Elsevier Editorial System(tm) for Journal of Hydrology or its open access mirror
Manuscript Draft

Manuscript Number: HYDROL33849R1

Title: Exploring a Long Short-Term Memory based Encoder-Decoder Framework for Multi-Step-Ahead Flood Forecasting

Article Type: Research paper

Keywords: Flood forecast; Encoder-Decoder (ED) model; Recurrent neural network (RNN); Long short-term memory (LSTM); Sequence-to-sequence

Corresponding Author: Professor Fi-John Chang, PhD

Corresponding Author's Institution: National Taiwan University

First Author: I-Feng Kao, Ms

Order of Authors: I-Feng Kao, Ms; Yanlai Zhou, PhD; Li-Chiu Chang, PhD; Fi-John Chang, PhD

Abstract: Operational flood control systems depend on reliable and accurate forecasts with a suitable lead time to take necessary actions against flooding. This study proposed a Long Short-Term Memory based Encoder-Decoder (LSTM-ED) model for multi-step-ahead flood forecasting for the first time. The Shihmen Reservoir catchment in Taiwan constituted the case study. A total of 12,216 hourly hydrological data collected from 23 typhoon events were allocated into three datasets for model training, validation, and testing. The input sequence of the model contained hourly reservoir inflows and rainfall data (traced back to the previous 8 hours) of ten gauge stations, and the output sequence stepped into 1- up to 6-hour-ahead reservoir inflow forecasts. A feed forward neural network-based Encoder-Decoder (FFNN-ED) model was established for comparison purposes. This study conducted model training a number of times with various initial weights to evaluate the accuracy, stability, and reliability of the constructed FFNN-ED and LSTM-ED models. The results demonstrated that both models, in general, could provide suitable multi-step ahead forecasts, and the proposed LSTM-ED model not only could effectively mimic the long-term dependence between rainfall and runoff sequences but also could make more reliable and accurate flood forecasts than the FFNN-ED model. Concerning the time delay between the time horizons of model inputs (rainfall) and model outputs (runoff), the impact assessment of this time-delay on model performance indicated that the LSTM-ED model achieved similar forecast performance when fed with antecedent rainfall either at a shorter horizon of 4 hours in the past (T-4) or at horizons longer than 7 hours in the past (> T-7). We conclude that the proposed LSTM-ED that translates and links the rainfall sequence with the runoff sequence can improve the reliability of flood forecasting and increase the interpretability of model internals.

Suggested Reviewers: Tanvir H. Bhuiyan PhD
Graduate Research Assistant, Civil & Environmental Engineering, University of Louisville
tanvir.h.bhuiyan@gmail.com

Figure

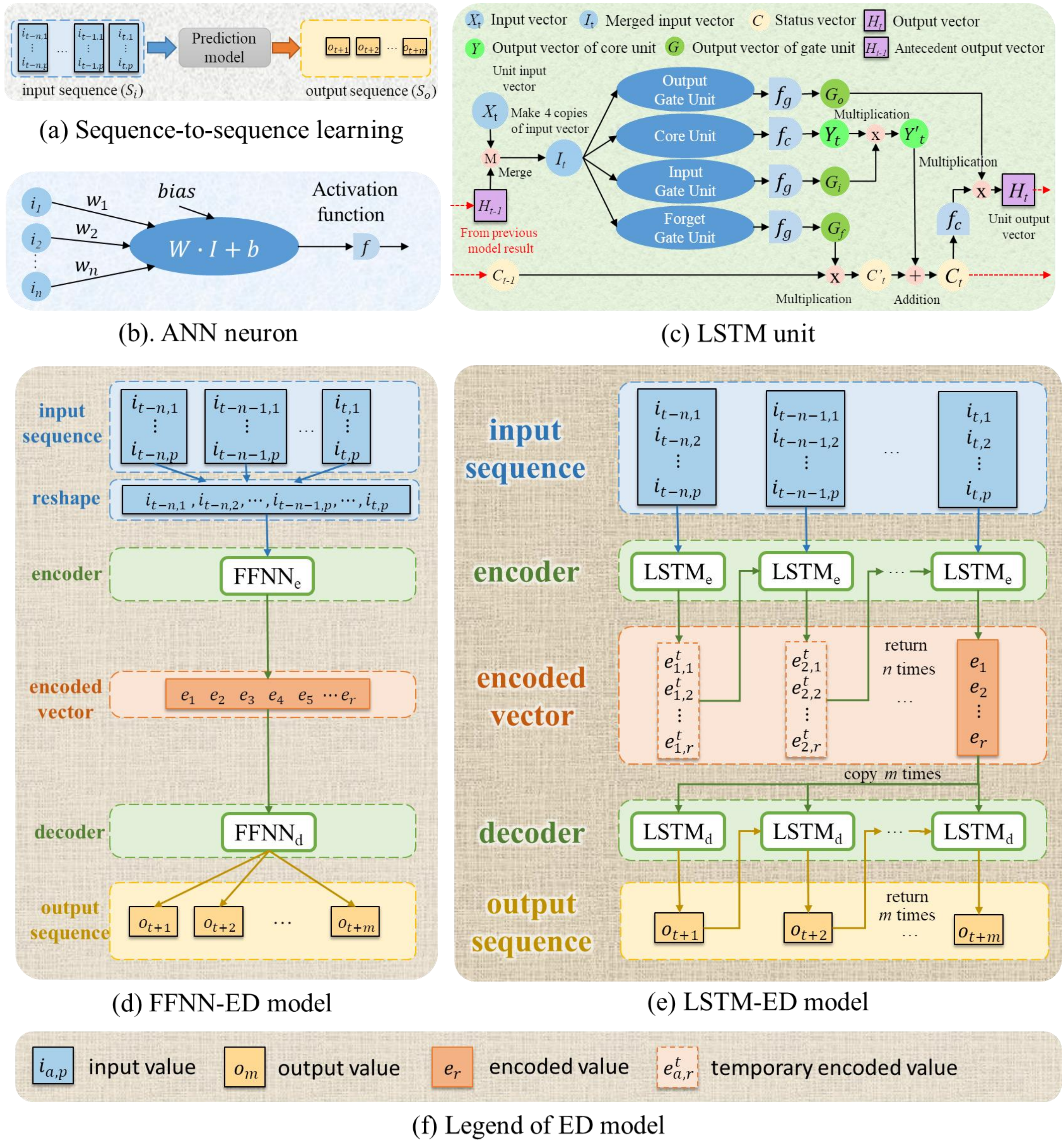


Figure 1 Architectures of the LSTM-ED and FFNN-ED models.

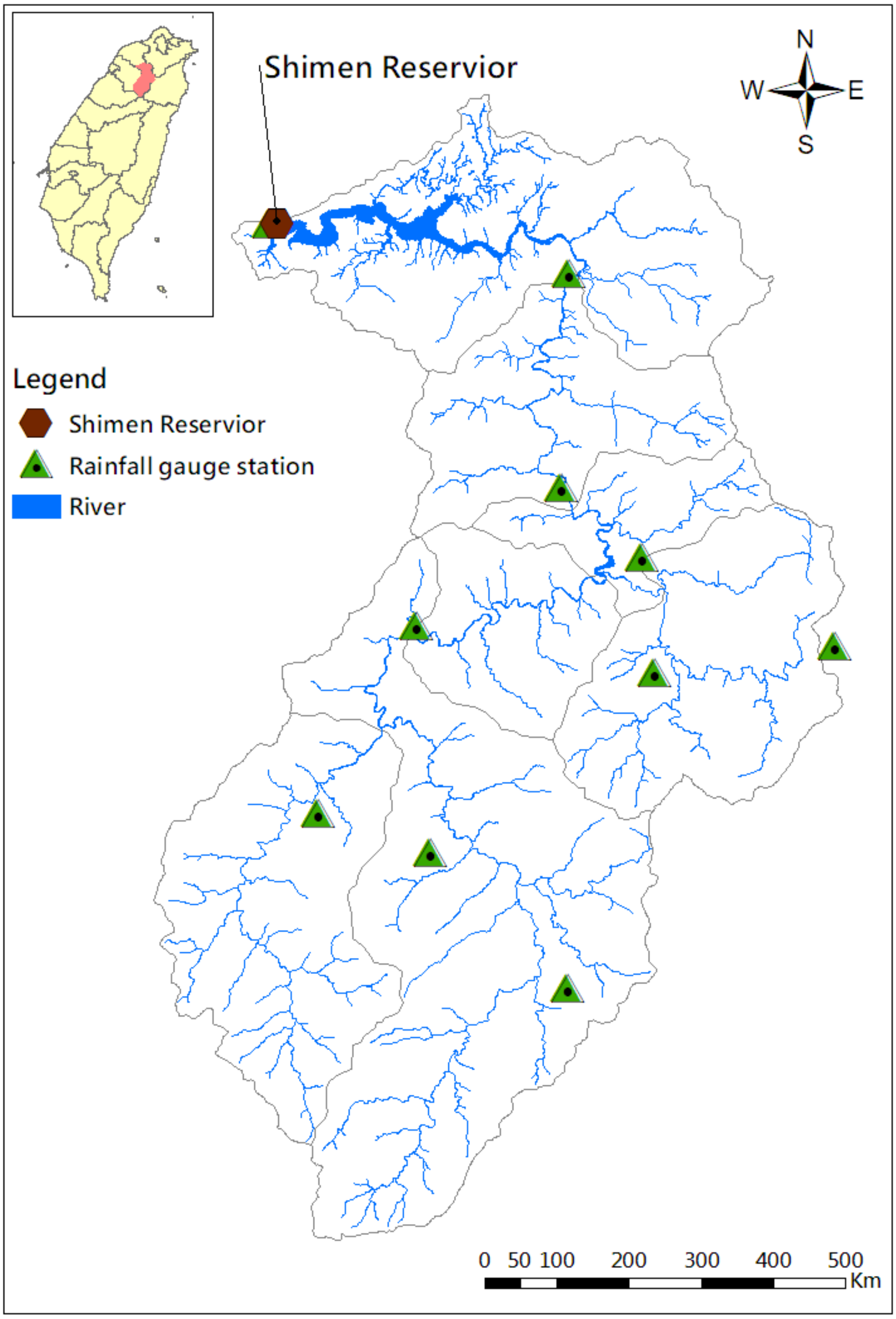
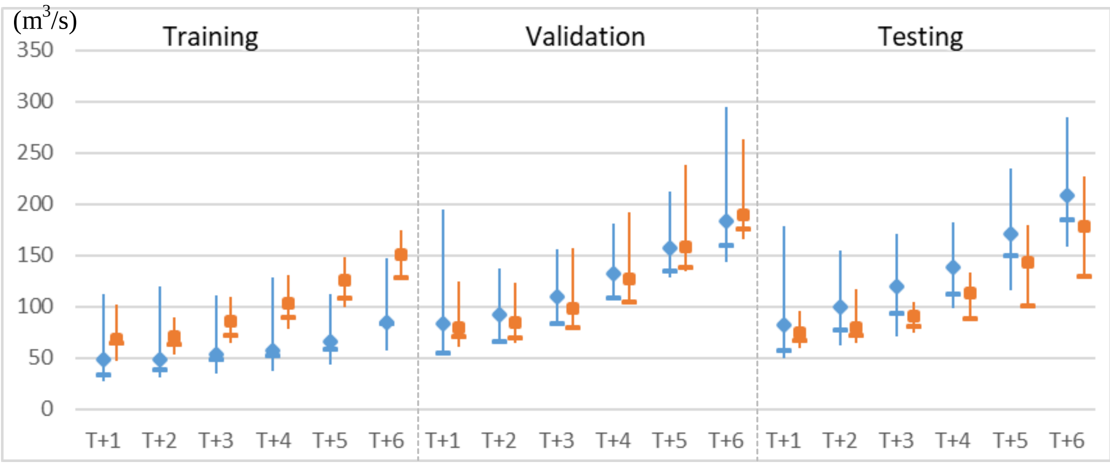
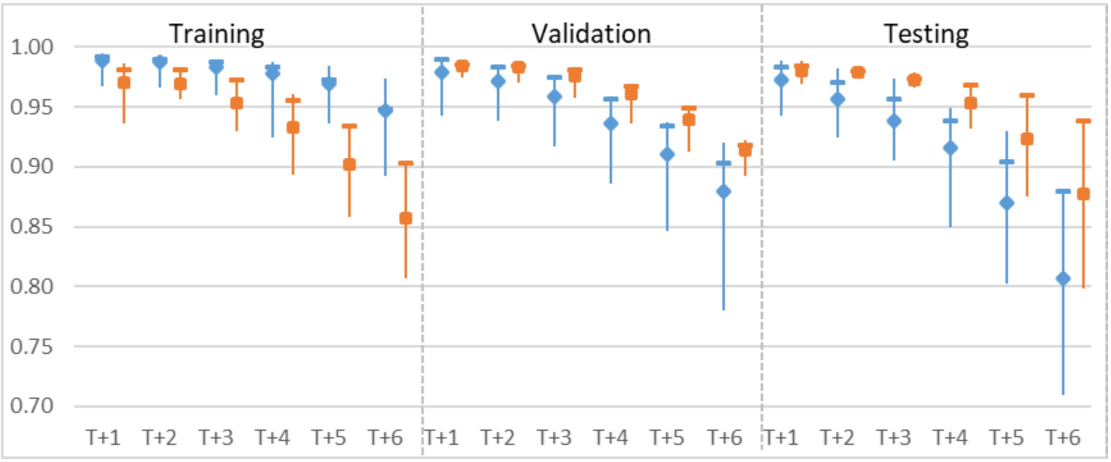


