The novelty of study is using the runoff generation and flow routing principals of the XAJ model to establish a physical-based XAJRNN neural network layer within an ordinary RNN framework by identifying model state variables and fluxes under a differential equation. A EDL model combining the XAJRNN layer and LSTM is constructed and tested in the Lushui River and Qingjiang River basins to demonstrate the advantages of the EDL model in flood simulation. The findings may offer a novel pathway for integrating DL models with hydrological physical mechanisms to improve flood forecasting accuracy.

The rest of this paper is organized as follows. The case study and materials are introduced in Section 2. Section 3 presents the methodologies. Section 4 evaluates and analyses the simulated results. Section 5 discusses the strengths as well as the weaknesses of the proposed model. Conclusions and outlook are

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

given in Section 6.

2 Study area and data

2.1 Study basin

(1) Lushui River basin

The Lushui River is a primary tributary of the middle Yangtze River, with a basin area of approximately 3,950 km^2. The basin's terrain slopes from high elevations in the southeast to lower areas in the northwest. The basin is located in a subtropical monsoon climate zone, characterized by warm and humid conditions, with an average annual temperature of approximately 15.5°C and an average annual rainfall of 1,550 mm. The Lushui River's annual runoff volume reaches 3.03 billion m^3, with rainfall concentrated from May to September, accounting for 70% of the annual total. At the river valley's outlet, the Lushui Reservoir has a total storage capacity of approximately 408 million m^3, with only 163 million m^3 allocated for flood control. In early July 1995, the reservoir experienced its largest flood event, with an inflow peak of 4,500 m^3/s and a three-day runoff of 500 million m^3. In 2017, six flood events occurred, with peak inflows exceeding 1,000 m^3/s, reaching a maximum inflow of 4,400 m^3/s (Cui et al., 2021; Xiang et al., 2024). The geographical location of Lushui Reservoir is shown in Figure 1.

(2) Qingjiang River basin

The Qingjiang River, a primary tributary of the Yangtze River in the middle reaches, has a basin area of approximately 17,000 km^2. The region receives an average annual rainfall of 1,460 mm, with the majority falling between April and September, representing 75%–78% of the annual total. Situated in the heavy rainfall region of western Hubei Province, the basin's terrain facilitates the uplift of warm, moist air and is frequently affected by southwest cyclones. With a natural elevation drop of 1,430 m, the area features steep terrain and a high river gradient, leading to swift water flow convergence and significant fluctuations in flood levels. Consequently, the area is susceptible to severe rainfall and flood disasters. Along the main stream of the Qingjiang River, three sizable reservoirs exist, with the Shuibuya Reservoir serving as the central hub for the basin's cascade development, overseeing an area of roughly 10,860 km^2. Located in Badong County, Hubei Province, the Shuibuya Reservoir plays a crucial role in the

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

hydropower development of the Qingjiang River. It greatly forms an integral component of the flood control system in the middle and lower reaches of the Yangtze River (Zhou et al., 2014). This study specifically focuses on the basin controlled by the Shuibuya Reservoir, as depicted in Figure 1.

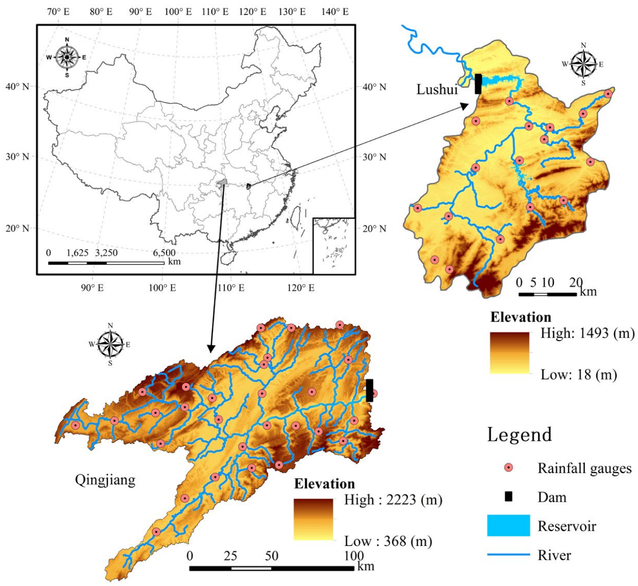


Figure 1: Sketch map of river networks and rainfall gauges in the Lushui River and Qingjiang River basins.

2.2 Data

The study collected flood season data (Lushui River basin: May 1 to October 31, 2012–2019; Qingjiang River basin: April 1 to October 31, 2012–2020) that includes rainfall, pan evaporation, and inflow datasets. For Lushui River basin, 3 h rainfall data from 17 gauges, 3 h pan evaporation, and 3 h inflow discharge was collected. The data from 2012 to 2016 were used for training, and the data from 2017 to 2019 for testing. For Qingjiang River basin, 6 h rainfall from 28 gauges, 6 h pan evaporation, and 6 h inflow discharge was gathered. The data from 2012 to 2016 were used for training, and the data from 2017 to 2020 for testing. The Thiessen polygon method was used to calculate the areal mean rainfall and pan evaporation for both basins.

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

3 Methodologies

3.1 XAJRNN neural network layer

3.1.1 XAJ model overview

In this study, the evapotranspiration of the XAJ model uses a three-layer soil moisture model. The runoff generation uses a tension water capacity curve (Zhao, 1992, 1993). In the runoff separation, the runoff is divided into three types using the free water capacity curve: surface runoff, interflow runoff, and groundwater runoff. The flow routing includes both hillslope routing and channel network routing submodules, which use linear reservoirs and Nash unit hydrographs (Singh, 1977), respectively. The logical structure of the XAJ model is shown in Figure 2.

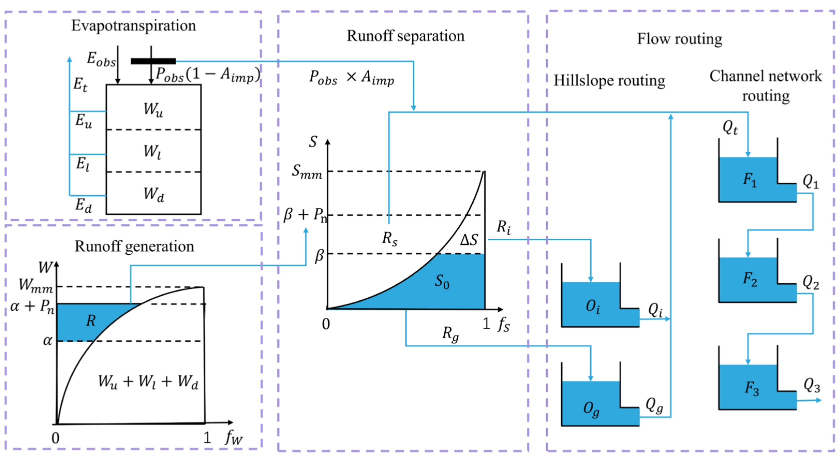


Figure 2: Structure diagram of the XAJ model.

The model consists of input variables, state variables, fluxes, output variables, and parameters, along with their corresponding mathematical equations. Input variables include areal mean rainfall and measured pan evaporation, while the output variable is the simulated runoff. State variables represent physical quantities that characterize the watershed's state, and their dimensions are independent of time (La Follette et al., 2021). The state variables of the XAJ model are shown in Table A1. Fluxes describe the exchange of water within the watershed or between the watershed and external environments. These can be expressed as functions of state variables or other fluxes, and their dimensions are time-dependent

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

(La Follette et al., 2021). The fluxes of the XAJ model are shown in Table A2. The mathematical equations can be divided into state variable control equations and constitutive equations for fluxes. The control equations describe how state variables evolve over time, while the constitutive equations establish the relationships between unknown fluxes and state variables or known fluxes. Detailed information on the XAJ model is provided in Text A1.

