JARQ 55 (1), 45-58 (2021) https://www.jircas.go.jp

Recurrent Neural Network Predictions for Water Levels at Drainage Pumping Stations in an Agricultural Lowland

Nobuaki KIMURA[1]*, Ikuo YOSHINAGA[1], Kenji SEKIJIMA[1], Issaku AZECHI[1], Hirohide KIRI[2] and Daichi BABA[3]

[1] Institute for Rural Engineering, National Agriculture & Food Research Organization, Tsukuba, Japan
[2] Agriculture, Forestry and Fisheries Research Council, Ministry of Agriculture, Forestry and Fisheries, Tokyo, Japan
[3] ARK Information Systems, Inc., Tokyo, Japan

Abstract

Drainage management in a complicated system in an agricultural lowland must operate pumps flexibly and quickly, based on the water level at the pumping station. A data-driven model without any physical-based information was implemented in a complicated drainage management system to predict the water level of a lagoon near a main drainage pumping station. We employed a long short-term memory (LSTM) model as an advanced neural network model to utilize the field datasets obtained from water-related facilities and sensors over about eight years as model input data. We performed sensitivity tests for model accuracy with different types of data and locations of data using cross-validation with an error quantity between observed and predicted water levels at the main drainage pumping station. The results showed that the LSTM model with the input of all available datasets predicted better than the models using several parts of datasets or it was roughly equivalent to those for water levels over the entire observed period in 3-h and 6-h lead times. In addition, the LSTM with only inputs of the water level and rainfall observed by drainage pumping stations performed better for the observed subperiod, including the severest flood event.

Discipline: Agricultural Engineering
Additional key words: complicated drainage management, *K*-fold cross-validation, multiple long short-term memory

Introduction

Drainage management in an agricultural lowland must facilitate the flexible and quick operation of pumps for the control of floods and droughts, which will be affected by climate change in the near future. Pumps are typically operated with on- or off-switching, depending on the water level of a water source near the pumping station. For flood control, pump operations have often been performed by skilled engineers who use their expertise to control the drainage management system. However, some negative factors recently reported in Japan could hinder effective and efficient drainage management. For example, skilled engineers may retire and the number of engineers could decrease due to high labor costs (e.g., Yamamoto et al. 2010). Artificial intelligence (AI) tools can be utilized to support these engineers or completely replace them with machines. In this study, an AI tool, an artificial neural network (ANN) model originally proposed by McCulloch & Pitts (1943), was developed. An ANN model was driven by a large amount of data without physical- and experimental-based parameters, unlike physical-based hydrological models. This study focused on the prediction of water level in a lagoon near a drainage pumping station using an ANN model.

There are several classes of ANNs, such as multilayer perceptron (MLP) and recurrent neural network (RNN). The MLP is a feedforward ANN, consisting of input, hidden, and output layers. Nodes on the layers are fully connected to the network with backpropagation for training (Rumelhart et al. 1986a).

*Corresponding author: kimuran590@affrc.go.jp
Received 21 January 2020; accepted 27 May 2020.

The RNN, which entails directed graph forms in sequence data, can be used to predict temporal dynamic behaviors (Hopfield 1982, Rumelhart et al. 1986b). An example of an advanced RNN is the long short-term memory (LSTM) network originally proposed by Hochreiter & Schmidhuber (1997). It can predict better on a temporal sequence or sequential letter recognition by holding long-term trends.

Previous studies employed an ANN model as a non-physical hydrological model to predict long-term water levels at a drainage pumping station in an agricultural lowland. For example, Kimura et al. (2018) reported that water-level and discharge predictions by an MLP model in a small lowland with a single drainage pumping station were in good agreement with referenced values produced by pseudo-rainfalls using a physical model calibrated to the small lowland for supplementing insufficient observed datasets. Another example is a comparison study between MLP and LSTM models in a complicated drainage management system in a mid-size lowland (Kimura et al. 2019). This study showed that the LSTM model predicted long-term water levels more accurately than the MLP model by several percent during the highest peak period. The comparison was made under a simplified condition of the drainage management system that only considered the main drainage pumping station in discharging water from the lowland to a river. However, an ANN model that considers multiple drainage pumping stations simultaneously has not been used in a real, complicated drainage management system.

LSTM models are currently well-accepted for time-series predictions. Several studies on water-level predictions have recently employed LSTM models with deep learning, particularly during riverine flood events (Yamada et al. 2018, Hu et al. 2018, Le et al. 2019). An LSTM inner structure is suitable for predicting time-series data as its memory function retains past sequential patterns (Hochreiter & Schmidhuber 1997, Gers et al. 2000). Therefore, the LSTM model was also employed in this study to predict water levels. This LSTM structure was extended with multiple hidden layers from a conventional hidden-layer structure of the LSTM described by Kimura et. al. (2019).

The purpose of this study was to evaluate the sensitivity of an LSTM model with multiple hidden layers that was implemented in a complicated drainage management system in an agricultural lowland to assist drainage management with long-term water-level predictions by selecting available field datasets. In this paper, the "Materials and methods" section describes the field site, LSTM structure, and setup of simulation cases. The "Result" section describes the LSTM predictions of water levels, and the "Discussion" section explains the evaluation of LSTM sensitivity among different input datasets and its accuracy compared with other studies. Then in summary, the conclusions of this study and proposed future work are presented.

Materials and methods

1. Field site

The target lowland (called Kamedagou) is located in the city of Niigata in the northern part of central Japan (37.9225°N, 139.0433°E). The Kamedagou basin covers an area of 99.8 km^2 and its northern side is near the Sea of Japan. It is bounded by two main rivers (Shinano and Agano) on its western and eastern sides, respectively, and by a tributary of the Agano River to the south. Located in the northwest part of the basin. Lagoon Toyanogata (1.58 km^2) is used to control the entire water volume of the area. Kamedagou has five drainage pumping stations and four main irrigation pumping stations (Table 1). Two of the drainage pumping stations (Oyamatsu and Toyanogata) are directly connected to the lagoon with a wide canal without slopes. Oyamatsu constantly operates to maintain a water level of approximately 2.5 m in the lagoon. The other drainage pumping stations potentially operate from April to October during heavy rainfall events. These stations only employ artificial drainage with pumps. The irrigation pumping stations operate from May to August for rice-paddy cultivation. In addition, there are six gates on the drainage canals (Honsho, downstream-Yokogoshi, mid-Yokogoshi, upstream-Yokogoshi, Kameda, and Oishi) and six stand-alone sensors for water level (Ohori, Kiyogoro, Yamasaki, Kameda, Yokogoshi, and Oishi). Each pumping station provides the datasets for water level, discharge that the pumps drain or draw, and rainfall. Each gate provides the water levels upstream and downstream from the gates. Each stand-alone sensor only provides the water level in one canal. Figure 1 shows a map of Kamedagou.

