water

MDPI

Article

Uncertainty Quantification in Machine Learning Modeling for Multi-Step Time Series Forecasting: Example of Recurrent Neural Networks in Discharge Simulations

Tianyu Song, Wei Ding *, Haixing Liu, Jian Wu, Huicheng Zhou and Jinggang Chu

School of Hydraulic Engineering, Dalian University of Technology, Dalian 116024, China; songtianyu@mail.dlut.edu.cn (T.S.); hliu@dlut.edu.cn (H.L.); wjdlut1992@163.com (J.W.); hczhou@dlut.edu.cn (H.Z.); jgchu@dlut.edu.cn (J.C.)

* Correspondence: weiding@dlut.edu.cn

Received: 8 February 2020; Accepted: 22 March 2020; Published: 23 March 2020

Abstract: As a revolutionary tool leading to substantial changes across many areas, Machine Learning (ML) techniques have obtained growing attention in the field of hydrology due to their potentials to forecast time series. Moreover, a subfield of ML, Deep Learning (DL) is more concerned with datasets, algorithms and layered structures. Despite numerous applications of novel ML/DL techniques in discharge simulation, the uncertainty involved in ML/DL modeling has not drawn much attention, although it is an important issue. In this study, a framework is proposed to quantify uncertainty contributions of the sample set, ML approach, ML architecture and their interactions to multi-step time-series forecasting based on the analysis of variance (ANOVA) theory. Then a discharge simulation, using Recurrent Neural Networks (RNNs), is taken as an example. Long Short-Term Memory (LSTM) network, a state-of-the-art DL approach, was selected due to its outstanding performance in time-series forecasting, and compared with simple RNN. Besides, novel discharge forecasting architecture is designed by combining the expertise of hydrology and stacked DL structure, and compared with conventional design. Taking hourly discharge simulations of Anhe (China) catchment as a case study, we constructed five sample sets, chose two RNN approaches and designed two ML architectures. The results indicate that none of the investigated uncertainty sources are negligible and the influence of uncertainty sources varies with lead-times and discharges. LSTM demonstrates its superiority in discharge simulations, and the ML architecture is as important as the ML approach. In addition, some of the uncertainty is attributable to interactions rather than individual modeling components. The proposed framework can both reveal uncertainty quantification in ML/DL modeling and provide references for ML approach evaluation and architecture design in discharge simulations. It indicates uncertainty quantification is an indispensable task for a successful application of ML/DL.

Keywords: uncertainty quantification; Machine Learning; Deep Learning; Long Short-Term Memory; time-series forecasting; discharge simulation

1. Introduction

Recently, Machine Learning (ML) techniques have obtained growing attention and Deep Learning (DL) techniques have led to substantial changes across many areas of study [1]. As a subfield of ML, DL is more concerned with datasets, algorithms and layered structures. Owing to the enhanced capability to characterize the system complexity and flexible structure, research and applications of ML/DL for time-series prediction are proliferating and promising. In hydrology, the advent of ML/DL techniques has encouraged novel applications or substantially improved old ones [2,3]. Though ML/DL

approaches are also fundamentally black-box methods, they can be developed with minimal inputs and applied easily, without considering the redundant physical mechanism of the watershed system. ML approaches and their hybrid application have performed well or comparable when applied to predict hydrological time series, compared to physically based models [4]. The continuous improvement of ML/DL approaches and architectures demonstrates their suitability for discharge simulation. However, the application of ML/DL techniques depends heavily on datasets, algorithms and layered structures, so it is necessary to quantify uncertainty contributions of each component in ML/DL modeling.

In ML modeling, quantifying uncertainty is an indispensable task to improve the validity and predictability of ML model applications. Various uncertainty sources can be mainly divided into three types: sample set partitioning, ML approach selection and ML architectures design. Most of the previous studies only focused on the single source of uncertainty. Jha et al. [5] investigated the impact of dataset uncertainties on the performance of the ML model predictions. Rahmati et al. [6] quantified the predictive uncertainty of three ML approaches with quantile regression and uncertainty estimation based on local errors and clustering. Li et al. [7] demonstrated the uncertainty quantification analysis of neural network structure (the number of layers and nodes in each layer). Most studies of hydrological time-series forecasting have focused on the application of algorithms and rarely discussed the importance of model architectures' design to simulation results [4]. Moreover, relatively few studies have been conducted to investigate the interaction among different uncertainty sources and their contributions to ML modeling, so it is necessary to establish a framework for quantifying uncertainty contributions of the sample set, ML approach, ML architectures and their interaction in ML modeling.

The choice of promising ML/DL approach is one of the most prominent activities for time-series forecasting. Among all ML approaches, Recurrent Neural Network (RNN) has established a reputation to deal with time series by its particular recurrent neural connections. Almost all exciting results based on RNNs have been achieved by Long Short-Term Memory (LSTM) network [8], so LSTM network has become the focus of DL. LSTM network was proposed by the German researchers Hochreiter and Schmidhuber as a solution to the long-term dependencies problem [9]. Due to the special network architecture, LSTM networks have great learning ability in dealing with time-series forecasting in many applications, such as speech recognition [10], stock price volatility [11], sentiment analysis [12], traffic forecast [13] and disease diagnosis [14]. Few attempts have been made to apply LSTM networks to hydrological problems [15]. In addition, most of the previous studies on evaluation of the LSTM network only focused on the comparison with other approaches, such as ANN [3], support vector regression [16] and RNNs [17]. However, there is no evaluation of the LSTM network based on the interactions between LSTM networks and other components of ML modeling.

To our knowledge, this idea serves as the first attempt to apply the analysis of variance (ANOVA) theory to the uncertainty quantification of ML modeling for multi-step time-series forecasting. The separate contributions and analysis to the total uncertainty budget can be estimated by ANOVA [18]. This paper focuses on comparing the uncertainty contributions ratio of components in ML modeling rather than evaluating the uncertainty of ML algorithms. The overall objectives of this paper are (1) to propose a framework to quantify uncertainty contributions of the sample set, ML approach, ML architecture and their interactions in ML modeling and evaluate multi-step time series forecasting models; and (2) to evaluate LSTM networks in discharge simulations under the proposed framework. In this paper, the standard LSTM cell is chosen because no variant can surpass the standard LSTM cell in all aspects [8].

To implement the proposed framework and evaluate LSTM networks, hourly flash-flood forecasting in small catchments is taken as a case study due to its more complex nonlinear relationship between rainfall and runoff in small temporal and spatial scales' simulation. Five sample sets are constructed in the case study. Simple RNN is chosen as a comparison. To evaluate the interactions between LSTM networks and ML architecture, a novel ML architecture is designed by combining the expertise of hydrology and stacked DL structure, and compared with another conventional architecture. Then the simulated results of five sample sets, two RNN approaches and two ML architectures are

evaluated and analyzed, respectively. Finally, we estimate the total ensemble uncertainty of the ML modeling and quantify the contributions of different uncertainty sources in discharge simulations.

2. Methodology

2.1. The Proposed Framework

Figure 1 shows the diagrammatic flowchart of the proposed framework for uncertainty quantification in ML modeling for multi-step time-series forecasting. This framework is composed of 8 steps: Steps 1–3 include three parts of ML modeling: partition sample sets, select ML approaches and design ML architectures. In Step 4, ML modeling combination scheme is established. Step 5 is to train all the combining schemes and simulate discharge. In Step 6, simulated results are compared to evaluate sample sets, ML approaches and ML architectures. The purpose of Step 7 is to quantify the individual and interactive contributions of different uncertainty sources, using ANOVA. Finally, Step 8 is to evaluate multi-step time-series forecasting models. In this paper, discharge simulations using by RNNs is taken as an example.