3.1.2 Derivation of XAJRNN

Establishing the XAJ model in the watershed can be considered a complete system that represents changes in state variables within the watershed, such as the variation in the average tension water storage of the upper soil layer. At the same time, the XAJ model also describes how the watershed system responds to specific input conditions. These responses can be expressed through a combination of ordinary differential equations (ODE) and output equations:

$$\begin{cases} \frac{d}{dt}h(t) = F(h(t), x(t); \varphi_f) \\ y(t) = G(h(t), x(t); \varphi_g) \end{cases} \tag{1}$$

where: $h(t)$ represents the state variables of the XAJ model (as shown in Table 1). $x(t)$ represents the input variables of the XAJ model (P_{obs} and E_{obs}). $y(t)$ represents the output of the XAJ model (Q). φ_f and φ_g are parameters in the XAJ model (as shown in Table 3). $F(\cdot)$ and $G(\cdot)$represent the mathematical equations and functions in the model. The above equations form explicit continuous equations, but in practice, implicit discrete equations are generally used to obtain numerical solutions:

$$\begin{cases} h(t) = f(h(t-1), x(t); \varphi_f) \\ y(t) = g(h(t), x(t); \varphi_g) \end{cases} \tag{2}$$

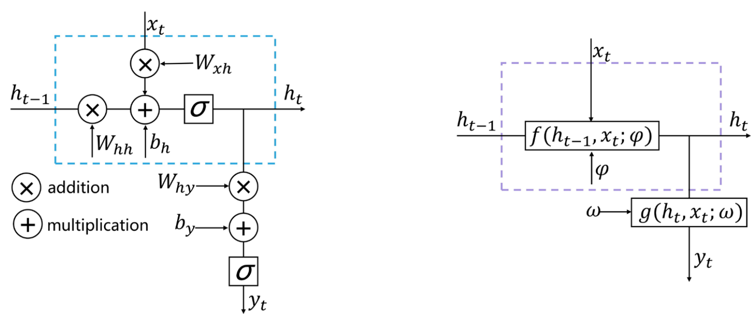
Ordinary RNN is neural network structures specifically designed for handling sequential data, as shown in Figure 3(a). RNN utilizes time-dependent relationships in sequences by storing previous state information to assist in current computations (Rumelhart et al., 1986). At tth time step, the calculation in an RNN unit can be divided into two steps: the first step is updating the hidden state (h_t), and the second step is calculating the output (y_t). The calculation formulas are as follows:

$$\begin{cases} h_t = \sigma(W_{xh} \cdot x_t + W_{hh} \cdot h_{t-1} + b_h) \\ y_t = \sigma(W_{hy} \cdot h_t + b_y) \end{cases} \tag{3}$$

where, h_t, x_t and y_t are the state, input, and output at tth time, respectively. W_{xh}, W_{hh} and W_{hy} are the weight parameters. b_h and b_y are bias parameters. σ is the nonlinear activation function.

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

It can be observed that Eq. (2) and (3) have a similar structure, as they both combine ODE and output equations. Therefore, in this study, we modify the ordinary RNN unit structure by replacing the original equations and parameters with those derived from the XAJ model, resulting in the XAJRNN. Similar to the ordinary RNN structure (Rumelhart et al., 1986), the backbone of the XAJRNN layer consists of recurrent units that provide memory of past sequences. In the XAJRNN layer structure, the connections between the recurrent units are represented by implicit discrete equations (Eq. 3), and the parameters (i.e., weight parameters and bias parameters) in the ordinary RNN unit are replaced by the physically meaningful parameters (as depict in Table 3) from the XAJ model. Niu et al. (2019) demonstrated the connection between RNN network architecture and numerical methods for ODE, theoretically supporting the use of XAJRNN for solving the dynamics of the XAJ model system. Table A4 summarizes the pseudocode for implementing the XAJRNN layer.



(a) Ordinary RNN unit **(b) Proposed XAJRNN unit**

Figure 3: The structures of ordinary RNN unit and proposed XAJRNN unit.

3.2 Model setup

3.2.1 EDL model

The proposed EDL model consists of the inputs, three neural network layers, and the outputs. First, the XAJRNN layer, as discussed in Section 3.1.2, processes the input data to generate outputs. This neural network layer follows the water balance principle and uses the physical sub-processes of the XAJ model to describe the runoff generation and routing process. The output variables, significantly influenced by the runoff process, are then passed to the Normalization layer. The purpose of this layer is to normalize

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

the data, helping the EDL model converge faster during training, increasing training stability, and reducing the impact of differences between features. Specifically, the normalization layer adjusts the data so that the mean is 0 and the standard deviation is 1. The normalized data is then passed into the LSTM layer for training. The choice of LSTM is based on the numerous studies demonstrating its ability to improve the performance of hydrological model simulations. Finally, the trained EDL model outputs the simulated runoff.

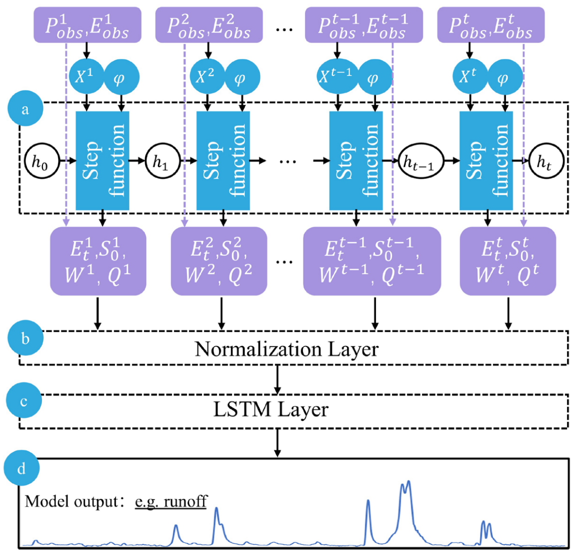


Figure 4: The structure of the proposed EDL model. (a) The structure network schematic graph of the XAJRNN layer. *h* represents the state variables in the XAJRNN layer (as shown in Table A1). φ represents the parameters in the XAJRNN layer (as shown in Table A3). (b), (c), and (d) represent the normalization layer, LSTM layer, and output results, respectively.

For the EDL model, similar to the traditional XAJ model, the XAJRNN layer takes areal mean rainfall (P_{obs}) and pan evaporation (E_{obs}) as input data, with a shape of [batch size, sequence length, 2 (input feature dimensions)]. The output physical quantities of interest are the actual evapotranspiration (E_t), the areal mean free water storage (S_0), the areal mean tension water storage (W), and outflow discharge of the basin (Q). These four physical quantities, along with the two input sequences, form a new sequence that serves as input for subsequent layers. The shape of the new input is [batch size, sequence length, 6 (new input feature dimensions)]. After passing through normalization layers and

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

LSTM layers, the final simulated flow sequence is obtained. The EDL model is trained using the *Adam* optimization algorithm, with *NSE* as the loss function and a learning rate of 0.001 for parameter estimation. The maximum number of iterations is set to 200, and training samples are reused in each training cycle until convergence is achieved (i.e., the absolute difference in *NSE* between consecutive cycles is less than 0.001).

3.2.1 Benchmark model

To compare the performance of the EDL model, two benchmark models are established. The first benchmark model is the LSTM model, which only uses the LSTM neural network layers. The input is the observed rainfall (P_{obs}) and pan evaporation (E_{obs}), the output is the simulated flow discharge. In order to reduce the impact of the training process on the model performance, the training process and hyperparameters of the LSTM model are the same as those of the EDL model.

The second benchmark model is the ordinary Xin'anjiang model, which also takes rainfall and evaporation as input to obtain the final simulated flow. Unlike the previous DL model, we use the genetic algorithm (GA) to calibrate model parameters. The GA searches in a population of points, uses the encoding of parameter sets, and uses probabilistic transition rules. There are four GA hyperparameters: crossover probability parameter (p_c), mutation probability parameter (p_m), population size parameter (p_{size}) and the maximum number of generation (T_{max}). Referring to the research results of Cheng et al. (2006), the above hyperparameters are set to, p_c=0.8, p_m=0.1, p_{size}=150, and T_{max}=1500.

3.3 Evaluation metrics

The overall performance of the models is evaluated using *NSE* (Nash and Sutcliffe, 1970), relative error (*RE*), and root mean squared error (*RMSE*). The calculation formulas are as follows:

$$NSE = 1 - \frac{\sum_{i=1}^{N}(Q_{o,i}-Q_{f,i})^2}{\sum_{i=1}^{N}(Q_{o,i}-\bar{Q}_o)^2} \tag{4}$$

$$RE = \frac{\sum_{i=1}^{N}Q_{f,i}-\sum_{i=1}^{N}Q_{o,i}}{\sum_{i=1}^{N}Q_{o,i}} \times 100\% \tag{5}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(Q_{f,i}-Q_{o,i})^2}{N}} \tag{6}$$

where N is the number of samples, Q_o, $\bar{Q}_o$ and $Q_{f,i}$ represent the observed inflows, mean value,

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

and simulated inflows, respectively.

To further illustrate the performance of the three models in flood simulation, the flood peak relative error (PRE) and the flood peak timing difference (ΔT) will be used to evaluate the simulating performance. The calculation formulas are as follows:

$$PRE = \frac{Q_{f,peak} - Q_{o,peak}}{Q_{o,peak}} \times 100\% \quad (7)$$

$$\Delta T = T_o - T_f \quad (8)$$

where $Q_{o,peak}$ and $Q_{f,peak}$ represent the observed and simulated peak inflow discharge. T_o and T_f are the observed and simulated times of peak discharges occurred. If ΔT is positive, the simulated peak discharge occurs early than the observed peak discharge; and vis vasa.