2. LSTM model

LSTM is a class of RNNs; it was created to train datasets with long-term trends and solve problems in vanishing and exploding gradients by introducing a memory cell that holds past trends. We explain the LSTM structure based on some past studies (Fischer & Krauss 2018, Hu et al. 2018, Sugomori 2017). The bold variables show the vectors in the following description. As shown in Figure 2 (a), input data that consists of multiple nodes at present time t are mixed with output data (e.g., water level) $\boldsymbol{h}_{t-1}$ that consists of multiple nodes at the previous

time step t-1. Note that a node roughly models a neuron in the brain. The input data move to three gates: forget, input, and output. The following equation defines the role of forget gate $\boldsymbol{f}_t$ that removes some information from the memory cell.

$$\boldsymbol{f}_t = \sigma(w_{f,h}\boldsymbol{h}_{t-1} + w_{f,x}\boldsymbol{x}_t + \boldsymbol{b}_f). \quad (1)$$

where $w_{f,h}$ and $w_{f,x}$ are the weighted coefficients in matrix form related to $\boldsymbol{x}_t$ and $\boldsymbol{h}_{t-1}$, $\boldsymbol{b}_f$ is a bias, and σ is a sigmoid as an activation function. The input gate mixes two types of information sources obtained from the feature quantities of the input data with two different activation functions. Equations (2) and (3) express the two sources. The input gate adds the mixed information to the memory cell.

$$\boldsymbol{i}_t = \sigma(w_{i,h}\boldsymbol{h}_{t-1} + w_{i,x}\boldsymbol{x}_t + \boldsymbol{b}_i), \quad (2)$$

$$\boldsymbol{z}_t = \tanh(w_{z,h}\boldsymbol{h}_{t-1} + w_{z,x}\boldsymbol{x}_t + \boldsymbol{b}_z). \quad (3)$$

where tanh is a hyperbolic tangent as an activation function, $w_{i,h}$, $w_{z,h}$, $w_{i,x}$ and $w_{z,x}$ are the weighted coefficients in matrix form, and $\boldsymbol{b}_i$ and $\boldsymbol{b}_z$ are biases. The variables $\boldsymbol{f}_t$, $\boldsymbol{i}_t$, and $\boldsymbol{z}_t$ are combined into the following equation with the state of a past trend at t-1 ($\boldsymbol{C}_{t-1}$), which the memory cell holds.

$$\boldsymbol{C}_t = \boldsymbol{f}_t \otimes \boldsymbol{C}_{t-1} \oplus \boldsymbol{i}_t \otimes \boldsymbol{z}_t, \quad (4)$$

where ⊗ and ⊕ are the multiplication and addition at each component between two vectors, whose symbols simply follow those in Figure 2 (a). The following equation defines output gate $\boldsymbol{o}_t$ that still involves the original feature quantities of the input data.

$$\boldsymbol{o}_t = \sigma(w_{o,h}\boldsymbol{h}_{t-1} + w_{o,x}\boldsymbol{x}_t + \boldsymbol{b}_o), \quad (5)$$

Where $w_{o,h}$ and $w_{o,x}$ are the weighted coefficients in matrix form, $\boldsymbol{b}_o$ is a bias. Finally, $\boldsymbol{C}_t$ from the memory cell is multiplied by $\boldsymbol{o}_t$ and updates the output ($\boldsymbol{h}_t$).

$$\boldsymbol{h}_t = \boldsymbol{o}_t \otimes \tanh(\boldsymbol{C}_t). \quad (6)$$

The network structure of the LSTM model has three layers: input, hidden, and output. The hidden layer can be extended to several layers of LSTM. We followed the Keras Documentation (https://keras.io/getting-started/sequential-model-guide) for the sequential model treatment of input and output. Figure 2 (b) shows the LSTM network structure, when variables A and B (e.g., water level and rainfall) are used as input data. Variables C and D are predicted at the forward time steps (t+1, t+2, …, t+q, where q = lead time) in the output layer of the network. Note that lead time is defined as the time when the model predicts. The training of the model uses the input data from past p h to 0 h and the output data of 1–q h lead times as a set. The model inner parameters, corresponding to the number of nodes, are tuned to obtain more optimal solutions of 1–q h lead times simultaneously. It suggests that the longer the lead time of q, the more difficult the accuracy of output. The model prediction uses the p–0 h input data and then outputs 1–q h lead times at the same time through a fully-connected layer after LSTM output.

3. LSTM simulation design

This study focused on the temporal prediction of

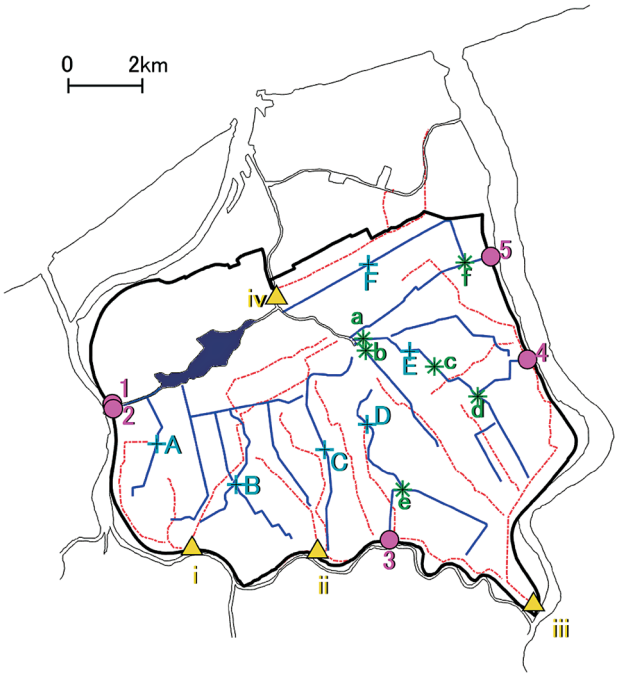

Fig. 1. Map of the field site
The enclosed deep blue area is a lagoon. Blue lines and red dotted lines indicate drainage canals and irrigation canals, respectively, and marks show the following locations: ● drainage pumping stations (1: Oyamatsu, 2: Toyanogata, 3: Nihongi, 4: Kuraoka, 5: Honsho); ▲irrigation pumping stations (i: Maigata, ii: Ryousen, iii: Soumi, iv: Takeo); * gates (a: Honsho, b: downstream-Yokogoshi, c: mid-Yokogoshi, d: upstream-Yokogoshi, e: Kameda, f: Oishi); + sensors (A: Ohori, B: Kiyogoro, C: Yamasaki, D: Kameda, E: Yokogoshi, F: Oishi). The enclosed bold line indicates the boundary of the Kamedagou basin. These locations are referenced from the homepage of the Kamedagou Land Improvement District (http://www.kamedagou.jp/yousui/index.html, in Japanese). Each number on the map refers to Tables 1 and 2.