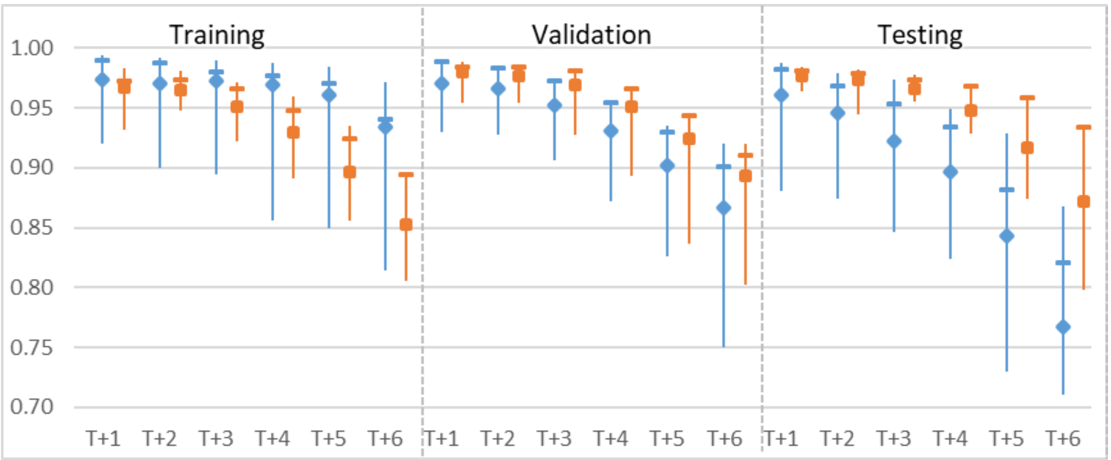
Figure 2 Locations of the Shihmen Reservoir catchment area and rainfall gauge stations.



(a) RMSE



(b) R^2



(c) NSE

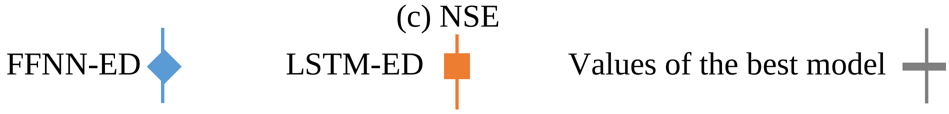
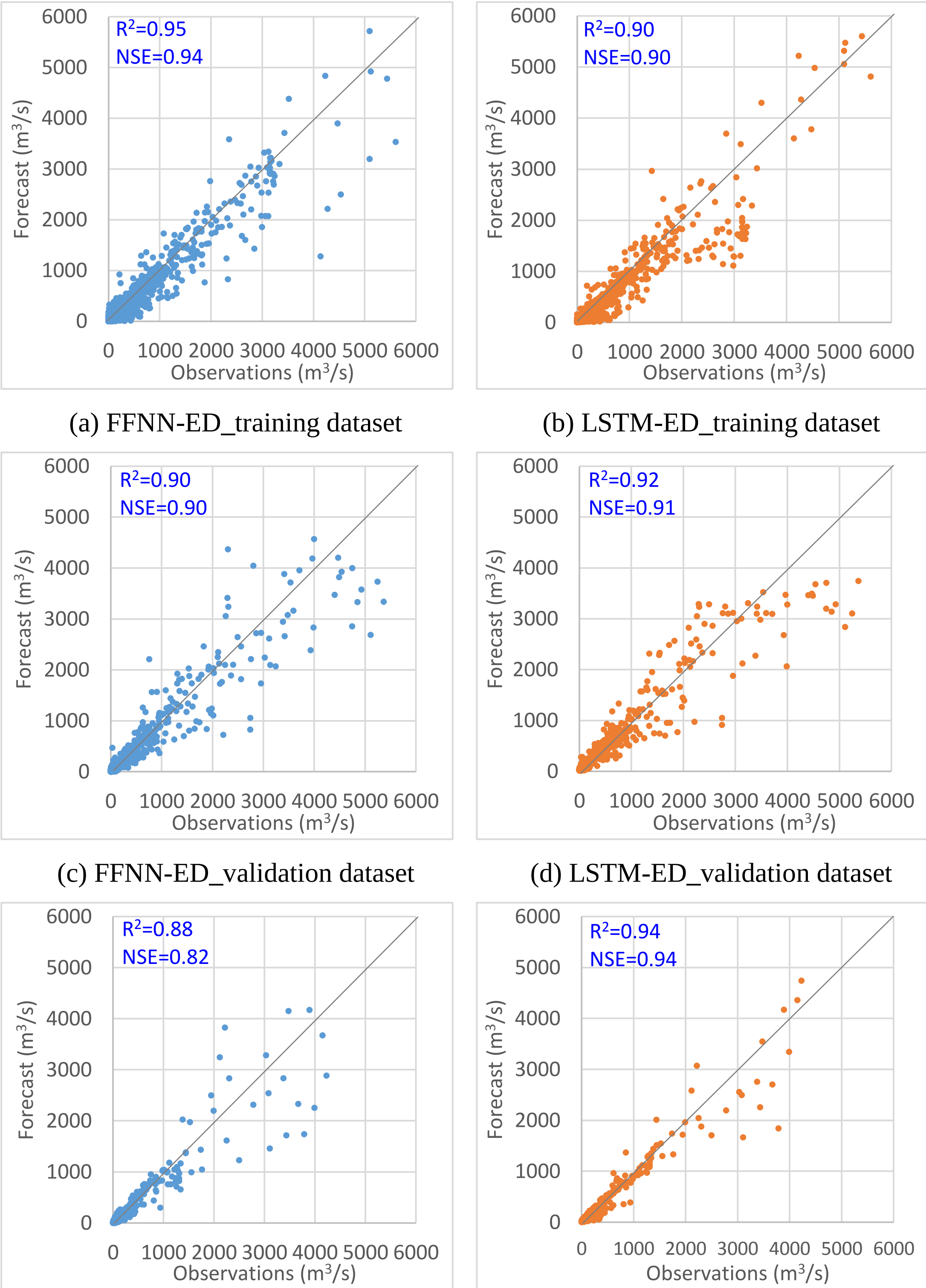


Figure 3 Performance of FFNN- and LSTM-ED models (each was performed 20 rounds). (a)-(c) RMSE, R^2 and NSE at horizons T+1−T+6, respectively. The range of an evaluation indicator is presented by a bar, where the mean and the value corresponding to best model are marked by a dot (diamond: FFNN-ED, and square: LSTM-ED) and a cross "+", respectively.



(a) FFNN-ED_training dataset

(b) LSTM-ED_training dataset

(c) FFNN-ED_validation dataset

(d) LSTM-ED_validation dataset

(e) FFNN-ED_test dataset

(f) LSTM-ED_test dataset

Figure 4 Scatter diagram of the best FFNN-ED and LSTM-ED models for T+6 forecasting.