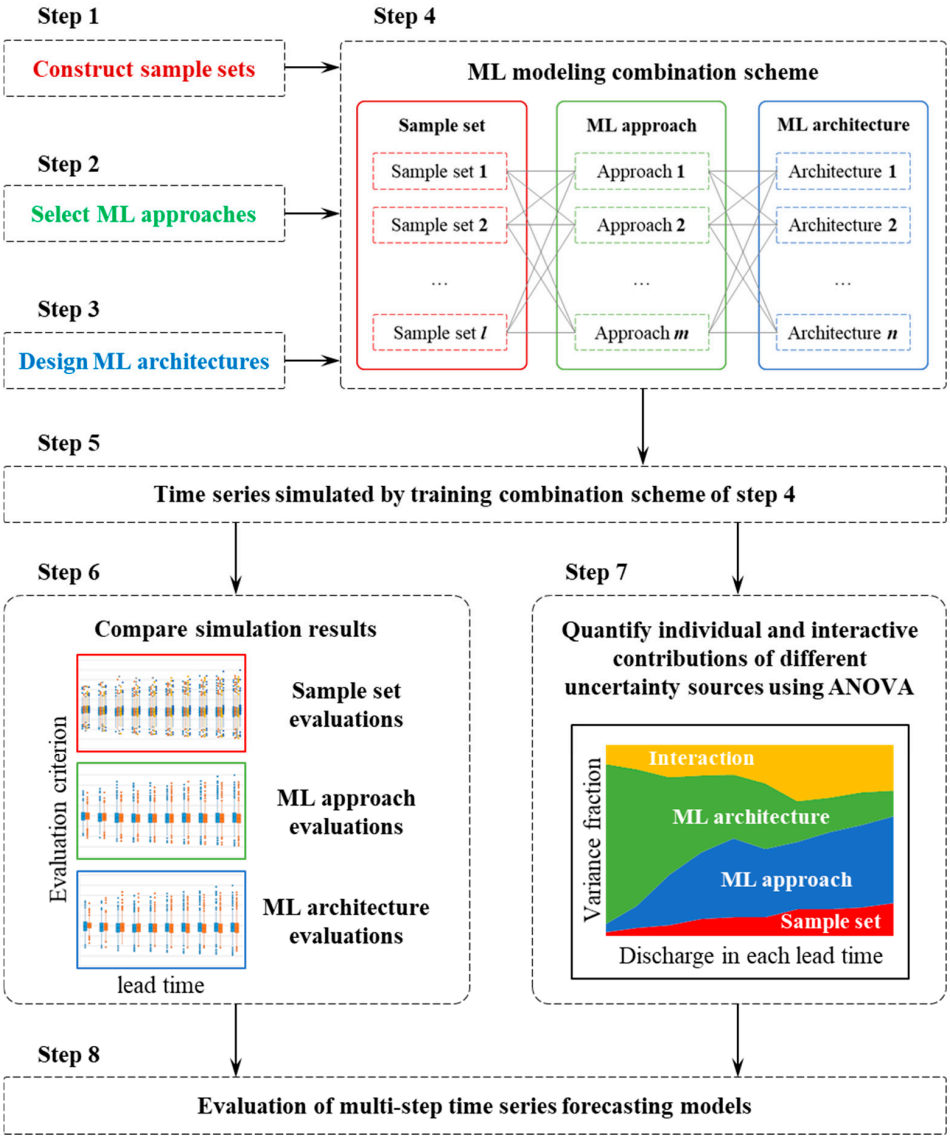


Figure 1. Diagrammatic flowchart of the proposed framework for uncertainty quantification in Machine Learning (ML) modeling for multi-step time-series forecasting.

2.2. RNN Approaches

Two RNN approaches were selected for discharge simulations in this paper. Figures 2 and 3 show the internals of a Simple RNN and LSTM memory cell, respectively.

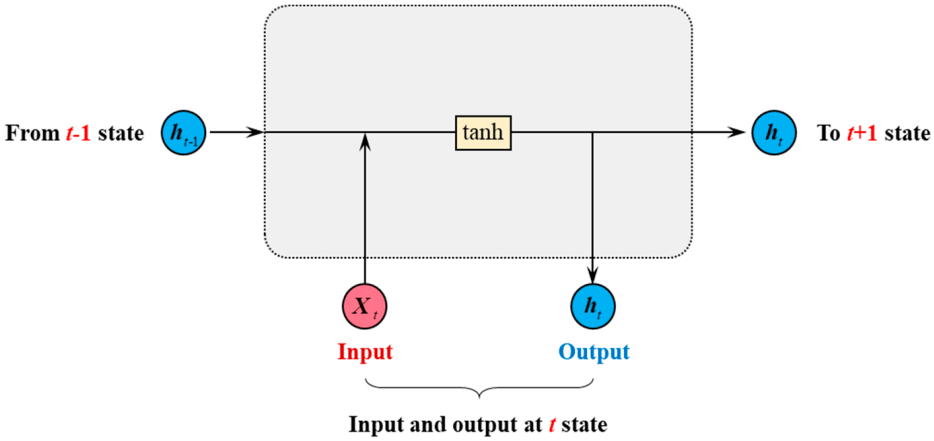


Figure 2. The internals of a Simple Recurrent Neural Network (RNN) memory cell.

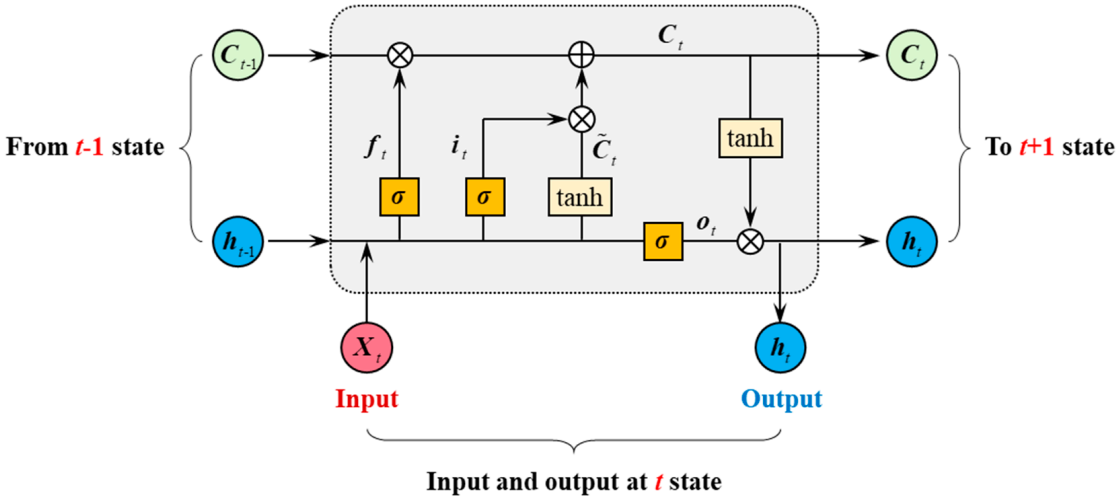


Figure 3. The internals of a Long Short-Term Memory (LSTM) memory cell.

2.2.1. Simple RNN

Simple RNN has the most basic structure in all RNNs [8]. In Figure 2, $\boldsymbol{X}_t$ is the input at time-step t; $\boldsymbol{h}_{t-1}$ is the hidden state and the output at time-step $t-1$; and $\boldsymbol{h}_t$ is the hidden state and output of time-step t. Moreover, $\boldsymbol{h}_t$ is calculated based on the previous hidden state $\boldsymbol{h}_{t-1}$, the current input $\boldsymbol{X}_t$ and activation function tanh().

2.2.2. LSTM Network

LSTM is an artificial RNN architecture which is suitable for processing and predicting the relatively long events in the time series [9]. Its particularity lies in that an LSTM memory cell is composed of four main elements: a forget gate, an input gate, a neuron with a self-recurrent connection and an output gate. In Figure 3, $\boldsymbol{X}_t$ is the input to the memory cell at time-step t. $\boldsymbol{C}_{t-1}$, $\boldsymbol{C}_t$ and $\boldsymbol{h}_{t-1}$, $\boldsymbol{h}_t$ are the cell states and hidden states at time-step $t-1$, t. Moreover, $\boldsymbol{f}_t$, $\boldsymbol{i}_t$, $\boldsymbol{o}_t$ are the states of forget gate, input gate and output gate; $\widetilde{\boldsymbol{C}}_t$ is the candidate state of $\boldsymbol{C}_t$; and σ represents the activation function of sigmoid().

2.3. Model Architecture Design

Two model architectures are proposed based on a stacked DL structure for discharge simulations in this paper. Architecture 1 adopts the most common modeling strategy of time-series prediction, and architecture 2 is designed based on the expertise of hydrology. Figures 4 and 5 show the data flow schematic diagram of model architectures. To make the data flow clearer, we only draw the Recurrent layers and omit other layers (i.e., Dense layers, TimeDistributed layers, Flatten layers and Concatenate layers). Recurrent layers are unrolled along the time dimension in Figures 4 and 5. Additionally, the inputs of these two model architectures have not taken evaporation, infiltration and storage into account, since their variations give no relevant contribution to the river flow rate in the short period of heavy rain [19]. To achieve a longer discharge lead-time, potential future rainfall during lead-time periods is considered.

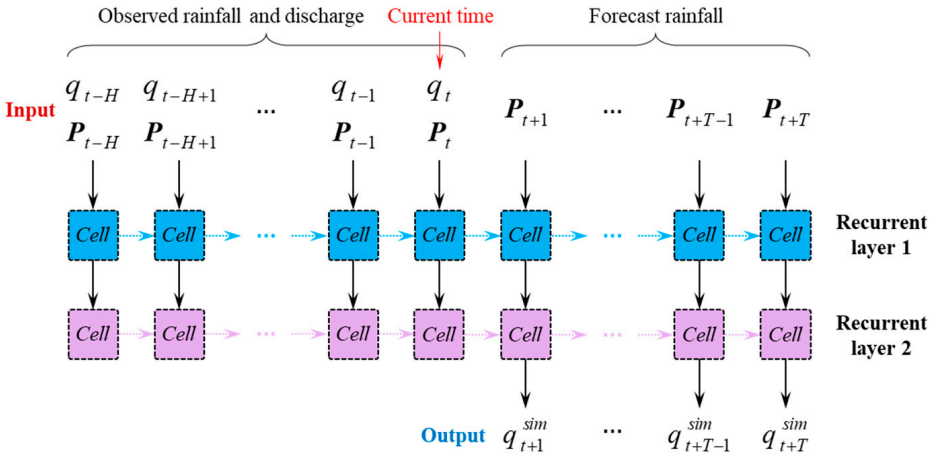


Figure 4. Data flow schematic diagram of architecture 1.