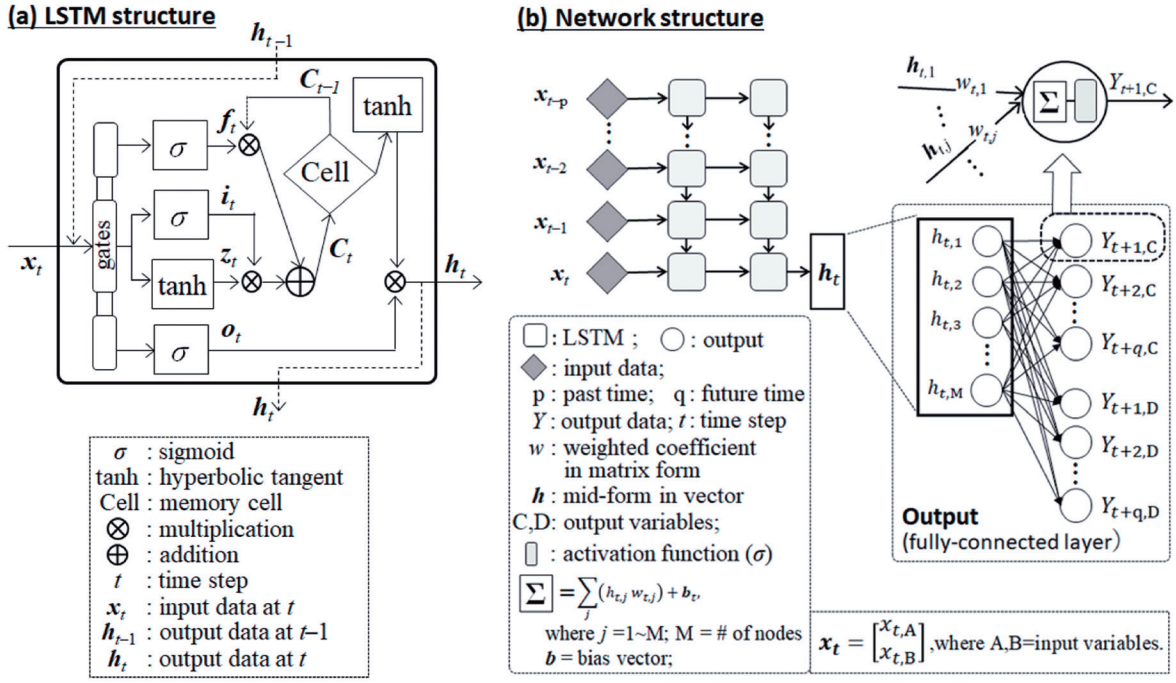

Fig. 2. LSTM and network structures

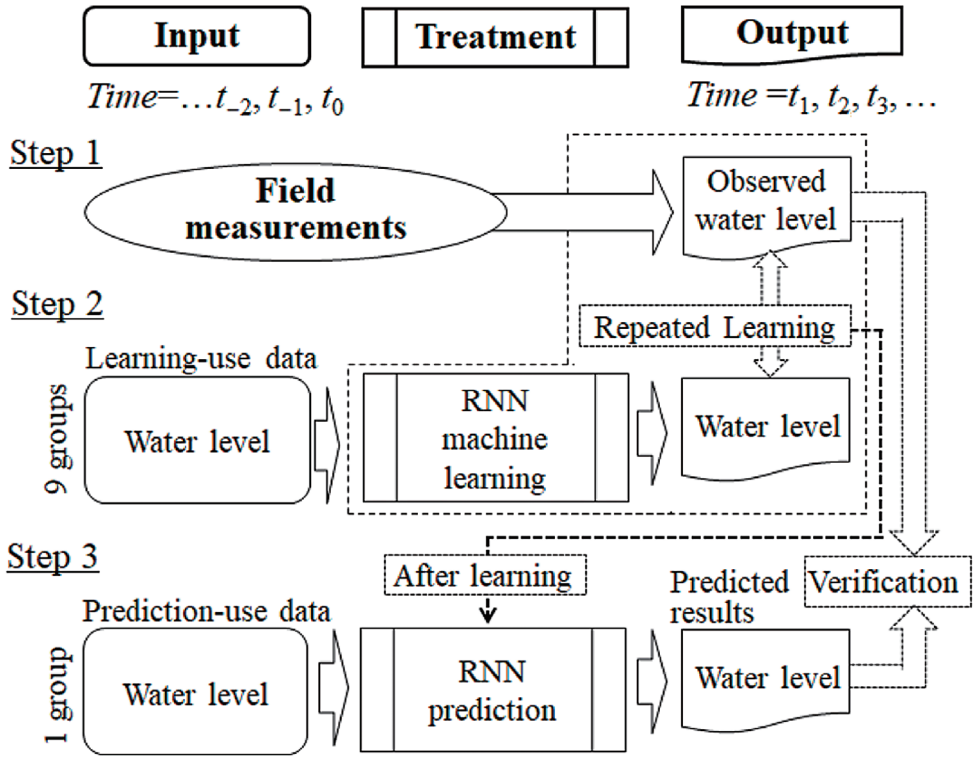

Fig. 3. Data flow for LSTM training and prediction

T_0 is present time, T_{-2}, T_{-1}… are past times, and T_1, T_2… are future times.

*Temporally predicted data indicate primary predictors from the RNN model in a learning process.

Table 1. Characteristics of drainage and irrigation pumping stations

No.	Name	Maximum flow rate (m^3/s)	Number of pumps
Drainage pumping station			
1	Oyamatsu	60.0	4
2	Toyanogata	40.0	Unknown
3	Nihongi	6.8	3
4	Kuraoka	12.8	3
5	Honsho	16.2	3
Irrigation pumping station			
i	Maigata	5.9	3
ii	Ryousen	4.2	2
iii	Soumi	8.4	2
iv	Takeo	2.2	2

water level at a lagoon connected by a canal without slopes to Oyamatsu, which operates continuously based on the water level. The water level at Oyamatsu was set up as a model output. The data on drainage and irrigation pumping stations, gates, and sensors were used for the sensitivity tests as model input. The basic case (Case 1) entailed the use of all input data. The same setup as that in Case 1 was also used for preliminary tests that determined appropriate hyperparameters such as the number of hidden layers and an optimizer, and other functions such as lead time and activation functions. The other cases were utilized as the sensitivity tests of input datasets, which were available in the field measurements. Cases 2 and 3 constituted the test that investigated the roles of facilities that control water volume in the lowland. Cases 4-6 tested the impact of the types of observed data (e.g., water level, rainfall) as related only to the drainage pumping stations. Cases 7-9 entailed a test similar to that of Cases 4-6, but only considered Oyamatsu, which always operates to ensure normal water-level control as well as flood control. The model was run for the nine cases listed in Table 2.

4. Data acquisition, model validation, and data flow

The observed data at all pumping stations, gates, and sensor locations were obtained at Kamedagou from March 2010 to November 2017. The data collected at each pumping station were pump discharge (hereafter "discharge"), rainfall, and water level. Upstream and downstream water levels were measured at each gate. The water level was measured at each sensor location. Data from the Toyanogata drainage pumping station were collected and preserved by Japan's Ministry of Land, Infrastructure, Transport, and Tourism (MLIT). The Kamedagou Land Improvement District, a local social association for land development, obtained the other data in cooperation with Niigata Prefecture and Japan's Ministry of Agriculture, Forestry and Fisheries (MAFF). These data were used to train the LSTM model and evaluate the errors of model predictions. *K*-fold cross-validation (Geisser 1993) was employed for the validation comparison. The number K was set to 10, and the observed data were separated into ten groups. Nine groups were used for training the model and one group was used for comparison with the model prediction. The one group for model prediction was exchanged with the other groups in order. Note that chronological-order periods during daily and flood pump operations are much shorter than a period of each group, although a nested cross-validation (Varma & Simon 2006) is typically used to capture progressive trends of time-series data. A quantitative error between observation and prediction was defined by root mean square error (RMSE). For a comparison among all cases listed in Table 2, a mean RMSE that is averaged over the RMSEs from ten time tests in each group was adopted. The second group (Group 2) of the ten groups was also considered for RMSE evaluation because it involves the highest flood peak during the observed period.

Observed data were obtained from field measurements in the first step. Parts of the observed data were used for the *K*-fold cross-validation in a learning process in the second step. In the third step, the remaining observed data were used for prediction. Figure 3 illustrates the data flow for the aforementioned procedures in LSTM learning and prediction. The program for the LSTM model was created using Python (version 3.6.4, http://www.python.org) incorporated with the Python deep learning libraries in Keras (http://keras.io/) that include the Tensorflow module on a Windows-OS PC with an Intel Core i7-4770K CPU at 3.50 GHz. The computational time averaged for all cases was approximately 1 h. Table 3 details the setups of several hyperparameters, such as batch size, epoch number, and activation functions.

Results

The field data in Kamedagou were obtained from March 2010 to November 2017. The data were screened by simply removing errors and interpolating between missing data. Figure 4 shows the water-level and rainfall data with the ten groups at the major drainage pumping station (Oyamatsu) for about eight years. These data indicate that several severe flood events occurred due to heavy rains. The largest flood event appears in Group 2 (Fig. 4 (b)). First, preliminary tests for the sensitivities of