4 Results

4.1 Comparison of model performance

Table 1 presents the evaluation metrics for flood simulation using three models – EDL, XAJ, and LSTM – across two different river basins, namely Lushui River and Qingjiang River basins. The evaluation metrics includes *NSE*, *RE*, and *RMSE* values for both training and test phases. When simulating floods in the Lushui River basin, the EDL model achieved an *NSE* of 0.98, an *RE* below 3%, and a significantly lower *RMSE* compared to benchmark models, demonstrating a clear advantage. The performance of the LSTM model was similar to that of the EDL model but slightly inferior in terms of *RMSE* and *NSE*. In contrast, the traditional XAJ model yielded lower *NSE* values, larger *RE* deviations, and the highest *RMSE* values during both training and test phases, indicating significantly poorer simulation accuracy and stability compared to the DL models.

When simulating floods in the Qingjiang River basin, the overall performance of the three models was not as strong as in the Lushui River basin. However, the EDL model still performed best during the training phase, achieving the highest *NSE* (0.95), the lowest *RE* (1.1%), and an *RMSE* of 104.09 m^3/s. During the testing period, its performance was comparable to that of the LSTM model. In conclusion, the EDL model demonstrated exceptional flood simulation capabilities under intricate basin conditions, highlighting its substantial potential in flood simulation and forecasting.

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

Table 1: Comparative analysis of model simulation accuracy evaluation metrics.

Basin	Model	Training period			Test period		
		NSE	*RE* (%)	*RMSE* (m^3/s)	*NSE*	*RE* (%)	*RMSE* (m^3/s)
Lushui River	EDL	0.98	1.59	34.11	0.98	-2.69	43.71
	XAJ	0.86	-26.07	93.83	0.9	-18.5	89.6
	LSTM	0.97	-1.9	44.87	0.96	-0.61	54.27
Qingjiang River	EDL	0.95	1.1	104.09	0.92	-8.74	167.94
	XAJ	0.85	5.91	182.05	0.85	-7.92	231.17
	LSTM	0.9	-4.16	147.89	0.93	-6.19	155.71

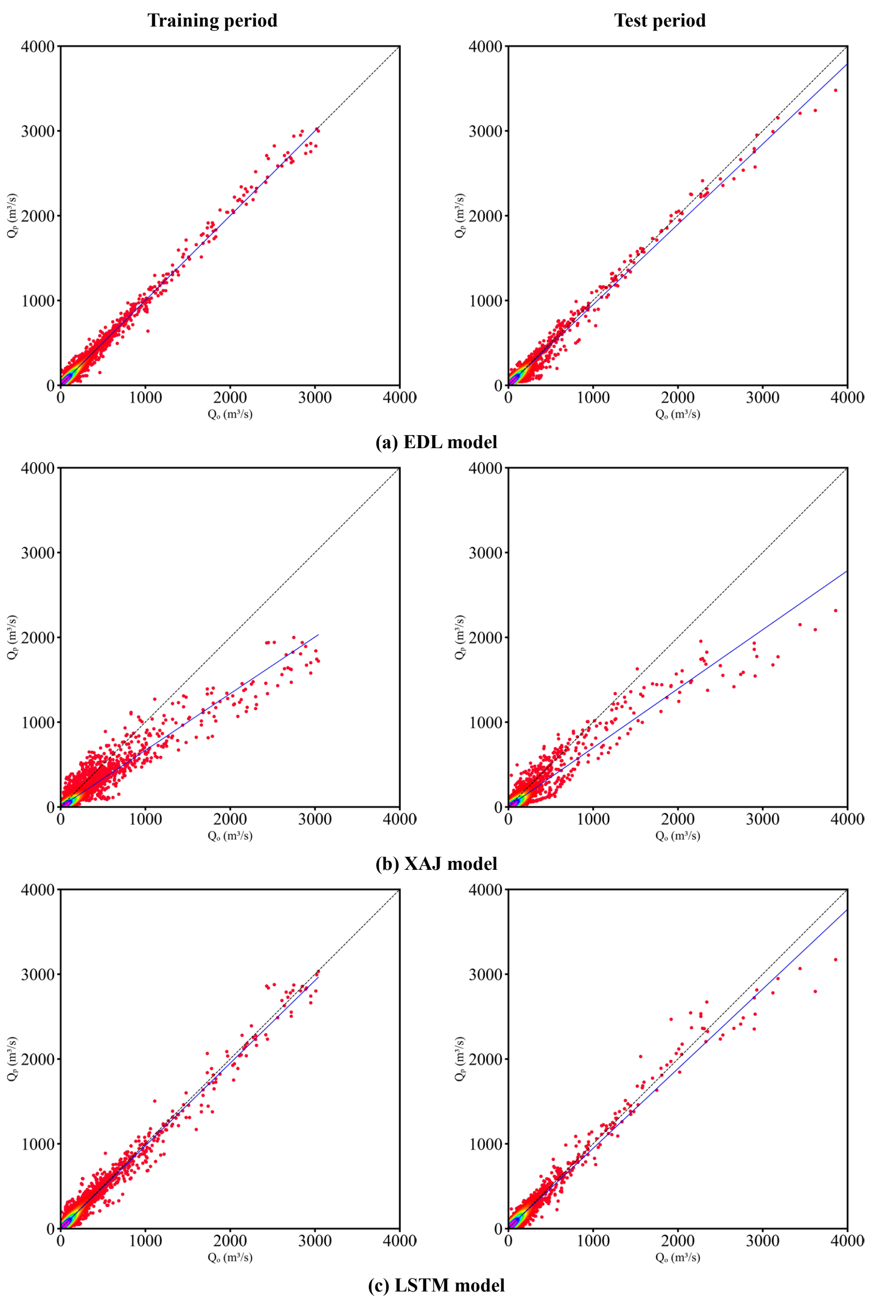
Figure 5 and Figure 6 respectively present the scatter plots of flood simulation results for the Lushui River and Qingjiang River basins using the EDL model, the XAJ model, and the LSTM model. In Figure 5(a), during the training period, the scatter points of the EDL model are tightly clustered and evenly distributed around the 1:1 ideal line. However, during the test period, in the high flow range, most of the scatter points are located below the 1:1 ideal line. As shown in Figure 5(b), the scatter points of the XAJ model are more dispersed in the low to medium flow range during the training period compared to the EDL model. In the high flow range, the scatter points are significantly below the 1:1 ideal line, with a similar distribution during the test period. In Figure 5(c), the scatter points of the LSTM model during the training period are evenly distributed around the 1:1 ideal line but are more dispersed than those of the EDL model. During the test period, the scatter points in the low to medium flow range are evenly distributed around the 1:1 ideal line, similar to the EDL model, but in the high flow range, most scatter points are below the 1:1 ideal line. Compared to the EDL model, the scatter points of the LSTM model deviate more noticeably from the 1:1 ideal line. Therefore, it can be concluded that the scatter plots of the EDL model are relatively better, while the scatter plots of the XAJ and LSTM models are relatively worse.

In Figure 6 (a), the scatter points of the EDL model are very compact and evenly distributed on both sides of the 1:1 ideal line during the training period. However, the scatter points are more loosely distributed, and some scatter points are obviously below the 1:1 ideal line during the test period. As shown in Figure 6 (b), the scatter points of the XAJ model during the training period are more scattered than the DEL model in the low to medium flow range, and the scatter points are obviously deviated from

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

the 1:1 ideal line in the high flow range. During the test period, the scatter point distribution of the XAJ model is looser, and the scatter points are farther from the 1:1 ideal line in the high flow range. As shown in Figure 6 (c), the scatter point distribution of the LSTM model during the training period is similar to that of the DEL model, but the scatter points deviate more obviously from the 1:1 ideal line in the high flow range. During the test period, it is also similar to the DEL model, but the scatter points are obviously below the 1:1 ideal line in the high flow range. Therefore, it can be concluded that the scatter plots of the DEL model are relatively better than these of the XAJ and LSTM models.