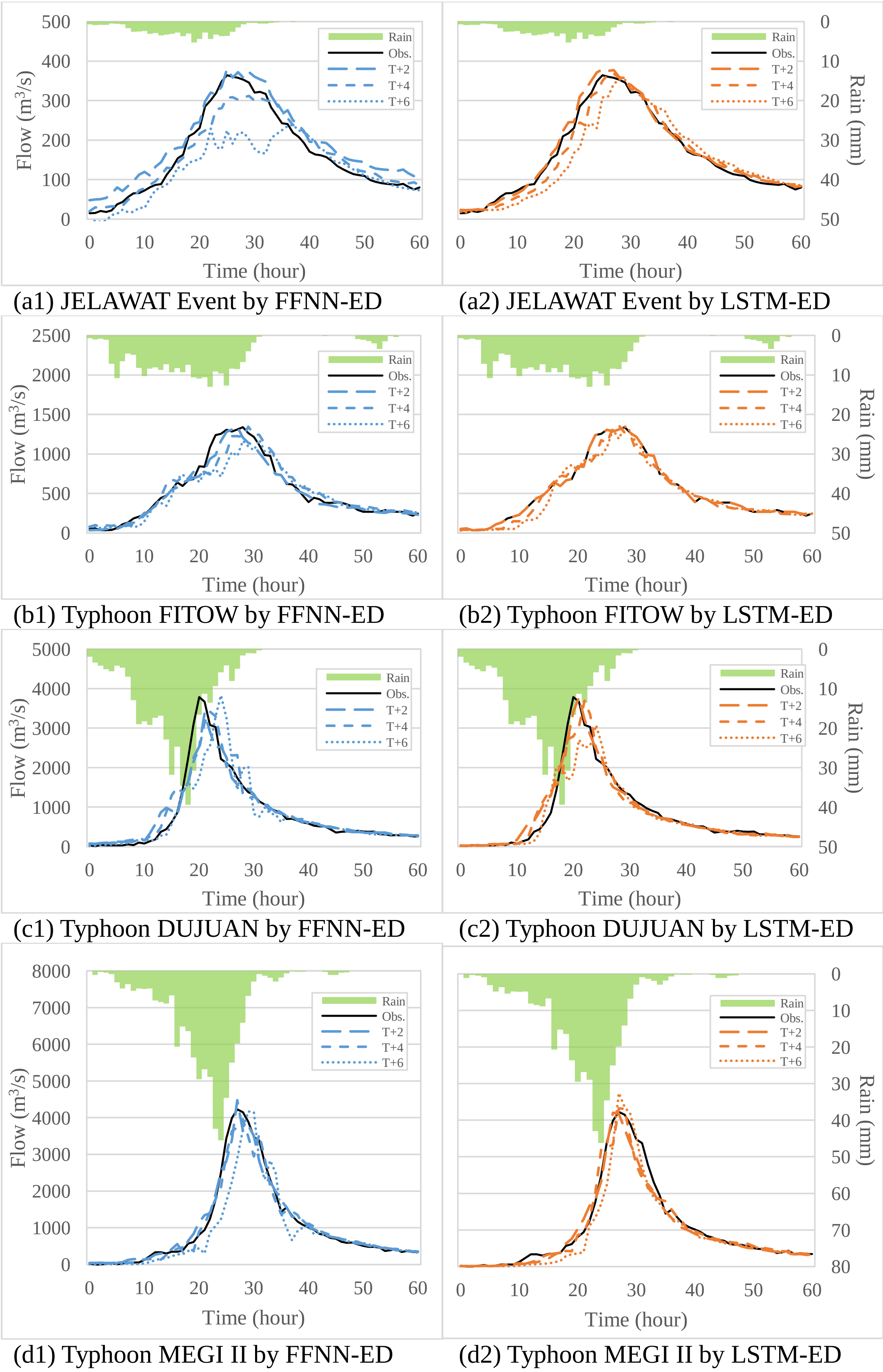


Figure 5 Comparison of observed and forecasted inflows obtained from the FFNN- and LSTM-ED models at horizons T+2, T+4 and T+6 for flood events corresponding to Typhoons JELAWAT, FITOW, DUJUAN and MEGI II.

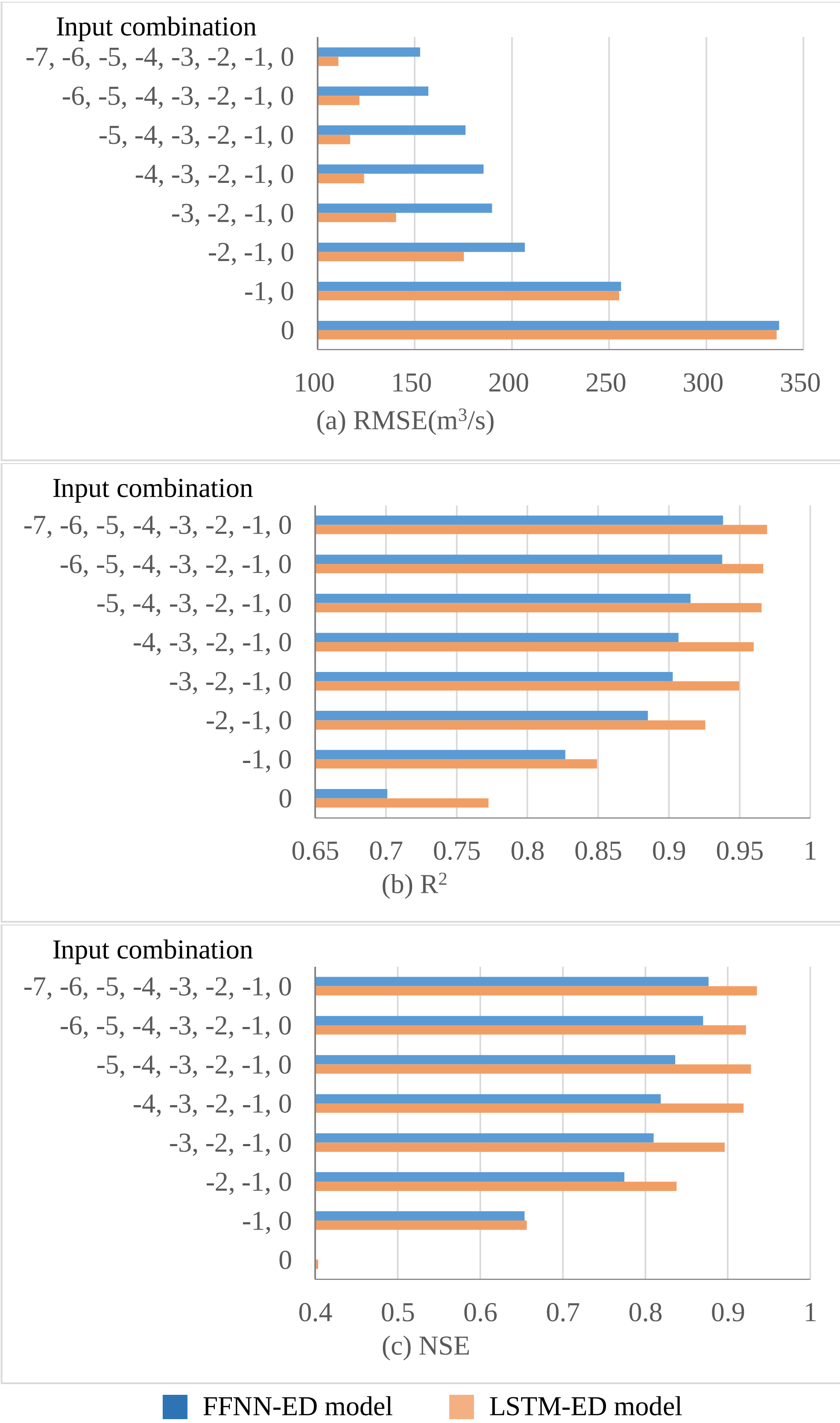


Figure 6 Performance of 6-step-ahead inflow forecasting using FFNN- and LSTM-ED models based on different numbers of antecedent (input) data, where 0 denotes data of the current time, and -n denotes data at horizon T-n (n=1-7, i.e. antecedent observed data). (a) RMSE. (b) R^2. (c) NSE.

Table 1 Typhoon events used in this study

Dataset	Typhoon	Max. inflow (m^3/s)	Year	Duration
Training	SEPAT	1,844	2007	08/07 - 09/12
	SINLAKU	3,447	2008	09/11 - 09/26
	JANGMI	3,292	2008	09/26 - 10/18
	MORAKOT	1,827	2009	08/04 - 08/24
	FANAPI	1,059	2010	09/17 - 10/07
	MEGI 1	843	2010	10/16 - 11/07
	MEARI	1,060	2011	06/23 - 07/30
	SOULIK	5,458	2013	07/12 - 07/26
	TRAMI	2,410	2013	08/20 - 09/18
	MATMO	1,180	2014	07/21 - 08/22
	FUNG-WONG II	323	2014	09/19 - 10/24
	CHAN-HOM	917	2015	07/09 - 08/06
	SOUDELOR	5,634	2015	08/06 - 09/12
Validation	WIPHA	2,788	2007	09/17 - 10/02
	KROSA	5,300	2007	10/03 - 10/24
	FUNG-WONG	2,040	2008	07/26 - 08/08
	PARMA	616	2009	10/03 - 10/31
	SAOLA	5,385	2012	07/29 - 09/03
	USAGI	1,195	2013	09/18 - 10/05
Testing	JELAWAT	439	2012	09/27 - 10/07
	FITOW	1,393	2013	10/05 - 10/24
	DUJUAN	3,786	2015	09/27 - 11/03
	MEGI II	4,227	2016	09/26 - 10/02

Table 2 Performance of the FFNN- and the LSTM-based ED models at horizons T+1−T+6 for training, validation and test datasets.