Table 2. Simulation cases

No.		Case 1	Case 2	Case 3	Case 4	Case 5	Case 6	Case 7	Case 8	Case 9
Drainage pumping stations										
1	Water level	○	○	○	○	○	○	○	○	○
	Rainfall	○	○	○	○	○		○	○	
	Discharge	○	○	○	○			○		
2	Water level	○	○	○	○	○	○			
	Rainfall	○	○	○	○	○				
	Discharge	○	○	○	○					
3	Water level	○	○	○	○	○	○			
	Rainfall	○	○	○	○	○				
	Discharge	○	○	○	○					
4	Water level	○	○	○	○	○	○			
	Rainfall	○	○	○	○	○				
	Discharge	○	○	○	○					
5	Water level	○	○	○	○	○	○			
	Rainfall	○	○	○	○	○				
	Discharge	○	○	○	○					
Irrigation pumping stations										
i	Water level	○	○	○						
	Rainfall	○	○	○						
	Discharge	○	○	○						
ii	Water level	○	○	○						
	Rainfall	○	○	○						
	Discharge	○	○	○						
iii	Water level	○	○	○						
	Rainfall	○	○	○						
	Discharge	○	○	○						
iv	Water level	○	○	○						
	Rainfall	○	○	○						
	Discharge	○	○	○						
Gates for water level										
a	Upstream	○	○							
	Downstream	○	○							
b	Upstream	○	○							
	Downstream	○	○							
c	Upstream	○	○							
	Downstream	○	○							
d	Upstream	○	○							
	Downstream	○	○							
e	Upstream	○	○							
	Downstream	○	○							
f	Upstream	○	○							
	Downstream	○	○							
Sensors for water level										
A		○								
B		○								
C		○								
D		○								
E		○								
F		○								

Table 3. Setups of LSTM hyperparameters and other functions

Hyperparameters	Values/function	Remarks
Number of hidden layers	2	
Number of vector dimensions in LSTM inner parameters (nodes)	20	e.g., C_t in Fig. 2a
Past and present time in input	−6 to 0	Time interval = h
Lead time in output	1 to 6	Same as above
Batch size	100	
Number of epochs	100	
Learning rate	0.01	Only for SGD
Dropout rate	0.0	
Reproducibility	None	
Optimizer	Stochastic gradient descent (SGD)	
Activation function	Sigmoid or hyperbolic tangent	
Loss function	Sum of squared residuals $=\frac{1}{2}\sum_{i=1}^{N1}(V_{ci}-V_{oi})^2$	c_i=model prediction, o_i=observed data, $N1$=the number of data
Error evaluation	Root mean square error (RMSE) $=\sqrt{\frac{1}{N1}\sum_{i=1}^{N1}(V_{ci}-V_{oi})^2}$	Same as above
	Nash-Sutcliffe coefficient (NS) $1-\sum_{i=1}^{N1}(V_{ci}-V_{oi})^2 / \sum_{i=1}^{N1}(V_{oi}-\langle V_o\rangle)^2$	Same as above, and <*> = average

the hyperparameters related to the LSTM network structure and calculational conditions were performed. From the preliminary results, the following hyperparameters were chosen for a more appropriate setup. The number of hidden layers in the network structure was set to two, which indicates the adoption of multiple hidden layers of LSTM. Time-series data in input were set up with the past 6 h and the present. Table 3 lists the other model setups including the hyperparameters selected in this study. Appendix I presents the major results of the preliminary tests.

This study focused on the sensitivity test for input datasets that consist of water level, discharge, and rainfall obtained from different locations in the LSTM model, which was implemented in a complicated drainage management system. We ran nine cases and compared their model predictions with the observed data by the mean of RMSEs among the ten groups (M-RMSE). Because the RMSE of Group 2 (G2-RMSE) potentially became worse than those of the other groups in the preliminary tests, the G2-RMSE and M-RMSE were chosen for the evaluation of model accuracy. The response time of water level to rainfall is typically 3-4 h; however, the drainage management system in the target lowland keeps monitoring the water level of the lagoon with up to 6 h of lead time (MAFF Rural Development Bureau). Therefore, G2-RMSE and M-RMSE were observed with 1, 3, and 6 h of lead time.

For Case 1, the predicted water levels for the lead times of 1, 3 and 6 h at the target station (Oyamatsu) showed M-RMSEs of 0.023, 0.034, and 0.046 m, respectively. When a relative M-RMSE is defined as M-RMSE divided by the variation between the maximum and minimum water levels during the observed period, the relative M-RMSEs of Case 1 were 1.81%, 2.70%, and 3.72% for lead times of 1, 3, and 6 h, respectively. The Nash-Sutcliffe coefficients (NSs) in Table 3 were 0.82, 0.62, and 0.26 for 1-, 3-, and 6-h lead times. It indicates that the predictions of Case 1 were good for 1- and 3-h lead times, but not for the 6-h lead time. In addition, the RMSEs of Group 2 were 0.03, 0.041, and 0.054 m, respectively, corresponding to the relative RMSEs of 2.58%, 3.57%, and 4.66% with respect to the variation between the maximum and minimum water levels during the Group 2 subperiod (Table 4). The NSs of Group 2 were 0.79, 0.59, and 0.30 for 1-, 3-, and 6-h lead times, respectively, which are similar to the mean NSs of all groups. Due to the poorer prediction accuracy among the ten groups, the water levels of Group 2 are drawn in Figure 5 with zoomed panels that indicate a typically operated period and a flood period. The prediction of 6-h lead time cannot determine the observations on typical

water-level control and flood control from the lines in the small panels. In particular, quick increases in water level at the starting point are shifted backward on a flood wave because a 6-h time lag exists. This poor prediction was caused by the LSTM model feature in which the model must obtain optimal solutions from 1-6 h simultaneously in the training. Given a longer time lag, the model cannot forecast a rising part for the waveform increase until a symptom of the increase appears.

Table 4 presents the RMSEs of the other cases. The M-RMSEs in 1-, 3-, and 6-h lead times are approximately 0.022, 0.035, and 0.047 m, respectively. The ranges of G2-RMSEs are more fragmented than those of M-RMSEs. For a quantitative comparison of the cases, the differences between the RMSE of Case 1 and those of the other cases were plotted (Fig. 6). The aforementioned difference is defined using the following equation.

$$Difference = 100 \times (RMSE_{Other\ case} - RMSE_{Case1}) / RMSE_{Case1}, \quad (7)$$

where RMSE is replaced by M-RMSE or G2-RMSE. The M-RMSEs of Cases 2-9 in a 1-h lead time are slightly improved, but those in 3- and 6-h lead times are worse than the M-RMSE of Case 1. In particular, Cases 8-9, only using the data at Oyamatsu, provide much worse errors. The G2-RMSE of Case 1 in 1-h lead time is more than 6% worse than those of the other cases, except for Case 3. Most G2-RMSEs in the 3- and 6-h lead times of Cases 2-9, excluding Cases 5 and 7, are more than 2% worse than that of Case 1. Case 5 has better output than the other cases in 1-, 3-, and 6-h lead times in Group 2. Figure 7 illustrates the predicted water levels of Case 5 in Group 2. The small panels in the typically operated and flood periods indicate that the water levels of Case 5 are greatly improved with better fits to observed data after wave peaks.

Discussion

The prediction accuracy of the LSTM model was evaluated among nine cases using RMSEs. A better prediction was the output of Case 1 that used all datasets from all observed locations for a comparison of M-RMSEs, for the output of 3-h and 6-h lead times (except the 1-h lead time at Oyamatsu) as input data. The worst result of Case 1 in 1-h lead time was potentially caused by time lags of water movement between the predicted location and other observed locations.