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025



344 **Figure 5: Scatter plots of observed (Q_o) and simulated (Q_p) flow discharges by three models in the Lushui**
345 **River basin.**

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

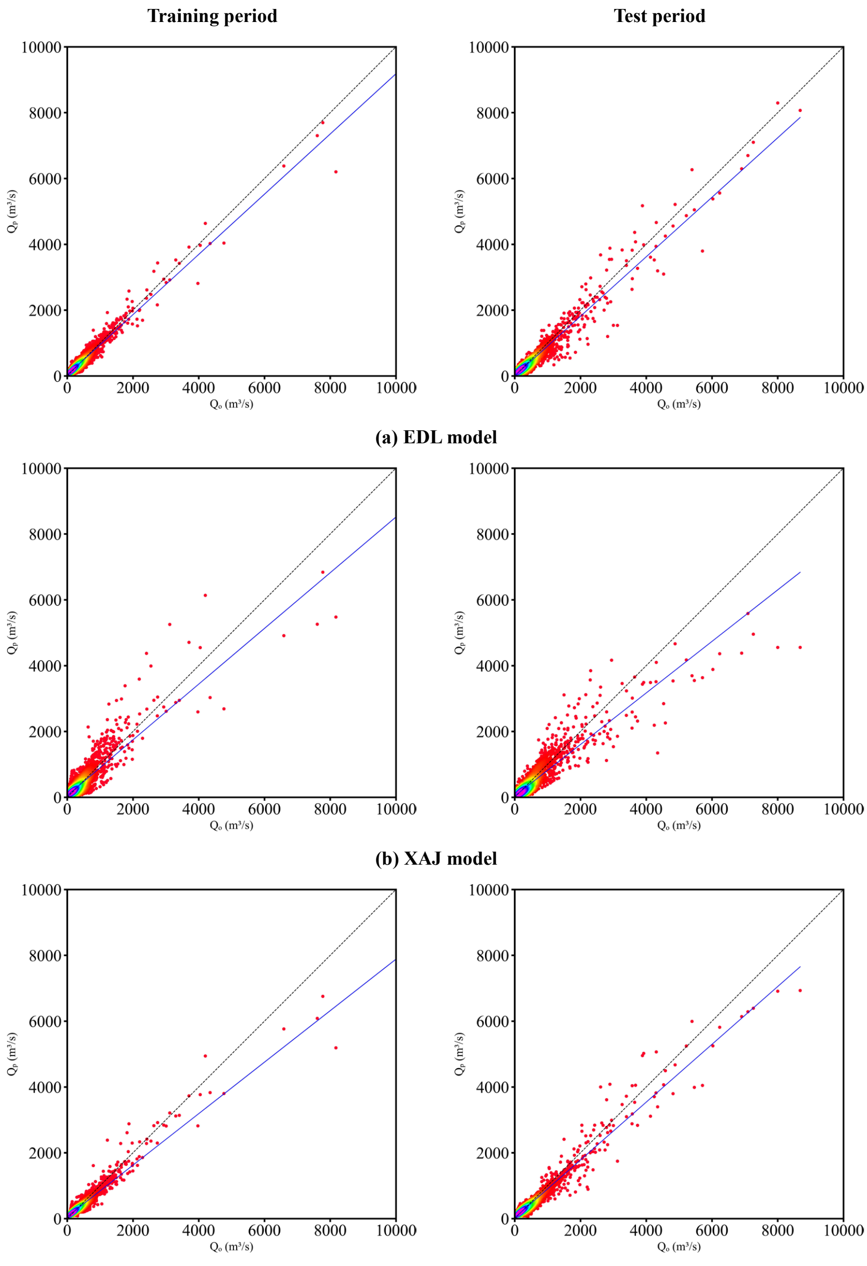


Figure 6: Scatter plots of observed (Q_o) and simulated (Q_p) flow discharge by three models in the Qingjiang River basin.

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

4.2 Comparison of the effectiveness of flood events simulation

Four major flood processes during the test period were selected in the Lushui River and Qingjiang River basins as case study. The simulation evaluation metrics of the three models (EDL model, XAJ model, and LSTM model) in these four flood processes are shown in Tables 2 and 3, respectively. These evaluation metrics include *NSE*, *RE*, *RMSE*, *PRE*, and ΔT.

In the Lushui River basin, the EDL model performed exceptionally well with high simulation accuracy, the *NSE* ranged from 0.97 to 0.99, *RE* ranged from -7.54% to -1.9%, and *RMSE* was as low as 71.66 m^3/s. Additionally, the EDL model's *PRE* was consistently below -12%, and ΔT remained at 0, highlighting its high reliability in simulating peak flow magnitude and timing. In contrast, the XAJ model's *NSE* ranged from 0.65 to 0.93, with significant *RE* deviations and *RMSE* values much higher than those of the EDL model, resulting in subpar overall performance. The LSTM model's *NSE* ranged from 0.91 to 0.97, close to that of the EDL model, but its *RE* and *RMSE* were slightly less favorable, resulting in marginally lower simulation accuracy.

In the Qingjiang River basin, the EDL model continued to demonstrate superior performance, with *NSE* exceeding 0.95 in all cases except for extreme events, *RE* ranging from -4.7% to -0.17% with minimal bias, and *RMSE* as low as 266.62 m^3/s. Although the EDL model's performance slightly declined during extreme events (e.g., 20200726), it still outperformed other models overall. The XAJ model's performance in the Qingjiang River basin was significantly inferior to the EDL model, with *NSE* varying widely, reaching as low as 0.32, *RE* deviations as high as 26.42%, and *RMSE* peaking at 1277.61 m^3/s, indicating its poor adaptability to complex events. The LSTM model's *NSE* ranged from 0.64 to 0.94, overall better than the XAJ model, but its accuracy and timeliness in peak flow simulation were insufficient during extreme events.

In summary, the EDL model exhibited the best overall performance in flood simulations for both the Lushui and Qingjiang River basins, with high accuracy, low bias, and excellent stability, particularly in regular flood events. Although the LSTM model's performance was close to that of the EDL model overall, it was slightly lacking in extreme events. In comparison, the XAJ model lagged significantly in both accuracy and adaptability, making it less suitable for precise flood simulation.

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

The outstanding performance of the EDL model highlights its immense potential in flood simulation, especially in complex basin conditions and extreme flood events. This further proves the advancement and feasibility of the model obtained by coupling deep learning technology with traditional hydrological models in the field of hydrological simulation, and provides strong tool support for solving flood forecasting problems.

Table 2: Results of flood simulation evaluation metrics for different events in the Lushui River basin.

Flood event	Model	NSE	RE (%)	$RMSE$(m^3/s)	PRE(%)	ΔT(h)
	EDL	0.98	-7.54	211.98	-11.99	0
20170624	XAJ	0.83	-19.31	561.31	-26.62	0
	LSTM	0.91	-12.29	421.53	-21.7	-3
	EDL	0.97	-3.53	138.01	-11.57	0
20170702	XAJ	0.65	-24.8	468.94	-37.34	30
	LSTM	0.93	0.76	204.43	-12.59	27
	EDL	0.99	-1.9	71.66	-0.9	0
20170813	XAJ	0.85	-18.28	351.25	-26.61	3
	LSTM	0.97	0.28	142.23	-7.31	0
	EDL	0.98	-3.85	85.92	-0.78	0
20190526	XAJ	0.93	1.23	175.86	-10.15	0
	LSTM	0.97	-4.49	109.74	0.45	0

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

Table 3: Results of flood simulation evaluation metrics for different events in the Qingjiang River basin.

Flood event	Model	*NSE*	*RE* (%)	*RMSE*(m^3/s)	*PRE*(%)	ΔT(h)
	EDL	0.95	-4.43	244.26	-7.64	0
20171003	XAJ	0.89	5.38	357.55	-8.59	0
	LSTM	0.85	-15.82	417.92	-27.02	0
	EDL	0.98	-0.17	266.62	-2.2	0
20200628	XAJ	0.95	5.48	467.13	-7.34	0
	LSTM	0.94	-3.58	506.72	-11.95	0
	EDL	0.95	-1.74	533.14	-4.48	6
20200717	XAJ	0.66	-23.15	1368.23	-28.61	0
	LSTM	0.91	-5.43	696.92	-20.18	0
	EDL	0.61	-4.7	966.73	-9.45	-6
20200726	XAJ	0.32	26.42	1277.61	-12.75	-6
	LSTM	0.64	1.84	931.87	-13.29	-6

To visually demonstrate the advantages of the EDL model, Figure 7 and 8 respectively present the hydrographs of four flood events in the Lushui River and Qingjiang River basins, comparing the simulation results of the EDL model, XAJRNN model, and LSTM model.

From Figure 7, it can be observed that the flood event in 20170624, all three models underestimated the peak flow to varying degrees, but the DEL model performed relatively better and more accurately simulated the timing of the peak flow. In the 20170702 and 20170813 compound flood events, the DEL model's simulated hydrograph during the peak flow phase was closer to the observed hydrograph compared to the XAJ model and the LSTM model. During flood event in 20190526, both the DEL model and the LSTM model simulated the peak flow phase well. However, in the 20170702 and 20190526 flood events, all three models exhibited delays, as evidenced by discrepancies in the rising speed during the flood rising phase compared to the observations. This may be related to the models' insufficient ability to simulate low flow conditions. Overall, the DEL model performed well in simulating the hydrographs of the Lushui River basin, accurately capturing both the peak flow and the timing of the peak.