Model	Dataset	Time step	RMSE Mean	RMSE (Max – Min)	R^2 Mean	R^2 (Max – Min)	NSE Mean	NSE (Max – Min)
FFNN-ED	Training	T+1	48	(112 - 27)	0.99	(0.99 - 0.97)	0.97	(0.99 - 0.92)
		T+2	49	(120 - 31)	0.99	(0.99 - 0.97)	0.97	(0.99 - 0.90)
		T+3	53	(111 - 34)	0.98	(0.99 - 0.96)	0.97	(0.99 - 0.89)
		T+4	57	(129 - 38)	0.98	(0.99 - 0.92)	0.97	(0.99 - 0.86)
		T+5	65	(113 - 43)	0.97	(0.98 - 0.94)	0.96	(0.98 - 0.85)
		T+6	85	(147 - 57)	0.95	(0.97 - 0.89)	0.93	(0.97 - 0.81)
	Validation	T+1	83	(194 - 54)	0.98	(0.99 - 0.94)	0.97	(0.99 - 0.93)
		T+2	93	(137 - 69)	0.97	(0.98 - 0.94)	0.97	(0.98 - 0.93)
		T+3	109	(156 - 83)	0.96	(0.97 - 0.92)	0.95	(0.97 - 0.91)
		T+4	133	(182 - 106)	0.94	(0.96 - 0.89)	0.93	(0.96 - 0.87)
		T+5	157	(212 - 129)	0.91	(0.94 - 0.85)	0.9	(0.94 - 0.83)
		T+6	183	(295 - 144)	0.88	(0.92 - 0.78)	0.87	(0.92 - 0.75)
	Testing	T+1	83	(179 - 49)	0.97	(0.99 - 0.94)	0.96	(0.99 - 0.88)
		T+2	99	(155 - 62)	0.96	(0.98 - 0.92)	0.95	(0.98 - 0.87)
		T+3	120	(171 - 71)	0.94	(0.97 - 0.91)	0.92	(0.97 - 0.85)
		T+4	139	(183 - 98)	0.92	(0.95 - 0.85)	0.9	(0.95 - 0.82)
		T+5	171	(235 - 116)	0.87	(0.93 - 0.80)	0.84	(0.93 - 0.73)
		T+6	208	(285 - 158)	0.81	(0.88 - 0.71)	0.77	(0.87 - 0.71)
LSTM-ED	Training	T+1	59	(88 - 41)	0.97	(0.99 - 0.94)	0.97	(0.99 - 0.93)
		T+2	61	(77 - 46)	0.97	(0.98 - 0.96)	0.97	(0.98 - 0.95)
		T+3	74	(94 - 56)	0.96	(0.98 - 0.93)	0.95	(0.97 - 0.92)
		T+4	89	(112 - 67)	0.93	(0.96 - 0.89)	0.93	(0.96 - 0.89)
		T+5	108	(128 - 86)	0.9	(0.94 - 0.86)	0.9	(0.94 - 0.86)
		T+6	129	(150 - 110)	0.86	(0.90 - 0.81)	0.85	(0.90 - 0.81)
	Validation	T+1	68	(107 - 52)	0.99	(0.99 - 0.98)	0.98	(0.99 - 0.96)
		T+2	73	(106 - 56)	0.98	(0.99 - 0.97)	0.98	(0.99 - 0.96)
		T+3	85	(135 - 68)	0.98	(0.98 - 0.96)	0.97	(0.98 - 0.93)
		T+4	109	(165 - 89)	0.96	(0.97 - 0.94)	0.95	(0.97 - 0.9)
		T+5	137	(205 - 116)	0.94	(0.95 - 0.91)	0.93	(0.95 - 0.84)
		T+6	163	(226 - 143)	0.92	(0.92 - 0.89)	0.89	(0.92 - 0.8)
	Testing	T+1	64	(82 - 51)	0.98	(0.99 - 0.97)	0.98	(0.99 - 0.97)
		T+2	68	(101 - 56)	0.98	(0.99 - 0.98)	0.97	(0.98 - 0.95)
		T+3	78	(90 - 64)	0.97	(0.98 - 0.97)	0.97	(0.98 - 0.96)
		T+4	98	(115 - 76)	0.95	(0.97 - 0.93)	0.95	(0.97 - 0.93)
		T+5	123	(154 - 87)	0.92	(0.96 - 0.88)	0.92	(0.96 - 0.88)
		T+6	153	(195 - 111)	0.88	(0.94 - 0.80)	0.87	(0.94 - 0.80)

Table 3 Performance of the FFNN- and the LSTM-based ED models for flood forecasting at horizons T+2, T+4 and T+6 in the test dataset based on four typhoon-induced flood events at different scales.

Flood event	Time step	RMSE (m^3/s)		R^2		NSE		Time shift in peak flow (hour)	
		FFNN	LSTM	FFNN	LSTM	FFNN	LSTM	FFNN	LSTM
JELAWAT[a]	T+2	42	20	0.93	0.96	0.80	0.96	0	0
	T+4	34	29	0.89	0.91	0.87	0.90	4	2
	T+6	51	40	0.79	0.83	0.70	0.82	12	3
FITOW[b]	T+2	43	25	0.99	0.99	0.96	0.99	-1	-1
	T+4	52	31	0.97	0.98	0.95	0.98	1	-1
	T+6	57	40	0.94	0.97	0.94	0.97	3	0
DUJUAN[c]	T+2	100	64	0.96	0.98	0.94	0.98	1	1
	T+4	111	89	0.94	0.96	0.93	0.96	2	2
	T+6	175	140	0.83	0.90	0.83	0.89	4	4
MEGI II[d]	T+2	133	145	0.98	0.98	0.98	0.97	0	-1
	T+4	118	164	0.99	0.97	0.98	0.97	1	0
	T+6	327	203	0.89	0.96	0.87	0.95	2	0

[a] Typhoon JELAWAT with total rainfall of 67 mm and maximal flow of 439 m^3/s.
[b] Typhoon FITOW with total rainfall of 255 mm and maximal flow of 1,393 m^3/s.
[c] Typhoon DUJUAN with total rainfall of 413 mm and maximal flow of 3,786 m^3/s.
[d] Typhoon MEGI II with total rainfall of 443 mm and maximal flow of 4,227 m^3/s.

Exploring a Long Short-Term Memory based Encoder-Decoder Framework for Multi-Step-Ahead Flood Forecasting

I-Feng Kao[a], Yanlai Zhou[b], Li-Chiu Chang[c] , Fi-John Chang[a,]*
[a] Department of Bioenvironmental Systems Engineering, National Taiwan University, Taipei, 10617, Taiwan
[b] Department of Geosciences, University of Oslo, P.O. Box 1047 Blindern, N-0316 Oslo, Norway.
[c] Department of Water Resources and Environmental Engineering, Tamkang University, New Taipei City, 25137, Taiwan
Correspondence to: Fi-John Chang (changfj@ntu.edu.tw)

Abstract

Operational flood control systems depend on reliable and accurate forecasts with a suitable lead time to take necessary actions against flooding. This study proposed a Long Short-Term Memory based Encoder-Decoder (LSTM-ED) model for multi-step-ahead flood forecasting for the first time. The Shihmen Reservoir catchment in Taiwan constituted the case study. A total of 12,216 hourly hydrological data collected from 23 typhoon events were allocated into three datasets for model training, validation, and testing. The input sequence of the model contained hourly reservoir inflows and rainfall data (traced back to the previous 8 hours) of ten gauge stations, and the output sequence stepped into 1- up to 6-hour-ahead reservoir inflow forecasts. A feed forward neural network-based Encoder-Decoder (FFNN-ED) model was established for comparison purposes. This study conducted model training a number of times with various initial weights to evaluate the accuracy, stability, and reliability of the constructed FFNN-ED and LSTM-ED models. The results demonstrated that both models, in general, could provide suitable multi-step ahead

forecasts, and the proposed LSTM-ED model not only could effectively mimic the long-term dependence between rainfall and runoff sequences but also could make more reliable and accurate flood forecasts than the FFNN-ED model. Concerning the time delay between the time horizons of model inputs (rainfall) and model outputs (runoff), the impact assessment of this time-delay on model performance indicated that the LSTM-ED model achieved similar forecast performance when fed with antecedent rainfall either at a shorter horizon of 4 hours in the past (T-4) or at horizons longer than 7 hours in the past (> T-7). We conclude that the proposed LSTM-ED that translates and links the rainfall sequence with the runoff sequence can improve the reliability of flood forecasting and increase the interpretability of model internals.

Keywords: Flood forecast, Encoder-Decoder (ED) model, Recurrent neural network (RNN), Long short-term memory (LSTM), Sequence-to-sequence

1. Introduction

Floods are one of the most dangerous natural disasters that notoriously threaten human life and property. The International Centre for Water Hazard and Risk Management (ICHARM) reported that floods accounted for about 30% of the total natural disasters and affected more than 48% of people worldwide over the last century (ICHARM, 2009). Floods are always a major concern in inundation prone areas. This is especially true in Taiwan because there are, on average, three typhoons to invade this island each year, and typhoon-induced heavy rainfalls usually cause severe flood inundation in various cities near estuaries. Therefore, flood forecasting plays a pivotal role in flood mitigation, floodplain management, agricultural cultivation, and human life protection. The development of early warning systems for flood defense encounters great challenges, which creates an outreach demand for reliable and accurate multi-step-ahead forecasts. This pinpoints the focus of scientific research for flood defense should be placed on increasing the reliability and accuracy of forecast models at longer horizons.

Artificial neural networks (ANNs) can adequately mimic highly non-linear complex systems and are widely used to tackle the modelling of complex systems in hydrological fields (e.g. Dawson & Wilby, 2001; Chau, 2006; Kalteh et al., 2008; Nourani et al., 2014; Chandwani, et al., 2015). For instance, precipitation or evapotranspiration prediction (e.g., Shafaei et al., 2016; Shenify et al., 2016; Valipour et al., 2016; Nourani et al., 2017; Nourani et al., 2020, Nourani et al.,2019), flood forecasting (e.g., Chen et al., 2013; Chang et al., 2014; Lohani et al., 2014; Taormina et al., 2015; Chang & Tsai, 2016; Noori & Kalin, 2016; Humphrey et al., 2016; Tan et al., 2018), and rainfall-runoff modeling (e.g., Abrahart et al., 2007; Nourani & Komasi, 2013; Badrzadeh et al., 2015; Nourani, 2017; Shoaib et al., 2018; Nourani et al.,2018). Various studies also adopted ANNs for deploying hydrological prediction during typhoons and storm events in Taiwan. For example, Tsai et al. (2014) combined radar reflectivity and ground rainfall data to predict reservoir inflows using the adaptive-network-based fuzzy inference system (ANFIS), and Chang et al. (2014) used recurrent neural networks to make real-time multi-step-ahead flood forecasts for a sewerage system in Taipei City.