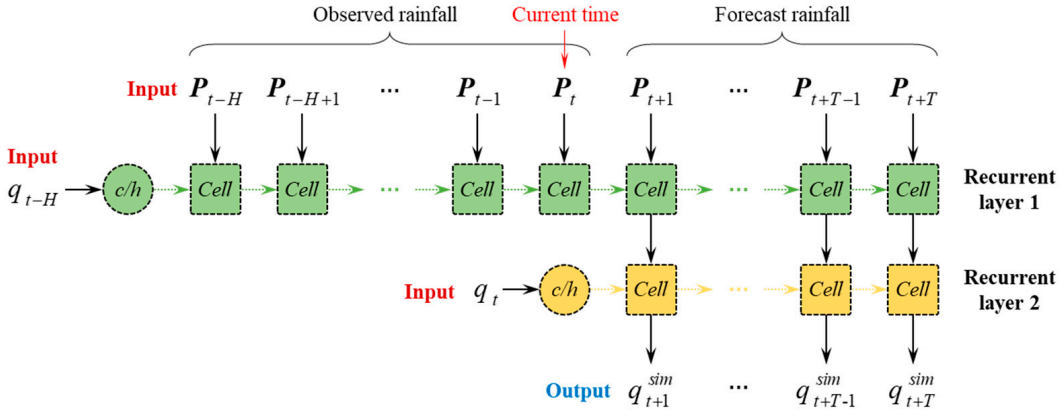


Figure 5. Data flow schematic diagram of architecture 2.

2.3.1. Model Architecture 1

As seen in Figure 4, two Recurrent layers are completely stacked. In layer 1, each time-step is a multivariate series comprised of discharge feature (q) and rainfall features ($\boldsymbol{P}$). All features take the observed value during the observed period, and $\boldsymbol{P}$ take predicted value during the lead-time periods. Moreover, t represents the current time; H represents the maximum time lag in the basin; and T represents the maximum lead-time. The outputs q_{t+1}^{sim}, ..., q_{t+T-1}^{sim}, q_{t+T}^{sim} are the forecasting discharges.

2.3.2. Model Architecture 2

As seen in Figure 5, two Recurrent layers are stacked in the lead-time periods; t, H, T and q_{t+1}^{sim}, …, q_{t+T-1}^{sim}, q_{t+T}^{sim} are the same as Figure 4. According to the expertise of hydrology, two crucial improvements are achieved in architecture 2. Firstly, rainfall and discharge features are no longer indiscriminately inputted into architecture 2. This is because the relationship between rainfall features, $\boldsymbol{P}$, and output is totally different from the relationship between discharge feature q and output. These two different features play their respective roles in discharge simulation, so exploring different relationships by the ML model should not be restricted by the same input form. Secondly, the discharge feature is no longer inputted into the model at each time-step, and only discharge feature q_{t-H} and q_t are inputted to layer 1 and layer 2 as the initial cell state, $\boldsymbol{c}$, and hidden state, $\boldsymbol{h}$, respectively. As seen in Figure 4, the input form at each time-step of architecture 2 is the same as the rainfall input of hydrological models. In addition, architecture 2 weakens the effect of the observed discharges trend on forecasting discharges by discarding the input of q_{t-H+1}, …, q_{t-2}, q_{t-1}. It emphasizes the direct driving effect of rainfall time series on forecasting discharges. In short, compared with architecture 1, architecture 2 is more in line with the hydrological model architecture.

In addition, we have tried a variety of architecture designs for discharge simulations, including different stacked architectures, different input modes and different numbers of Recurrent layers. The results of the case study indicate that architectures 1 and 2 are most representative, because (1) architecture 1 adopts the simplest design, in which both rainfall and discharge features are inputted at each time-step; (2) architecture 2 combines the expertise of hydrology and outperforms other architectures, in which discharge features are inputted as initial states. In other words, other architectures could be regarded as variants of architecture 1 or architecture 2, so architectures 1 and 2 are only chosen for variance decomposition in this study.

2.4. Criteria for Accuracy Assessment

Uncertainties of multi-step time-series forecasting models are evaluated based on discharge observations. The following evaluation criteria are used in discharge error assessment: Relative Peak-Discharge Error (*RPE*), Nash–Sutcliffe coefficient of efficiency (*NSE*), Mean Absolute Error (*MAE*) and Mean Square Error (*MSE*) [20]. In addition, *NSE* and *RPE* are suggested by the Chinese flood-forecasting guidelines and often used in flood-forecasting studies [21,22].

$$RPE = \frac{q_{peak}^{sim} - q_{peak}^{obs}}{q_{peak}^{obs}} \times 100\% \tag{1}$$

$$NSE = 1 - \frac{\sum_{t=1}^{N}\left(q_t^{obs} - q_t^{sim}\right)^2}{\sum_{t=1}^{N}\left(q_t^{obs} - \overline{q}^{obs}\right)^2} \tag{2}$$

$$MAE = \frac{1}{N} \cdot \sum_{t=1}^{N}\left|q_t^{obs} - q_t^{sim}\right| \tag{3}$$

$$MSE = \frac{1}{N} \cdot \sum_{t=1}^{N}\left(q_t^{obs} - q_t^{sim}\right)^2 \tag{4}$$

where q_{peak}^{obs} and q_{peak}^{sim} are the observed and simulated peak-discharge; q_t^{obs} and q_t^{sim} are the observed and simulated discharge; $\overline{q}^{obs}$ is the average value of the observed discharge; and N is the total number of observations.

2.5. Variance Decomposition

Figure 6 depicts the ML time-series forecasting modeling combination scheme employed in this study. It has three parts: sample set (SS), RNN approach (RA) and Model architecture (MA). In the variance decomposition, we aimed to decompose the total ensemble uncertainty into contributions from different elements of the ML modeling and interactions among them. The ML modeling ensemble consists of 10 combinations. For each of them, we estimated *MAE*, *MSE*, *NSE* and *RPE*. In the following, we describe these evaluation criteria as the general variable Y. To relate variable Y to the uncertainty sources, we use superscripts in $Y^{j,\,k,\,l}$, with j, k and l representing the different options of SS, RA and MA, respectively. The subsampling of the training sets and the ANOVA approach are explained in detail, as follows.

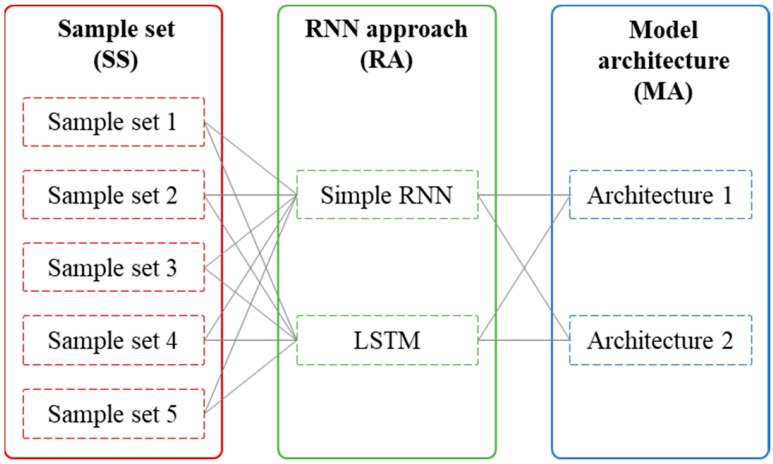


Figure 6. Machine-learning time-series forecasting modeling combination scheme.