Compared to the Lushui River basin, the simulation results of the three models in the Qingjiang River basin showed certain limitations, which were particularly evident in the 20200726 flood event as shown in Figure 8. All three models underestimated the peak flow, and the simulated peak was significantly delayed compared to the observed peak, indicating potential errors in the models' calculation

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

of the flow routing process, especially under complex terrain conditions. For the 20201003 and 20200717 flood events, the DEL model's simulated hydrograph was closer to the observed hydrograph compared to the XAJ model and the LSTM model. For flood event in 20200628, the LSTM model performed better during the recession phase but significantly underestimated the peak flow and failed to accurately simulate the rising phase. In contrast, the DEL model performed better during the rising and peak phases but exhibited delays during the recession phase. Overall, although some deviations in peak flow and timing exist, the DEL model still effectively captures the general flood trends in the Qingjiang River basin.

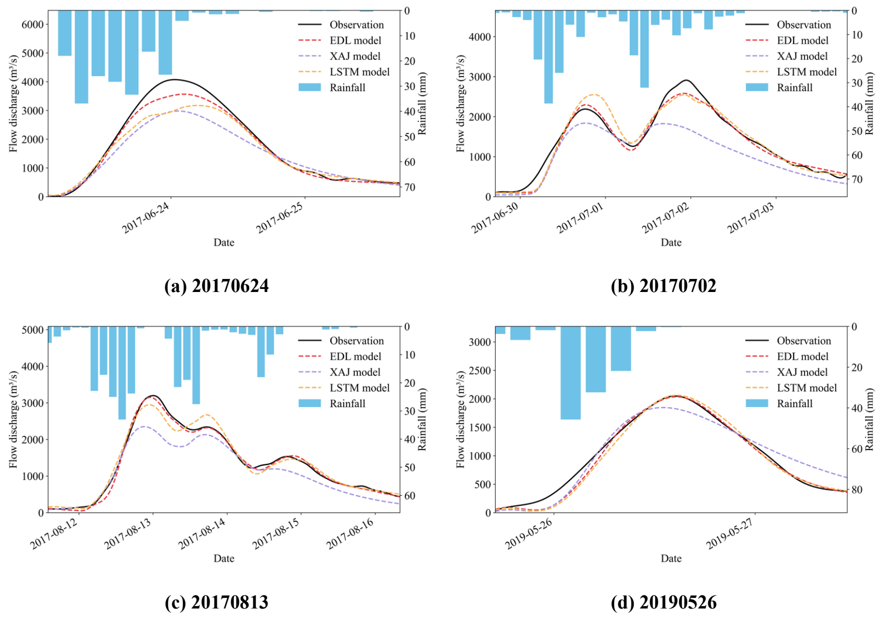


Figure 7: Comparison of observed and simulated flood hydrographs in the Lushui River basin.

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

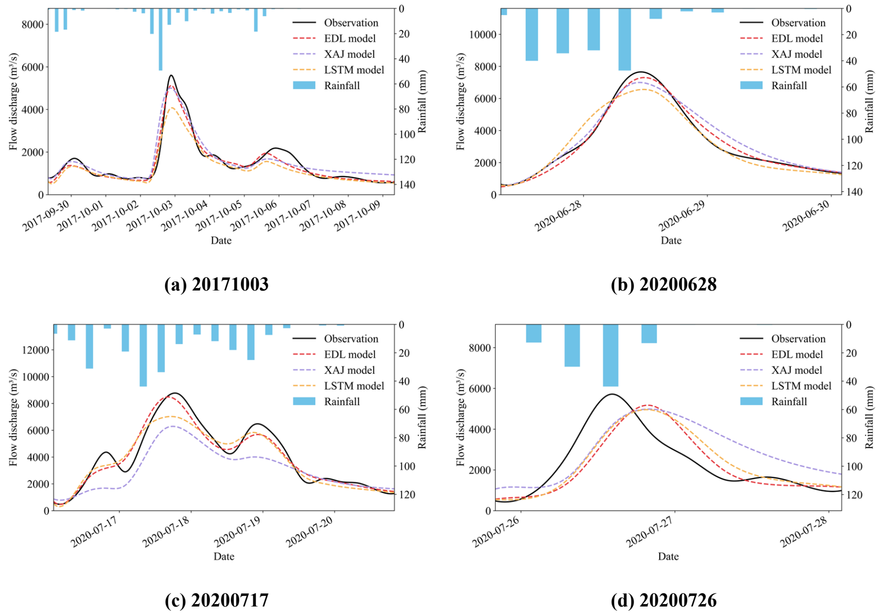


Figure 8: Comparison of observed and simulated flood hydrographs in the Qingjiang River basin.

5 Discussion

The XAJ model is a well-established hydrological model utilized for generalizing hydrological processes in a basin, including runoff generation and routing. However, when used in isolation, the model may struggle to adequately capture intricate nonlinear relationships, particularly evident in flood peak simulating where it might not fully account for the impacts of real-time meteorological changes. On the other hand, DL models, especially time-series models like LSTM, are adept at capturing complex nonlinear relationships within time series data. Nevertheless, they may encounter delays in accurately simulating flood peak timings (Chen et al., 2022; Cui et al., 2021; Xiang et al., 2024). To address this limitation, a fusion of the XAJ model and DL models can mitigate the weaknesses inherent in each approach. Specifically, the conventional hydrological model offers a foundation for physical processes that enhance the simulation of watershed hydrological responses, while the DL model can refine the output of the hydrological model, particularly in terms of temporal accuracy and the comprehension of nonlinear relationships. This hybrid approach allows the XAJ model to capture long-term dependencies in hydrological processes while enabling the DL model to make more precise simulations regarding flood

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

peak timing, thus effectively minimizing delays in flood peak simulation.

In traditional processes, DL models such as LSTM are commonly employed as post-processors to rectify the outcomes of hydrological models (Cho and Kim, 2022; Cui et al., 2021; Frame et al., 2021; Han and Morrison, 2022). Nevertheless, a significant drawback of this methodology stems from the inconsistent parameter tuning. Typically, the parameters of the hydrological model are initially fine-tuned to achieve the best simulation results, followed by the utilization of an LSTM model for refinement. However, as the parameter adjustments for the hydrological and DL models are carried out independently, this lack of synchronization in tuning may result in a less-than-optimal parameter combination, subsequently impacting the accuracy of the final simulations. In contrast, the EDL model we have devised undergoes synchronized training within the DL framework, where all parameters are collectively optimized. This strategy ensures that the parameters of the hydrological and DL models are simultaneously fine-tuned during training, leading to the optimal parameter combination and enhancing simulation accuracy. By tackling the inherent parameter mismatch issue seen in conventional approaches, this methodology elevates the overall performance of the model.

6 Conclusion

The present study proposes a novel EDL model that combines the physics-driven XAJRNN layer with the LSTM layer, successfully achieving accurate simulation of flood processes in the Lushui River basin and Qingjiang River basin. This model leverages the physical mechanisms of the XAJ model and the nonlinear representation capabilities of DL, demonstrating strong simulation performance. The key findings of this study are as summarized as follows:

(1) The EDL model demonstrates superior performance in both simulation accuracy and error control. It achieves an average *NSE* of 0.98 in the Lushui River basin and 0.94 in the Qingjiang River basin, demonstrating its outstanding fitting capability. Its average *RMSE* is 38.91 m^3/s in the Lushui River basin and 136.02 m^3/s in the Qingjiang River basin, significantly lower than that of benchmark models, highlighting its superior simulation accuracy. Although the *RE* is slightly higher during the testing phase, the combined analysis of the training phase RE shows that the EDL model consistently outperforms its

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

counterparts with stable error control.

(2) The EDL model demonstrates the highest stability in most flood simulations. Compared to the XAJ and LSTM models, the EDL model achieves smaller *PRE* values, indicating its superior accuracy in simulating flood peak magnitudes. Moreover, except for a few rare cases, the EDL model's ΔT is nearly zero, showcasing its unparalleled precision in simulating the timing of flood peaks.

(3) Compared to traditional single models, the EDL model not only significantly improves simulation accuracy but also enhances interpretability by integrating physical mechanisms. This innovative approach paves the way for the seamless integration of deep learning with hydrological physical mechanisms, advancing research in the field.

This study demonstrates that the proposed EDL model that combine physical mechanisms with DL is an effective way to improve flood simulation and forecasting accuracy. Future research should further explore how to more closely integrate complex physical mechanisms with neural network models to achieve higher simulative capabilities and scientific discovery potential.

Code availability

The code used to support the findings of this study are available from the corresponding author upon request.