The attractiveness of ANNs comes mainly from the remarkable characteristics of data mining, such as learning ability, noise tolerance, and generalizability. Nevertheless, different types of ANNs do have their own merits and limitations in modeling complex systems. For instance, the feed forward neural network (FFNN) fails to suitably manage time-series data because the state of the network is erased after processing each data, i.e. information about the sequential order of the inputs is discarded, which is not desirable when handling inherently interrelated data points. Besides, the FFNN implements a fixed-sized sliding window protocol, which restrains the model from learning or capturing the long-term dependencies between input and output. On the other hand, recurrent neural networks (RNNs) are designed to capture temporal dynamics by sequentially processing the inputs for modelling the nonlinear relationship between input and output via cycles formed by the hidden nodes in the network. In recent years, Deep Learning (DL) has gained a lot of attention. Deep Neural Networks (DNNs) are powerful tools and achieve excellent performance on difficult tasks (e.g., Sainath et al., 2015; Liu et al., 2017; Zhou et al., 2019). The long short-term memory (LSTM) proposed by Hochreiter and Schmidhuber (1997) is a type of DNNs configured with an RNN architecture. The LSTM is used to deal with the exploding and vanishing gradient problems that may occur when training traditional RNNs with long-term lags. Recently, LSTMs have been implemented to explore its capability in time series forecasting of river flood (Le et al., 2019) and water table depth (Zhang et al., 2018; Jeong & Park 2019) as well as to learn long-term dependencies, e.g., storage effects within hydrological catchments (Kratzert et al., 2018) and model rainfall-runoff processes (Sezen et al., 2019).

For neural networks, the sequence-to-sequence learning trains models by converting sequences from one domain into another domain (Sutskever et al., 2014). Sequence-to-sequence models have recently achieved significant performances on complex tasks like machine translation, video to text, and question answering (Bengio et al., 2015; Venugopalan et al., 2015; Wiseman et al., 2016; Chiu et al., 2018). Sequence-to-sequence models configured with a LSTM unit have gained marvelous achievements in various fields, like anomaly detection (Fengming et al., 2017), image segmentation (Marmanis et al., 2018), video recognition (Zhu et al., 2017; Zhu et al., 2018), machine translation (Audhkhasi et al., 2017; Malinowski et al., 2017; Costa-Jussa, 2018), and air pollution forecasting (Freeman et al., 2018; Zhou et al., 2019). From the perspective of data science, hydrological analyses involve many physical processes similar to sequence-to-sequence problems. For instance, rainfall-runoff processes can be considered as the conversion of rainfall sequences into watershed discharge sequences. This provides merit to explore in-depth how the rainfall sequence can be mapped onto a runoff sequence through DNN models for reliably and accurately making multi-step-ahead flood forecasts.

This study proposes a LSTM-based Encoder-Decoder (LSTM-ED) model that integrates a sequence-to-sequence learning, two LSTM units, and an Encoder-Decoder scheme to make reliable and accurate multi-step-ahead flood forecasts for the first time. In the beginning, the sequence-to-sequence learning is employed to establish a multi-input and multi-output model structure. Then, the two LSTM units and the sequence-to-sequence learning are fused into the Encoder-Decoder scheme for constructing a multi-output deep learning neural network (i.e., LSTM-ED). To demonstrate the applicability of the LSTM-ED model in multi-step-ahead flood forecasting, this study utilizes an inflow series of the Shihmen Reservoir in Taiwan as a case study. The remainder of this study is organized as follows. Section 1 introduces the study background and makes a literature review. Section 2 presents the framework of the proposed model. Section 3 introduces the case study and materials. Section 4 presents the results and discussion of the methods applied to multi-step-ahead flood forecasting. Conclusions are then drawn in Section 5.

2. Methodology

This study proposes a LSTM-ED model to improve the reliability and accuracy of multi-step-ahead flood forecasts. For comparison, a feed-forward neural network-based Encoder-Decoder (FFNN-ED) is also constructed. Fig. 1 illustrates the architecture of the LSTM-ED and FFNN-ED models, where Fig. 1(a) presents the sequence-to-sequence learning, Fig. 1(b) presents a prototype of an ANN neuron, Fig. 1(c) presents the LSTM unit, and Figs. 1(d) and 1(e) present the frameworks of the LSTM-ED and FFNN-ED, respectively. The methods adopted in this study are briefly introduced as follows.

2.1 Sequence-to-sequence learning

Sequence prediction is commonly centered on forecasting the succeeding value in an observed sequence. Time series prediction problems usually concern either of the two frameworks: 1) a sequence of one input time step converted to a sequence of one output time step, or 2) a sequence of multiple input time steps converted to a sequence of one output time step. It will be more challenging to make a sequence prediction when taking a sequence as the input, which is termed as a sequence-to-sequence prediction problem. A sequence-to-sequence prediction problem involves an input sequence (S_i) and an output sequence (S_o). The input sequence contains known information, and the output sequence is the prediction target. Fig. 1(a) illustrates the sequence-to-sequence learning. Input and output sequences generally have different lengths, and the implementation process will require the entire input sequence as soon as the prediction of the target start. This study establishes a prediction model M to convert the input sequence into the output sequence. A sequence (S) is defined as a set of vectors (V_n) with time series relationship.

Definition 1: Sequence

$$\mathrm{S} = \{V_1, V_2, \cdots, V_n\} \tag{1}$$

$$V_x = \left\{v_{n,1}, v_{n,2}, \cdots, v_{n,p}\right\} \tag{2}$$

where n is the length of the input time series (the lookback length of time) and p is the number of elements (variables) in a vector.

Definition 2: Prediction model

$$S_o = M(S_i) \tag{3}$$

In this study, the input sequences contain hourly data of ten rainfall gauge stations and the inflow data of the Shihmen Reservoir collected from the horizon t-n to the current time t. The output sequence is the multi-step-ahead reservoir inflow. That is to say, this study intends to establish a rainfall-runoff model for making reservoir inflow forecasts based on antecedent rainfall and inflow data.

Definition 3: rainfall-runoff model

$$S_i = \{I_{t-n}, I_{t-n-1}, \dots, I_{t-1}, I_t\} \tag{4}$$

$$I_t = \{i_{t,1}, i_{t,2}, \dots, i_{t,p}\} \tag{5}$$

$$S_o = \{O_{t+1}, O_{t+2}, \dots, O_{t+m}\} \tag{6}$$

$$t, p, n, m \in N \tag{7}$$

where *I* denotes a vector of the input sequence S_i, *O* denotes a vector of the output sequence S_o, *t* is the current time, *n* is the lookback length of time, *m* is the forecast horizon, and *p* is the number of gauge stations (rainfall or inflow in this study).

2.2 Long short-term memory (LSTM) unit

The LSTM units have several architectures. A common architecture comprises a core unit (the memory part) and three gate units (input, output and forget gates) that direct the information flow inside the LSTM unit (Fig. 1(c)). The computation steps of the LSTM are shown in Eqs. (8)-(16), referred from Hochreiter and Schmidhuber (1997).

(1) Combine the antecedent output vector with the input vector.

$$I_t = H_{t-1} + X_t \tag{8}$$

where I_t is the merged input vector that combines the antecedent output vector H_{t-1} with the input vector X_t.

(2) Calculate the output vector of the core unit.

$$Y_t = f_c(W_c \cdot I_t + b_c) \tag{9}$$

where Y_t and f_c are the output vector and the activation function of the core unit, respectively, W_c is the connection weight, and b_c is the bias of the core unit.

(3) Calculate the output vectors corresponding to the units of the input gate, the forget gate and the output gate.

$$G_i = f_g(W_i \cdot I_t + b_i) \tag{10}$$

$$G_f = f_g\left(W_f \cdot I_t + b_f\right) \tag{11}$$

$$G_o = f_g(W_o \cdot I_t + b_o) \tag{12}$$

where G_i, G_f and G_o are the output vectors (gate values) obtained from the input gate, forget gate and output gate units, respectively. The weights (W_i, W_f, W_o) and bias (b_i, b_f, b_o) are the parameters corresponding to the three gate units. f_g denotes the activation function of a gate unit, and its output value falls between zero and one.

(4) Calculate the new cell state vector of long-term memory.

$$Y'_t = G_i \cdot Y_t \tag{13}$$

$$C'_t = G_f \cdot C_{t-1} \tag{14}$$

$$C_t = Y'_t + C'_t \tag{15}$$

where Y'_t is the raw output of the LSTM unit, and C'_t is the antecedent cell state vector (C_{t-1}) that is finely tuned by the forgot gate value (G_f). C_t is the new cell state vector of long-term memory, and it will return to the LSTM unit when being reused. In this step, the cell state vector of long-term memory gains new information but forgets some old information.

(5) Calculate the output vector of the LSTM unit.

$$H_t = f_c(C_t) \cdot G_o \tag{16}$$

where H_t is the output vector of the LSTM unit, and f_c is the same activation function as the one used in the core unit. The activation function can stabilize the output value after the LSTM unit are reused many times. The output gate value (G_o) can control whether the LSTM unit should produce an output or not. In addition, the cell state vector is not affected by f_c in this step such that it is much easy to keep the raw output (Y'_t) of this LSTM unit for the next reuse.

2.3 Encoder-Decoder model

Encoder-Decoder (ED) models have been developed to effectively tackle the challenging sequence-to-sequence prediction problems lately. From the perspective of model architecture, an ED model has two implementation phases: the first is to read the input sequence and encode it into a fixed-length vector, and the second is to decode the fixed-length vector and output the predicted sequence. The innovation of the ED model is that the model facilitates a fixed-sized internal representation such that input sequences are read to and output sequences are read from. It was noticed that an ED model configured with LSTM was developed to cope with natural language processing problems and achieved state-of-the-art performance in the text translation field. This study intends to implement the ED architecture for translating the rainfall sequence into the runoff sequence, where the lengths of the input sequence and the output sequence are fixed. The two ED models with different encoders and decoders are introduced as follows.