2.5.1. Subsampling Approach

The ANOVA approach could underestimate the variance in small sample sizes [23]. To diminish the effect of the sample size on the quantification of the variance contribution, Bosshard et al. [24] added the subsampling method to ANOVA approach. The purpose of the subsampling approach is to keep the same number of sample sizes for SS, RA and MA in each variance decomposition. In this paper, we subsample the five different SSs. In each subsampling iteration i, we select two SSs out of the five SSs, which results in a total of 10 possible SS pairs in Equation (5). To prevent confusion between the full set of five SSs and the subsampled SS pair, we replace the superscript j with $g(h, i)$. As seen in Equation (5), the superscript g is a 2×10 matrix that contains the selected SSs for the particular subsampling iteration i ($i = 1, 2, \ldots, 10$). In each iteration i, there are two SSs ($h = 1, 2$).

$$g = \begin{pmatrix} \text{SS1} & \text{SS1} & \text{SS1} & \text{SS1} & \text{SS2} & \text{SS2} & \text{SS2} & \text{SS3} & \text{SS3} & \text{SS4} \\ \text{SS2} & \text{SS3} & \text{SS4} & \text{SS5} & \text{SS3} & \text{SS4} & \text{SS5} & \text{SS4} & \text{SS5} & \text{SS5} \end{pmatrix} \tag{5}$$

2.5.2. ANOVA Approach

In this study, we referred to an application of ANOVA in hydrological climate-impact projections [24]. According to the ANOVA theory, in each iteration i, the total sum of the squares (SST_i) can be split into sums of squares due to the individual effects (SSA_i, SSB_i and SSC_i) and their interactions (SSI_i) as follows:

$$SST_i = SSA_i + SSB_i + SSC_i + SSI_i \tag{6}$$

where SST_i represents the total variance from two SSs, two RAs and two MAs in iteration i; SSA_i represents the variance from two SSs in iteration i; SSB_i represents the variance from two RAs in iteration i; SSC_i represents the variance from two MAs in iteration i; SSI_i represents the variance from

the interaction among two SSs, two RAs and two MAs in iteration i. The terms in Equation (6) are estimated by the subsampling procedure, as follows:

$$SST_i = \sum_{h=1}^{H}\sum_{k=1}^{K}\sum_{l=1}^{L}\left(Y^{g(h,\,i),\,k,\,l} - Y^{g(\circ,\,i),\,\circ,\,\circ}\right)^2 \tag{7}$$

$$SSA_i = K \cdot L \cdot \sum_{h=1}^{H}\left(Y^{g(h,\,i),\,\circ,\,\circ} - Y^{g(\circ,\,i),\,\circ,\,\circ}\right)^2 \tag{8}$$

$$SSB_i = H \cdot L \cdot \sum_{k=1}^{K}\left(Y^{g(\circ,\,i),\,k,\,\circ} - Y^{g(\circ,\,i),\,\circ,\,\circ}\right)^2 \tag{9}$$

$$SSC_i = H \cdot K \cdot \sum_{l=1}^{L}\left(Y^{g(\circ,\,i),\,\circ,\,l} - Y^{g(\circ,\,i),\,\circ,\,\circ}\right)^2 \tag{10}$$

$$SSI_i = \sum_{h=1}^{H}\sum_{k=1}^{K}\sum_{l=1}^{L}\left(Y^{g(h,\,i),\,k,\,l} - Y^{g(h,\,i),\,\circ,\,\circ} - Y^{g(\circ,\,i),\,k,\,\circ} - Y^{g(\circ,\,i),\,\circ,\,l} + 2Y^{g(\circ,\,i),\,\circ,\,\circ}\right)^2 \tag{11}$$

where variable Y represents evaluation criteria or simulated discharges; H, K and L represent the number of SS, RA and MA in iteration i. H, K and L are all equal to 2 in each iteration i. The symbol $\circ$ indicates averaging over the particular index. For example, in Equation (7), $Y^{g(\circ,\,i),\,\circ,\,\circ}$ equals to the average of $Y^{g(1,\,i),\,1,\,1}$, $Y^{g(1,\,i),\,1,\,2}$, $Y^{g(1,\,i),\,2,\,1}$, $Y^{g(1,\,i),\,2,\,2}$, $Y^{g(2,\,i),\,1,\,1}$, $Y^{g(2,\,i),\,1,\,2}$, $Y^{g(2,\,i),\,2,\,1}$, $Y^{g(2,\,i),\,2,\,2}$. Then, each variance fraction, η^2, is derived as follows:

$$\eta^2_{SS} = \frac{1}{I} \cdot \sum_{i=1}^{I} \frac{SSA_i}{SST_i} \tag{12}$$

$$\eta^2_{RA} = \frac{1}{I} \cdot \sum_{i=1}^{I} \frac{SSB_i}{SST_i} \tag{13}$$

$$\eta^2_{MA} = \frac{1}{I} \cdot \sum_{i=1}^{I} \frac{SSC_i}{SST_i} \tag{14}$$

$$\eta^2_{Interactions} = \frac{1}{I} \cdot \sum_{i=1}^{I} \frac{SSI_i}{SST_i} \tag{15}$$

where η^2 has a range of 0 to 1 and describes a 0–100% contribution to the total uncertainty of simulated discharges. According to the matrix (5), I is equal to 10 in this study. The sum of η^2_{SS}, η^2_{RA}, η^2_{MA}, $\eta^2_{Interactions}$ is equal to 1, and the larger the value of η^2 is, the relatively greater uncertainty contribution of corresponding compositions in ML modeling.

3. Case Study

3.1. Study Area

Anhe catchment, located in the province of Jiangxi, in humid Southeastern China, with a drainage area of 251 km^2 and mean annual rainfall of 1426 mm, is taken as the case study. The rainfall data from eight rain stations and discharge data from one hydrology station were used in this study, as shown in Figure 7. According to statistical analysis, the maximum basin time lag of Anhe is 6 h. In this paper, observed rainfall measurements are taken as the perfect forecast for 1–10 h lead-time. Thus, we take H as 5 h and T as 10 h in the two model architectures (see Figures 4 and 5).

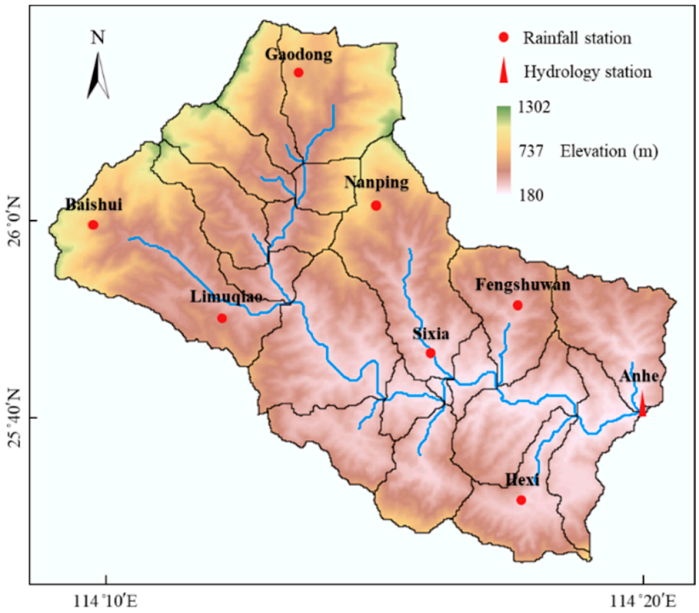
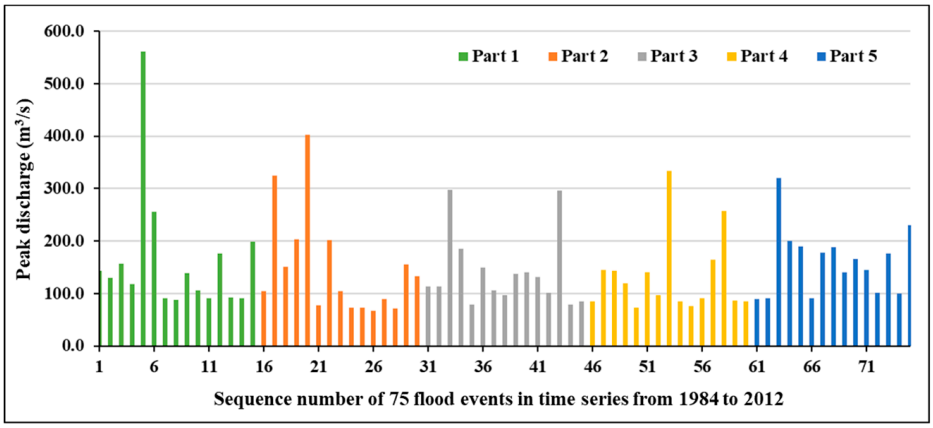


Figure 7. Study catchment with the hydrology station and rain stations.

The hourly time-series data for twenty-nine years (from 1984 to 2012; 75 flood events) are included in the modeling process. The beginning and the end of a flood event are taken as the rising point from base flow before the main rainfall and the recession point to base flow, respectively. In light of the time sequence of flood events and the representativeness of peak-discharge, 75 flood events are divided into five parts, and each part has 15 flood events (see Figure 8). Then, five sample sets are constructed, and each sample set is composed of a training set (four parts) and a validation set (one part). The training set is used to fit model parameters, and the validation set is used to tune the parameters for preventing overfitting or underfitting.



Sample set	Training set	Validation set
SS1	Part 2, 3, 4, 5	Part 1
SS2	Part 1, 3, 4, 5	Part 2
SS3	Part 1, 2, 4, 5	Part 3
SS4	Part 1, 2, 3, 5	Part 4
SS5	Part 1, 2, 3, 4	Part 5

Figure 8. Partition of sample set 1–5.