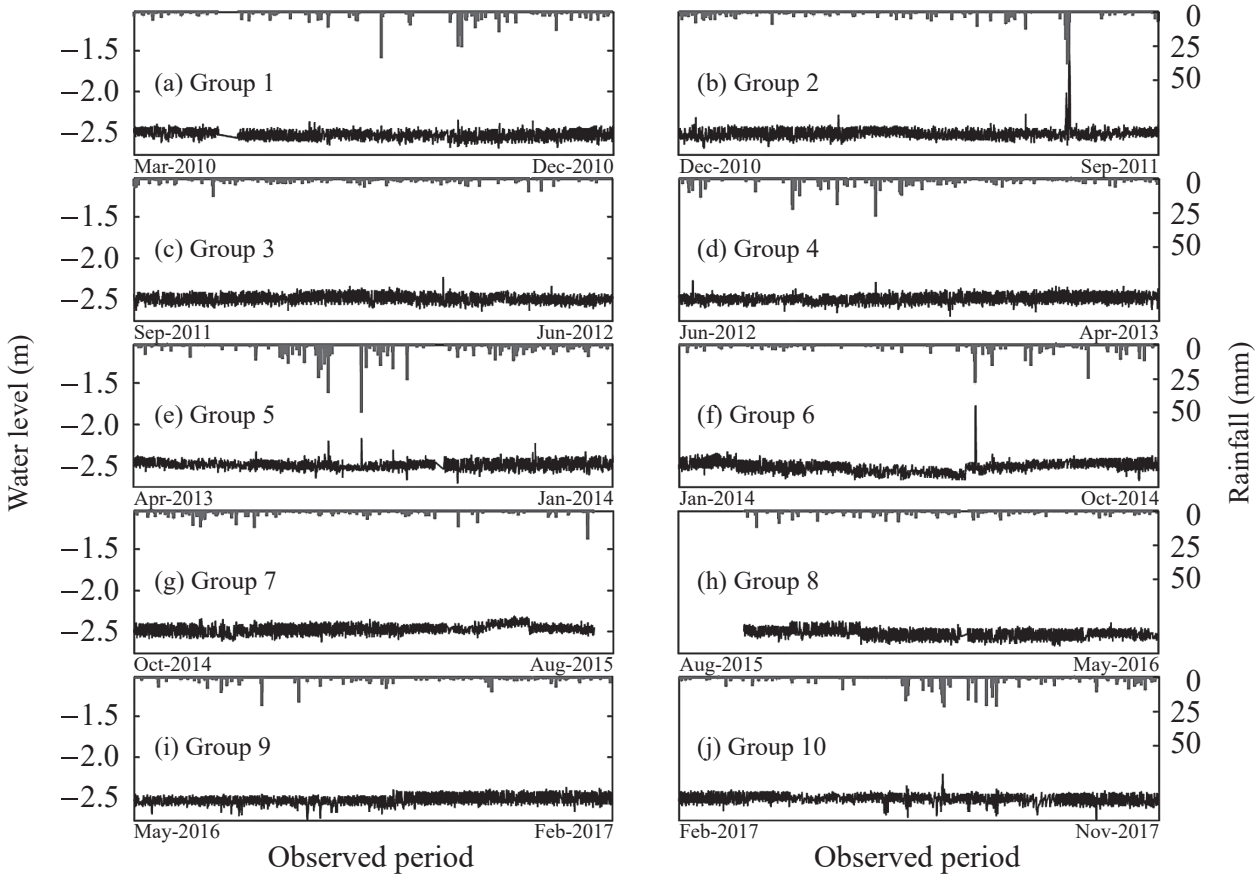

Fig. 4. Water level and rainfall observed at Oyamatsu during each subperiod

This implies that the model in Case 1 inappropriately trained the time lags due to poor determination of numerous relationships among datasets. The outputs of longer lead times suggest that a complicated drainage management system similar to the one in this study may require numerous datasets that cover information related to the water volume of the lowland as much as possible. As in the comparison of M-RMSEs, Case 1 was worse in 1-h lead time and better in longer lead times for G2-RMSEs, except for Case 5. The model in Case 5 that considered only drainage pumping stations excluding the discharge data made the best predictions for all lead times. The reason is that the predicted water level at Oyamatsu is mainly affected by the drainage pumping stations that transfer the water volume from the area to the rivers during a flood event, and the severest flood

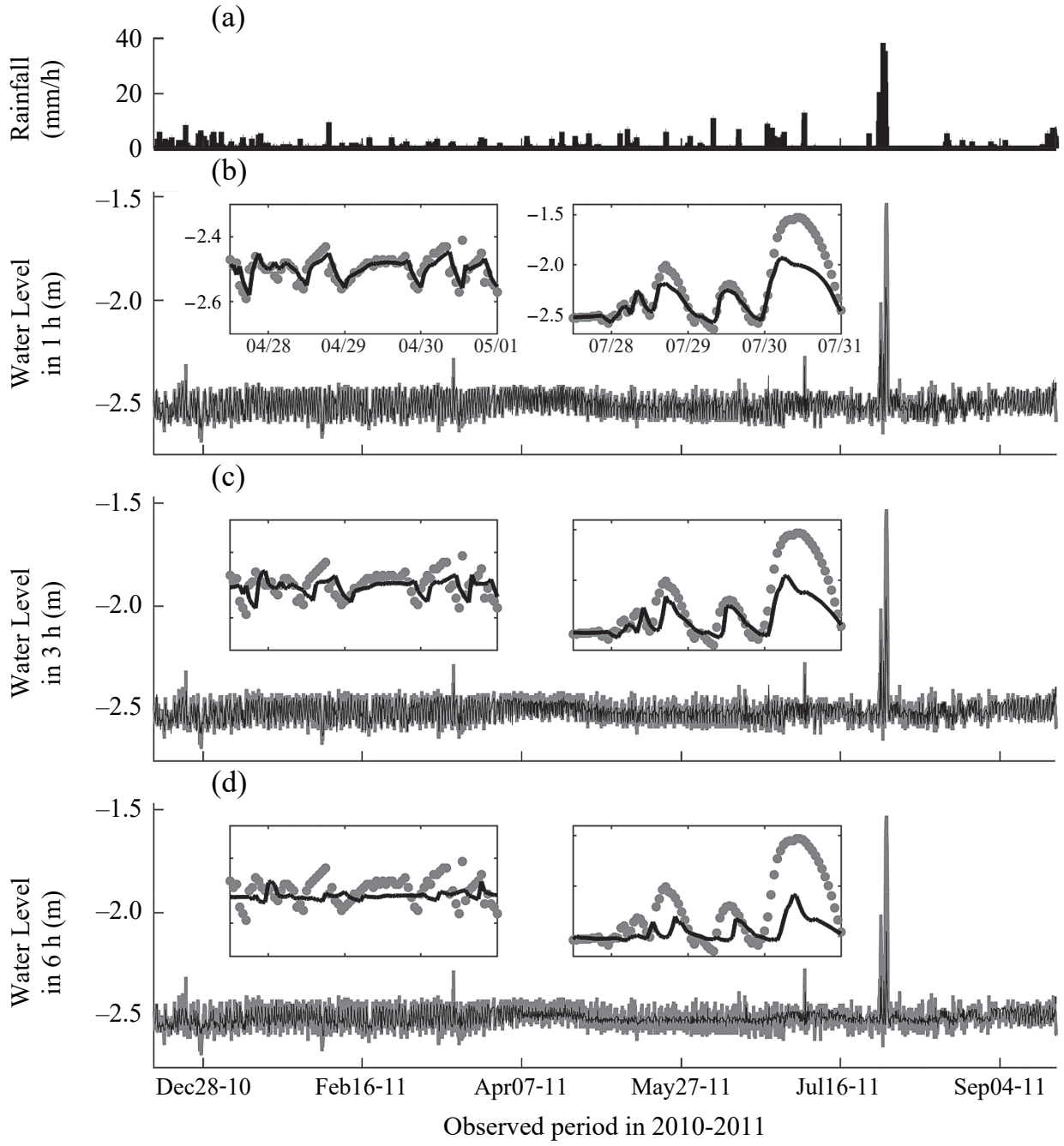

Fig. 5. Predicted water levels of Case 1 compared with observed data at Oyamatsu in Group 2 with lead times of 1, 3, and 6 h

Gray and black lines indicate observed data and predicted data of water level, respectively. The two small panels show a typically operated period and a flood period, respectively.

Table 4. Prediction accuracy among cases in lead time

Case name	M-RMSE (m)			G2-RMSE (m)			Remarks
	1 h[a]	3 h[a]	6 h[a]	1 h[a]	3 h[a]	6 h[a]	
Case 1	0.0225	0.0335	0.0461	0.0299	0.0414	0.0541	All data from fields
Case 2	0.0218	0.0338	0.0469	0.0281	0.0417	0.0565	All data from facilities
Case 3	0.0219	0.0336	0.0460	0.0325	0.0438	0.0555	Data from only stations
Case 4	0.0215	0.0335	0.0462	0.0294	0.0418	0.0559	Only drainage consideration
Case 5	0.0224	0.0357	0.0477	0.0275	0.0406	0.0538	Same as above
Case 6	0.0222	0.0355	0.0476	0.0281	0.0437	0.0585	Same as above
Case 7	0.0210	0.0338	0.0472	0.0264	0.0406	0.0566	Only Oyamatsu consideration
Case 8	0.0222	0.0367	0.0499	0.0278	0.0432	0.0579	Same as above
Case 9	0.0222	0.0367	0.0496	0.0280	0.0434	0.0576	Same as above

[a] 1, 3, and 6 h indicate lead times.

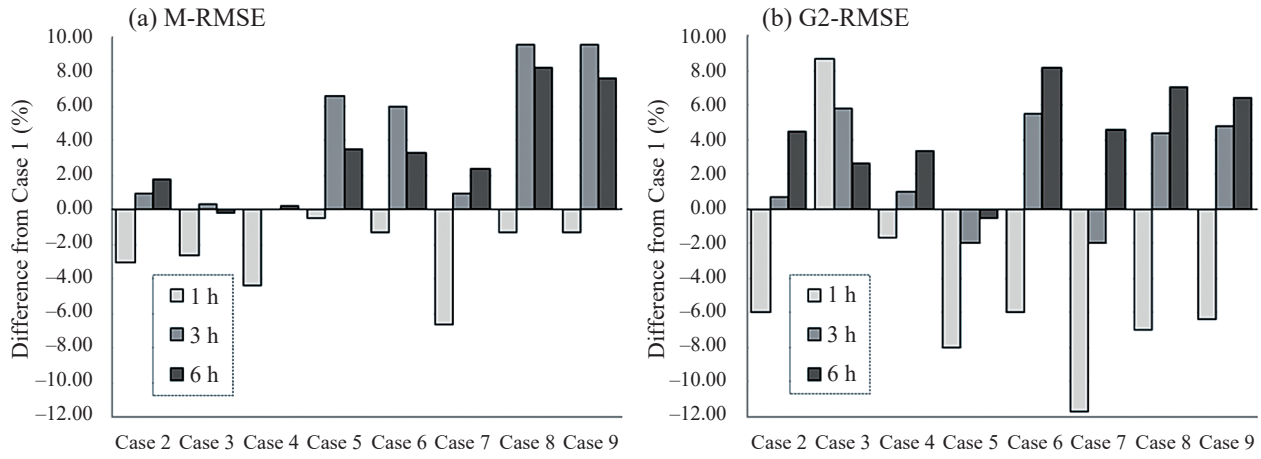

Fig. 6. RMSE differences between Case 1 and the other cases
The negative values indicate the reduction in RMSE compared to Case 1.

event is a major characteristic of data patterns in Group 2. However, the data on discharge excluded in Case 5 were not countable, although discharge is a main factor when performing flood control. The Group 2 result of Case 5 shows that water level and rainfall for input data were the major factors of better prediction for a longer lead time when compared with those of Cases 4 and 6 (Fig. 6 (b)).

This study with an ANN-based model is the first, to the authors' knowledge, to perform sensitivity tests related to input datasets from the field in a complicated drainage management system in a lowland setting. However, some studies have reported long-term predictions of water level, including riverine flood events. Yamada et al. (2018) reported that the LSTM model performed well for long-term prediction of water levels, including multiple riverine flood events. Their model setups were similar to those in the present study. For example, 95% of 18 years was used for training and the rest was used for prediction. A difference in the setup is that the targeted water level was a river in a watershed that ranges from mountain highland to 0 m lowland. However, the targeted water level in our study was a lagoon in a lowland under a 0 m area. The RMSE for their model accuracy was 0.18 m in a 6-h lead time at the downstream location. The M-RMSEs in the present study had a similar order based on the evaluation of model accuracy, although their RMSE was approximately 40% of our M-RMSEs.

Less accurate predictions in this study may have occurred for the following reasons, however. The observed water levels were affected by pump operations based on human judgment during flood events and by inhomogeneous time delays due to inertial force when switching the pumps on or off. These observed data might contain fewer periodic patterns than the number the LSTM model needs to learn. Another reason could be the model setup by Yamada et al. (2018). The data obtained along a main river can be beneficial to train temporal variations of water level, including flood events.

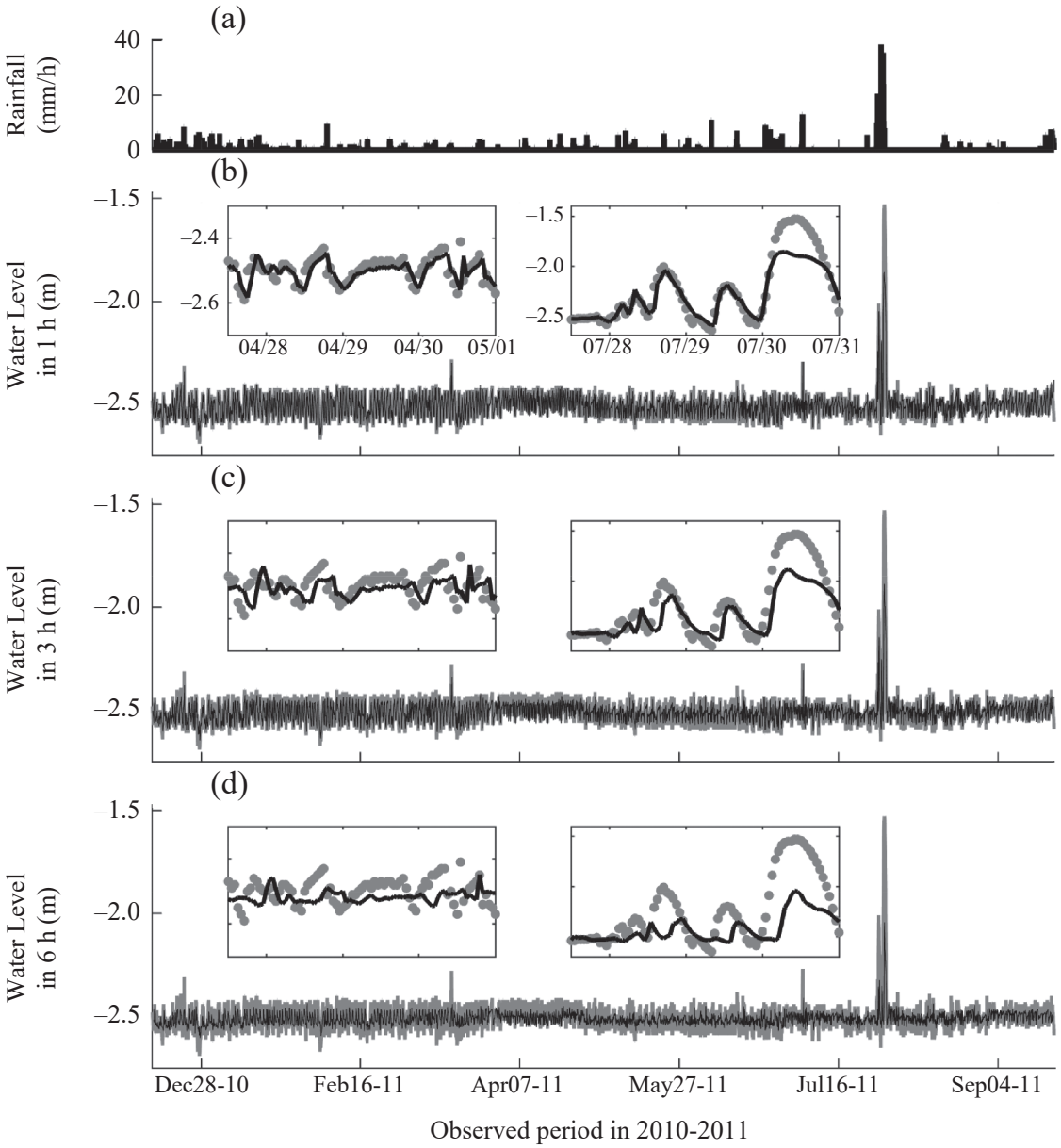

Fig. 7. Predicted water levels of Case 5 compared with observed data at Oyamatsu in Group 2 with lead times of 1, 3, and 6 h
Lines and small panels indicate the same as in Fig. 5.