Data availability

The data generated and/or analyzed during the current study are not publicly available for legal/ethical reasons but are available from the corresponding author on reasonable request.

Author contributions

Xin Xiang and Shenglian Guo conceived and designed the experiments; Xin Xiang performed the experiments and wrote the manuscript draft; Xin Xiang, Shenglian Guo, Chenglong Li, and Yun Wang reviewed and edited the manuscript.

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

Competing interests

The authors declare that they have no conflict of interest.

Acknowledgments

This study was financially supported by the National Natural Science Foundation of China (No. U2340205). The authors would like to thank the editor and anonymous reviewers whose comments and suggestions help to improve the manuscript.

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

Appendix A. Supplementary tables and texts

Table A1 describes the state variables of the XAJ model.

Table A2 describes the flux of the XAJ model.

Table A3 describes parameters and their value ranges of the XAJ model.

Table A4 describes the pseudocode of the XAJRNN layer.

Text A1 describes the details of the XAJ model.

Table A1: State variables of the XAJ model.

Module	State variable	Meaning	Unit
Evapotranspiration	W_u	Areal mean tension water storage of the upper soil layer	mm
	W_l	Areal mean tension water storage of the lower soil layer	
	W_d	Areal mean tension water storage of the deep soil layer	
Runoff separation	S_0	Areal mean free water storage	
Flow routing	O_i	Water storage of the interflow linear reservoir	
	O_g	Water storage of the groundwater linear reservoir	
	F_1	Water storage of the first reservoir in the Nash unit hydrograph	
	F_2	Water storage of the second reservoir in the Nash unit hydrograph	
	F_3	Water storage of the third reservoir in the Nash unit hydrograph	

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

491 **Table A2: Flux of the XAJ model.**

Module	Flux	Meaning	Unit
Evapotranspiration	P_{obs}	Areal mean rainfall	$mm/\Delta t$
	E_{obs}	Measured pan evaporation	
	P	Areal mean rainfall of the impervious area	
	R_{imp}	Runoff directly from the impervious area	
	P_n	Areal mean net rainfall	
	E_p	Potential evapotranspiration	
	E_u	Actual evapotranspiration of the upper soil layer	
	E_l	Actual evapotranspiration of the lower soil layer	
	E_d	Actual evapotranspiration of the deep soil layer	
	E_t	Actual evapotranspiration	
Runoff generation	R	Runoff produced from the previous area	$mm/\Delta t$
Runoff separation	R_s	Surface runoff	$mm/\Delta t$
	R_i	Interflow runoff	
	R_g	Groundwater runoff	
Flow routing	Q_i	Outflow of the interflow linear reservoir	$mm/\Delta t$
	Q_g	Outflow of the groundwater linear reservoir	
	Q_t	Total inflow to channel network	
	Q_1	Outflow of the first reservoir in the Nash unit hydrograph	
	Q_2	Outflow of the second reservoir in the Nash unit hydrograph	
	Q_3	Outflow of the third reservoir in the Nash unit hydrograph	
	Q	Outflow discharge of the basin	m^3/s

492

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

493 **Table A3: Parameters and their value ranges of the XAJ model.**

Module	Parameter	Meaning	Value range	Unit
Evapotranspiration	K_c	Ratio of potential evapotranspiration to pan evapotranspiration	[0.6,1.5]	-
	c	Coefficient of deep evapotranspiration	[0.01,0.2]	-
	W_{um}	Areal mean tension water capacity of the upper soil layer	[5,30]	mm
	W_{lm}	Areal mean tension water capacity of the lower soil layer	[60,90]	mm
	W_{dm}	Areal mean tension water capacity of the deep soil layer	[15,60]	mm
	A_{imp}	Ratio of the impervious area	[0.01,0.2]	-
Runoff generation	b	Exponent of the tension water capacity curve	[0.1,0.4]	-
Runoff separation	S_m	Areal mean of the free water capacity of the surface soil layer	[10,50]	mm
	ex	Exponent of the free water capacity curve	[1,1.5]	-
	K_i	Outflow coefficient of the free water storage to interflow	[0.1,0.55]	-
	K_g	Outflow coefficient of the free water storage to groundwater	[0.7-K_i]	-
Flow routing	c_i	Recession constant of interflow storage	[0.1,0.9]	-
	c_g	Recession constant of groundwater storage	[0.9,0.988]	-
	K_f	Storage-discharge coefficient of the linear reservoir in the Nash unit hydrograph	[0.1,10]	-

494

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

Table A4: Pseudocode of the XAJRNN layer.

Algorithm: the XAJRNN layer

Input: Sequences of observed rainfall $\{P_{obs}\}$ and observed evapotranspiration $\{E_{obs}\}$

State initialization: $W_u^{(0)}=0$, $W_l^{(0)}=0$, $W_d^{(0)}=0$, $S_0^{(0)}=0$, $O_i^{(0)}=0$, $O_g^{(0)}=0$, $O_s^{(0)}=0$, $F_1^{(0)}=0$, $F_2^{(0)}=0$ and $F_3^{(0)}=0$

Parameters: K_c, c, W_{um}, W_{lm}, W_{dm}, A_{imp}, b, S_m, ex, K_i, K_g, c_i, c_g, K_f, n

function step_function ([$P^{(i)}$, $E^{(i)}$], [$W_u^{(i-1)}$, $W_l^{(i-1)}$, $W_d^{(i-1)}$, $S_0^{(i-1)}$, $O_i^{(i-1)}$, $O_g^{(i-1)}$, $O_s^{(i-1)}$, $F_1^{(i-1)}$, $F_2^{(i-1)}$, $F_3^{(i-1)}$], parameters):

    Calculate $R^{(i)}$, $R_{imp}^{(i)}$, $E_t^{(i)}$, $P_n^{(i)}$, $W_u^{(i)}$, $W_l^{(i)}$, $W_d^{(i)}$ via Eqs. (A1) – (A20)
    Calculate $R_s^{(i)}$, $R_i^{(i)}$, $R_g^{(i)}$, $S_0^{(i)}$ via Eqs. (A21) – (A26)
    Calculate $Q_i^{(i)}$, $Q_g^{(i)}$, $Q_t^{(i)}$, $O_i^{(i)}$, $O_g^{(i)}$ via Eqs. (A27) – (A29)
    Calculate $Q_1^{(i)}$, $Q_2^{(i)}$, $Q_3^{(i)}$, $F_1^{(i)}$, $F_2^{(i)}$, $F_3^{(i)}$ via Eqs. (A30) – (A32)

return $W_u^{(i)}$, $W_l^{(i)}$, $W_d^{(i)}$, $S_0^{(i)}$, $O_i^{(i)}$, $O_g^{(i)}$, $O_s^{(i)}$, $F_1^{(i)}$, $F_2^{(i)}$ and $F_3^{(i)}$

do RNN (step function, [$\{P\}$, $\{E\}$], [$W_u^{(0)}$, $W_l^{(0)}$, $W_d^{(0)}$, $S_0^{(0)}$, $O_i^{(0)}$, $O_g^{(0)}$, $O_s^{(0)}$, $F_1^{(0)}$, $F_2^{(0)}$, $F_3^{(0)}$]) to obtain sequences of $\{W_u\}$, $\{W_l\}$, $\{W_d\}$, $\{S_0\}$, $\{O_i\}$, $\{O_g\}$, $\{O_s\}$, $\{F_1\}$, $\{F_2\}$, $\{F_3\}$

Calculate sequence of $\{Q\}$ via Eq. (A33)

Output: The sequence of runoff at the catchment outlet $\{Q\}$

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

Text A1

In the evapotranspiration, considering the uneven vertical distribution of soil, the XAJ model divides the soil into three layers and calculates the actual evapotranspiration using a three-layer soil moisture model. The calculation principle is as follows: The upper layer evaporates according to its evapotranspiration capacity. When the upper layer's water content is insufficient, the remaining evapotranspiration capacity is supplied by evaporation from the lower layers. The evaporation from the lower layers is proportional to the water storage in those layers. The ratio of the evaporation from the lower layer to the remaining evapotranspiration capacity must not be less than the coefficient of deep evapotranspiration (c). Otherwise, the lower layer water storage will supply the insufficient portion. If the lower layer water storage is not sufficient to compensate, the deep layer water storage will provide the remainder. The calculation formula is as follows:

$$P = P_{obs}(1 - A_{imp}) \tag*{（A1）}$$

$$R_{imp} = P_{obs} \times A_{imp}$$

$$E_p = K_c E_{obs} \tag*{（A2）}$$

(1) When $W_u + P \geq E_p$,

$$E_u = E_p;\ E_l = 0;\ E_d = 0 \tag*{(A2)}$$

(2) When $W_u + P < E_p$ and $W_l \geq c \times W_{lm}$,

$$E_u = W_u + P; E_l = (E_p - E_u) \times W_l / W_{lm};\ E_d = 0 \tag*{(A3)}$$

(3) When $W_u + P < E_p$ and $c \times (E_p - E_u) \leq W_l < c \times W_{lm}$,

$$E_u = W_u + P;\ E_l = c \times (E_p - E_u);\ E_d = 0 \tag*{(A4)}$$

(4) When $W_u + P < E_p$ and $c \times (E_p - E_u) > W_l$,

$$E_u = W_u + P;\ E_l = W_l;\ E_d = c \times (E_p - E_u) - E_l \tag*{(A5)}$$

$$E_t = E_u + E_l + E_d \tag*{（A6）}$$

$$P_n = \begin{cases} P - E_t, & P \geq E_t \\ 0, & P < E_t \end{cases} \tag*{（A7）}$$