3.2. Model Training

(1) Data normalization: Observed rainfall, p^1, p^2, …, p^8, from eight rainfall stations, and observed discharge, q, from Anhe hydrology station are normalized in the range of 0 to 1, as Equation (16). $\hat{x}$ is the normalized value; x is the observed value; $x_{\max}$ and $x_{\min}$ are the maximum and minimum observed value.

$$\hat{x} = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{16}$$

Rainfall and discharge data are normalized separately due to their different physical significance and dimensions.

(2) Establish a dataset: For each sample set, the training set (60 flood events) and validation set (15 flood events) are restructured to a supervised learning dataset by sliding window method [25]. In the supervised learning dataset, the sample size of the supervised learning training set is about 9000, and the sample size of the supervised learning validation set is about 2000. The sample size is slightly different in the supervised learning datasets for sample sets 1–5 because of different duration of 75 flood events.

As seen in Figures 4 and 5, the time-step of each sample is 16 ($H = 5$, $T = 10$). In architecture 1, the input of each time-step, i, is composed of nine features: (a) q_i, p_i^1, p_i^2, …, p_i^8 before the current time, and (b) 0, p_i^1, p_i^2, …, p_i^8 after current time; in architecture 2, the input includes (a) q_{t-5}, q_t, and (b) eight features p_i^1, p_i^2, …, p_i^8 for each time-step, i.

(3) Open-source software of the Deep-Learning framework TensorFlow and Keras in Python 3.6 was chosen in this study. Adaptive Moment Estimation was chosen as an optimization algorithm. *MSE* was determined as the loss function.

(4) The hyper-parameters in the ML models are determined through trial and error. The trial-and-error results are as follows: units in recurrent layer 1 and layer 2 are both 5; batch-size is 64; and the epoch of each ML model is different and determined by the learning curve. Figure 9 shows the visualization of tensor flow in four ML models by Keras. In Figure 9a,b, the numbers 16, 9, 10 and 5 represent the number of total time-steps, features, the forecasting time-steps and units, respectively. Moreover, 6, 8 and 1 represent the numbers of observed time-steps, rainfall features and discharge features in Figure 9c,d. In the red part, q_{t-5} is inputted to simple_rnn_1 (or lstm_1) and turned into an initial hidden state (or both cell state and hidden state); in the green part, q_t is inputted to simple_rnn_2 (or lstm_2) and turned into an initial hidden state (or both cell state and hidden state); the yellow part presents the inputs of eight rainfall features of 16 time-steps.

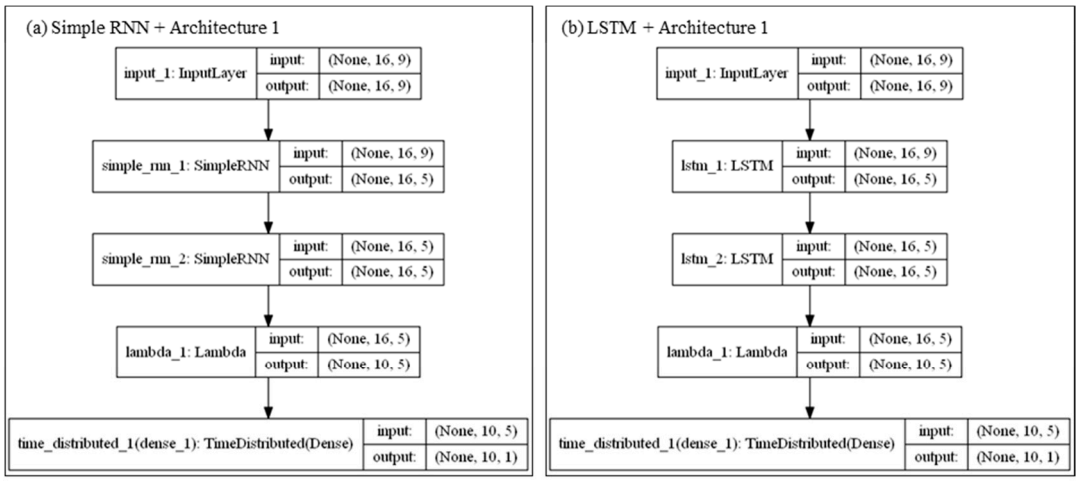


Figure 9. *Cont.*

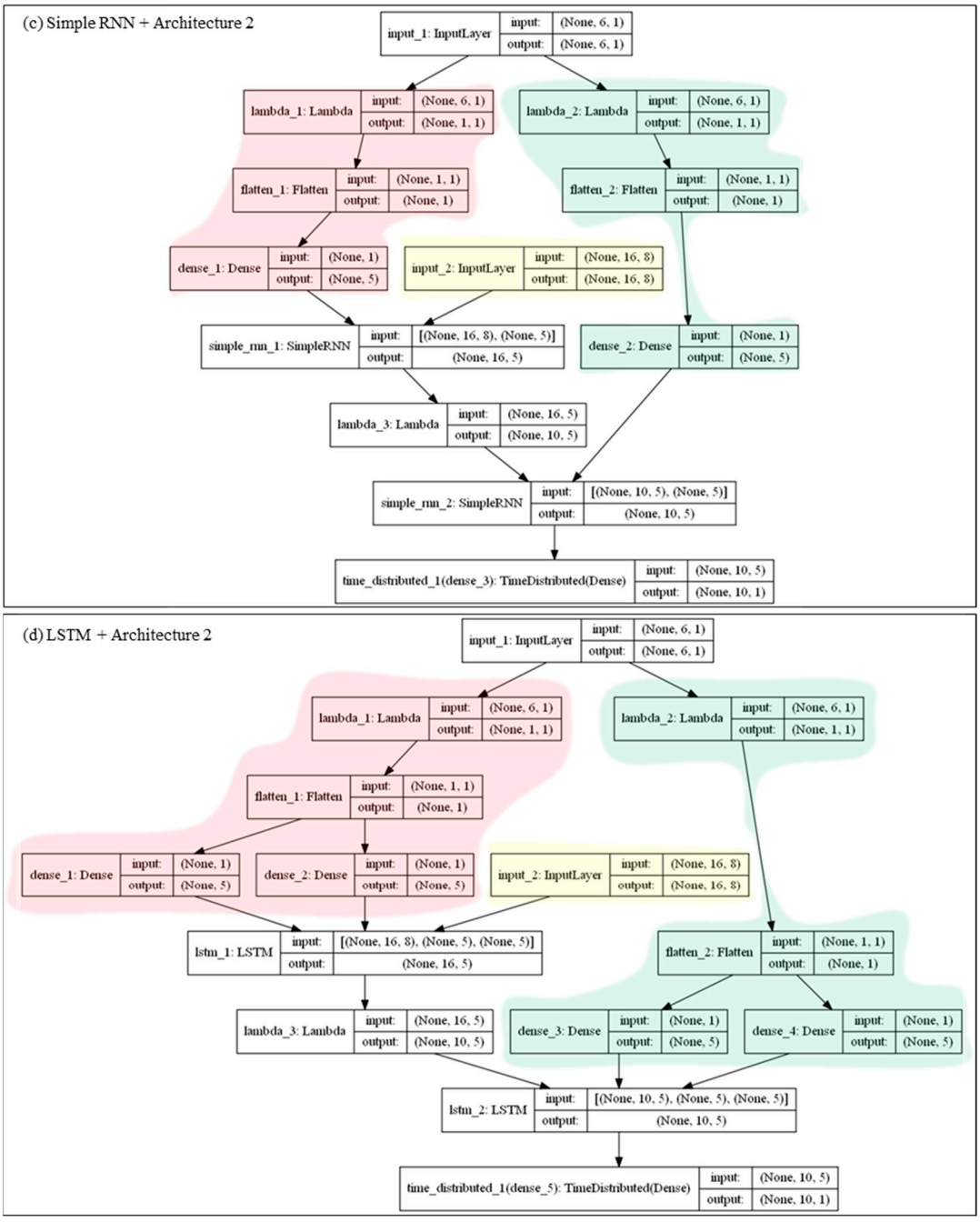


Figure 9. Visualization of tensor flow in ML models.

4. Results and Discussion

4.1. Sample Set Evaluations

The *NSE* and *RPE* for different sample sets at 1–10-h lead-time are summarized in box-and-whisker plots in Figures 10 and 11, and each boxplot includes 300 simulated results from 75 flood events by four combination schemes, which consist of two RNN approaches and two model architectures. The minimum, maximum, 25th percentile (Q_1), 75th percentile (Q_3) and the median value (Q_2) are shown by the lower and upper whiskers, lower and upper bounds of the box, and the line within the box, respectively. The lower and upper whiskers indicate the values $(Q_1 - 1.5 \times \text{IQR})$ and $(Q_3 + 1.5 \times \text{IQR})$, where IQR is the Inter-Quartile Range $(Q_3 - Q_1)$.

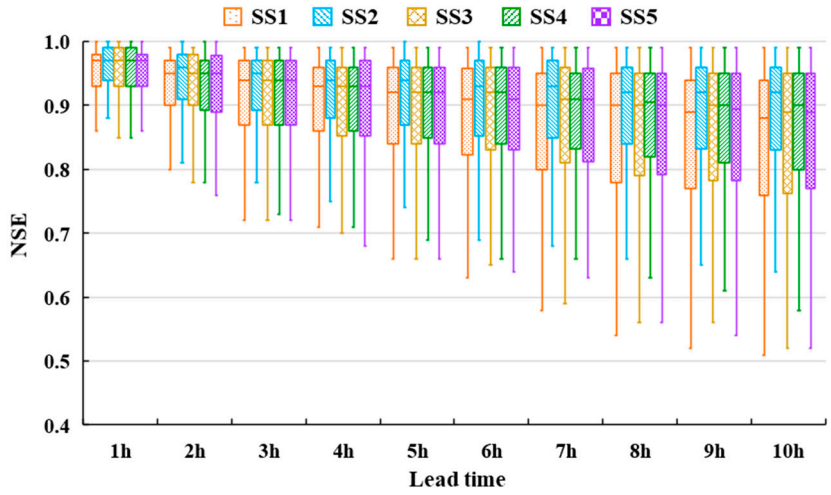


Figure 10. Nash–Sutcliffe coefficient of efficiency *(NSE)* for different sample sets at 1–10-h lead-time.

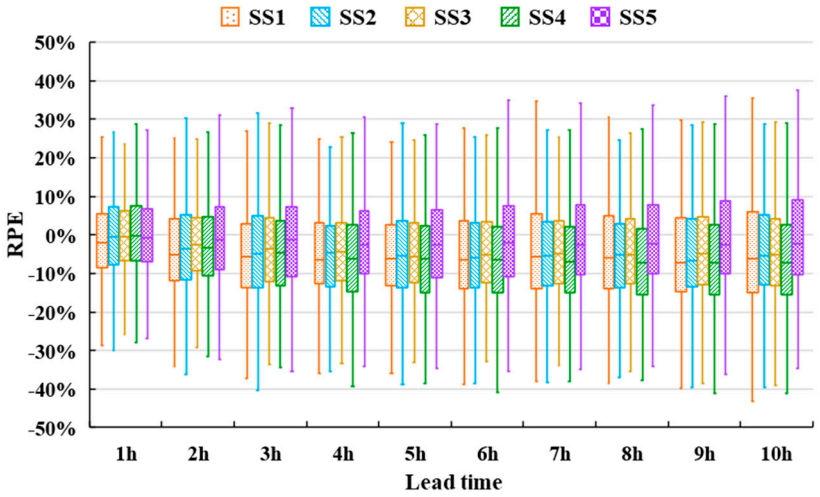


Figure 11. Relative Peak-Discharge Error *(RPE)* for different sample sets at 1–10-h lead-time.

As shown in Figure 10, at least 75% of *NSE* is greater than 0.8 at 1–7-h lead-time. Different SSs have almost the same upper whisker, Q_2 and Q_3 at each lead-time. For Q_1 and lower whisker, the differences between SSs are greater with the increase of lead-time. The *NSE* of SS2 is slightly better than others. Figure 11 indicates the *RPE* distributions of SS1–SS4 are almost the same. Compared with SS1–SS4, SS5 shows a slightly different distribution range of *RPE*. Figures 10 and 11 illustrate that no SS outperforms others obviously in the statistic distribution of *NSE* and *RPE*, so it is feasible to divide the sample set by a time sequence of flood events and the representativeness of peak-discharge.

4.2. RNN Approach Evaluations

Figures 12 and 13 show box-and-whisker plots of *NSE* and *RPE* for different RNN approaches at 1–10-h lead-time, and each boxplot includes 750 simulated results from 75 flood events by 10 combination schemes, which consist of five sample sets and two model architectures.

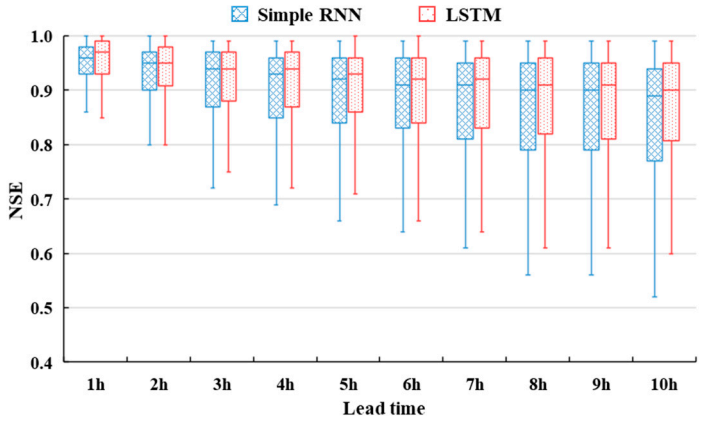


Figure 12. *NSE* for different RNN approaches at 1–10-h lead-time.

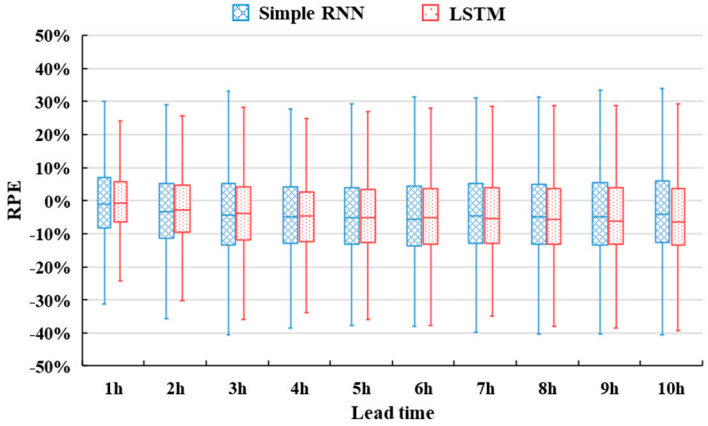


Figure 13. *RPE* for different RNN approaches at 1–10-h lead-time.

Figure 12 illustrates LSTM has higher Q_1, Q_2, Q_3 and whisker of *NSE* than that of Simple RNN. Except for 1, 2-h lead-time, LSTM shows a more compact *NSE* distribution (shorter distance between Q_1 and Q_3). It indicates LSTM takes obvious advantage with the increase of lead-time. Its advantage is mainly reflected in the flood events, which are simulated relatively well at short lead-times, and the flood events, which are simulated relatively poorly at long lead-times. Figure 13 illustrates the simulated peak discharges of LSTM have more compact *RPE* distribution at each lead-time. Both Figures 12 and 13 indicate LSTM outperforms Simple RNN in discharge simulations, especially in the simulation of flood process at relatively long lead-times. LSTM indeed demonstrates its powerful ability for processing multi-step time-series discharge simulations due to its special gates structure.

4.3. Model Architecture Evaluations

Figures 14 and 15 compared box-and-whisker plots of *NSE* and *RPE* for different model architectures at 1–10-h lead-time, and each boxplot includes 750 simulated results from 75 flood events by 10 combination schemes, which consist of five sample sets and two RNN approaches.

Figure 14 illustrates that architecture 2 has higher Q_1, Q_2 and Q_3 and lower whisker of *NSE* than that of architecture 1. At 1–3-h lead-time, the advantages of architecture 2 are especially obvious. For *RPE* distribution in Figure 15, architecture 2 also has a distinct advantage at 1, 2-h lead-time. As the lead-time increases, the difference in *RPE* distribution between architecture 1 and 2 gets smaller. Both Figures 14 and 15 indicate that Architecture 2 has absolute superiority in simulating peak discharge at 1-h lead-time and flood process at all lead-times, because a shorter distance between Q1 and Q3 means the quartile data are bunched together.

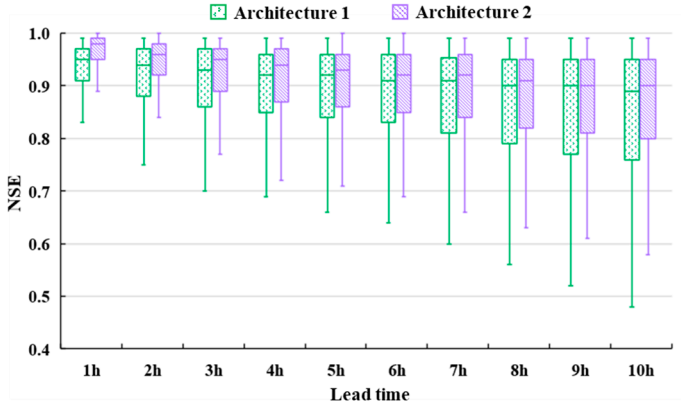


Figure 14. *NSE* for different model architectures at 1–10-h lead-time.

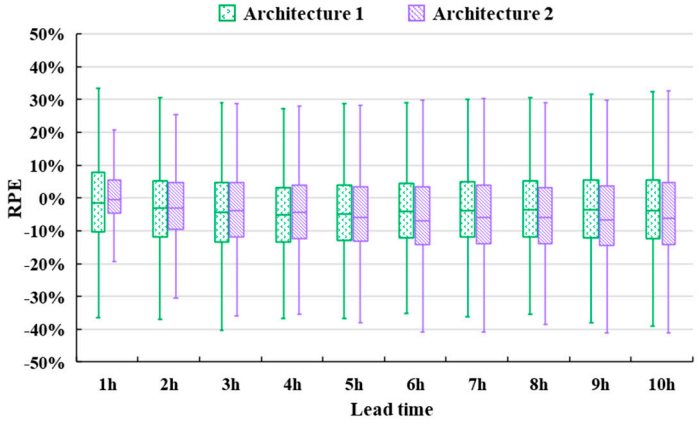


Figure 15. *RPE* for different model architectures at 1–10-h lead-time.

The results reveal that the strategy, q_t, is inputted as an initial cell state, $\boldsymbol{c}$, and hidden state, $\boldsymbol{h}$, to Recurrent layers and improves the simulation accuracy of both flood hydrograph and peak. As the lead-time increases, the difference of simulated discharges between two architectures decreases due to the similar architecture at long lead-time (see Figures 4 and 5). In brief, the multi-step time-series forecasting models could obtain better performance by inputting the discharge feature and rainfall features separately. It suggests that the ML model might be improved by combining the expertise of hydrology.

4.4. Uncertainty Source Quantification

As shown in Figures 16 and 17, the stacked area chart is used to show the ratio of uncertainty contributions, which is calculated by Equations (12)–(15), at each lead-time or in the discharge quantile interval.

Figure 16a–c shows the ratio of the contribution of uncertainty sources to discharge simulations by *MAE*, *MSE* and *NSE* at 1–10-h lead-time. The MA is the dominant source at 1–3-h lead-time. The uncertainty contribution ratio of MA reaches 0.8 at 1-h lead-time. It is attributed to the strategy that q_t is inputted as an initial state to Recurrent layers in architecture 2 because of the strong correlation between q_t and q_{t+1}. Separately inputted discharge feature plays a major role at short lead-time, and the strong correlation between q_t and q_{t+1}, q_{t+2}, q_{t+3} weakens gradually with the increase of lead-time. At 4–10-h lead-time, the contributions of SS and RA gradually increase, while the contributions of MA gradually decrease. It indicates the conclusions of variance decomposition are consistent with the previous analysis of Sections 4.2 and 4.3: (1) The outperformance of LSTM is manifested in the simulation of flood process at long lead-time, because LSTM could solve the problem that the Simple RNN would not learn long-term dependencies (see Figure 11); (2) the outperformance of architecture 2

is mainly manifested at short lead-time, because MAs have similar architecture at long lead-time (see Figures 4 and 5).

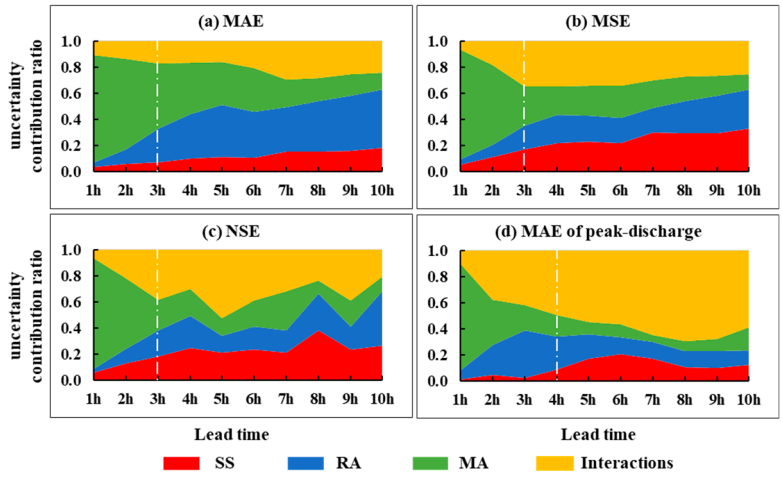


Figure 16. The ratio of the contributions of uncertainty sources at 1–10-h lead-time.

Figure 17. The ratio of the contributions of uncertainty sources in different discharge quantiles.

By comparing Figure 16a and Figure 16b, we see the uncertainty contribution ratio of SS by *MSE* is greater than that of *MAE*. We concluded that the SS contributes more uncertainty in simulated outliers, because *MAE* has better robustness than *MSE* for outliers. On the contrary, the RA contributes less uncertainty in simulated outliers. Therefore, we suspect the simulated results of LSTM and Simple RNN might be similar for partial outliers, which are caused by the partition of SS.

Despite the contribution of uncertainty sources that fluctuate a little bit in Figure 16c, the overall trend is consistent with Figure 16a,b. Figure 16d shows the ratio of the contribution of uncertainty sources to peak-discharge simulations by *MAE* at 1–10-h lead-time.

Combining the previous analysis of Section 4.3, it indicates that (1) LSTM and architecture 2 shows their advantages of peak-discharge simulations mainly at 1–4-h lead-time; (2) the SS is the least important individual source of uncertainty for peak-discharge simulations at 1–4-h lead-time. Compared with Figure 15a–c, the uncertainty contribution of the interaction on peak-discharge simulations is much greater than other flood processes. It indicates that peak-discharge is more difficult to simulate, because the rainfall process with variable space–time distribution always occurs before the flood peak.

Figure 17 depicts the ratio of the contributions of uncertainty sources in different discharge quantiles. We take the mean of the variance decomposition for each quantile interval and divide the quantile bins into three parts: a low range (5–75%, 9.6–47.3 m^3/s), an intermediate range (75–95%, 47.3–189.4 m^3/s) and a high range (95–99.9%, 189.4–333.5 m^3/s).

In the low quantile range, the SS is the dominant source of uncertainty at 2–10-h lead-time. It suggests that the different combinations of RA and MA make little difference for low discharge simulations. Small discharges mainly come from base flows or recession flows, which are relatively stable, barely affected by rainfall and much easier to simulate, whereas at 1-h lead-time, the MA and interactions contribute the majority of uncertainty sources due to the strong correlation between q_t and q_{t+1}.

In the intermediate range, the contribution of the MA gradually decreases with the increase of lead-time, and the contribution of the SS and RA is relatively stable. Combined with the analysis of Figures 14 and 15, it indicates that architecture 2 is more helpful to improve the discharge simulations in the intermediate and high quantile ranges. LSTM has more advantages in intermediate and high quantile ranges, in which the contribution ratio of RA is stable at 0.2. However, LSTM does not show any advantage compared to Simple RNN in the low quantile range.

Compared with the intermediate range, the contribution of the SS increases in the high range, because the division of training set and validation set is according to the value of peak-discharge. Moreover, the relatively small sample size in the 99–99.9% quantile also tends to increase the uncertainty contribution of the SS. The contribution of the interaction term to the total uncertainty varies throughout the different discharge quantiles between 0.4 and 0.5 and more stable at long lead-time.

The results of variance decomposition indicate that the contribution of individual uncertainty source quantification varies on different lead-time and different discharge quantiles. In addition, the contribution of interactions should not be ignored. Although the proposed framework cannot take all the uncertainties into account, or gives no further explanation of interactions, it is valuable to evaluate the multi-step time-series forecasting models. It suggests the combination of LSTM and model architecture 2 is superior to others in discharge simulations.

Finally, through the result analysis of model evaluation and uncertainty source quantification, several suggestions are put forward to the ML modeling for multi-step discharge simulations: (1) LSTM network demonstrates its superiority in discharge simulations, and ML architecture design is as important as ML approach selection; (2) it is essential to input discharge feature properly for short-term discharge forecasting; (3) for low discharge simulations, different combinations of RA and MA make little difference, and the sample set partitioning dominates the uncertainty in modeling.

5. Conclusions

A framework is proposed for uncertainty quantification of ML modeling and evaluation of multi-step time-series forecasting models. Discharge simulation using RNN networks and stacked DL structure in Anhe catchment of Southeastern China is taken as an example to quantify uncertainty contributions of each component (five different SSs, two RAs and two MAs). To evaluate the interactions between LSTM networks and ML architecture, a novel ML architecture is designed by combining

the expertise of hydrology and stacked DL structure, and compared with another conventional architecture. We estimated the total ensemble uncertainty of the ML modeling and quantified the contributions of different uncertainty sources in discharge simulations. The proposed framework can reveal discharge-simulation uncertainties in ML multi-step time-series forecasting models based on ANOVA theory. Moreover, it also provides references for ML technology selection and architecture design. The results indicated that uncertainty quantification is an indispensable task that must be performed in multi-step time-series discharge simulations, for a successful application of ML. Conclusions from this study include the following:

(1) The results of variance decomposition indicate that the contribution of individual uncertainty source quantification varies at different lead-times and different discharge quantiles, and the contribution of interactions should not be ignored. With the increase of lead-time, the contributions of SS and RA gradually increase, while the contributions of MA gradually decrease. The SS is the dominant source of uncertainty in the low quantile range. In the intermediate and high quantile range, the contribution of the MA gradually decreases with the increase of lead-time, and the contribution of the SS and RA is relatively stable. Interactions contribute a lot to the total uncertainty in discharge simulations, and interactive impacts are influenced by discharge magnitude and lead-time. The evaluation results of the proposed framework demonstrate the model combining LSTM and architecture 2 is superior to others in the statistic distribution of *NSE* and *RPE*.

(2) As expected, the LSTM network outperforms Simple RNN in both flood processes and peak discharge. According to the analysis results of the proposed framework, the advantages of the LSTM network are mainly embodied in intermediate and high quantile ranges due to the dominant uncertainty contribution of SS in low quantile range. Comparing with ML architecture, the LSTM network shows limited advantage at short lead-time. In short, the LSTM network demonstrates its powerful ability for processing and predicting the relatively long events for multi-step time-series discharge simulations. It has more practical value to evaluate the LSTM network based on the interactions between LSTM networks and other components of ML modeling.

(3) ML architecture design is as necessary as ML approaches selection for multi-step time-series forecasting model. Architecture 2 has absolute superiority in simulating peak discharge at 1-h lead-time and flood process at all lead-times, especially in intermediate and high quantile ranges. It is mainly due to the strategy that incorporating hydrological expertise into stacking DL architecture.

(4) The SS is the dominant source of uncertainty in the low quantile range (except 1-h lead-time). In other words, different combinations of RA and MA make little difference for low discharge simulations. No one SS is better than the others in the statistic distribution of *NSE* and *RPE*. Therefore, it is feasible to divide the sample set by time sequence of flood events and the representativeness of peak-discharge.

Although this study only investigated a discharge simulation with two ML approaches and two ML architectures, the proposed framework is versatile and adjustable to research and applications of ML techniques in other areas of study. In addition, further research should also focus on the hydrological interpretation of ML models, such as the relationship between the dynamical changes of LSTM cell state and the soil moisture content.

Author Contributions: T.S. conceptualized, designed the model and wrote the paper; W.D. and H.L. reviewed and edited the paper; H.Z. conceptualized the model; J.W. and J.C. validated the results. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the National Natural Science Foundation of China (Grant No. 2018YFC0407705, 51709036, 91747102 and 51879029).

Acknowledgments: We truly appreciate the insightful comments and suggestions from the editor and reviewers, which have helped us to improve this paper significantly. We also want to thank the developers of Python and TensorFlow.

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

References

1. Shrestha, A.; Mahmood, A. Review of deep learning algorithms and architectures. *IEEE Access* **2019**, *7*, 53040–53065. [CrossRef]
2. Qin, J.; Liang, J.; Chen, T.; Lei, X.; Kang, A. Simulating and predicting of hydrological time series based on tensorflow deep learning. *Pol. J. Environ. Stud.* **2019**, *28*, 795–802. [CrossRef]
3. Hu, C.; Wu, Q.; Li, H.; Jian, S.; Li, N.; Lou, Z. Deep learning with a long short-term memory networks approach for rainfall-runoff simulation. *Water* **2018**, *10*, 1543. [CrossRef]
4. Mosavi, A.; Ozturk, P.; Chau, K.W. Flood prediction using machine learning models: Literature review. *Water* **2018**, *10*, 1536. [CrossRef]
5. Jha, A.; Chandrasekaran, A.; Kim, C.; Ramprasad, R. Impact of dataset uncertainties on machine learning model predictions: The example of polymer glass transition temperatures. *Model. Simul. Mater. Sci. Eng.* **2019**, 27. [CrossRef]
6. Rahmati, O.; Choubin, B.; Fathabadi, A.; Coulon, F.; Soltani, E.; Shahabi, H.; Mollaefar, E.; Tiefenbacher, J.; Cipullo, S.; Bin Ahmad, B.; et al. Predicting uncertainty of machine learning models for modelling nitrate pollution of groundwater using quantile regression and uneec methods. *Sci. Total Environ.* **2019**, *688*, 855–866. [CrossRef]
7. Li, Y.M.; Xiao, W.R.; Wang, P.F. *Uncertainty Quantification of Artificial Neural Network Based Machine Learning Potentials*; Amer Soc Mechanical Engineers: New York, NY, USA, 2019.
8. Yu, Y.; Si, X.; Hu, C.; Zhang, J. A review of recurrent neural networks: Lstm cells and network architectures. *Neural Comput.* **2019**, *31*, 1235–1270. [CrossRef]
9. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef]
10. Hsu, W.N.; Zhang, Y.; Lee, A.; Glass, J.; Int Speech Commun, A. Exploiting depth and highway connections in convolutional recurrent deep neural networks for speech recognition. In Proceedings of the 17th annual conference of the international speech communication association, San Francisco, CA, USA, 8–12 September 2016; pp. 395–399.
11. Kim, H.Y.; Won, C.H. Forecasting the volatility of stock price index: A hybrid model integrating lstm with multiple garch-type models. *Expert Syst. Appl.* **2018**, *103*, 25–37. [CrossRef]
12. Palangi, H.; Deng, L.; Shen, Y.; Gao, J.; He, X.; Chen, J.; Song, X.; Ward, R. Deep sentence embedding using long short-term memory networks: Analysis and application to information retrieval. *IEEE ACM Trans. Audio Speech Lang. Process.* **2016**, *24*, 694–707. [CrossRef]
13. Zhao, Z.; Chen, W.; Wu, X.; Chen, P.C.Y.; Liu, J. Lstm network: A deep learning approach for short-term traffic forecast. *IET Intell. Transp. Syst.* **2017**, *11*, 68–75. [CrossRef]
14. Tan, J.H.; Hagiwara, Y.; Pang, W.; Lim, I.; Oh, S.L.; Adam, M.; Tan, R.S.; Chen, M.; Acharya, U.R. Application of stacked convolutional and long short-term memory network for accurate identification of cad ecg signals. *Comput. Biol. Med.* **2018**, *94*, 19–26. [CrossRef] [PubMed]
15. Kratzert, F.; Klotz, D.; Brenner, C.; Schulz, K.; Herrnegger, M. Rainfall-runoff modelling using long short-term memory (lstm) networks. *Hydrol. Earth Syst. Sci.* **2018**, *22*, 6005–6022. [CrossRef]
16. Zhang, D.; Lin, J.; Peng, Q.; Wang, D.; Yang, T.; Sorooshian, S.; Liu, X.; Zhuang, J. Modeling and simulating of reservoir operation using the artificial neural network, support vector regression, deep learning algorithm. *J. Hydrol.* **2018**, *565*, 720–736. [CrossRef]
17. Tian, Y.; Xu, Y.P.; Yang, Z.; Wang, G.; Zhu, Q. Integration of a parsimonious hydrological model with recurrent neural networks for improved streamflow forecasting. *Water* **2018**, *10*, 1655. [CrossRef]
18. Committee, A.M. Unbalanced robust anova for the estimation of measurement uncertainty at reduced cost. *Anal. Methods* **2014**, *6*, 7110–7111.
19. Campolo, M.; Andreussi, P.; Soldati, A. River flood forecasting with a neural network model. *Water Resour. Res.* **1999**, *35*, 1191–1197. [CrossRef]
20. Roberts, W.; Williams, G.P.; Jackson, E.; Nelson, E.J.; Ames, D.P. Hydrostats: A python package for characterizing errors between observed and predicted time series. *Hydrology* **2018**, *5*, 66. [CrossRef]
21. Liu, P.; Zhang, X.; Zhao, Y.; Deng, C.; Li, Z.; Xiong, M. Improving efficiencies of flood forecasting during lead times: An operational method and its application in the baiyunshan reservoir. *Hydrol. Res.* **2019**, *50*, 709–724. [CrossRef]

22. Song, T.; Ding, W.; Wu, J.; Liu, H.; Zhou, H.; Chu, J. Flash flood forecasting based on long short-term memory networks. *Water* **2020**, *12*, 109. [CrossRef]
23. Deque, M.; Rowell, D.P.; Luethi, D.; Giorgi, F.; Christensen, J.H.; Rockel, B.; Jacob, D.; Kjellstrom, E.; de Castro, M.; van den Hurk, B. An intercomparison of regional climate simulations for europe: Assessing uncertainties in model projections. *Clim. Chang.* **2007**, *81*, 53–70. [CrossRef]
24. Bosshard, T.; Carambia, M.; Goergen, K.; Kotlarski, S.; Krahe, P.; Zappa, M.; Schaer, C. Quantifying uncertainty sources in an ensemble of hydrological climate-impact projections. *Water Resour. Res.* **2013**, *49*, 1523–1536. [CrossRef]
25. Liang, C.; Li, H.; Lei, M.; Du, Q. Dongting lake water level forecast and its relationship with the three gorges dam based on a long short-term memory network. *Water* **2018**, *10*, 1389. [CrossRef]