This is because the data pattern of the upstream river must be similar to that of the predicted location downstream. As a result, the data pattern upstream strongly reflects the pattern at the predicted location, although a certain time lag exists. Water levels at observed upstream locations can enhance a similar water level in a learning process. Therefore, it is possible that their model accuracy was much better than that in the present study.

Conclusion

In this study, sensitivity tests of the LSTM model implemented in a complicated drainage management system in a lowland area were performed to predict long-term water levels in a lagoon near a main station where the water volume is always controlled. Nine case predictions based on the type and location of data were conducted using datasets obtained from the sensors and facilities that control water movement and volume in the area. The findings related to model accuracy using M-RMSE and G2-RMSE are as follows:

1. The error evaluation for the entire simulation period (i.e., Groups 1-10) revealed that Case 1 containing all datasets from the field had relatively better or roughly equivalent accuracy for the water level in 3- and 6-h

lead times when compared with the other cases, but provided poorer accuracy in a 1-h lead time than the other cases.

2. The error evaluation for the subperiod simulation including the severest flood event (i.e., Group 2) showed that Case 5, containing the dataset of water levels and rainfalls only from the drainage pumping stations, provided the most accurate prediction of water levels in 1-, 3-, and 6-h lead times.

   For the prediction of long-term water levels in a longer lead time, it is reasonable that the use of all datasets over the field (Case 1) provided better performance due to the available information that may include the factors of time lags. For the predicted water level in the subperiod that includes the severest flood event, the result of Group 2 in Case 5 was better. However, the physical reason why is unknown due to a data-driven model. Therefore, in future work, we need to study the implications of physical mechanisms from the datasets by using methods that evaluate major factors based on the contribution to the model's predictions, such as Local Interpretable Model-agnostic Explanations (LIME, Ribeiro et al. 2016) and SHapley Additive exPlanations (SHAP, Lundberg & Lee 2017).

Acknowledgments

This research was supported by grants from the project of the NARO Bio-oriented Technology Research Advancement Institution (as part of a research program on the development of innovative technology). We greatly appreciate the Kamedagou Land Improvement District, Niigata Prefecture, MAFF, and MLIT for the acquisition of observed data.

References

Dozat, T. (2016) Incorporating Nesterov momentum into Adam. *In* proceedings of the International Conference on Learning Representations (ICLR) 2016.

Duchi, J. et al. (2011). Adaptive subgradient methods for online learning and stochastic optimization. *J Mach. Learn. Res.*, **12**, 2121-2159.

Fischer, T. & Krauss, C. (2018) Deep learning with long short-term memory networks for financial market predictions. *Eur. J. Oper. Res.*, **270**, 654-669.

Geisser, S. (1993) *Predictive inference: An introduction*, *Monographs on statistics and applied probability 55*. Chapman and Hall, NY, USA.

Gers, F. A. et al. (2000) Learning to forget: continual prediction with LSTM. *Neural Comput.*, **12**, 2451-2471. DOI: https://doi.org/ 10.1162/089976600300015015.

Hinton, G. *Overview of mini-batch gradient descent*. https://www.cs.toronto.edu/~tijmen/csc321/slides/lecture_slides_lec6.pdf (January 2020).

Hochreiter, P. & Schmidhuber, J. (1997) Long short-term memory, *Neural Comput.*, **9**, 1735-1780. DOI: https://doi.org/10.1162/neco.1997.9.8.1735.

Hopfield, J. J. (1982) Neural networks and physical systems with emergent collective computational abilities. *Proc. Natl. Acad. Sci.*, **79**, 2554-2558. DOI: https://doi.org/10.1073/pnas.79.8.2554.

Hu, C. et al. (2018) Deep learning with a long short-term memory networks approach for rainfall-runoff simulation. *Water*, **10**, 1543. DOI: https://doi.org/10.3390/w10111543.

Kimura, N. et al. (2018) Prediction on water levels in a wet pond for a drainage system using an artificial neural network model. *Bull. NARO Rural Engr.*, **3**, 71-80 [In Japanese with English summary].

Kimura, N. et al. (2019) Water level prediction at drainage pump station in low lands using LSTM model. *J. Jpn Soc. Civil Eng. (JSCE) Ser. B1 (hydraul. engr.)*, **75**, I_235-I_240 [In Japanese with English summary].

Kingma, D. P. & Ba, J. L. (2015) Adam: a method for stochastic optimization. *In* Proceeding of the International Conference on Learning Representations (ICLR), San Diego, USA, 1-13.

Le, X. et al. (2019) Application of long short-term memory (LSTM) neural network for flood forecasting. *Water*, **11**, 1387. DOI: https://doi.org/10.3390/w11071387.

Lundberg, S. M. & Lee, S.-I. (2017) A unified approach to interpreting model predictions. *In* Proceeding of the 31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

McCulloch, W. S. & Pitts, W. (1943) A logical calculus of the ideas immanent in nervous activity. *Bull. Math. Biophys.*, **5**, 115-133.

Ribeiro, M. T. et al. (2016) Why should I trust you? Explaining the predictions of any classifier. *In* Proceedings of the 22nd International Conference on the ACM Special Interest Group on Knowledge Discovery and Data mining (SIGKDD), San Francisco, CA, USA, 1135-1144. DOI: https://doi.org/10.1145/2939672.2939778.

Robbins, H. & Monro, S. (1951) A stochastic approximation method. *Ann. Math. Stat.*, **22**, 400-407. doi:10.1214/aoms/1177729586.

Rumelhart, D. E. et al. (1986a) Learning internal representations by error propagation. *In* Parallel distributed processing, eds. Rumelhart, D. E. & McClelland, J. L., MIT Press, USA, pp. 318-362.

Rumelhart, D. E. et al. (1986b) Learning representations by back-propagating errors. *Nature*, **323**, 533-536. DOI: https://doi.org/10.1038/323533a0.

Rural Development Bureau, MAFF, Japan. *Preparedness and countermeasures against heavy rain -examples for drainage pumping stations.* https://www.maff.go.jp/j/nousin/kantai/tekiou/pdf/haisuiki_jyou_sankou.pdf (January 2020) [In Japanese].

Sugomori, Y. (2017) *Detailed explanation - time series data processing with TensorFlow and Keras*. Mynavi Publishing, Tokyo, Japan [In Japanese].

Varma, S. & Simon, R. (2006) Bias in error estimation when using cross-validation for model selection. *BMC Bioinformatics*, **7**: 91. DOI: https://doi.org/10.1186/1471-2105-7-91.

Yamada, K. et al. (2018) A case study of flood water level prediction in the Tokoro River in 2016 using recurrent neural networks. *J. Jpn Soc. Civil Eng.* (*JSCE) Ser. B1* (*hydraul. engr.*), **74**, I_1369-I_1374 [In Japanese with English summary].

Yamamoto, T. et al. (2010) Consideration of factors to promote the participation in maintenance works of irrigation and drainage facilities. *Jpn. J. Rural Plann. Assoc*., **29**, 101-106 [In Japanese with English summary].

Zappone, A. et al. (2019) Wireless networks design in the era of deep learning: model-based, AI-based, or both? *IEEE T. Commun*., **67**, 7331-7376. DOI: https://doi.org/10.1109/TCOMM.2019.2924010.

Appendix I

The preliminary sensitivity tests were performed for hyperparameters to choose a better setup of the LSTM model. Major tests included the number of hidden layers in the LSTM network structure, a selection from several optimizers in Keras that search for a local minimum of the loss function, and error evaluation of the ten groups that were separated from the observed period. The learning rate controls how much the weights of networks are adjusted to the gradient of the loss function. As a result of the first test related to the number of hidden layers, two layers can be appropriate for a longer lead time (after 3 h), although the mean RMSE of one layer is similar to that of the two layers (Fig. A1). Moreover, a deep-learning approach of more than three layers might not be trained appropriately in the learning process due to a limited number of data (Zappone et al. 2019). Stochastic gradient descent (SGD) was employed as an optimizer in this test.

In the second test for a better optimizer, SGD, Adagrad, Adam, Adamx, Nadam, and root mean square propagation (RMSProp) were selected. SGD is an iterative optimizing method with a smoothing approach and a constant learning rate based on a substantially stochastic approximation (Robbins & Monro 1951). Adagrad is a modified version of SGD with a per-parameter learning rate (Duchi et al. 2011), and Adam is an extended version of SDG that is designed particularly for a deep learning approach (Kingma & Ba 2015). Adamax is an extended method of Adam based on the Euclidean norm of individual present and past gradients (Kingma & Ba 2015). Nadam is a modified version of Adam incorporating Nesterov momentum based on a moving window of gradient updates (Dozat 2016), and RMSProp is a method that determines a specific learning rate for each parameter (Hinton). The number of hidden layers was set to two. The mean RMSEs from these optimizers similarly increase with respect to lead time from 1 to 10 h (Fig. A2). The mean RMSE of SGD is slightly better than those of the other optimizers from 1 to 4 h. A lead time of 3-4 h is roughly equivalent to a maximum response time of water level to rainfall in the target lowland, from past flood events.

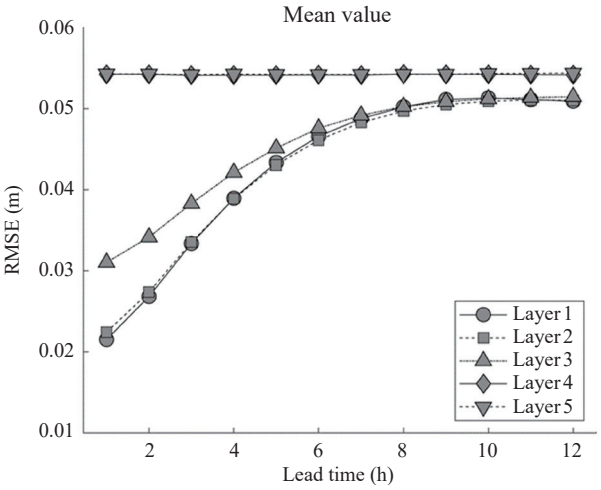

Fig. A1. Comparisons among M-RMSEs from the hidden layer test

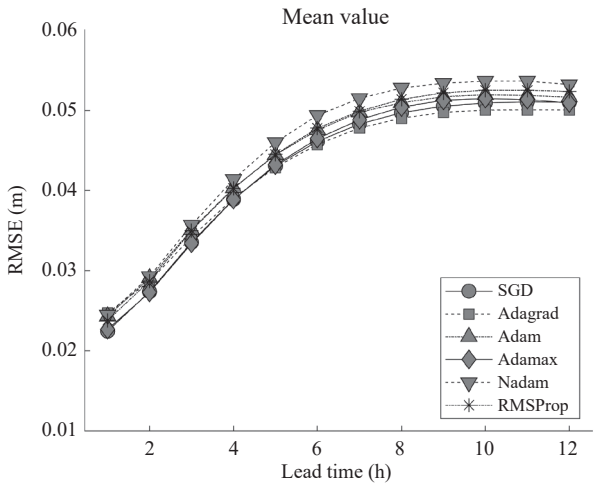

Fig. A2. Comparisons among M-RMSEs from the optimizer test

The third test evaluated the differences in RMSEs among the groups with up to a 4-h lead time. The number of hidden layers was two, and SGD was selected as a better optimizer. Group 2 had a higher RMSE up to a 4-h lead time in Fig. A3, because it involves a maximum water level caused by a more severe flood event. As a result, for the RMSE in Group 2, it is important to measure the worst error in the LSTM model as well as the mean RMSE.

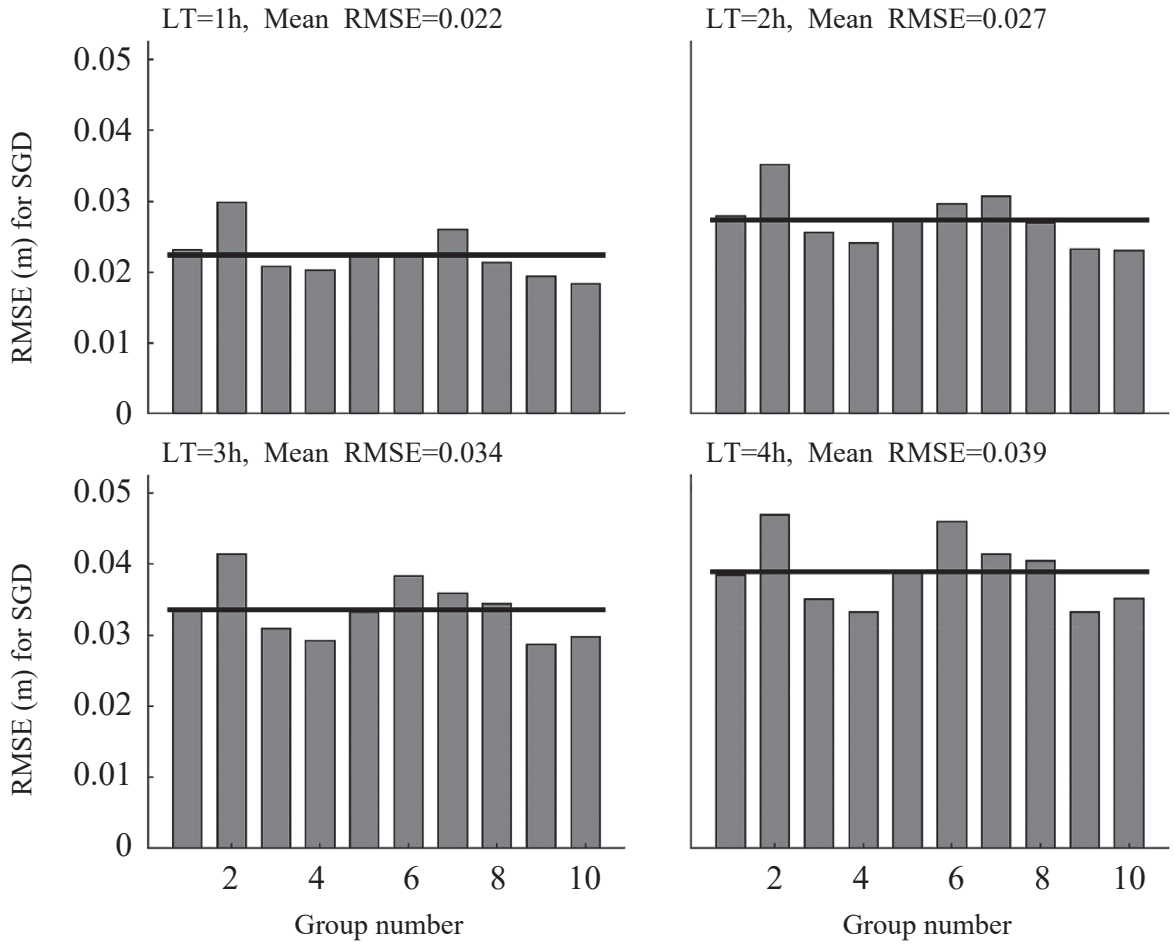

Fig. A3. Error evaluation of the ten groups: RMSEs from 1- to 4-h lead time (LT)
Solid lines indicate a mean of the RMSEs among the groups.