2.3.1 FFNN-ED model

Fig. 1(d) illustrates the structure of the FFNN-ED model, which uses the FFNN in the encoder and decoder schemes. The input sequences are reshaped to a 1-dimensional vector before entering the encoder. Then the encoder generates a 1-dimensional encoded vector (error vector) and feeds it to the decoder. Finally, the decoder produces a 1-dimensional vector of the output sequence. It is noted that the FFNN-ED model servers as a comparative model in this study.

2.3.2 LSTM-ED model

The structure of the proposed LSTM-ED model is shown in Fig. 1(e). This study utilizes the LSTM unit in the encoder and decoder schemes for improving the learning of the continuity in input and output sequences. The LSTM unit will be reused many times for “reading” the input sequence and “writing” the output sequence sequentially. The numbers of times to reuse the LSTM units in encoding and decoding schemes depend on the lengths of the input sequence and the output sequence, respectively. For the encoding phase, the LSTM unit serves as a “collector” for accumulating rainfall information. The LSTM unit can well simulate the physical mechanism of the rainfall-runoff process, as shown in the previous studies (e.g., Kratzert, et al., 2018). The process of reading a vector in the input sequence one-by-one is similar to the way that rain falls to the ground sequentially. Integrating information through the recurrent architecture is similar to the concentration of river flow with a time lag. Discarding the previous input information by the forget gate (i.e., the LSTM computation step (4) in Section 2.2) is similar to the hydrological phenomenon of precipitation loss and infiltration during the rainfall-runoff process. When the encoder reads a vector, the LSTM unit will generate a temporary encoded vector. The encoding process will repeat n times so that all the input vectors enter the LSTM to produce their corresponding encoded vectors. For the decoding phase, the LSTM unit generates the output value of forecasted discharge (i.e., the reservoir inflow) sequentially. The input to the LSTM unit during the decoding phase is a vector that merges the final encoded vector and the output value (reservoir inflow) of the previous LSTM. It is noted that the currently observed reservoir inflow is used to produce the output value of the LSTM at horizon of 1 hour ahead (T+1) because there is no antecedent forecasted reservoir inflow at the beginning of the decoding phase. The recurrent and sequential processes (features) of the decoding phase that generates the output sequence is similar to the continuity of river flow in a watershed. The LSTM unit fed with the previous flow information can maintain the continuous feature of flows, which is not available in the FFNN unit of the FFNN-ED model. The advantage of the LSTM-ED model is that it can produce more stable and less fluctuated output values. Therefore, This study expects the LSTM-ED model can perform better than the FFNN-ED model.

2.4 Evaluation of model performance

This study adopts the root mean square error (RMSE), the coefficient of determination (R^2), and the Nash–Sutcliffe model efficiency coefficient (NSE) to evaluate the forecast results of the two ED models. The RMSE value represents the error between the forecasted and the observed values, and its unit is the same as the output value of the model. The RMSE value ranges from 0 to infinity. A model with its RMSE value closer to 0 implies that it can produce more accurate forecasts. The RMSE can be calculated by the following equation.

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{t=1}^{N}(d_i - y_i)^2} \tag{17}$$

where N is the number of samples, d_i is the target output value, and y_i is the model output value.

The R^2 value is the proportion of the variance in the dependent variable that is predictable by the independent variable(s), and it is commonly used to evaluate the linear correlation between model outputs and target outputs. The R^2 value ranges from 0 to 1. A model with its R^2 value closer to 1 implies it can predict more accurately.

The R^2 value can be calculated by the following equation.

$$\mathrm{R}^2 = \left[\frac{\sum_{t=1}^{N}(d_i - \bar{d})(y_i - \bar{y})}{\sqrt{\sum_{t=1}^{N}(d_i - \bar{d})^2 \sum_{t=1}^{N}(y_i - \bar{y})^2}}\right]^2 \tag{18}$$

where $\bar{d}$ is the mean of target outputs, and $\bar{y}$ is the mean of model outputs. Other symbols are consistent with those of Eq. (17).

The NSE is commonly used to evaluate hydrological prediction models. The NSE value ranges from negative infinity to 1. A model with NSE value closer to 1 implies it can predict more accurately. A model with its NSE value less than 0 reveals it performs worse than a model that produces mean values only. The NSE value can be calculated by the following equation.

$$\mathrm{NSE} = 1 - \frac{\sum_{t=1}^{N}(d_i - y_i)^2}{\sum_{t=1}^{N}(d_i - \bar{d})^2} \tag{19}$$

where all the symbols are consistent with those of Eqs. (17) and (18).

3. Case study and materials

3.1 Study area

The Shihmen Reservoir basin with an area of 763.4 km^2 is located in northern Taiwan (Fig. 2). It has an annual average rainfall of 2,504 mm and an annual inflow of 1.47 billion m^3. In this basin, 76% of rainfall occurs within six months (May-October), with a high incidence of typhoon events (Water Resources Agency, Taiwan, 2016). This is consistent with typical rainfall-runoff characteristics in Taiwan.

3.2 Observational data

This study collected the monitoring records associated with 23 typhoon events occurring from 2007 to 2016, including hourly rainfall data of ten rainfall gauge stations and the inflow data of the Shihmen Reservoir. Table 1 shows the information of typhoon events used in this study. A total of 12,216 hourly data were allocated into three datasets for model training (8,232 from 13 events), validation (2,688 from 6 events), and testing (1,296 from 4 events). The training dataset was used to adjust model parameters such as the weights and bias of the neural network. The validation dataset was used to verify whether a model is undertrained or overfitting. The test dataset was used to evaluate model performance.

3.3 Model construction

After data pre-processing, the observational data were organized into an input sequence and an output sequence. According to historical rainfall-runoff data of this basin, the longest flood propagation time was 8 hours. Therefore, the input sequence contained reservoir inflows and hourly data (traced back to the previous 8 hours of the current time) of ten rainfall gauge stations. Considering the demand for the flood control of the Shihmen Reservoir, the output sequence stepped into 1- up to 6-hour-ahead reservoir inflow.

The FFNN-ED model behaved in a similar way to the BPNN model with 30 neurons in the hidden layer (i.e., the length of the encoded vector), and it was trained by the Levenberg-Marquardt optimizer using MATLAB 2018b. The number of neurons was determined by trial and error. For comparison purposes, the length of the encoder vector for the LSTM-ED model was also set as 30. The LSTM-ED model was implemented in Python, where the Python library Keras and the Adam optimizer compiling were used in the training stage, and the dropout regularization was adopted to avoid overfitting.

4. Results and discussion

Three evaluation indicators were conducted to evaluate the performance of the LSTM-ED and FFNN-ED models. To verify model reliability, this study also evaluated the model performance of four test flood events. Finally, the impacts of the number of antecedent observed data (model inputs) on model performance were investigated.

4.1 Evaluation of model performance

It was worth mentioning that the structures of LSTM-ED and FFNN-ED models were different, so as their training algorithms. Therefore, this study investigated the effectiveness and reliability of both models. Considering the FFNN-ED model had no recurrent structure, the Levenberg-Marquardt optimizer with the second-order training characteristics was implemented because it could reduce errors faster than the gradient descent optimizer with the first-order training characteristics. In contrast, the LSTM-ED model has a complex recurrent structure, an optimizer (such as Adam) with the first-order training characteristics can reduce the complexity of the training algorithm and make the model easy to train. The first-order training algorithm, however, required more iterations, and therefore the training time of the LSTM-ED model was much longer than the FFNN-ED model. It is noticed that the computation time of the LSTM-ED model is, on average, about 20 times longer than that of the FFNN-ED model (Computer specifications: Intel i7-6700 CPU, 16GB Memory, and 1 TB Storage. FFNN-ED: Matlab 2018b, Levenberg-Marquardt Optimizer, and 3-5 minute training time per round. (2) LSTM-ED: Python 3.6 with Keras 2.2.4, Adam Optimizer, and 60-100 minutes training time per run). The training time, however, is not the main issue to prohibit the utilization of these models. According to the runtime records of the test case, the computation time of the two constructed Encoder-Decoder models (FFNN-ED and LSTM-ED) for on-line forecasting is less than 1 minute. This study raised more concerns about the accuracy, stability, and reliability of the constructed models instead. Therefore, both models were trained 20 rounds (with different initial weights) using the training datasets, and then model performances were evaluated by validation and test datasets. The best model of each framework was determined as the model that produced the highest R^2 value averaging over six time steps in the validation stages. Finally, the best FFNN-ED model was compared with the best LSTM-ED model.

The results (maximum, minimum, and mean values over 20 rounds) of the FFNN-ED and the LSTM-ED models at each of the six horizons in all three stages are shown in Table 2 and Fig. 3. It appears that both models, in general, could be trained almost perfectly, in terms of very small RMSE values and very high R^2 and NSE values at each horizon in the training stages. In addition, the forecast errors of both models increased as the forecast horizon increased, which was caused by the lack of future rainfall information in the long forecast horizons. The results of performance show that the FFNN-ED models, in general, perform better than the LSTM-ED models in the training stages, but this is not the case in validation and testing stages (in fact, their performances are quite the opposite). The FFNN-ED models produced much larger error ranges than the LSTM-ED models in all three stages. For the FFNN-ED models, their mean values of the RMSE in the validation and testing stages at the six horizons are 50%−250% higher than those of the training stages. For the LSTM-ED model, the RMSE values are only slightly higher in the validation and testing stages than in the training stage. The results of performance showed that the LSTM-ED model reduced forecast errors (RMSE) by 3% up to 38% in the testing stages at horizons 1 to 6 hours ahead (T+1−T+6), as compared to the FFNN-ED model. Fig. 3 explicitly presents the detailed results (maximum, mean, and minimum over 20 rounds) of both models at each of the six horizons in all three stages. The results (20 rounds) of the constructed LSTM-ED models are much more consistent than those of the constructed FFNN-ED models. The results of performance also showed that the LSTM-ED model produced higher R^2 and NSE values than the FFNN-ED model, especially true at long horizons (> 2 hours) in the validating and testing stages. These results support that the LSTM-ED model outperforms the FFNN-ED model, in terms of model stability, reliability, and accuracy.