The runoff generation calculation uses the tension water capacity curve. First, it is necessary to calculate the areal mean tension water storage (W) and the areal mean tension water capacity (W_m):

$$W = W_u + W_l + W_d \tag*{（A8）}$$

$$W_m = W_{um} + W_{lm} + W_{dm} \tag*{(A9)}$$

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

The vertical coordinate value (α) corresponding to the areal mean tension water storage (W) on the tension water capacity curve is calculated as:

$$\alpha = W_m \times (b+1) \times \left[1 - \left(1 - \frac{W}{W_m}\right)^{\frac{1}{1+b}}\right] \quad \text{(A11)}$$

Calculate the runoff produced from the previous area:

$$R = \begin{cases} P_n + W - W_m + W_m\left(1 - \frac{P_n+\alpha}{W_m \times (b+1)}\right)^{b+1}, & P_n + \alpha \le W_m \times (b+1) \\ P_n + W - W_m, & P_n + \alpha > W_m \times (b+1) \end{cases} \quad \text{(A12)}$$

Finally, update the areal mean tension water storage of the upper, lower, and deep soil layer of the watershed at the end of the current period, which will serve as the initial values for the next period:

$$W_u = W_u + P - E_t - R \quad \text{(A13)}$$

$$W_l = W_l - E_l \quad \text{(A14)}$$

$$W_d = max(W_d - E_d, 0) \quad \text{(A15)}$$

When $W_u > W_{um}$,

$$W_l = W_l + W_u - W_{um} \quad \text{(A16)}$$

$$W_u = W_{um} \quad \text{(A17)}$$

When $W_l > W_{lm}$,

$$W_d = W_d + W_l - W_{lm} \quad \text{(A18)}$$

$$W_l = W_{lm} \quad \text{(A19)}$$

$$W_d = min(W_d, W_{dm}) \quad \text{(A20)}$$

The runoff separation uses the free water capacity curve. The vertical coordinate value (β) corresponding to the areal mean free water storage (S_0) is:

$$\beta = S_m \times (ex+1) \times [1 - (1 - S_0/S_m)^{\frac{1}{1+ex}}] \quad \text{(A21)}$$

Therefore, the surface runoff (R_s) is:

$$R_s = \begin{cases} R + \{S_0 - S_m + S_m[1 - \frac{(P_n+\beta)}{S_m \times (ex+1)}]^{ex+1}\}\frac{R}{P_n}, & P_n + \beta \le S_m \times (ex+1) \\ R + (S_0 - S_m)\frac{R}{P_n}, & P_n + \beta > S_m \times (ex+1) \end{cases} \quad \text{(A22)}$$

$$R_s = R_s + R_{imp} \quad \text{(A23)}$$

The interflow runoff (R_i):

$$R_i = K_i \times S_0 \times \frac{R}{P_n} \quad \text{(A24)}$$

The groundwater runoff (R_g):

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

$$R_g = K_g \times S_0 \times \frac{R}{P_n} \quad \text{(A25)}$$

Calculate the areal mean free water storage (S_0) at the end of the current period, which will serve as the initial value for the next period, as:

$$S_0 = S_0 + \left(R - R_s - R_i - R_g\right) \times \frac{P_n}{R} \quad \text{(A26)}$$

The flow routing module consists of two submodules: hillslope and channel network routing. The hillslope routing adopts a linear reservoir approach, while the channel network routing uses the Nash unit hydrograph. The calculation formula for the linear reservoir is as follows:

$$Q_i = -O_i \times \ln c_i \quad \text{(A27)}$$

$$Q_g = -O_g \times \ln c_g \quad \text{(A28)}$$

The total inflow to channel network is equal to the sum of the surface runoff (R_s), the outflow of the interflow linear reservoir (Q_i), and the outflow of the groundwater linear reservoir (Q_g). The calculation formula is as follows:

$$Q_t = R_s + Q_i + Q_g \quad \text{(A29)}$$

The calculation formula for the Nash unit hydrograph reservoir is as follows:

$$Q_1 = F_1/K_f \quad \text{(A30)}$$

$$Q_2 = F_2/K_f \quad \text{(A31)}$$

$$Q_3 = F_3/K_f \quad \text{(A32)}$$

The calculation formula for the outflow discharge of the basin (Q) is as follows:

$$Q = Q_3 \times 1000 \times F/\Delta t \quad \text{(A33)}$$

where F is the watershed area，km^2. Δt is the input time step, s.

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

Reference

Bindas, T., Tsai, W., Liu, J., Rahmani, F., Feng, D., Bian, Y., Lawson, K., and Shen, C.: Improving river routing using a differentiable Muskingum-Cunge model and physics-informed machine learning, Water Resour. Res., 60, e2023WR035337, https://doi.org/10.1029/2023WR035337, 2024.

Boucher, M. A., Quilty, J., and Adamowski, J.: Data assimilation for streamflow forecasting using extreme learning machines and multilayer perceptrons, Water Resour. Res., 56, e2019WR026226, https://doi.org/10.1029/2019WR026226, 2020.

Chen, C., Jiang, J., Liao, Z., Zhou, Y., Wang, H., and Pei, Q.: A short-term flood prediction based on spatial deep learning network: A case study for Xi County, China, J. Hydrol., 607, 127535, https://doi.org/10.1016/j.jhydrol.2022.127535, 2022.

Cheng, C., Zhao, M., Chau, K. W., and Wu, X.: Using genetic algorithm and TOPSIS for Xinanjiang model calibration with a single procedure, Journal of Hydrology, 316, 129–140, https://doi.org/10.1016/j.jhydrol.2005.04.022, 2006.

Cho, K. and Kim, Y.: Improving streamflow prediction in the WRF-Hydro model with LSTM networks, J. Hydrol., 605, 127297, https://doi.org/10.1016/j.jhydrol.2021.127297, 2022.

Cui, Z., Zhou, Y., Guo, S., Wang, J., Ba, H., and He, S.: A novel hybrid XAJ-LSTM model for multi-step-ahead flood forecasting, Hydrol. Res., 52, 1436–1454, https://doi.org/10.2166/nh.2021.016, 2021.

De la Fuente, L. A., Ehsani, M. R., Gupta, H. V., and Condon, L. E.: Toward interpretable LSTM-based modeling of hydrological systems, Hydrol. Earth Syst. Sci., 28, 945–971, https://doi.org/10.5194/hess-28-945-2024, 2024.

Feng, D., Liu, J., Lawson, K., and Shen, C.: Differentiable, learnable, regionalized process-based models with multiphysical outputs can approach state-ff-the-art hydrologic prediction accuracy, Water Resour. Res., 58, e2022WR032404, https://doi.org/10.1029/2022WR032404, 2022.

Feng, D., Beck, H., Lawson, K., and Shen, C.: The suitability of differentiable, physics-informed machine learning hydrologic models for ungauged regions and climate change impact assessment, Hydrol. Earth Syst. Sci., 27, 2357–2373, https://doi.org/10.5194/hess-27-2357-2023, 2023.

Frame, J. M., Kratzert, F., Raney II, A., Rahman, M., Salas, F. R., and Nearing, G. S.: Post-processing the national water model with long short-term memory networks for streamflow predictions and model diagnostics, J. Am. Water Resour. Assoc., 57, 885–905, https://doi.org/10.1111/1752-1688.12964, 2021.

Frame, J. M., Kratzert, F., Gupta, H. V., Ullrich, P., and Nearing, G. S.: On strictly enforced mass conservation constraints for modelling the Rainfall-Runoff process, Hydrol. Process., 37, e14847, https://doi.org/10.1002/hyp.14847, 2023.

Frank, E. S., Zhen, Y., Han, F., Shailesh, T., and Matthias, D.: An introductory review of deep learning for prediction models with big data, Front. Artif. Intell., 3, 4, https://doi.org/10.3389/frai.2020.00004, 2020.

Guido, B. I., Popescu, I., Samadi, V., and Bhattacharya, B.: An integrated modeling approach to evaluate the impacts of nature-based solutions of flood mitigation across a small watershed in the southeast United States, Nat. Hazard. Earth Sys., 23, 2663–2681, https://doi.org/10.5194/nhess-23-2663-2023, 2023.

Han, H. and Morrison, R. R.: Improved runoff forecasting performance through error predictions using a deep-learning approach, J. Hydrol., 608, 127653, https://doi.org/10.1016/j.jhydrol.2022.127653, 2022.

Hirabayashi, Y., Mahendran, R., Koirala, S., Konoshima, L., Yamazaki, D., Watanabe, S., Kim, H., and Kanae, S.: Global flood risk under climate change, Nat. Clim. Chang., 3, 816–821, https://doi.org/10.1038/nclimate1911, 2013.

Hochreiter, S. and Schmidhuber, J.: Long short-term memory, Neural Comput., 9, 1735–1780, https://doi.org/10.1162/neco.1997.9.8.1735, 1997.

Hoedt, P. J., Kratzert, F., Klotz, D., Halmich, C., Holzleitner, M., Nearing, G., Hochreiter, S., and Klambauer, G.: MC-LSTM: Mass-conserving LSTM, in: International Conference on Machine Learning, International Conference on Machine Learning, San Diego, Web of Science ID: WOS:000683104604027, 2021.

Jiang, S., Zheng, Y., and Solomatine, D.: Improving AI system awareness of geoscience knowledge: Symbiotic integration of physical approaches and deep learning, Geophys. Res. Lett., 47, e2020GL088229, https://doi.org/10.1029/2020GL088229, 2020.

La Follette, P. T., Teuling, A. J., Addor, N., Clark, M., Jansen, K., and Melsen, L. A.: Numerical daemons of hydrological models are summoned by extreme precipitation, Hydrol. Earth Syst. Sci., 25, 5425–5446, https://doi.org/10.5194/hess-25-5425-2021, 2021.

Li, H., Zhang, C., Chu, W., Shen, D., and Li, R.: A process-driven deep learning hydrological model for daily rainfall-runoff simulation, J. Hydrol., 637, 131434, https://doi.org/10.1016/j.jhydrol.2024.131434, 2024.

Nash, J. E. and Sutcliffe, J. V.: River flow forecasting through conceptual models part I - A discussion of principles, J. Hydrol., 10, 282–290, https://doi.org/10.1016/0022-1694(70)90255-6, 1970.

Nearing, G. S., Kratzert, F., Sampson, A. K., Pelissier, C. S., Klotz, D., Frame, J. M., Prieto, C., and Gupta, H. V.: What role does hydrological science play in the age of machine learning? Water Resour. Res., 57, e2020WR028091, https://doi.org/10.1029/2020WR028091, 2021.

Niu, M. Y., Horesh, L., and Chuang, I.: Recurrent neural networks in the eye of differential equations, https://doi.org/10.48550/arXiv.1904.12933, 29 April 2019.

Pokharel, S., Roy, T., and Admiraal, D.: Effects of mass balance, energy balance, and storage-discharge constraints on LSTM for streamflow prediction, Environ. Modell. Softw., 166, 105730, https://doi.org/10.1016/j.envsoft.2023.105730, 2023.

Roy, A., Kasiviswanathan, K. S., Patidar, S., Adeloye, A. J., Soundharajan, B. S., and Ojha, C. S. P.: A physics-aware machine learning-based framework for minimizing prediction uncertainty of hydrological models, Water Resour. Res., 59, e2023WR034630, https://doi.org/10.1029/2023WR034630, 2023.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J.: Learning representations by back-propagating errors, Nature, 323, 533–536, https://doi.org/10.1038/323533a0, 1986.

Shen, C.: A transdisciplinary review of deep learning research and its relevance for water resources scientists, Water Resour. Res., 54, 8558–8593, https://doi.org/10.1029/2018WR022643, 2018.

Shen, C., Appling, A. P., Gentine, P., Bandai, T., Gupta, H., Tartakovsky, A., Baity-Jesi, M., Fenicia, F., Kifer, D., Li, L., Liu, X., Ren, W., Zheng, Y., Harman, C. J., Clark, M., Farthing, M., Feng, D., Kumar, P., Aboelyazeed, D., Rahmani, F., Song, Y., Beck, H. E., Bindas, T., Dwivedi, D., Fang, K.,

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

Höge, M., Rackauckas, C., Mohanty, B., Roy, T., Xu, C., and Lawson, K.: Differentiable modelling to unify machine learning and physical models for geosciences, Nat. Rev. Earth Environ., 4, 552–567, https://doi.org/10.1038/s43017-023-00450-9, 2023.

Singh, V. P.: Estimation of parameters of a uniformly nonlinear surface runoff model, Hydrol. Res., 8, 33–46, https://doi.org/10.2166/nh.1977.0003, 1977.

Song, Y., Knoben, W. J. M., Clark, M. P., Feng, D., Lawson, K., Sawadekar, K., and Shen, C.: When ancient numerical demons meet physics-informed machine learning: adjoint-based gradients for implicit differentiable modeling, Hydrol. Earth Syst. Sci., 28, 3051–3077, https://doi.org/10.5194/hess-28-3051-2024, 2024.

Thaisiam, W., Yomwilai, K., and Wongchaisuwat, P.: Utilizing sequential modeling in collaborative method for flood forecasting, J. Hydrol., 636, 131290, https://doi.org/10.1016/j.jhydrol.2024.131290, 2024.

Tsai, W. P., Feng, D., Pan, M., Beck, H., Lawson, K., Yang, Y., Liu, J., and Shen, C.: From calibration to parameter learning: Harnessing the scaling effects of big data in geoscientific modeling, Nat. Commun., 12, 5988, https://doi.org/10.1038/s41467-021-26107-z, 2021.

Wang, C., Jiang, S., Zheng, Y., Han, F., Kumar, R., Rakovec, O., and Li, S.: Distributed hydrological modeling with physics-encoded deep learning: A general framework and its application in the Amazon, Water Resour. Res., 60, e2023WR036170, https://doi.org/10.1029/2023WR036170, 2024.

Wang, N., Zhang, D., Chang, H., and Li, H.: Deep learning of subsurface flow via theory-guided neural network, J. Hydrol., 584, 124700, https://doi.org/10.1016/j.jhydrol.2020.124700, 2020.

Worland, S. C., Steinschneider, S., Asquith, W., Knight, R., and Wieczorek, M.: Prediction and inference of flow duration curves using multioutput neural networks, Water Resour. Res., 55, 6850–6868, https://doi.org/10.1029/2018WR024463, 2019.

Xiang, X., Guo, S., Cui, Z., Wang, L., and Xu, C. Y.: Improving flood forecast accuracy based on explainable convolutional neural network by Grad-CAM method, J. Hydrol., 642, 131867, https://doi.org/10.1016/j.jhydrol.2024.131867, 2024.

Xie, K., Liu, P., Zhang, J., Han, D., Wang, G., and Shen, C.: Physics-guided deep learning for rainfall-runoff modeling by considering extreme events and monotonic relationships, J. Hydrol., 603,

https://doi.org/10.5194/egusphere-2025-279
Preprint. Discussion started: 7 February 2025

127043, https://doi.org/10.1016/j.jhydrol.2021.127043, 2021.

Yann, L., Yoshua, B., and Geoffrey, H.: Deep learning, Nature, 521, 436–444, https://doi.org/10.1038/nature14539, 2015.

Zhao, R.: The Xinanjiang model applied in China, J. Hydrol., 135, 371–381, https://doi.org/10.1016/0022-1694(92)90096-E, 1992.

Zhao, R.: A non-linear system model for basin concentration, J. Hydrol., 142, 477–482, https://doi.org/10.1016/0022-1694(93)90024-4, 1993.

Zhong, L., Lei, H., Li, Z., and Jiang, S.: Advancing streamflow prediction in data-scarce regions through vegetation-constrained distributed hybrid ecohydrological models, J. Hydrol., 645, 132165, https://doi.org/10.1016/j.jhydrol.2024.132165, 2024a.

Zhong, L., Lei, H., and Yang, J.: Development of a distributed physics-informed deep learning hydrological model for data-scarce regions, Water Resour. Res., 60, e2023WR036333, https://doi.org/10.1029/2023WR036333, 2024b.

Zhou, Y., Guo, S., Liu, P., and Xu, C.: Joint operation and dynamic control of flood limiting water levels for mixed cascade reservoir systems, J. Hydrol., 519, 248–257, https://doi.org/10.1016/j.jhydrol.2014.07.029, 2014.