Fig. 4 shows the scatter diagrams of the best FFNN-ED and LSTM-ED models for T+6 forecasting in the training, validating, and testing stages, respectively. The results of T+6 forecasting show that both models, in general, fit well to the observed data in all three stages, and the LSTM-ED model has better performance (in terms of higher R^2 and NSE values and narrowly dispersed points) than the FFNN-ED model in the validating and testing stages. This is especially true in the testing cases, as the study can easily conclude that the LSTM-ED model can make more accurate T+6 forecasting, especially under the conditions of high flow (>2000 cms), than the FFNN-ED model.

4.2 Evaluation of model reliability

According to the flood forecast results of the four test events shown in Table 3, the LSTM-ED model is superior to the FFNN-ED model with respect to RMSE, R^2, and NSE values. The hydrographs (near the peak flow) of observations and model forecasts at horizons T+2, T+4, and T+6 are illustrated in Fig. 5. The first flood event induced by Typhoon JELAWAT (total rainfall < 67 mm, maximal inflow=439 m^3/s) had the smallest magnitude. The performances of both models for this event, however, are the worst, as compared to those of the other three test events. As shown in Figs. 5(a1) and 5(a2), both models under-estimated peak flows.

The second flood event induced by Typhoon FITOW was also a small-scale flood event, which was considered less hazardous to the Shihmen Reservoir. Its maximal flow was 1,393 m^3/s, and the accumulated rainfall in the basin during the first 48 hours of the typhoon period was 255 mm. Figs. 5(b1) and 5(b2) indicate that the LSTM-ED model performs better in flow peak at horizons T+2, T+4, and T+6. Besides, the LSTM-ED model maintains similar performance at all the three forecast horizons, yet the forecast error of the FFNN-ED model increases significantly.

Moreover, the LSTM-ED model can accurately forecast the peak flow, whereas the FFNN-ED model underestimates the peak flow.

The third flood event induced by Typhoon DUJUAN was a large-scale flood event, and it was considered moderately hazardous. Its maximal flow reached 3,225 m^3/s, and the accumulated rainfall in the basin during the first 48 hours of the typhoon period achieved 389 mm. Because there were multiple peaks in the rainfall distribution, the forecasts obtained from both models were unstable and undulate in the rising limb of the flood. The forecast results of both models at horizons T+2, T+4, and T+6 illustrated in Figs. 5(c1) and 5(c2) display unstable forecasts and multiple peaks. The results of this flood event forecasting show that the forecasting at horizons T+4, unexpectedly, performs better than the forecasting at horizons T+2 and T+6. It is observed from Figs. 5(c1) and 5(c2) that the interval between peaks in the rainfall distribution spans approximately 4 hours. This information may be the key to solving flood forecasting problems suffering from multi-peak rainfall distribution, which will be investigated in future research.

The fourth flood event caused by Typhoon MEGI II was a large-scale flood event, and it was considered highly hazardous. Its maximal flow reached 4,227 m^3/s, and the accumulated rainfall in the basin during the first 48 hours of the typhoon period achieved 443 mm. Table 3 indicates that the LSTM-ED model is superior to the FFNN-ED model at horizons T+2, T+4 and T+6. The RMSE value of the LSTM-ED model was about 50% smaller than that of the FFNN-ED model at each horizon. In addition, the R^2 and NSE values of the LSTM-ED model still exceeded 0.95 for all the three horizons. Figs. 5(d1) and 5(d2) clearly show that the LSTM-ED model produces more accurate forecasts of peak flow than the FFNN-ED model.

Overall, the LSTM-ED model not only can produce more accurate forecasts on high flow, especially true for flood events induced by single-peak rainfall distributions (e.g., MEGI II, FITOW), but also can produce more stable forecasts on flood events of multi-peak rainfall distributions (e.g., JELAWAT, DUJUAN), as compared with the FFNN-ED model. The FFNN-ED model could easily learn the linear correlation exhibiting in the rainfall-runoff process but failed to simulate the dynamics of the system effectively. Therefore, the FFNN-ED model either seriously over-estimated or under-estimated peak flow and had an obvious time-delay (time shift) problem. As for the LSTM-ED model, the output flow value (e.g., T+*i*) of a LSTM decoder is recurrently fed into the same decoder unit for making the forecast at the next horizon (e.g., T+*i*+1). Therefore, the flow forecasts correlate with their previous output flow. As described in Section 2.3, the process of information flow of the LSTM structure is similar to the rainfall-runoff process. The forecast reliability of the LSTM-ED model is significantly higher than that of the FFNN-ED model throughout the rising limb, peak flow, and the recession limb of a flood. In short, the LSTM-ED model not only achieves a better outcome than the FFNN-ED model in simulating complex rainfall-runoff processes but also improves the reliability and accuracy of multi-horizon forecasting of flood events.

4.3 Impact assessment of input combination on model performance

This study reduced the length of the input sequence and identified the impact of the length reduction on the two ED models. Fig. 6 illustrates the T+6 forecast performance of the FFNN-ED and LSTM-ED models with different input combinations. The results show that the two models experience a continuous decrease in performance as the length of the input sequence decreases, and this situation is notably worse for the FFNN-ED model. The results of impact assessment show that there is no significant difference in the performances of the LSTM-ED model with input information spanning 8 (T-7, ..., 0) down to 5 (T-4, ..., 0) continuous hours. Besides, the FFNN-ED model performs inferior to the LSTM-ED model under the same scenarios. Comparing input information spanning 4 (T-3, …, 0) and 8 (T-7, …, 0) continuous hours, the RMSE value increases by 20% while the R^2 and NSE values decreases by 10% for the FFNN-ED model. In contrast, the RMSE value decreases by 10% while the R^2 and NSE values make no significant changes for the LSTM-ED model. The results indicate that the LSTM-ED model is able to achieve similar forecast performance with less input information while the FFNN-ED model does have difficulty in making such achievement. This study speculates that this is because the recurrent architecture of the LSTM unit feeds the next input vector with the output vector of the previous unit such that the model can learn the temporal pattern in a continuous way.

5. Conclusions

This study proposes a LSTM-based Encoder-Decoder (LSTM-ED) framework to model multi-step-ahead flood forecasting. The results reveal that fusing the LSTM unit with sequence-to-sequence learning into the ED model not only can improve the accuracy and reliability of flood forecasting but also increase the interpretability of the framework through translating the rainfall sequence to the runoff sequence. Besides, the LSTM-ED model can better learn the rainfall-runoff process and provide more reliable and accurate multi-step ahead forecasts than the FFNN-ED. The findings of this study are summarized below.

(1) The FFNN-ED model can produce a small error and consume less time in convergence during model training, but it suffers from unstable (wide variability) and overfitting problems. The LSTM-ED model can reduce multi-step-ahead

forecast error and significantly mitigate the overfitting problem to provide more stable performance. Still, it demands more time in training the model.

(2) In the flood forecasting of four test events, the time-delay at the horizon of 6 hours ahead (T+6) for the LSTM-ED model is much shorter than that of the FFNN-ED model. The LSTM-ED model not only can make more accurate forecasts on high flow of flood events induced by single-peak rainfall distribution but also can make more stable forecasts on flood events induced by multi-peak rainfall distribution, taking the FFNN-ED model as the benchmark.

(3) The LSTM-ED model plays an important role in modeling the rainfall-runoff process for multi-step ahead flood forecasts, where the LSTM unit in the encoder can effectively integrate sequential rainfall patterns with watershed discharge while the LSTM decoder can systematically and precisely forecast the flow sequence in a continuous way.

(4) According the impact assessment of the length of the input sequence on model performance, the LSTM-ED model can produce much better performances than the FFNN-ED model, especially when being fed with less input information. This study speculates that this is because the architecture of the LSTM unit feeds the next input vector with the output vector of the previous unit such that the model can learn the temporal pattern in a continuous way.

A barrier to applying the ANNs (or DNNs) is their black-box nature that could not provide explicit internal representation of hydrologic processes. In this study, the input sequence was translated into the output sequence by configuring them into the LSTM-based Encoder-Decoder learning framework and the implementation process of the LSTM-ED model was linked with hydrological processes (i.e. the rainfall-runoff process), as discussed briefly in Sect. 2.3.2. We believe that improving the reliability and accuracy of model performance and increasing the interpretability of the network internals would increase the trust in data-driven approaches and lead to more practices in hydrologic sciences.

There are quite many sequence-to-sequence problems encountered in hydrological fields. This study is only a case that applies the LSTM-ED to modeling the rainfall-runoff problem. More extensive research on hydrological disasters (e.g., regional flooding or drought) and water resources management (e.g., inflow forecasting and groundwater estimation) using the proposed methods can be explored in the future.

Acknowledgments

The Ministry of Science and Technology supported this research, Taiwan (Grant numbers: MOST 107-2621-M-002-004-MY3, and MOST 108-2911-I-002-507). The data provides by the Northern Region Water Resources Office, Water Resources Agency, Taiwan is much appreciated.

References

1. Abrahart, R. J., Heppenstall, A. J., See, L. M., 2007. Timing error correction procedure applied to neural network rainfall—runoff modelling. Hydrological sciences journal 52(3), 414-431.
2. Adikari, Y., Yoshitani, J. 2009. Global Trends in Water-Related Disasters: An Insight for Policymakers, International Centre for Water Hazard and Risk Management (ICHARM). The United Nations World Water Development Report 3, Tsukuba, Japan.
3. Audhkhasi, K., Rosenberg, A., Sethy, A., Ramabhadran, B., Kingsbury, B., 2017.

End-to-end ASR-free keyword search from speech. IEEE Journal of Selected Topics in Signal Processing 11(8), 1351-1359.

4. Badrzadeh, H., Sarukkalige, R., Jayawardena, A. W., 2015. Hourly runoff forecasting for flood risk management: Application of various computational intelligence models. Journal of Hydrology 529, 1633-1643.
5. Bengio, S., Vinyals, O., Jaitly, N., Shazeer, N., 2015. Scheduled sampling for sequence prediction with recurrent neural networks. In Advances in Neural Information Processing Systems, 1171-1179.
6. Chandwani, V., Vyas, S. K., Agrawal, V., Sharma, G., 2015. Soft computing approach for rainfall-runoff modelling: A review. Aquatic Procedia 4, 1054-1061.
7. Chang, F. J., Chen, P. A., Lu, Y. R., Huang, E., Chang, K. Y., 2014. Real-time multi-step-ahead water level forecasting by recurrent neural networks for urban flood control. Journal of Hydrology 517, 836-846.
8. Chang, F. J., Tsai, M. J., 2016. A nonlinear spatio-temporal lumping of radar rainfall for modeling multi-step-ahead inflow forecasts by data-driven techniques. Journal of Hydrology 535, 256-269.
9. Chang, L. C., Shen, H. Y., Chang, F. J., 2014. Regional flood inundation nowcast using hybrid SOM and dynamic neural networks. Journal of hydrology 519, 476-489.
10. Chau, K. W., 2006. A review on integration of artificial intelligence into water quality modelling. Marine pollution bulletin 52(7), 726-733.
11. Chen, P. A., Chang, L. C., Chang, F. J., 2013. Reinforced recurrent neural networks for multi-step-ahead flood forecasts. Journal of Hydrology 497, 71-79.
12. Chiu, C. C., Sainath, T. N., Wu, Y., Prabhavalkar, R., Nguyen, P., Chen, Z., Kannan, A., Weiss, R. J., Rao, K., Gonina, E., Jaitly, N., Li, B., Chorowski, J.,

Bacchiani, M., 2018. State-of-the-art speech recognition with sequence-to-sequence models. In 2018 IEEE International Conference on Acoustics, Speech and Signal Processing, 4774-4778.

13. Costa-Jussa, M. R., 2018. From Feature to Paradigm: Deep Learning in Machine Translation. Journal of Artificial Intelligence Research 61, 947-974.
14. Dawson, C. W., Wilby, R. L., 2001. Hydrological modelling using artificial neural networks. Progress in physical Geography 25(1), 80-108.
15. Fengming, Z., Shufang, L., Zhimin, G., Bo, W., Shiming, T., Mingming, P., 2017. Anomaly detection in smart grid based on encoder-decoder framework with recurrent neural network. The Journal of China Universities of Posts and Telecommunications 24(6), 67-73.
16. Freeman, B. S., Taylor, G., Gharabaghi, B., Thé, J., 2018. Forecasting air quality time series using deep learning. Journal of the Air & Waste Management Association 68(8), 866-886.
17. Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural computation 9(8), 1735-1780.
18. Humphrey, G. B., Gibbs, M. S., Dandy, G. C., Maier, H. R., 2016. A hybrid approach to monthly streamflow forecasting: integrating hydrological model outputs into a Bayesian artificial neural network. Journal of hydrology 540, 623-640.
19. Jeong, J., Park, E., 2019. Comparative applications of data-driven models representing water table fluctuations. Journal of Hydrology 572, 261-273.
20. Kalteh, A. M., Hjorth, P., Berndtsson, R., 2008. Review of the self-organizing map (SOM) approach in water resources: Analysis, modelling and application. Environmental Modelling & Software 23(7), 835-845.

21. Kratzert, F., Klotz, D., Brenner, C., Schulz, K., Herrnegger, M., 2018. Rainfall–runoff modelling using long short-term memory (LSTM) networks. Hydrology and Earth System Sciences 22(11), 6005-6022.
22. Le, X. H., Ho, H. V., Lee, G., Jung, S., 2019. Application of Long Short-Term Memory (LSTM) Neural Network for Flood Forecasting. Water 11(7), 1387.
23. Liu, W., Wang, Z., Liu, X., Zeng, N., Liu, Y., Alsaadi, F. E., 2017. A survey of deep neural network architectures and their applications. Neurocomputing 234, 11-26.
24. Lohani, A. K., Goel, N. K., Bhatia, K. K. S., 2014. Improving real time flood forecasting using fuzzy inference system. Journal of hydrology 509, 25-41.
25. Malinowski, M., Rohrbach, M., Fritz, M., 2017. Ask your neurons: A deep learning approach to visual question answering. International Journal of Computer Vision 125(1-3), 110-135.
26. Marmanis, D., Schindler, K., Wegner, J. D., Galliani, S., Datcu, M., Stilla, U., 2018. Classification with an edge: improving semantic image segmentation with boundary detection. ISPRS Journal of Photogrammetry and Remote Sensing 135, 158-172.
27. Noori, N., Kalin, L., 2016. Coupling SWAT and ANN models for enhanced daily streamflow prediction. Journal of Hydrology 533, 141-151.
28. Nourani, V., Komasi, M., 2013. A geomorphology-based ANFIS model for multi-station modeling of rainfall–runoff process. Journal of Hydrology 490, 41-55.
29. Nourani, V., Baghanam, A. H., Adamowski, J., Kisi, O., 2014. Applications of hybrid wavelet–artificial intelligence models in hydrology: a review. Journal of Hydrology 514, 358-377.

30. Nourani, V., 2017. An emotional ANN (EANN) approach to modeling rainfall-runoff process. Journal of Hydrology 544, 267-277.

31. Nourani, V., Sattari, M. T., Molajou, A., 2017. Threshold-based hybrid data mining method for long-term maximum precipitation forecasting. Water Resources Management 31(9), 2645-2658.

32. Nourani, V., Elkiran, G., & Abdullahi, J., 2020. Multi-step ahead modeling of reference evapotranspiration using a multi-model approach. Journal of Hydrology 581, 124434.

33. Nourani, V., Molajou, A., Uzelaltinbulat, S., Sadikoglu, F., 2019. Emotional artificial neural networks (EANNs) for multi-step ahead prediction of monthly precipitation; case study: northern Cyprus. Theoretical and Applied Climatology, 1-16.

34. Nourani, V., Partoviyan, A., 2018. Hybrid denoising-jittering data pre-processing approach to enhance multi-step-ahead rainfall–runoff modeling. Stochastic environmental research and risk assessment 32(2), 545-562.Sainath, T. N., Kingsbury, B., Saon, G., Soltau, H., Mohamed, A. R., Dahl, G., Ramabhadran, B., 2015. Deep convolutional neural networks for large-scale speech tasks. Neural Networks 64, 39-48.

35. Sezen, C., Bezak, N., Bai, Y., Šraj, M., 2019. Hydrological modelling of karst catchment using lumped conceptual and data mining models. Journal of Hydrology 576. 98-110.

36. Shafaei, M., Adamowski, J., Fakheri-Fard, A., Dinpashoh, Y., Adamowski, K., 2016. A wavelet-SARIMA-ANN hybrid model for precipitation forecasting. Journal of Water and Land Development 28(1), 27-36.

37. Shenify, M., Danesh, A. S., Gocić, M., Taher, R. S., Wahab, A. W. A., Gani, A.,

Shamshirband S., Petković, D., 2016. Precipitation estimation using support vector machine with discrete wavelet transform. Water resources management 30(2), 641-652.

38. Shoaib, M., Shamseldin, A. Y., Khan, S., Khan, M. M., Khan, Z. M., Sultan, T., Melville, B. W., 2018. A comparative study of various hybrid wavelet feedforward neural network models for runoff forecasting. Water resources management 32(1), 83-103.

39. Sutskever, I., Vinyals, O., Le, Q. V., 2014. Sequence to sequence learning with neural networks. In Advances in neural information processing systems, 3104-3112.

40. Tan, Q. F., Lei, X. H., Wang, X., Wang, H., Wen, X., Ji, Y., Kang, A. Q., 2018. An adaptive middle and long-term runoff forecast model using EEMD-ANN hybrid approach. Journal of hydrology 567, 767-780.

41. Taormina, R., Chau, K. W., & Sivakumar, B., 2015. Neural network river forecasting through baseflow separation and binary-coded swarm optimization. Journal of Hydrology 529, 1788-1797.

42. Tsai, M. J., Abrahart, R. J., Mount, N. J., Chang, F. J., 2014. Including spatial distribution in a data- driven rainfall- runoff model to improve reservoir inflow forecasting in Taiwan. Hydrological Processes 28(3), 1055-1070.

43. Valipour, M., 2016. Optimization of neural networks for precipitation analysis in a humid region to detect drought and wet year alarms. Meteorological Applications 23(1), 91-100.

44. Venugopalan, S., Rohrbach, M., Donahue, J., Mooney, R., Darrell, T., Saenko, K., 2015. Sequence to sequence-video to text. In Proceedings of the IEEE international conference on computer vision, 4534-4542.

45. Wiseman, S., Rush, A. M., 2016. Sequence-to-sequence learning as beam-search optimization. arXiv 1606.02960v2.

46. Zhang, J., Zhu, Y., Zhang, X., Ye, M., Yang, J., 2018. Developing a Long Short-Term Memory (LSTM) based model for predicting water table depth in agricultural areas. Journal of Hydrology 561, 918-929.

47. Zhou, Y., Chang, F. J., Chang, L. C., Kao, I. F., Wang, Y. S., 2019. Explore a deep learning multi-output neural network for regional multi-step-ahead air quality forecasts. Journal of Cleaner Production 209, 134-145.

48. Zhu, L., Xu, Z., Yang, Y., Hauptmann, A. G., 2017. Uncovering the temporal context for video question answering. International Journal of Computer Vision 124(3), 409-421.

49. Zhu, Y., Zabaras, N., 2018. Bayesian deep convolutional encoder–decoder networks for surrogate modeling and uncertainty quantification. Journal of Computational Physics 366, 415-447.

*Credit Author Statement

Credit Author Statement

Conceptualization: I-Feng Kao, Li-Chiu Chang, Fi-John Chang;
Data curation: I-Feng Kao, Li-Chiu Chang;
Formal analysis: I-Feng Kao, Yanlai Zhou, Li-Chiu Chang, Fi-John Chang;
Funding acquisition: Fi-John Chang, Li-Chiu Chang;
Investigation: Fi-John Chang, Li-Chiu Chang;
Methodology: I-Feng Kao, Yanlai Zhou;
Project administration: Fi-John Chang;
Resources: I-Feng Kao;
Software: I-Feng Kao, Li-Chiu Chang;
Supervision: Fi-John Chang, Li-Chiu Chang;
Validation: I-Feng Kao, Yanlai Zhou;
Visualization: I-Feng Kao;
Roles/Writing - original draft: I-Feng Kao;
Writing - review & editing: Fi-John Chang

***Declaration of Interest Statement**

Declaration of interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: