



Deep learning algorithm development for river flow prediction: PNP algorithm

Gwiman Bak¹ · Youngchul Bae²

Accepted: 13 April 2023 / Published online: 13 May 2023
© The Author(s) 2023

Abstract

Deep learning algorithms developed in recent decades have performed well in prediction and classification using accumulated big data. However, as climate change has recently become a more serious global problem, natural disasters are occurring frequently. When analyzing natural disasters from the perspective of a data analyst, they are considered as outliers, and the ability to predict outliers (natural disasters) using deep learning algorithms based on big data acquired by computers is limited. To predict natural disasters, deep learning algorithms must be enhanced to be able to predict outliers based on information such as the correlation between the input and output. Thus, algorithms that specialize in one field must be developed, and specialized algorithms for abnormal values must be developed to predict natural disasters. Therefore, considering the correlation between the input and output, we propose a positive and negative perceptron (PNP) algorithm to predict the flow rate of rivers using climate change-sensitive precipitation. The PNP algorithm consists of a hidden deep learning layer composed of positive and negative neurons. We built deep learning models using the PNP algorithm to predict the flow of three rivers. We also built comparative deep learning models using long short-term memory (LSTM) to validate the performance of the PNP algorithm. We compared the predictive performance of each model using the root mean square error and symmetric mean absolute percentage error and demonstrated that it performed better than the LSTM algorithms .

Keywords Deep learning · PNP · LSTM · River flow · Prediction

1 Introduction

Deep learning technology is emerging all around us, strengthening the technical aspects of modern society, from time-series prediction to image analysis to natural language processing (Ren and Xu 2015; Ciregan et al. 2012; Krizhevsky et al. 2012; Bordes et al. 2012; Cireşan et al. 2012; Mikolov et al. 2013; Hadsell et al. 2008). This technology has been widely used recently in biological mechanisms,

helping to solve problems that have previously been difficult to solve (Asgari Taghanaki et al. 2021; Tunyasuvunakool et al. 2021; Quang et al. 2015; Wang et al. 2020; Jaganathan et al. 2019). In addition, better algorithms have been developed for time-series prediction, such as recurrent neural networks (RNN), long short-term memory (LSTM), and gated recurrent units (GRU), which have shown strength in time-series predictions using deep learning based on big data accumulated in recent years (Oord et al. 1609; Bai et al. 1803; Borovykh et al. 1703; Mudassir et al. 2020; Li et al. 1707; Mudelsee 2019). To date, deep learning algorithms have been developed by focusing on the structure of the human brain, and the algorithms thus developed have generally performed well. The learning method of deep learning varies depending on the composition of the data because the method creates an optimal model based on the data. In the process of acquiring data, abnormal data may occur, such as those resulting from errors in sensors or abnormalities in the acquisition of the

✉ Youngchul Bae
ycbae@chonnam.ac.kr

Gwiman Bak
qkrrlend@naver.com

¹ Department of Electrical and Semiconductor, Chonnam National University, Yeosu 59626, Korea

² Division of Electrical and Computer Engineering, Chonnam National University, Yeosu 59626, Korea

data. These are called outliers. Deep learning models are affected by learning performance if there is an outlier in the learning data. Many researchers training deep learning models remove outliers in the data to improve the learning performance of the models (Gharaei et al. 2022; Hendrycks et al. 1812; Pang et al. 2021). From an analyst's point of view in the environmental field, natural disasters are judged to be outliers because the period of data acquisition using sensors is much shorter than the longest patterns of the earth. These natural disasters occurred once in 100 years in the past, but they have occurred more frequently in recent years due to climate change. If deep learning expert judges natural disaster data as outliers and delete them to train deep learning model, that cannot predict natural disasters.

Therefore, we need a wisdom which we do not treat as outlier for natural disaster data, and we also need to develop deep learning model to analyze and predict natural disasters to consider the frequency and damage scale of natural disasters. These novel deep learning model should be designed by cause information of natural disasters and the correlation between inputs and outputs. The deep learning model also should not be judged as outlier even though the data from natural disasters come in.

In addition, to predict outliers (natural disasters), deep learning algorithms specializing in a single field and specialized algorithms for outlier values must be developed. As the climate changes, natural disasters such as floods and droughts occur more frequently, and these disasters seriously impact cities around rivers. Climate change is one of the major problems facing humans because the changing climate patterns are expected to increase the frequency of extreme weather events (Field et al. 2012; Palmer and Räisänen 2002). In recent years, the frequency of droughts caused by global warming has been increasing. River and groundwater levels are declining due to the increased water demand (Lauer et al. 2018). As a result, water shortages, ground subsidence, and penetration into the groundwater layer of seawater may occur (Shankar et al. 2011; Malyan et al. 2019; Abidin et al. 2001). In addition, global warming can add to the collision of cold and dry air currents at high latitudes and warm and humid air currents at low latitudes, causing frequent torrential rains and floods worldwide (Fowler and Hennessy 1995; Papalexiou and Montanari 2019; Myhre et al. 2019; Hoegh-Guldberg et al. 2018). For this reason, it is necessary to analyze and predict the flow of rivers to prevent damage from water depletion and natural disasters from floods. Therefore, in this paper, we propose novel PNP deep learning model to predict outliers (natural disasters) in the river flow prediction.

Since the PNP deep learning algorithm induces more accurate flow prediction by applying the correlation between precipitation and flow, we can apply this in the river flow prediction.

This paper is organized as follows. Section 2 introduces the deep learning algorithm used in time-series prediction and the study of predicting river flow through deep learning. Section 3 describes the introduction, composition, and formula of the PNP algorithm. Section 4 describes the input and flow data to be used in Sects. 5 and 6. Section 5 measures the performance of the PNP algorithm, proving that it outperforms the LSTM algorithm. Section 6 predicts the river flow using not only the PNP algorithm but also the LSTM based on several variables. Finally, Sect. 7 comments on the predictive results and provides directions for future research.

2 Related works

In this section, we introduce the deep learning algorithm used in time-series prediction and the study of predicting river flow through deep learning. The artificial neural network (ANN), the most basic concept of deep learning, was inspired by biological processes in the 1960s, when it was discovered that different visual cortical cells were activated when cats visualize different objects (Hubel and Wiesel 1962, 1959).

The study reviewed the links between the eyes and cells in the visual cortex and showed that information was processed hierarchically in the visual system. The ANN mimicked the recognition of objects by connecting artificial neurons inside layers that could extract object features (Rosenblatt 1958).

An ANN does not provide information on the sequential order of the input, so there are limitations in processing time-series data with dependencies between the data. Time-series data or strings are generally used to predict later data by the data entered earlier. Therefore, it is difficult to accurately predict time-series data using an ANN, which only disseminates the current input data in the order of input, hidden, and output layers. To address these problems, the RNN was proposed, an ANN designed to process temporally continuous data, such as time-series data (Elman 1990).

The RNN adds loop connections to each neuron in the hidden layer to input the hidden layer output from previous data back into the hidden layer neurons when predicting data from the present time. This allows an RNN to make predictions about current data simultaneously based on the data entered at a previous time. However, when back-propagation algorithms learn data covering a long period, the gradient can be abnormally increased or reduced. LSTM was proposed to solve the gradient problems that occur with RNNs, and most RNN-based applications proposed to date have been implemented using LSTM (Hochreiter and Schmidhuber 1997).

LSTM adds new elements such as a forget gate, input gate, and output gate to each neuron in the hidden layer to solve gradient-related problems. As the layer deepens, LSTM causes information loss because the encoder has too much information to compress and tends to use the information compressed by the encoder only for the initial prediction. Thus, bidirectional RNN (BRNN) and bidirectional LSTM (BLSTM) were developed (Schuster and Paliwal 1997).

Two-way BRNN and BLSTM add a backward processing layer to the existing RNN and LSTM layers, where the input data are transmitted to both forward and backward learning. This does not degrade the performance, even if the data are lengthy. LSTM has become a model that performs well even with data with long sequences while also solving the long-term dependence problem, but it requires more parameters than an RNN due to its complex structure. Overfitting can occur when there are not enough learning data. GRUs have been proposed to improve on these shortcomings (Chung et al. 2014).

DeepAR is an autoregressive RNN-based forecasting method that learns a global model from the historical data of all-time series in the dataset. This algorithm learns seasonal behaviors and dependencies on given covariates across time series; minimal manual intervention in providing covariates is needed in order to capture complex and group-dependent behavior (Salinas et al. 2020).

Temporal Fusion Transformer (TFT) is based on deep neural network (DNN) architecture for multi-horizon forecasting that achieves high performance while enabling new forms of interpretability. For multi-layer horizontal prediction, various types of data are used as TFT input. In addition, it was designed to consider various types of input values by supplementing the limitations of the existing model (Lim et al. 2021) (Table 1).

The following is a list of flow prediction studies using machine learning.

Fathian et al. analyzed time series using self-exciting threshold autoregressive (SETAR) and generalized autoregressive conditional heteroscedasticity (GARCH) models (Fathian et al. 2019). Then, they used multivariable adaptive regression splines (MARS) and random forest (RF) models to predict the monthly river flow of the Grand River's Brantford and Galt Observatory in Canada. In addition, they developed hybrid models by combining MARS and RF models with SETAR and GARCH models, and they showed that the RF-SETAR models had the highest accuracy of the hybrid models.

Musarat et al. proposed a machine learning (ML) approach to predict water levels to prevent the devastation caused by extreme water levels rising in the Kabul River (Musarat et al. 2021). They used a variety of machine learning models, among used ML model, and they showed that ARIMA model has the highest performance.

Ghimire, S. et al. developed a new AI model based on LSTM and Convolution neural network (CNN) for hourly flow forecasts of the Brisbane River and Teewah Creek in Australia (Ghimire et al. 2021). They designed a prediction model based on six preceding values through statistical self-analysis of the time series from time-series data for the river flow. They also set different time intervals, such as one week, two weeks, four weeks, and nine months.

Huang et al. proposed deep learning models for flow prediction in astrophysics as an alternative to physically based models because the physically based models for flow prediction were relatively slow and costly (Huang et al. 2021). Their study used the SOBEK model which is physically based model and three neural network models: ANN, LSTM, and adaptive neural fuzzy inference system (ANFIS). They showed that the LSTM model had the highest accuracy.

Debbarma et al. utilized the gamma memory neural network (GMN) and genetic algorithm-gamma GMN (GA-GMN) to predict the daily flow rate (Debbarma and

Table 1 Related deep learning algorithms

Num	Year	Algorithm	Abbreviation	References
1	1958	Multi-layer perceptron	MLP	Rosenblatt (1958)
2	1990	Recurrent neural network	RNN	Elman (1990)
3	1997	Long short-term memory	LSTM	Hochreiter and Schmidhuber (1997)
4	1997	Bidirectional recurrent neural network	BRNN	Schuster and Paliwal (1997)
		Bidirectional long short-term memory	BLSTM	
5	2014	Gated recurrent unit	GRU	Chung et al. (2014)
6	2020	DeepAR	–	Salinas et al. (2020)
7	2021	Temporal Fusion Transformer	TFT	Lim et al. (2021)
8	2022	Positive & negative perceptron	PNP	This paper

Choudhury 2020). They showed that the GA-GMN model performed better than the GMN model.

Senent-Aparicio et al. suggested that instantaneous peak flow (IPF) is very important for reducing flood damage and is combined with ML models and soil and water assessment tool (SWAT) simulations to suggest an approach to estimating IPF (Senent-Aparicio et al. 2019). They applied an ANN, ANFIS, support vector machine (SVM), and extreme learning machine (ELM) as the ML models and compared the performance.

Mehedi, Md Abdullah AI et al. predicted river flow in East Branch Delaware in United States (Mehedi et al. 2022). This paper predicted discharge time series up to 7 days in advance using a LSTM neural network regression model trained using over 80 years of daily data.

Le, Xuan-Hien, et al. constructed several models of LSTM and have applied these models to forecast discharge at the Hoa Binh Station on the Da River, one of the largest river basins in Vietnam (Le et al. 2019). LSTM models have been developed for one-day, two-day, and three-day flowrate forecasting in advance at Hoa Binh Station. Rainfall and flowrate at meteorological and hydrological stations in the study area are input data of the model.

Dodangeh, E. et al. proposed novel integrative models by coupling the resampling algorithms with ML models (to provide the more reliable flood susceptibility predictions compared to the one-time splitting train-test data set) with a case study at Ardabil province near the coastal margins of Caspian Sea (Dodangeh et al. 2020). This paper used random subsampling (RS) and bootstrapping (BT) algorithms, integrated with machine learning models: generalized additive model (GAM), boosted regression tree (BTR) and multivariate adaptive regression splines (MARS).

Kim, Donghyun, et al. used hydrological data observed in the upper Bokha interchange of the river to predict the flood level of Heungcheon Station located downstream of the Bokha River (Kim et al. 2022). Data on rainfall, water level, and discharge volume from 2005 to 2020 were collected from two observation stations for each upstream and downstream basin. In addition, we perform flood water level predictions at downstream points using machine learning models such as Gradient Boosting (GB), SVM, and LSTM.

Studies on river flow prediction using deep learning (Fathian et al. 2019; Musarat et al. 2021; Ghimire et al. 2021; Huang et al. 2021; Debbarma and Choudhury 2020; Senent-Aparicio et al. 2019; Mehedi et al. 2022; Le et al. 2019; Dodangeh et al. 2020; Kim et al. 2022) show that the algorithms used to predict river flow are limited in predicting outliers using data-based algorithms. Therefore, we propose the PNP algorithm, which can predict the flow of

river using climate change-sensitive precipitation (Table 2).

3 PNP algorithm

3.1 Characteristics of the PNP algorithm

On Earth, water always flows, and the process of water completing its cycle is called water circulation. This circulation is the continuous flow of water above and below the Earth's surface. It is usually caused by water evaporating from the sea and forming clouds, which transfer the water to land through rain. Once on land, the water flows over the surface of the earth to a lower place on the surface, forming a stream or a river. The stream's water flows downstream into the ocean, completing the water circulatory system. The time when water stays in this circulatory system is called the residence time. The average residence time in Korea is 1.5 weeks in the atmosphere and two weeks in the river. Even in the absence of precipitation, the river flow characteristics require a deep learning algorithm that can identify the characteristics of the water circulation system to predict the river flow. Therefore, we present the PNP algorithm, a new deep learning algorithm that considers the water circulation system and the residence time. The deep learning model consists of input, hidden, and output layers. The input data are fed into the input layer, and the output layer produces the model value. Hidden layers are made up of neurons composed of deep learning algorithms. The PNP algorithm can consist of neurons in the deep learning model, such as MLP, LSTM, and other deep learning algorithms. Figure 1 shows the layers and neurons of the deep learning model.

The PNP algorithm focused on river flow that we present is divided into water entering the stream (positive water) and water exiting to the sea (negative water). If there is more positive water than negative water, the river becomes too full, which causes flooding, while more negative water than positive water leads droughts. We applied positive and negative water to the PNP algorithm, corresponding to the positive and negative neurons, respectively, and we also applied features from past precipitation data to consider the residence time.

The PNP algorithm is organized in a hidden deep learning layer, such as LSTM or RNN, consisting of positive and negative neurons in the hidden layer. Figure 2 shows the PNP algorithm concept. Both neurons receive the precipitation as input. Positive neurons analyze the increasing factor of the water flow in the river, while negative neurons analyze the decreasing factor. There is a

Table 2 Related river flow prediction using deep learning

Num	Country	Algorithm	Description
1	Canada	MARS, RF	This work predicted monthly river flows using three AI approaches: ANN, MARS, and RF. Among the hybrid models developed, the RF-SETAR models were generally the most accurate, improving the river flow modeling.
2	Afghanistan	ARIMA	This approach used an ML tool known as an autoregressive integrated moving average for statistical methodological analysis to predict stream flow.
3	Australia	CNN-LSTM	CNN layers were used to extract the time-series river flow features, while LSTM networks used these features from the CNN for river flow time-series prediction. River flow prediction was conducted for different time intervals, including one week, two weeks, four weeks, and nine months. The results showed that the proposed CNN-LSTM model based on the novel framework yielded more accurate predictions. Thus, CNN-LSTM has significant practical value in river flow prediction.
4	China	LSTM, ANFIS, SOBEK	This work focused on evaluating the reliability of three neural network models (ANN, LSTM, ANFIS) and one physically based model (SOBEK) in terms of efficiency and accuracy for average and peak streamflow simulation. LSTM models can generally compete with physically based models in flow simulations of complex urban river systems by providing fast flow predictions with acceptable accuracy.
5	India	GMN, GA-GMN	The authors showed how to predict daily river flows using a memory-based ANN. They chose two different networks: GMN and GA-GMN. The GA-GMN model gave better results for both datasets. Therefore, it was chosen as an effective tool to predict the flow features of the Dholai River.
6	Spain	ANN, ANFIS, SVM, ELM	This work proposed an approach to instantaneous peak flow estimation by combining SWAT simulations and ML models. The SWAT model was used to estimate the maximum average daily flow, and the ML model was used to estimate the instantaneous peak flow based on the maximum average daily flow.
7	United States	LSTM	This study contributes to a reproducible template to investigate the uniqueness of the temporal dynamics of river discharge through extensive EDA. The hidden pattern of the distribution of discharge values through over 80 years of data is discovered in various up-to-date data exploration tools, which is a mandatory requirement for the satisfactory training of the LSTM algorithm.
8	Vietnam	LSTM	This work has proposed an effective approach to flood forecasting based on the data-driven method. The LSTM neural network model was constructed and carefully assessed to forecast one-day, two-day, and three-day flood flow at the Hoa Binh Station on the Da River.
9	Iran	RS,BT, GAM,BTR, MARS	This work proposed a novel integrated model by combining the resampling algorithm with an ML model with a case study in Ardabil province near the Caspian coast edge. The main purpose of this work is to evaluate the performance of three ML models: GAM, BRT, and MARS for river flow sensitivity prediction, and to integrate independent ML models (GAM, BRT, and MARS models) with new RS-GAM and RS-GARS models.
10	Korea	GB, SVM, LSTM	This work collected rainfall, water level, and discharge data from 2005 to 2020 for each upstream and downstream basin, and in particular, the collected rainfall data were classified into 53 rainfall events using IETD (Interevent Time Definition) analysis of the rainfall data. In addition, flood water level forecasting at downstream point was performed using machine learning models such as GB (Gradient Boosting), SVM (Support Vector Model), and LSTM (Long Short-Term Memory). Also, the SFM which is a rainfall-runoff model was used for the forecasting.
11	Korea	PNP	In the current paper, we propose an algorithm that can be applied to the water circulatory system and water retention times. We use not only weather input variables but also upstream operating data as input variables to predict the river flow.

Fig. 1 Layers and neurons of the deep learning model

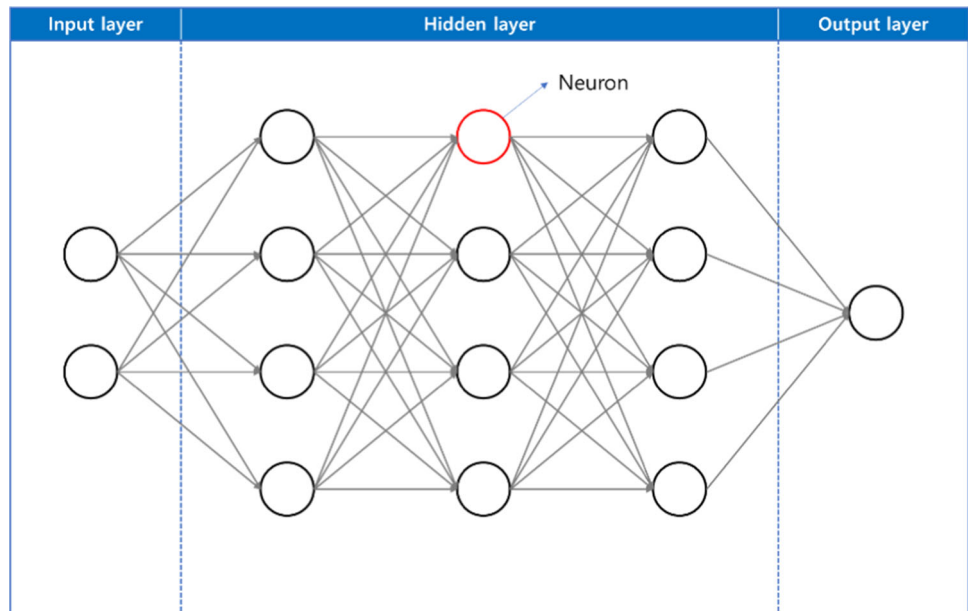
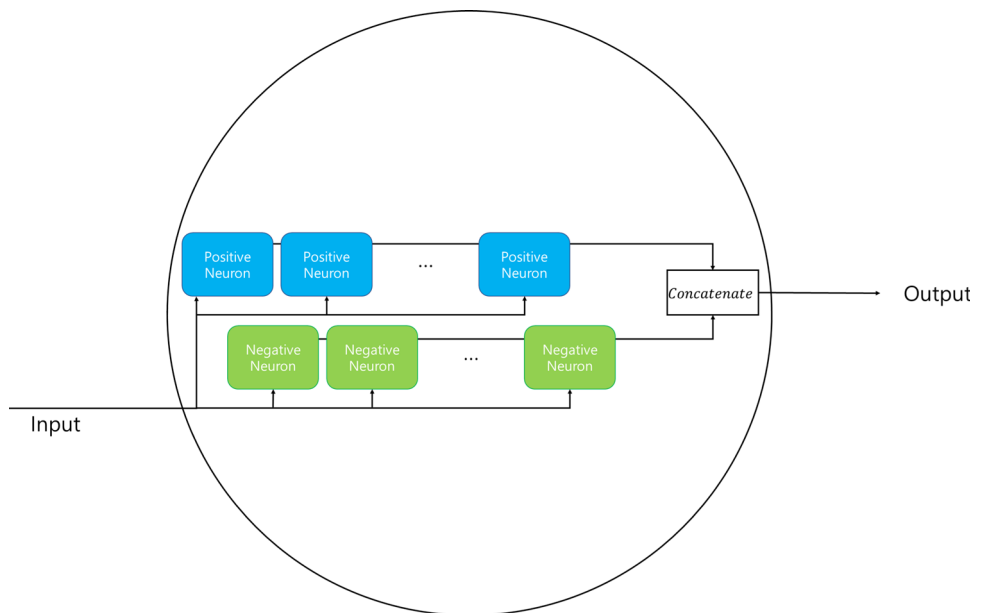


Fig. 2 PNP algorithm concept

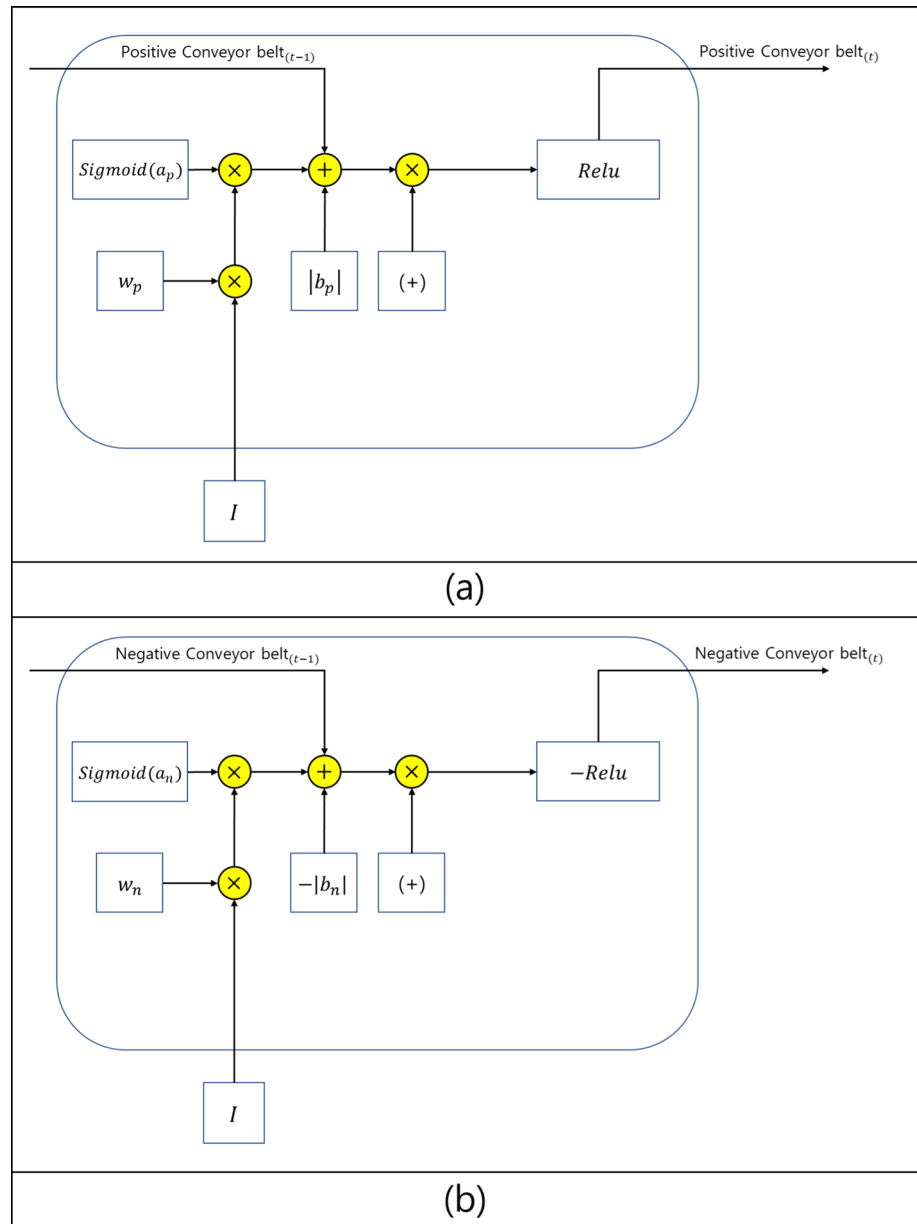


line connected between the negative and positive neurons called the conveyor belt, which delivers the results of each neuron to the next node and provides past precipitation information. Through this process, we can reflect the amount of river emissions during the residence time. The conveyor belt provides the calculated value from the previous neuron to the next neuron. This is similar to the cell state in LSTM, which completely removes the cell state's information. However, unlike the cell state, the conveyor belt maintains the information and transfers it to the next neuron. Finally, when the last positive and negative

neurons finish calculating, two results are generated from one node of the PNP hidden layer. Figure 3 shows the positive and negative neuron configuration diagram.

The positive and negative neurons are calculated in four steps: input, judgment, application, and transfer. The first step of the positive neuron is the input step, which is given to the neuron and multiplied by the weighted value w . The second step is the judgment by applying the *sigmoid* function, using the value one if the need is high and zero if it is low. The third step is the application step, which adds the value of the sigmoid function, the positive bias term,

Fig. 3 Positive and negative neuron configuration diagram



b_p , and the value from the previous positive conveyor belt. Finally, in the fourth step, we remove a negative element by applying a rectified linear unit (ReLU) function to the value calculated in the third step. The values removed by the ReLU are transferred to the conveyor belt and the next neuron. The negative neuron follows the same process through the second computational step, but in the third step, b_n is added to the negative bias term rather than the positive bias term, b_p . The fourth step is applied to prevent the quantity value. Equations (1)–(4) show the calculation of positive neurons, while Eqs. (5)–(8) show the calculation of negative neurons.

$$fs_p = I \times w_p \tag{1}$$

$$ss_p = \text{sigmoid}(a_p) \times fs_p \tag{2}$$

$$ts_p = ss_p + |b_p| + \text{PCb}_{(t-1)} \tag{3}$$

$$Ls_p = \text{Relu}(ts_p) \tag{4}$$

$$fs_n = I \times w_n \tag{5}$$

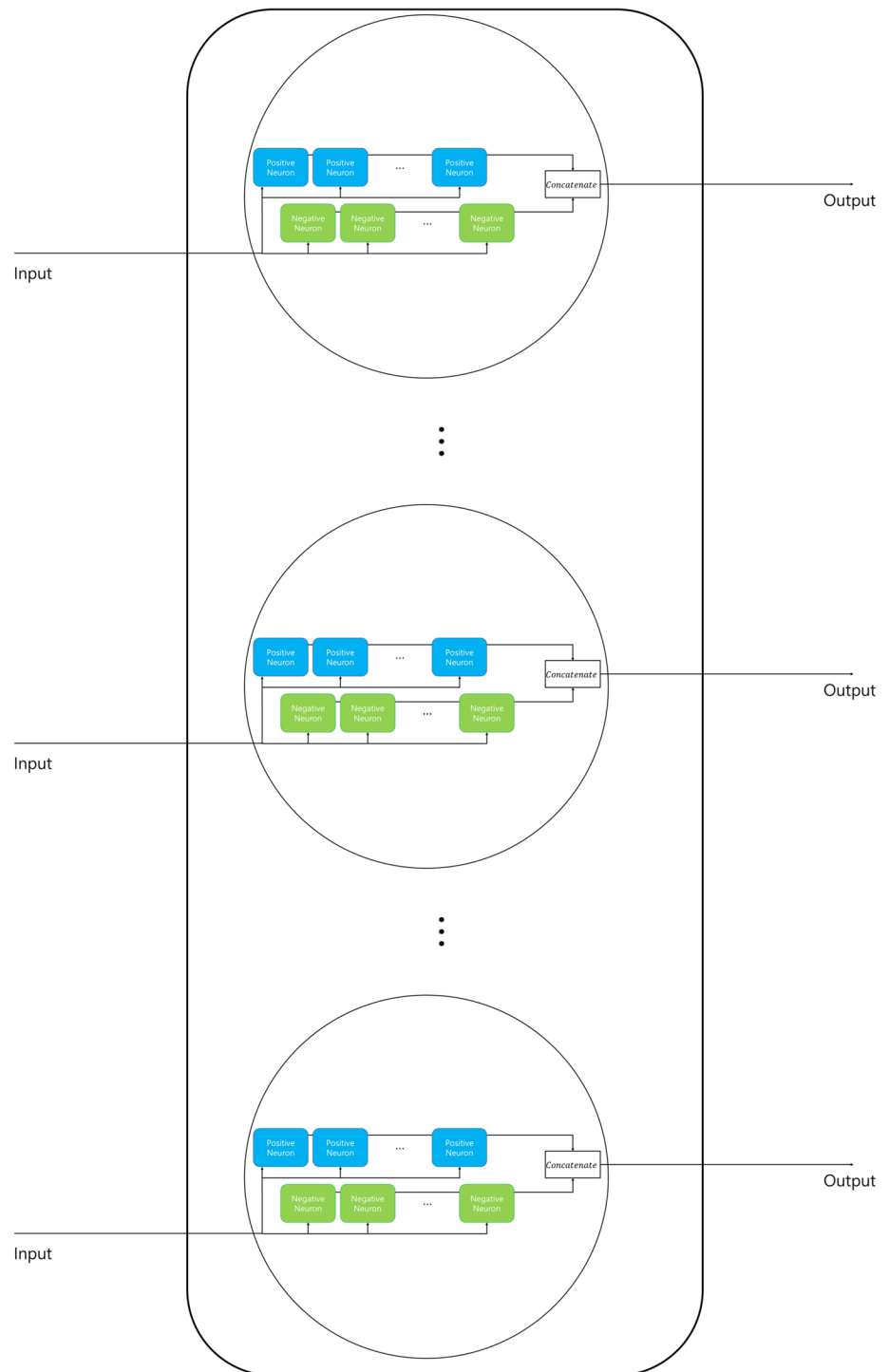
$$ss_n = \text{sigmoid}(a_n) \times fs_n \tag{6}$$

$$ts_n = ss_n + |-b_n| + \text{NCb}_{(t-1)} \tag{7}$$

$$Ls_n = -\text{Relu}(-ts_n) \tag{8}$$

In Eqs. (1)–(8), fs_p and fs_n are the first step in the PNP algorithm computational process, ss_p and ss_n are the second

Fig. 4 Hidden layers and internal nodes comprising the PNP algorithm



step, ts_p , ts_n are the third step, and LS_p and LS_n are the final step. w_p and w_n are the positive and negative weighted values, respectively. a_p and a_n are the positive and negative sigmoid functions, respectively. b_p and b_n are the positive and negative bias terms, respectively. I is the input data, $PCb_{(t-1)}$ represents the previous positive conveyor belt value, and $NCb_{(t-1)}$ represents the previous negative conveyor belt value.

3.2 Deep learning configuration of PNP algorithm

PNP algorithms can be configured as hidden ANN layers, and the number of nodes can be adjusted, as in other deep learning algorithms such as MLP, LSTM, and GRU. Figure 4 shows the hidden layers and internal nodes comprising the PNP algorithm. The input value of the PNP

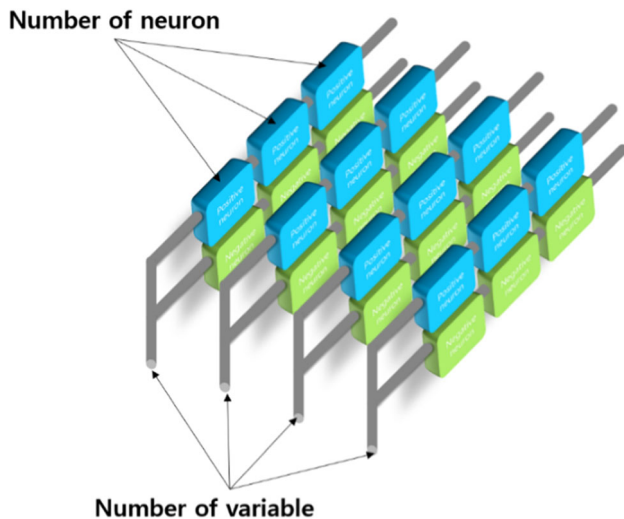


Fig. 5 Algorithm structure according to the numbers of variables and neurons

algorithm is entered in three dimensions, as with the LSTM algorithm. The input data dimensions consist of the data size, variable, and time step. The data size is the amount of data in time units used in the algorithm. The variable represents the number of variables. The time step is the number of past data points used, for which we must consider the residence time of the river flow. We can designate the number of neurons in positive and negative time steps through the time step value. The neuron array is organized in a parallel format based on the number of variables. The neuron array has a two-dimensional structure composed in a series according to the time step. Figure 5 shows the algorithm structure according to the number of variables and neurons.

4 Acquired data characteristics

To predict the river flow through deep learning, we must consider the correlation between the target value and the input variables. Because the river flow includes the circulation process of water and the characteristic of flowing from upstream to downstream, we should select input variables that have such a characteristic of water or affected. The water circulation process is largely determined by weather (water quantity, temperature, etc.) and the upstream impact of dams, depending on how the dams operate. Therefore, we collected river flow data, weather observation data closest to the flow measurement point, and

upstream dam operation data to predict the river flow. In addition, we obtained datasets from three regions (Hangang Bridge, Yeosu Bridge, and Gangchung Bridge) to generalize the performance evaluation of the algorithms.

4.1 River flow

We acquired data measured at the Hangang, Yeosu, and Gangchung Bridges in Korea. The Hangang Bridge ($37^{\circ}31'03.3''\text{N}$, $126^{\circ}57'31.7''\text{E}$), located over the Han River in Seoul, has a total length of 1,005 m. The Han River is the second longest river in South Korea (494 km) and the largest river in terms of flow and basin. The Han River has hundreds of tributaries, including the Bukhan and Imjin Rivers, across four regions: Gyeonggi, Gwandong, Haeseo, and Hoseo. The Yeosu Bridge ($37^{\circ}29'81.7''\text{N}$, $127^{\circ}64'81.4''\text{E}$) is located at the southern end of the Namhan River from Sangdong to Cheonsong-dong, Yeosu-si, Gyeonggi-do. The Gangchung Bridge ($36^{\circ}82'16.1''\text{N}$, $126^{\circ}93'36.6''\text{E}$), which crosses the Gokgyo river in Yeomchi, is 325 m long and 10 m wide. The blue color in Fig. 6a indicates the Han River, and the red checkpoints indicate the locations of the Hangang and Yeosu Bridges. The blue color in (b) indicates the Gokgyo river, and the red checkpoint indicates the location of the Gangchung Bridge. The Hangang and Yeosu Bridges are located in the same Han River area, but there are differences in upstream and downstream, and the Gangchung Bridge is located in a different basin. The Hangang Bridge is the largest by region, followed by the Yeosu Bridge, and finally, the Gangchung Bridge is the smallest.

The river flow data were provided by the Han River Flow Control Office for the period 1/1/2010–12/31/2020. The unit of the data is one day and is in the form of 4018 int. Table 3 shows the information of the acquired river flow data, and Fig. 7 shows the obtained river flow graph.

4.2 Weather

As Korea is in the mid-latitudes of the northern hemisphere, changes are evident in all four seasons, with periodic characteristics throughout the year as the seasons change. We used the weather observation data closest to the river flow measurement area and collected data from the Korea Meteorological Administration. Figure 8 shows the locations of the weather station and the flow station, and Table 4 summarizes the weather data.

Fig. 6 **a** Map of the Han River basin and location of the Hangang and Yeosu Bridges; **b** map of the Gokgyo River basin and location of the Gangchung Bridge

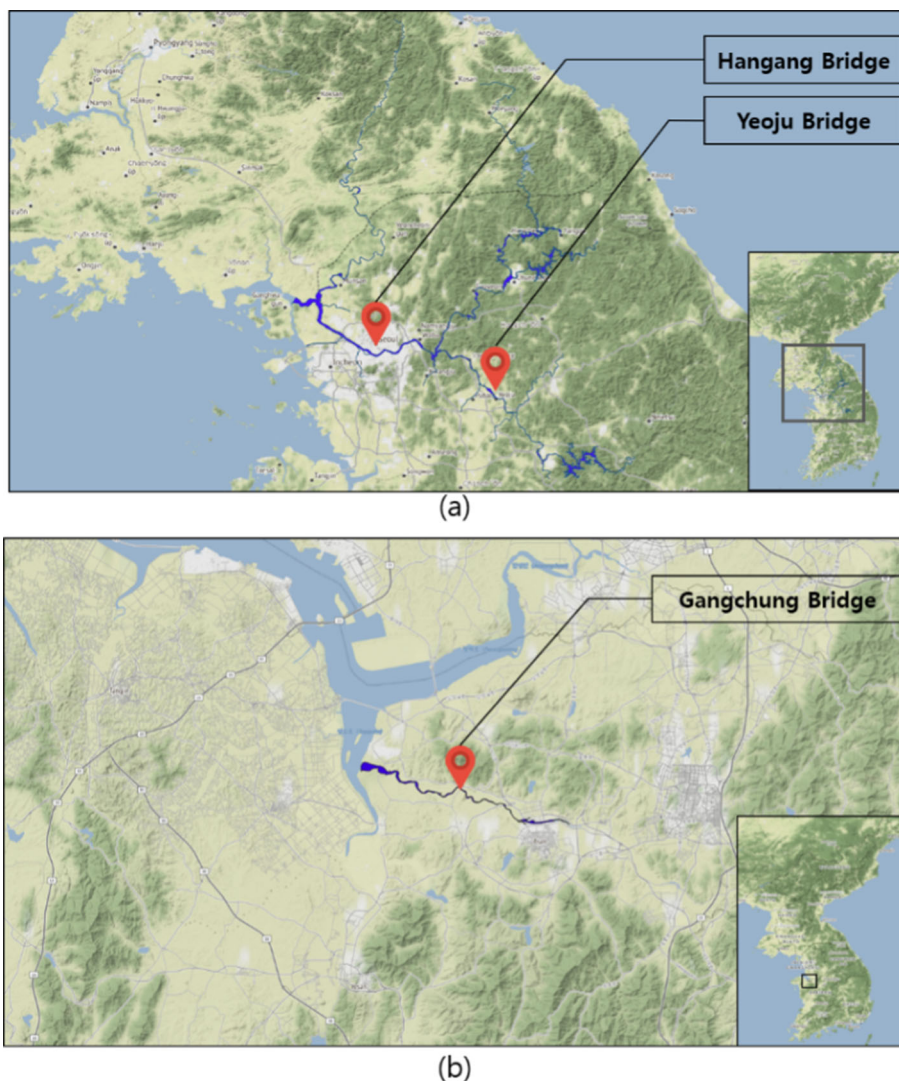


Table 3 Analysis of river flow (m^3/s) data obtained

Location	Period	Min	Max	Mean	Std
Hangang Bridge	1/1/2010–12/31/2020	12.78	20680.17	484.188	1113.083
Yeosu Bridge	1/1/2010–12/31/2020	11.06	5493.91	231.759	418.218
Gangchung Bridge	1/1/2010–12/31/2020	4.4	696.22	55.4	41.798

4.3 Upstream dam hydrological data

Dams reduce the peak flood of downstream rivers during flooding and provide water for continuous river maintenance during droughts. They affect downstream changes in spill characteristics in the short term. On the other hand, they also increase flows in the dry season for the long term caused by increased low flows. For this reason, we must use upstream floodgate information as an input variable to predict the flow of the stream. Thus, we collected upstream dam operation data around the flow observation point. We

used hydrological data provided by the Korea Water Resources Corporation and Korea Hydro & Nuclear Power for the dams upstream on the Han River. Figure 9 shows the location of the dams on the Han River, and Table 5 summarizes the data from the dam.

There are 10 dams upstream from the Hangang Bridge, three dams upstream from the Yeosu Bridge (the Hoengseong, Chungju, and Goesan Dams), and none upstream from the Gangchung Bridge.

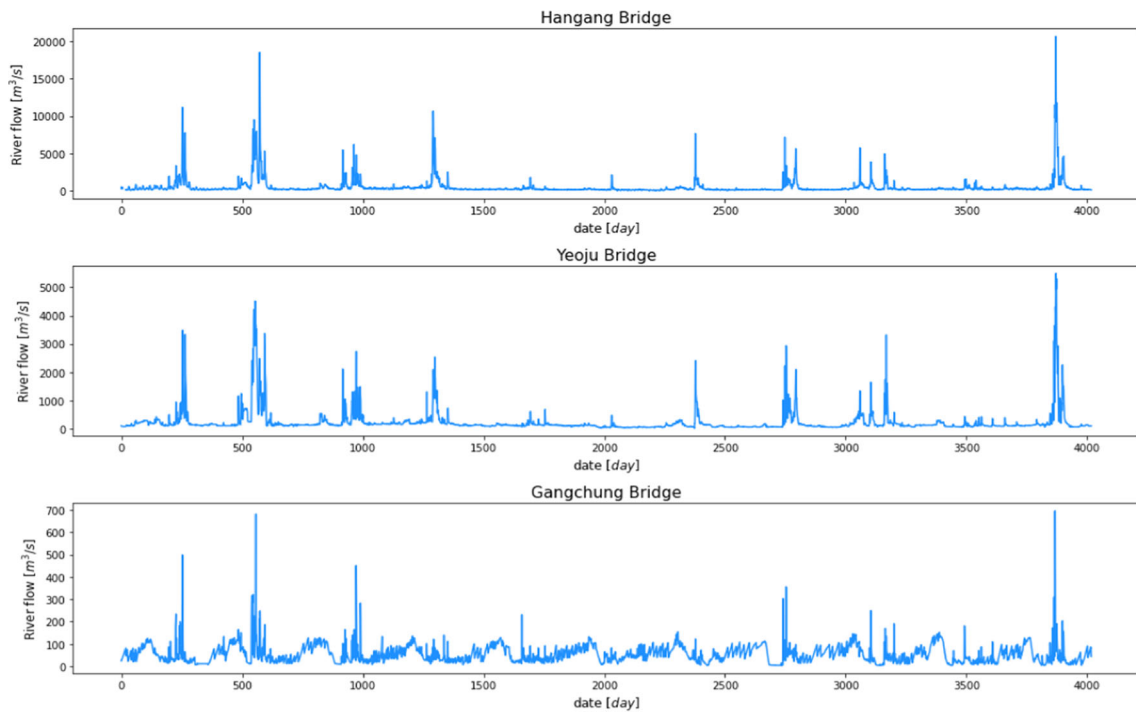
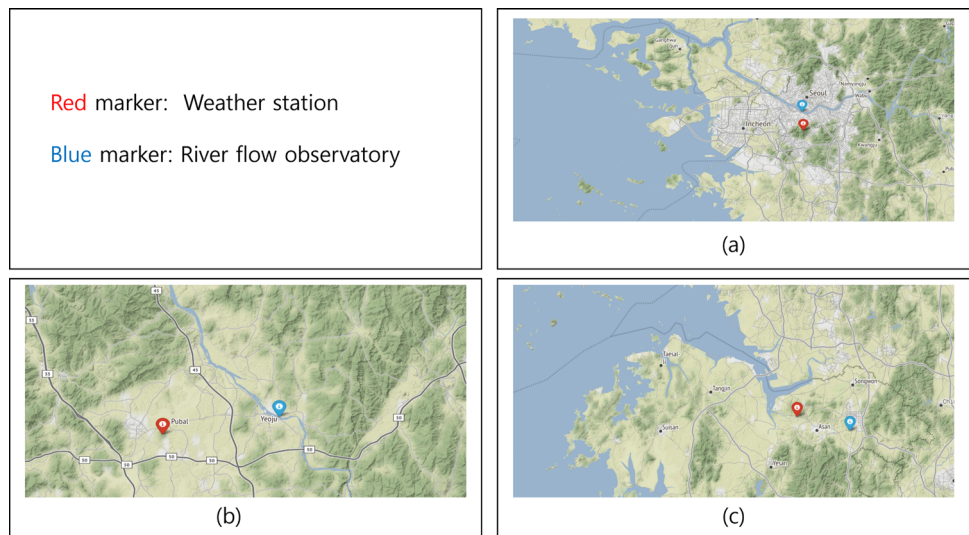


Fig. 7 River flow graph (2010–2020)

Fig. 8 Locations of river flow observatories and weather stations: a Hangang Bridge, b Yeosu Bridge, and c Gangchung Bridge



5 Performance verification of the PNP algorithm

5.1 Algorithm performance verification pipeline configuration

We built a pipeline to verify the algorithm performance. The pipeline sequence is divided primarily into the data and deep learning processes. Figure 10 indicates the configuration of the algorithm performance verification pipeline. The first step of the data process divides the acquired

data into learning and test data in an 80:20 ratio. The learning data are used to train the deep learning model, and the test data are used to verify the model performance. We also applied K-fold cross-validation to prevent data with high noise values from being concentrated on one side when dividing the learning and test data and overfitting to a specific dataset. K-fold cross-validation creates K divisions of the learning and test data. In this paper, we organized five types of K-folds, and we also designated the same ratio for the learning and test data. However, we designated

Table 4 Analysis of weather data acquired

Location	Period	Sort	Min	Max	Mean	Std
Hangang Bridge	1/1/2010–12/31/2020	Temperature[°C]	-14.8	33.7	13.0	10.8
		Precipitation[m ³ /s]	0.0	301.5	3.68	14.401
		Humidity[%]	17.9	99.8	59.7	15.1
Yeosu Bridge	1/1/2010–12/31/2020	Temperature[°C]	-14.8	31.6	11.9	10.7
		Precipitation[m ³ /s]	0.0	247.5	3.5	12.9
		Humidity[%]	15.6	100	65.5	15.0
Gangchung Bridge	1/1/2010–12/31/2020	Temperature[°C]	-13.7	30.8	12.3	10.4
		Precipitation[m ³ /s]	0.0	232.7	3.3	12.1
		Humidity[%]	16.3	99.3	68.7	13.4

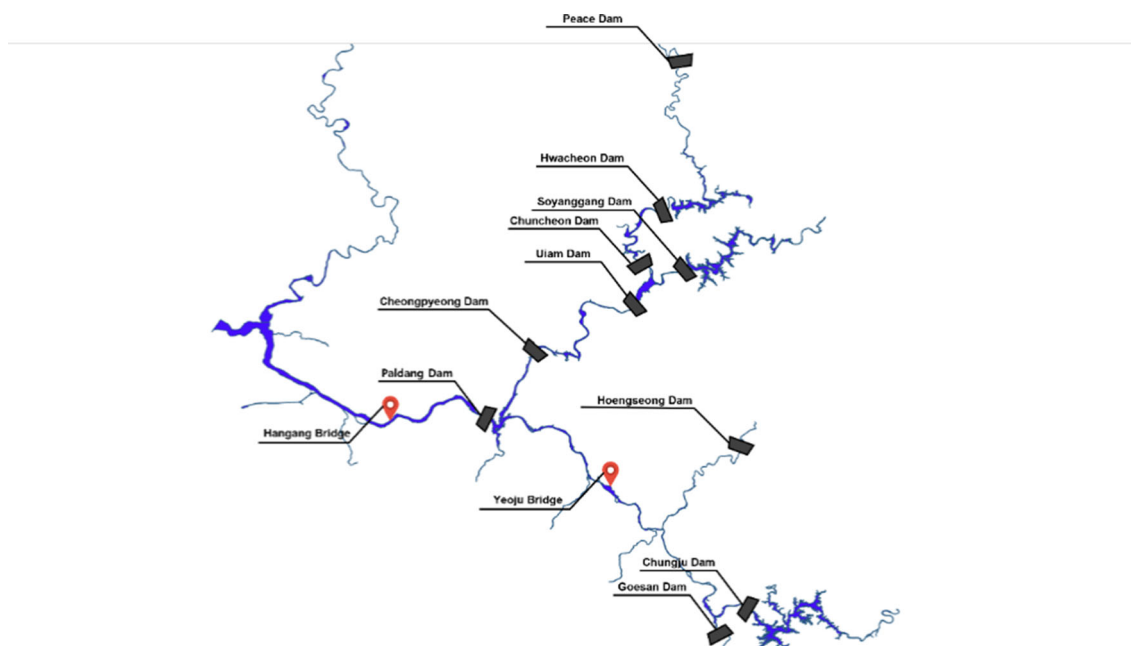


Fig. 9 Location of the dams in the Han River basin

Table 5 Analysis of discharge (m³/s) data of the dams upstream from the Hangang Bridge

Location	Period	Min	Max	Mean	Std
Paldang dam	1/1/2010–12/31/2020	0.0	15870.583	422.778	932.7
Cheongpyeong dam	1/1/2010–12/31/2020	0.0	9985.75	181.625	436.982
Uiam dam	1/1/2010–12/31/2020	1.1	9948.75	142.506	344.196
Chuncheon dam	1/1/2010–12/31/2020	0.0	4118.625	68.696	202.649
Soyanggang dam	1/1/2010–12/31/2020	2.728	2423.94	62.963	97.428
Hwacheon dam	1/1/2010–12/31/2020	0.0	2945.458	49.229	202.649
Peace dam	1/1/2010–12/31/2020	0	2179.22	52.365	114.605
Hoengseong dam	1/1/2010–12/31/2020	0.036	363.322	4.467	12.463
Chungju dam	1/1/2010–12/31/2020	0.0	3960.37	136.99	235.694
Goesan dam	1/1/2010–12/31/2020	0.0	684.042	12.239	34.539

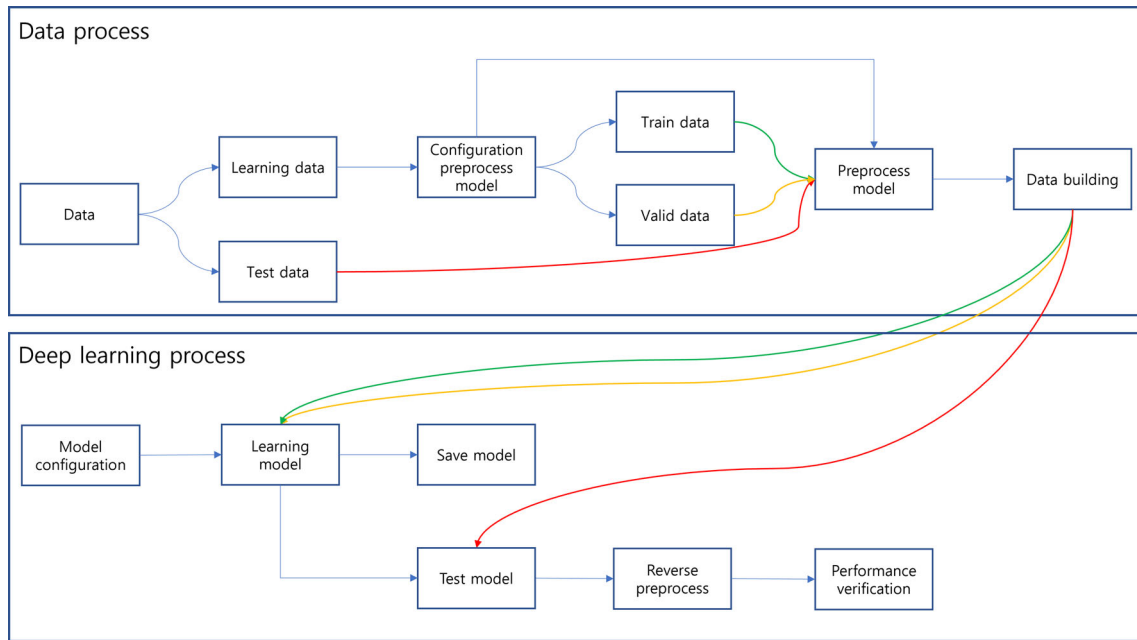


Fig. 10 Performance verification of the algorithm pipeline configuration

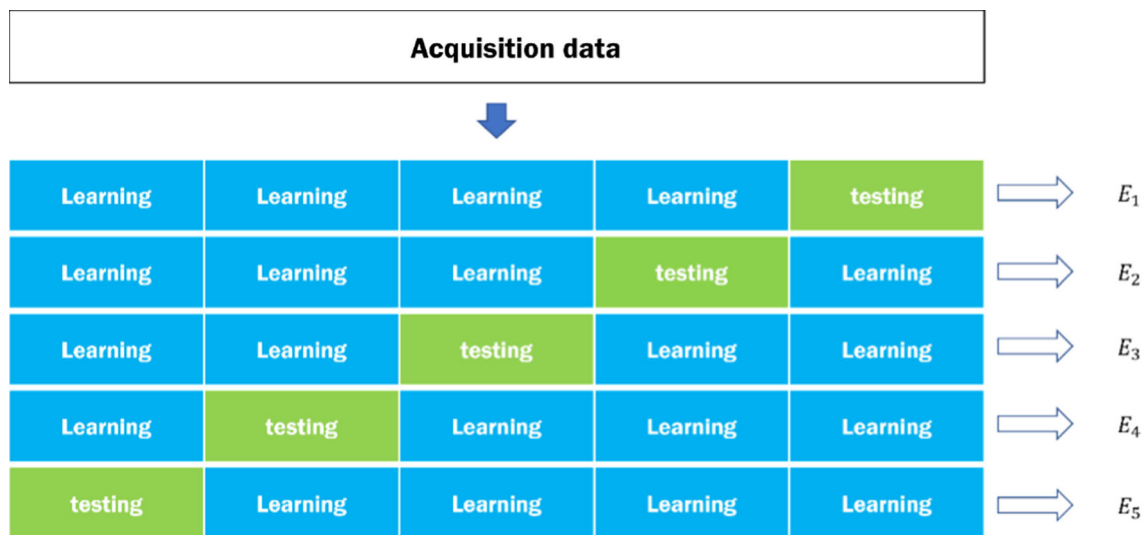


Fig. 11 K-Fold cross-validation where K = 5

different locations for the test data. Figure 11 illustrates the K-fold cross-validation where K = 5.

We organized a preprocessing model for the learning data. The preprocessing model is for standardization, using the learning data divided between learning and validation in a 60:20 ratio. Figure 12 shows the division of the acquired data.

We divided the acquired data into learning and testing data and the learning data into training data and validation data. The learning data were used to construct

preprocessing models, and the training data were used to train the models. The validation and testing data were used to measure the performance of the deep learning model, with the validation data used in constructing or training the model. Validation data were used for preprocessing and training the deep learning model, for which the testing data were not involved. After applying the training, validation, and test data to the preprocessing model, we built a dataset to fit the input of the deep learning model. We organized the data building in three dimensions, namely, the data

Fig. 12 Division of acquired data

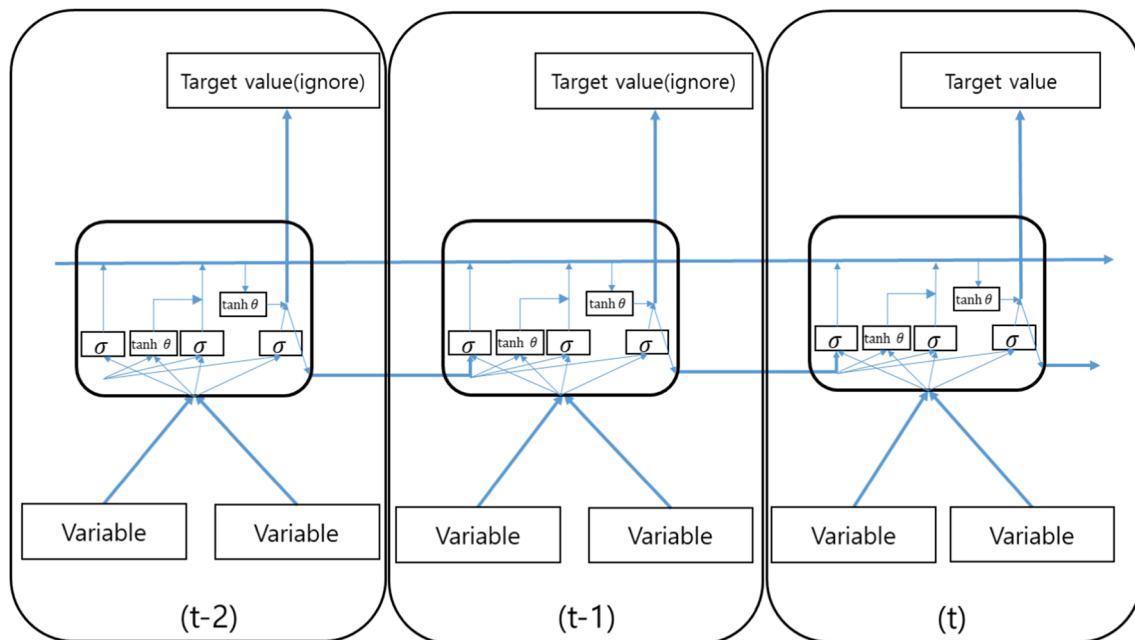
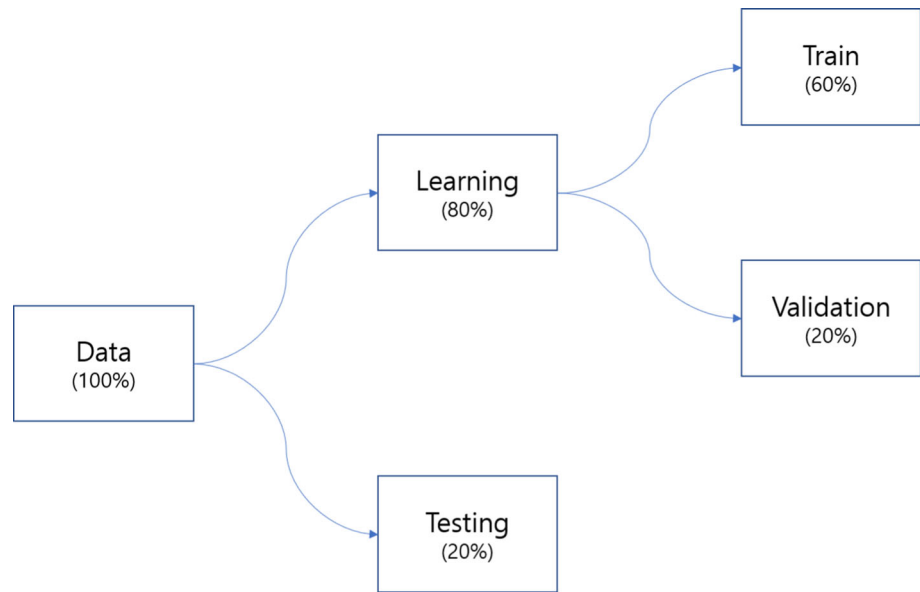


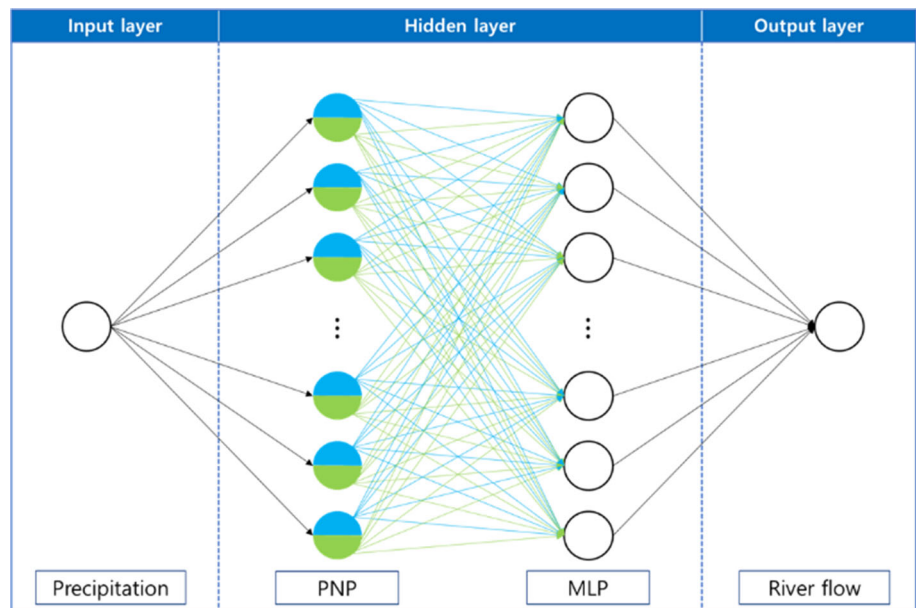
Fig. 13 Structure of LSTM

size, variable, and time step, to fit the PNP algorithm input requirements. The time step must be selected considering the residence time in the river. The average residence time in Korean rivers is 14 days; thus, we designated 14 days as the time step. In this study, we designated the variable value as one because the input value to be entered into the PNP algorithm is one of the closest precipitation points to the river flow observation point. The total data size is 3977 data records based on the past data considering the time

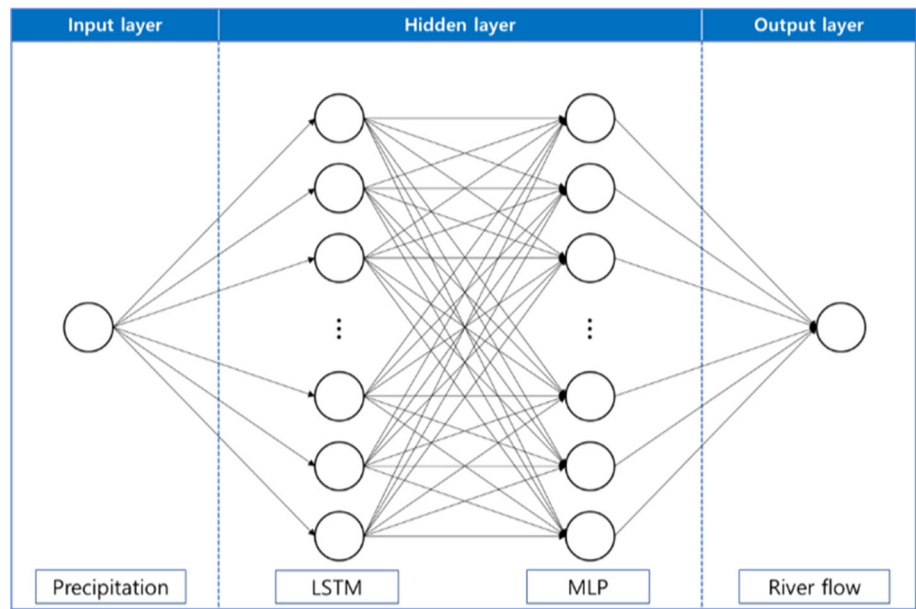
step. Therefore, the size of the input data of the training, verification, and testing datasets was [2386, 1, 14], [795, 1, 14], [796, 1, 14].

The first stage of the deep learning process is the model configuration. We constructed the model with the PNP and MLP algorithms, designating 50 nodes for each algorithm. We used the Adam optimizer and mean squared error for error measurement. We used another deep learning which

Fig. 14 Model composition using **a** the PNP algorithm and **b** the LSTM algorithm



(a)



(b)

LSTM algorithm with excellent performance in time-series prediction to compare the performance.

LSTM algorithm is a derivative RNN algorithm, which consists of three gates: Input Gate, Output Gate, and Forget Gate. If the $(t-1)$ hidden layer of Fig. 13 is affected by the (t) hidden layer, it adds a calculation that erases it. We constructed a model like the PNP and MLP algorithm, and this model is composed of the LSTM and MLP algorithm. This model consisted of LSTM and MLP algorithms, with

the number of nodes set to 50. Figure 14 shows the configuration of the model, with Fig. 14a representing the model using the PNP algorithm and Fig. 14b showing the model using the LSTM algorithm.

After constructing the model, we used the training and validation data to train it. We input the trained model as test data to predict the river flow and then applied reverse preprocessing. We then measured the error by comparing the reverse-prediction result with the actual river flow

Table 6 Comparison of model prediction errors

Location	K-fold	RMSE(m^3/sec)		sMAPE(%)	
		PNP	LSTM	PNP	LSTM
Hangang Bridge	E_1	332.934	367.537	24.979	25.555
	E_2	217.206	310.295	26.461	29.886
	E_3	239.045	317.881	30.888	32.239
	E_4	291.571	328.284	23.830	27.781
	E_5	431.208	477.596	25.617	29.137
	E_{mean}	302.392	360.319	26.355	28.920
Yeoju Bridge	E_1	136.686	166.380	25.686	26.232
	E_2	141.359	164.857	30.581	33.161
	E_3	95.066	136.162	26.861	34.055
	E_4	143.296	162.093	26.860	29.384
	E_5	174.871	294.072	24.579	26.889
	E_{mean}	138.256	184.713	26.913	29.944
Gangchung Bridge	E_1	31.457	47.717	32.814	41.450
	E_2	33.050	49.687	30.737	40.433
	E_3	26.550	47.430	24.989	37.393
	E_4	28.640	46.968	26.632	28.030
	E_5	34.306	58.868	29.663	42.033
	E_{mean}	30.801	50.134	28.967	37.868

Bold values indicate the emphasizes that the error values (RMSE, sMAPE) for both the PNP and LSTM algorithms are lower compared to the mean value of Location

value using RMSE and sMAPE for error measurement. We compared the predictive results of the two models with two error-measurement methods to demonstrate the PNP algorithm performance. Equation (9) represents the preprocessing, and Eq. (10) represents the reverse preprocessing. Equation (11) represents RMSE, and Eq. (12) represents sMAPE.

$$\text{Preprocess} = \frac{ID - LD_{mean}}{LD_{std}} \tag{9}$$

$$\text{Reverse preprocess} = PD \times LD_{std} + LD_{mean} \tag{10}$$

$$RMSE = \sum \sqrt{(PD - RD)^2} \tag{11}$$

$$sMAPE = \frac{|PD - RD|}{(|PD| + |RD|)/2} \times 100 \tag{12}$$

In Eqs. (9)–(12), ID is the training, verification, and test data, while LD_{mean} represents the average value of the learning data and LD_{std} represents the standard deviation. PD indicates the prediction value of the model using the test data, and RD indicates the river flow data.

5.2 Verification results

Table 6 gives the RMSE and sMAPE prediction results for the two models. E_1, E_2, E_3, E_4, E_5 represent the K-fold

cross-validation in Fig. 11, respectively, while E_{mean} represents the average value of E_1, E_2, E_3, E_4, E_5 . The error of the PNP model was lower than that of LSTM for the Hangang, Yeoju, and Gangchung Bridges. Thus, we proved that the PNP algorithm performed better than the LSTM algorithm in predicting the river flow using precipitation.

Table 7 represents calculation of RMSE and sMAPE based on average value of river flow (Hangang, Yeoju, Gangchung). Through Table 7, we can compare the prediction performance of PNP and LSTM model in case of natural disasters (flood) and non-natural disasters (daily life).

As a result of the measurement, the prediction performance of river flow of the PNP model was higher than LSTM model, except for cases, the average value of the river flow at the Hangang Bridge below.

For sMAPE, the size of river is larger; we can confirm that the prediction performance of PNP model is relatively higher than LSTM model when the river flow is below the average value.

This means that as the size of a river is larger, there are more tributaries. The more tributaries mean that there are more areas to consider precipitation. However, in this paper, we used one precipitation as input, so the error of the PNP algorithm was measured to be larger than the LSTM algorithm.

Table 8 shows sMAPE according to river flow range; we can judge the total prediction performance according to river flow range through Table 8. When the river flow is above the average value, the PNP model showed 10% lower error than the LSTM model. Also, when the river flow is below the average value, the PNP model showed a little higher error than LSTM model, about 3%. From this, we can see that when the river flow is above the average value, the prediction performance of PNP model is excellent compared to LSTM model.

Figures 15, 16, and 17 show the prediction of river flow of the PNP&LSTM algorithm for the Hangang Bridge, Yeoju Bridge, and Gangchung Bridge, respectively.

We confirmed the overall high predictive performance of models using the PNP algorithm. Figure 18 shows a predictive error comparison graph of the two models where (a) shows the RMSE and (b) shows the sMAPE. To verify the performance of the model, we measured the prediction errors in both models: for the Hangang Bridge, the RMSE and sMAPE were 57.927 and 2.565, respectively; for the Yeoju Bridge, the RMSE and sMAPE were 46.457 and 3.031, respectively; and for the Gangchung Bridge, the RMSE and sMAPE were 19.333 and 8.901, respectively.

Table 7 Prediction errors according to river flow range

Location	River flow range	K-fold	RMSE(m^3/sec)		sMAPE(%)		
			PNP	LSTM	PNP	LSTM	
Hangang Bridge	<i>Mean</i> \geq	E_1	1442.929	1869.447	28.906	38.985	
		E_2	795.508	1187.133	32.151	44.240	
		E_3	726.806	1399.42	22.525	43.105	
		E_4	797.455	883.390	34.828	41.474	
		E_5	1220.295	1429.923	40.400	44.812	
		E_{mean}	996.599	1353.863	31.762	42.523	
		<i>Mean</i> $<$	E_1	173.452	151.745	24.415	23.625
	E_2		101.022	134.133	25.317	27.002	
	E_3		236.267	228.450	36.873	35.480	
	E_4		170.869	147.874	43.269	23.330	
	E_5		147.032	131.295	31.813	23.437	
	E_{mean}		165.728	158.699	32.337	26.575	
	Yeosu Bridge		<i>Mean</i> \geq	E_1	524.877	750.349	30.673
		E_2		422.226	593.469	31.460	46.572
E_3		308.436		364.001	31.600	42.450	
E_4		406.436		446.575	37.298	43.233	
E_5		406.679		525.835	33.975	43.010	
E_{mean}		413.731		536.046	33.001	43.648	
<i>Mean</i> $<$		E_1		85.237	88.685	25.022	24.004
		E_2	82.093	74.416	30.396	30.332	
		E_3	78.183	118.133	26.486	33.390	
		E_4	62.530	74.776	23.657	25.133	
		E_5	61.367	64.909	19.978	18.995	
		E_{mean}	73.882	84.184	25.108	26.371	
		Gangchung Bridge	<i>Mean</i> \geq	E_1	44.472	58.125	26.485
E_2				21.7613	46.350	19.280	27.582
E_3	28.391			45.431	19.797	29.168	
E_4	30.218			44.451	18.202	28.947	
E_5	50.568			72.643	29.063	38.059	
E_{mean}	35.082			53.400	22.565	31.590	
<i>Mean</i> $<$	E_1			24.986	42.542	35.960	45.133
	E_2		38.6633	52.761	36.433	52.267	
	E_3		25.150	48.950	28.937	43.648	
	E_4		29.193	48.911	33.143	45.043	
	E_5		21.786	48.263	30.125	45.093	
	E_{mean}		27.956	48.285	32.920	46.237	

Bold values indicate the emphasizes that the error values (RMSE, sMAPE) for both the PNP and LSTM algorithms are lower compared to the mean value of Location

6 Prediction of river flow

In this section, we constructed a prediction model of the river flow using temperature, humidity, and upstream dam hydrological data as well as precipitation. We composed

two models to compare the performance, as shown in Fig. 19, including the PNP (a) and LSTM (b) algorithms.

Model (a) in Fig. 19 uses the PNP algorithm, which consists of three input layers: precipitation, weather (excluding precipitation), and upstream dam hydrological data. The PNP uses the hidden layer connected from the

Table 8 sMAPE according to river flow range

Error	Location	K-fold	River flow range			
			<i>Mean</i> ≥		<i>Mean</i> <	
			PNP	LSTM	PNP	LSTM
sMAPE (%)	Hangang bridge	E_1	28.906	38.985	24.415	23.625
		E_2	32.151	44.240	25.317	27.002
		E_3	22.525	43.105	36.873	35.480
		E_4	34.828	41.474	43.269	23.330
		E_5	40.400	44.812	31.813	23.437
		E_{mean}	31.762	42.523	32.337	26.575
	Yeoju bridge	E_1	30.673	42.975	25.022	24.004
		E_2	31.460	46.572	30.396	30.332
		E_3	31.600	42.450	26.486	33.390
		E_4	37.298	43.233	23.657	25.133
		E_5	33.975	43.010	19.978	18.995
		E_{mean}	33.001	43.648	25.108	26.371
	Gangchung bridge	E_1	26.485	34.192	35.960	45.133
		E_2	19.280	27.582	36.433	52.267
		E_3	19.797	29.168	28.937	43.648
		E_4	18.202	28.947	33.143	45.043
		E_5	29.063	38.059	30.125	45.093
		E_{mean}	22.565	31.590	32.920	46.237
	Total		29.110	39.254	30.122	33.061

input precipitation layer, while the LSTM uses the hidden layer connected to the weather and upstream dam hydrological data input layers. Because the PNP algorithm is based on the correlation between precipitation and river flow, we do not use the PNP algorithm for the hidden layers of weather and upstream dam hydrological data. There are 10 dams upstream of the Hangang Bridge and three upstream of the Yeoju Bridge. There is no upstream dam input layer for the Gangchung Bridge because it has no upstream dam. We used the PNP algorithm for the hidden precipitation layers, while we used the LSTM algorithm for the hidden layers of the remaining weather (excluding precipitation) and operational information regarding the upstream dams. Finally, we organized a deep learning model with two hidden MLP layers and one output layer. The model in Fig. 19b used the LSTM algorithm rather than PNP, and the remainder of the deep learning configuration was identical to the model in Fig. 19a. We divided the data into a 60:20:20 ratio for training, verification, and test data. Figure 20 shows the flow data from the Hangang, Yeoju, and Gangchung Bridges into training, verification, and test data. Table 9 represents the output shape of the deep learning layer of the models in Fig. 1a and b.

Table 10 shows the model’s prediction error, and Figs. 21 and 22 show the model’s prediction and flow measurement.

As a result of the experiment, it can be determined that both the PNP model and the LSTM model have p -value < 0.05, which means that the prediction of both models is significant. And we concluded that the PNP model performs better than LSTM, according to both RMSE and sMAPE. Figure 23 shows a predictive error comparison graph of the two models, where (a) shows the RMSE and (b) shows the sMAPE. To verify the performance of the model, we measured the river flow prediction performance by constructing a model using the LSTM algorithm and measured the prediction errors in the two models: for the Hangang Bridge, the RMSE and sMAPE were 18.352 and 0.13, respectively; for the Yeoju Bridge, the RMSE and sMAPE were 6.125 and 0.628, respectively, and for the Gangchung Bridge, the RMSE and sMAPE were 0.382 and 0.035, respectively. We confirmed that overall, the prediction error of the model using the PNP algorithm was low.

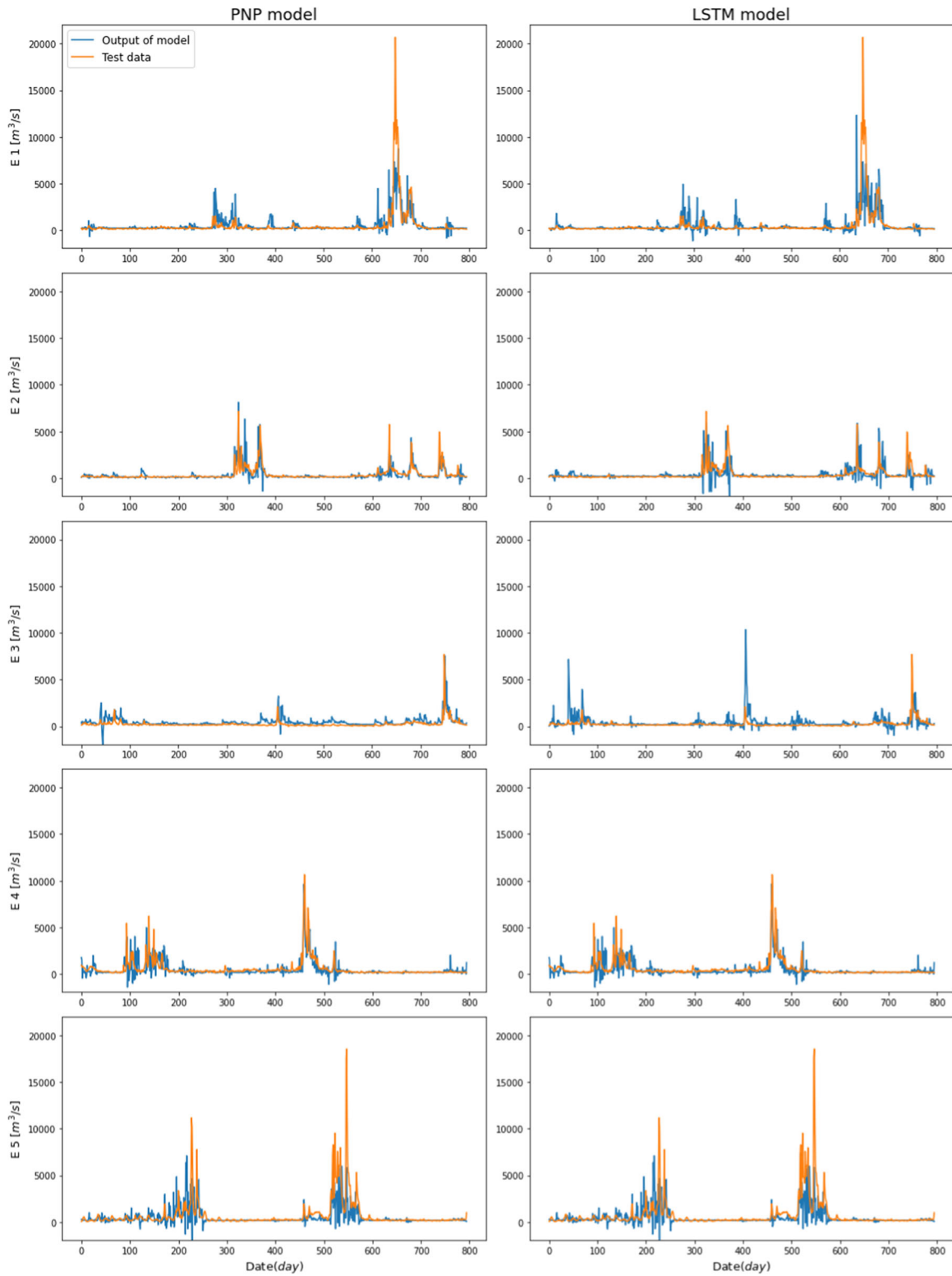


Fig. 15 Prediction of deep learning algorithms (PNP & LSTM) and test data of the Hangang Bridge

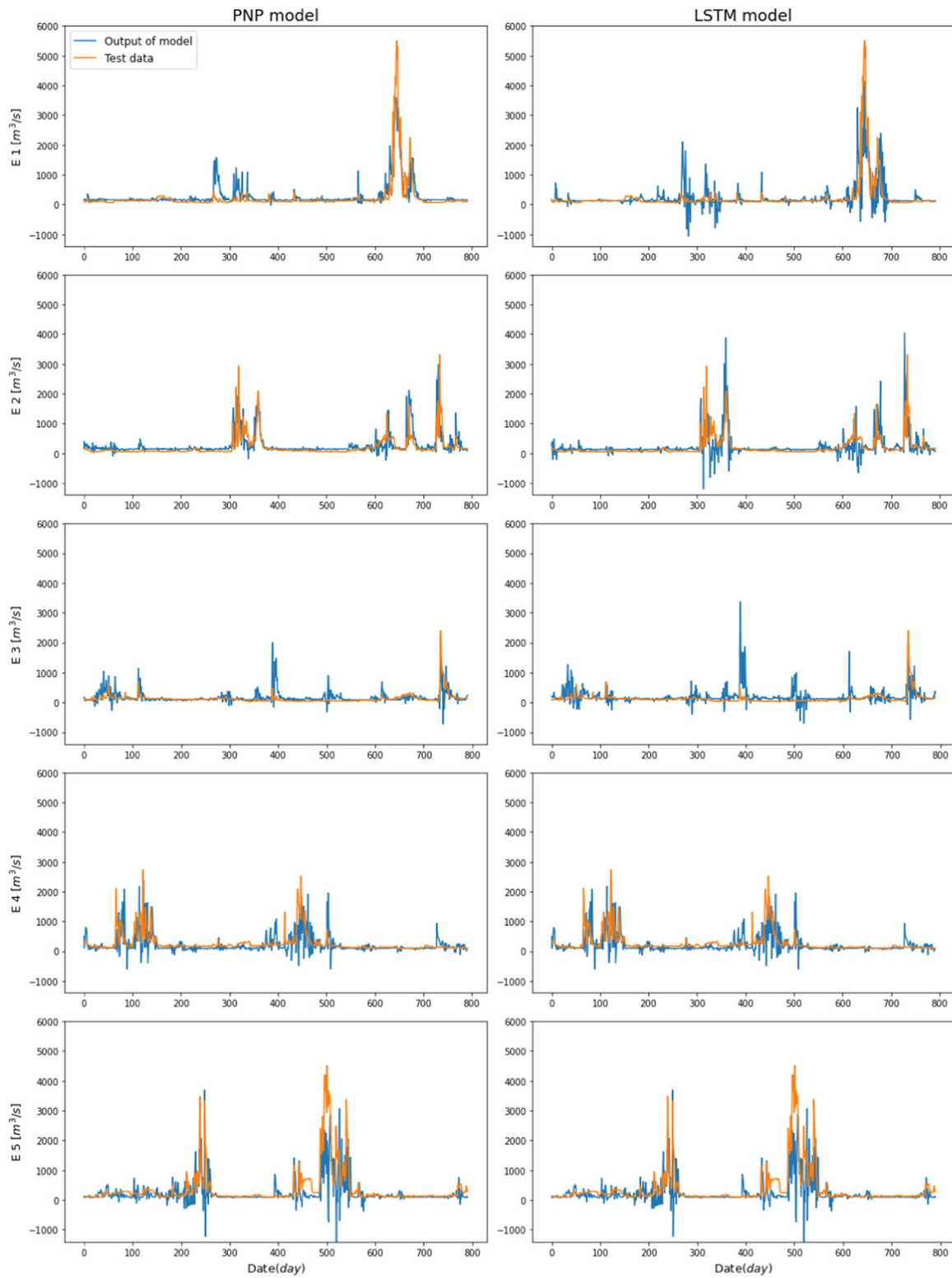


Fig. 16 Prediction of deep learning algorithm (PNP & LSTM) and test data of the Yeosu Bridge

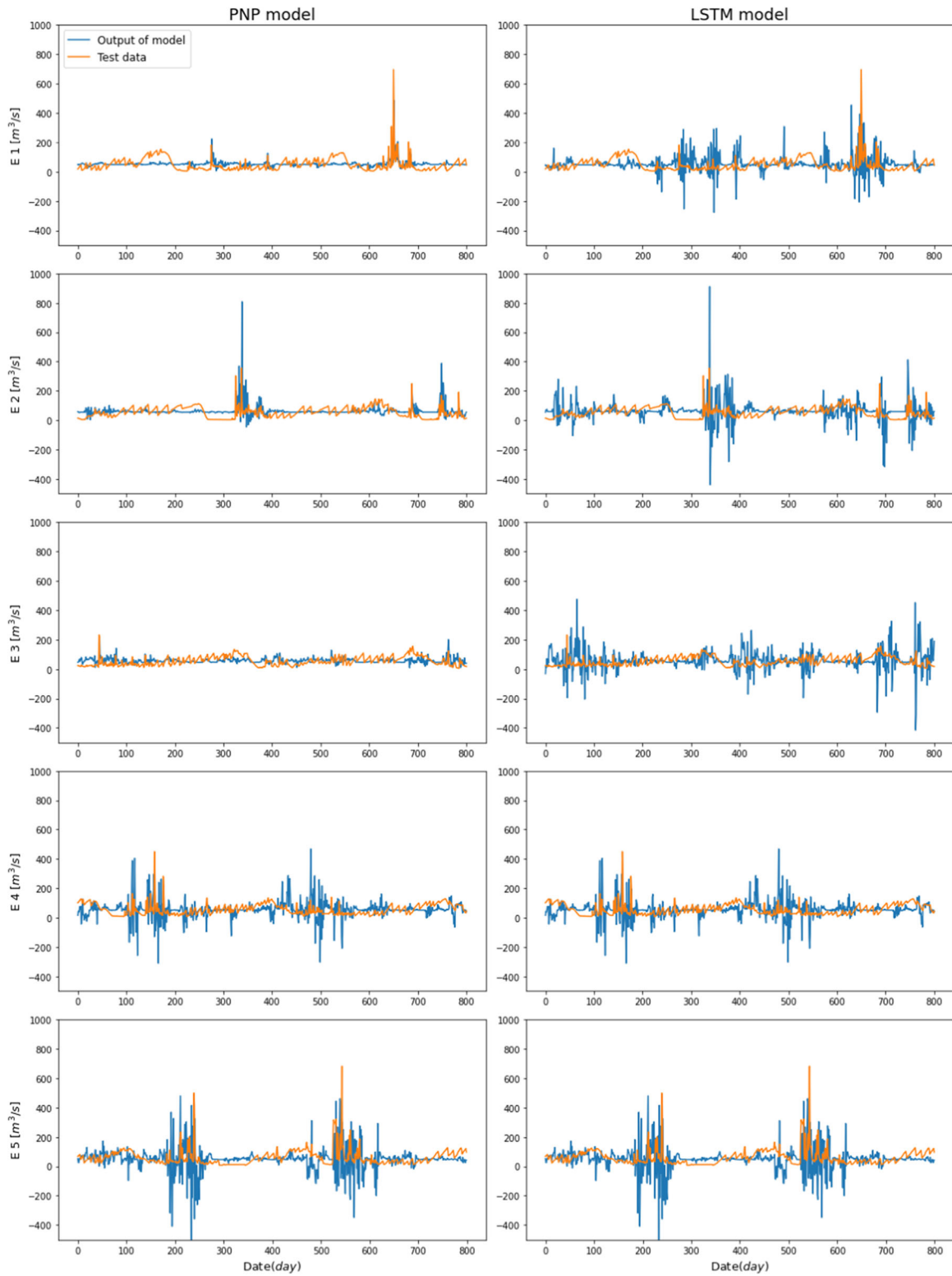
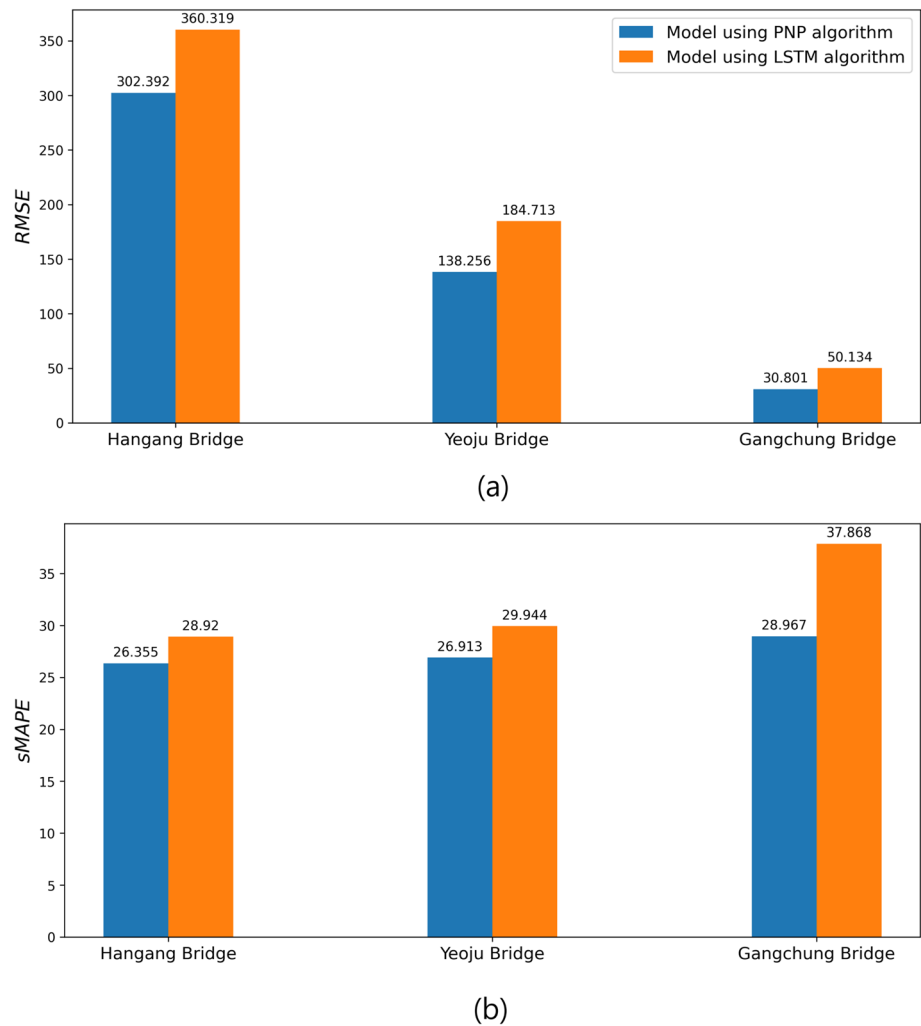


Fig. 17 Prediction of deep learning algorithm (PNP & LSTM) and test data of the Gangchung Bridge

Fig. 18 Results of RMSE and sMAPE Errors



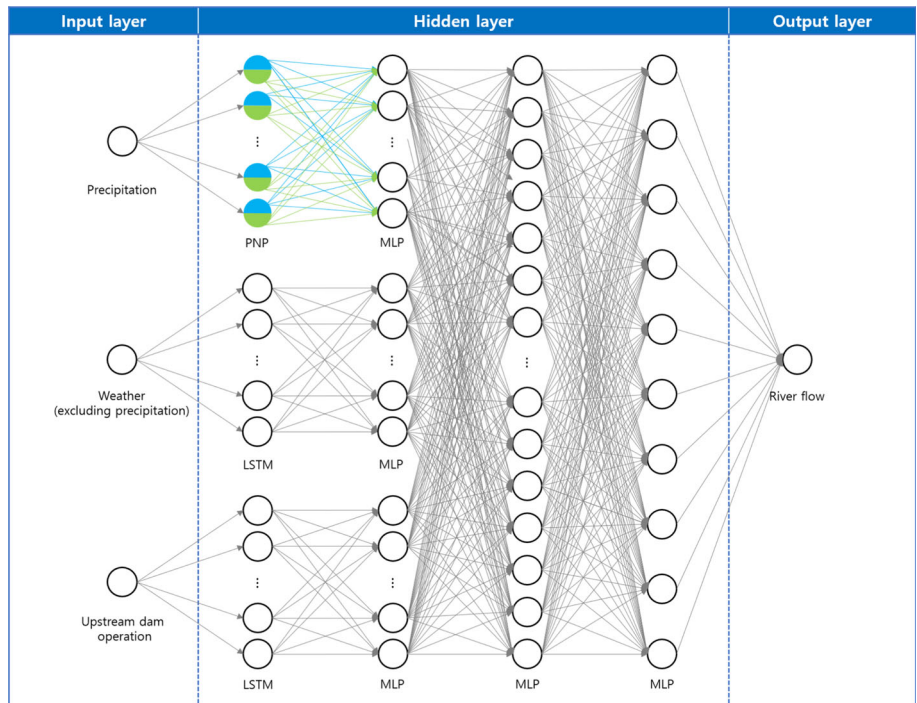
7 Conclusion

This paper presented a PNP algorithm based on an understanding of hydrology to predict river flow. We developed the PNP algorithm considering precipitation with the river flow, the length of water residence time, and other variables. The PNP algorithm consists of positive and negative neurons that consider the water circulatory system. We applied a conveyor belt, which uses past data to consider the residence time in the river. The input value of the PNP algorithm is entered in three dimensions, as with LSTM, and the dimensions of the input data are the data size, variable, and time step. We acquired input and output data to train the deep learning model. The output is the river flow data measured at the Hangang Bridge, Yeosu Bridge, and Gangchung Bridge, and the input data use the weather data from the near river flow measurement point

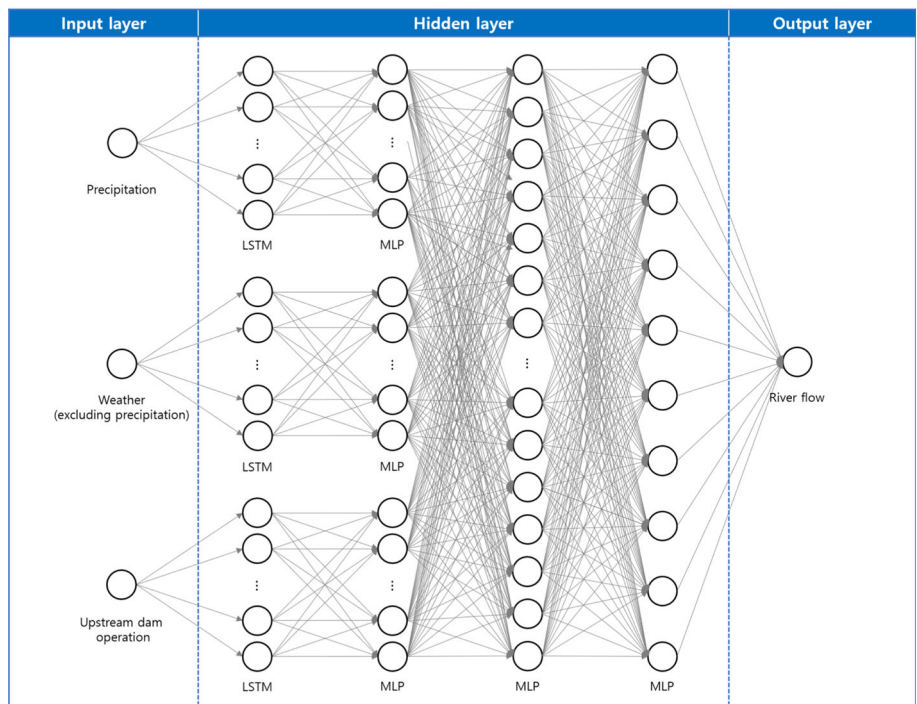
and the operational data of upstream dams. We also used K-fold cross-validation to measure the model performance accurately. We constructed two models to verify the performance of the PNP algorithm. One was a model composed of PNP algorithms, and the other was a model composed of LSTM algorithms. We compared the average values by measuring the performance of the models with RMSE and sMAPE with five datasets generated by K-fold cross-validation.

We measured and compared RMSE and sMAPE based on the average value of river flow for three sectors (Hangang Bridge, Yeosu Bridge, Gangchung Bridge). As a compared result, we can see that the prediction performance of river flow for PNP model is superior to LSTM model except for cases, the average value of the river flow at the Hangang Bridge below. The greater the river flow, the more tributaries, and the more tributaries means that

Fig. 19 Composition of the two models



(a)



(b)

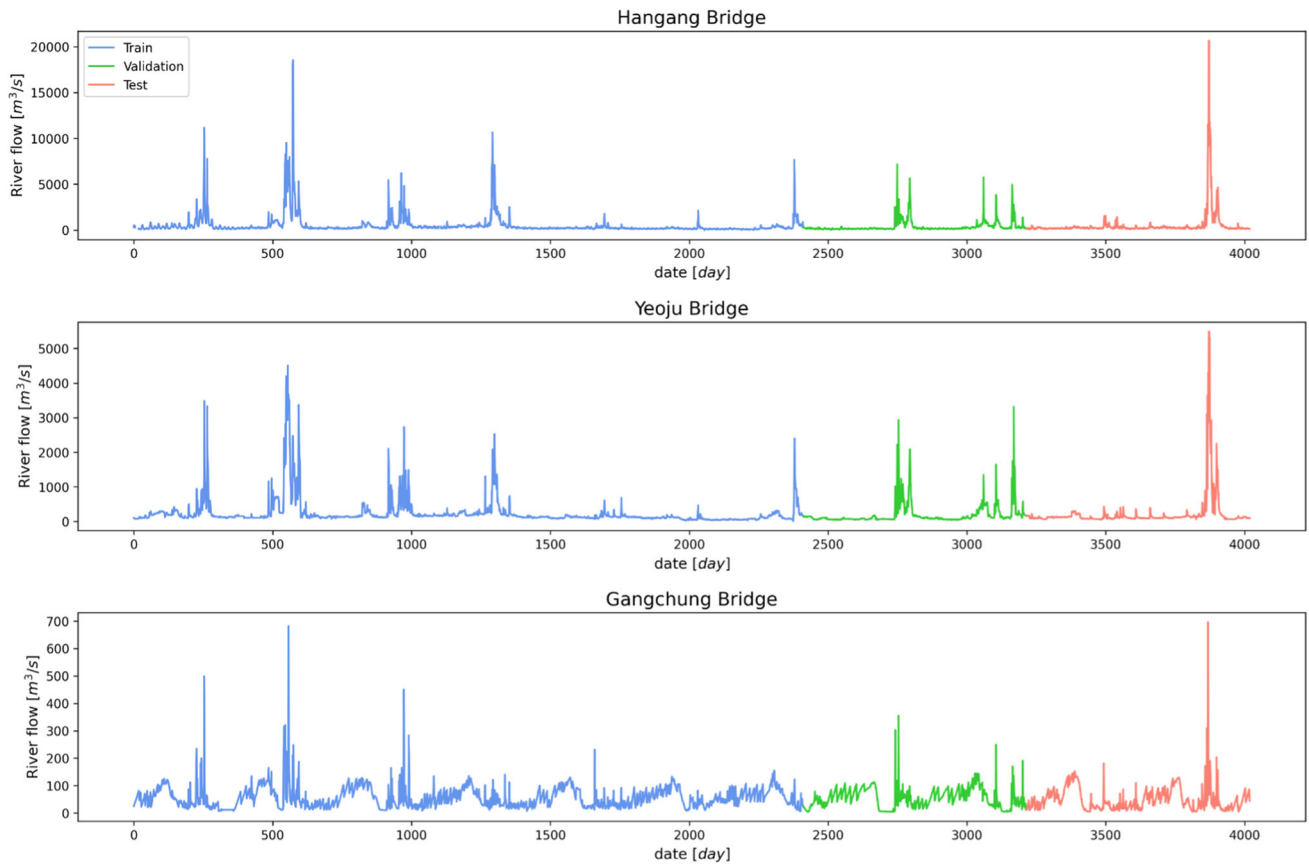


Fig. 20 Training, verification, and testing precipitation data

Table 9 Type of input data and output shape of the model

Location	Input layer	Variable of input data	Data shape
Hangang Bridge	Precipitation	Precipitation	(4004, 1, 14)
	Weather	Temperature, humidity	(4004, 2, 14)
	Upstream dam operation	Paldang dam, Cheongpyeong, and others	(4004, 10, 14)
Yeosu Bridge	Precipitation	precipitation	(4004, 1, 14)
	Weather	Temperature, humidity	(4004, 2, 14)
	Upstream dam operation	Geosan dam, Chungju dam, Hoengseong dam	(4004, 3, 14)
Gangchung Bridge	Precipitation	Precipitation	(4004, 1, 14)
	Weather	Temperature, humidity	(4004, 2, 14)

Table 10 Prediction error of the model

Location	RMSE(m^3/sec)		sMAPE(%)		<i>p</i> -value	
	PNP	LSTM	PNP	LSTM	PNP	LSTM
Hangang Bridge	198.106	216.458	22.121	22.251	3.25×10^{-295}	3.61×10^{-180}
Yeosu Bridge	87.340	93.465	16.173	16.801	4.11×10^{-221}	3.02×10^{-190}
Gangchung Bridge	27.888	28.270	31.504	31.539	4.45×10^{-55}	1.67×10^{-41}

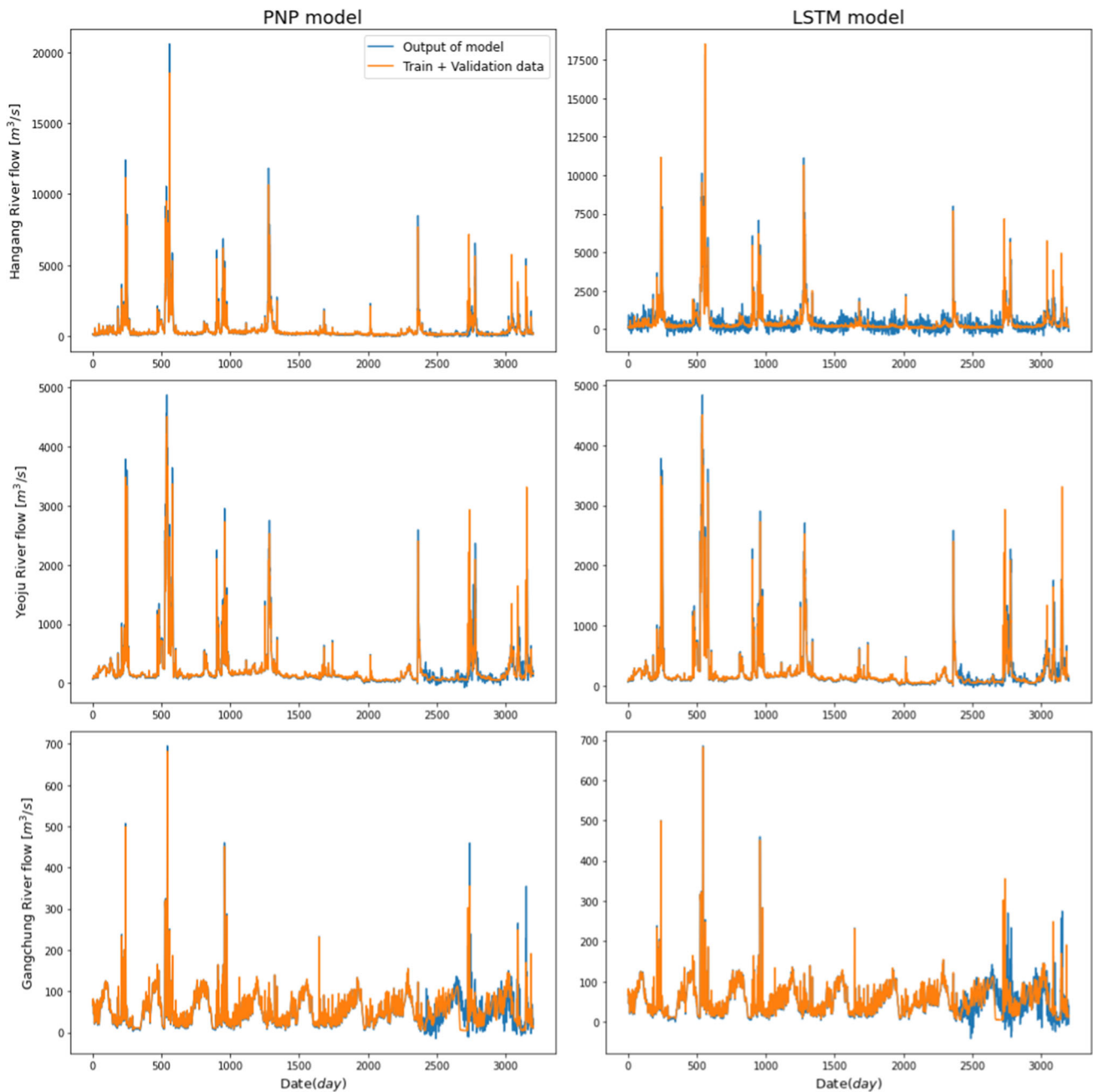


Fig. 21 Prediction and observation values of the training and validation data

there are more areas to consider the amount of precipitation. However, since one amount of precipitation is input in this study, the error of the PNP algorithm is judged to be larger than that of the LSTM algorithm.

When the river flow is less than the average value, we confirmed the prediction of errors for large scale of river has a little higher error than LSTM model. Also, when we

compared the total item by using sMAPE, we confirmed the PNP model has a 10% lower error than the LSTM model when the river flow is above the average value. In addition, when the river flow is below the average value, the PNP model showed a little higher error than LSTM model, about 3%. Through this, we confirmed that the PNP model has better prediction performance than other deep

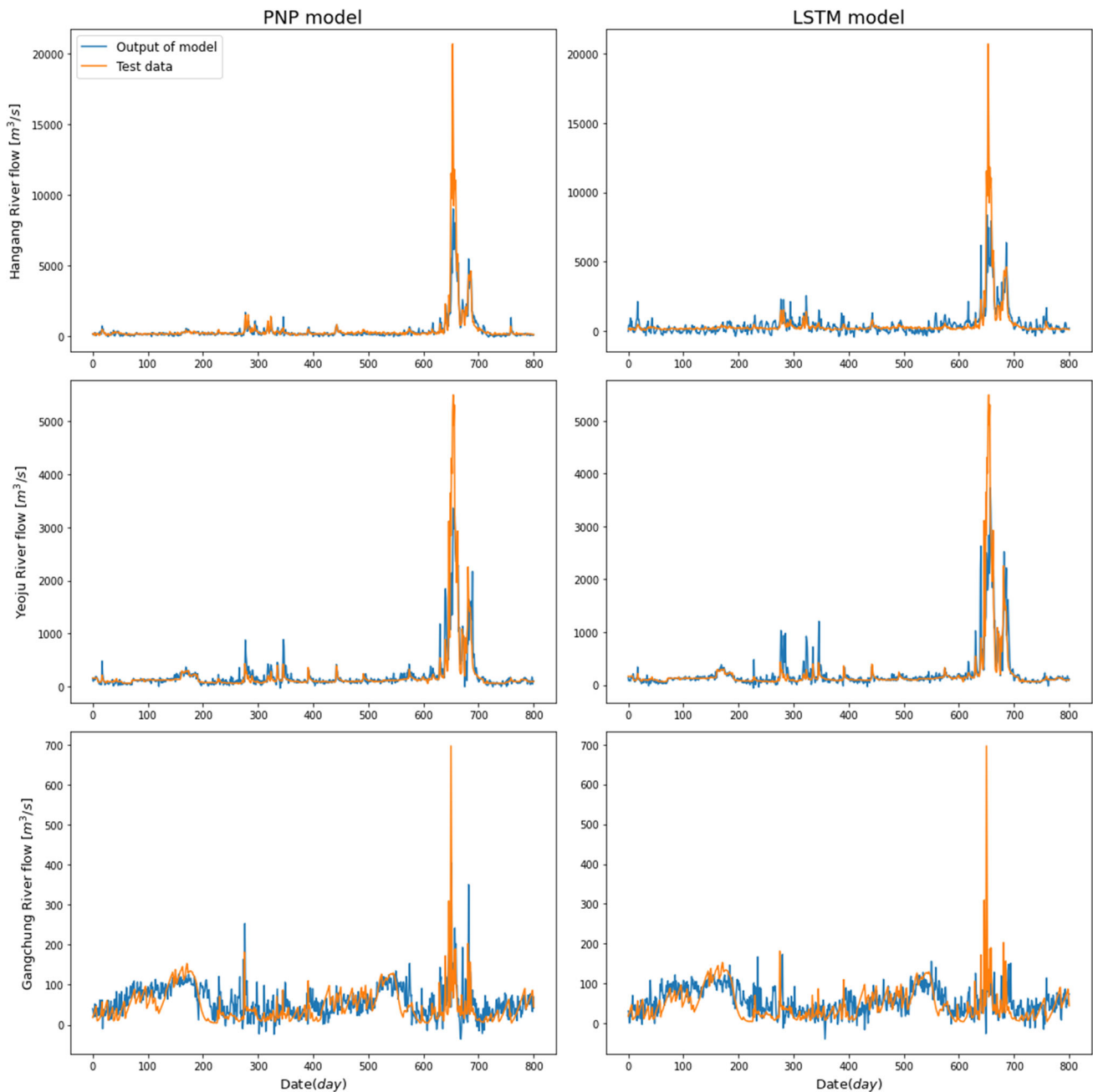


Fig. 22 Prediction and observation values of the test data

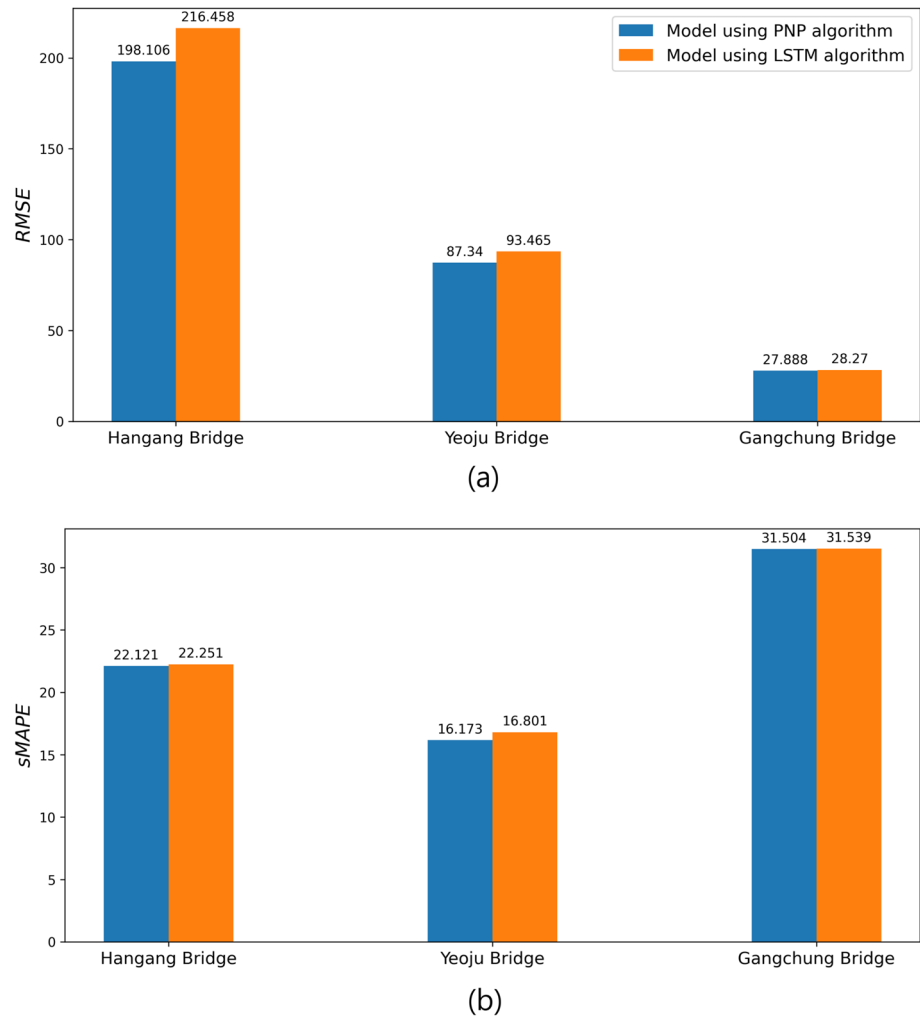
learning model such as LSTM. Thus, we confirmed that the prediction performance is higher when the river flow is high (natural disaster). Through this paper, we proved that the performance of the PNP algorithm, which is designed for the correlation between precipitation and flow, was excellent compare to LSTM.

Partially, the PNP algorithm sometimes performs worse than the LSTM algorithm because it only calculates the precipitation in the target area. In order to solve this problem, it is necessary to consider not only the

precipitation data at the flow prediction point but also the precipitation data at the upper point of the river.

In the future, to upgrade the PNP algorithm, we need to design it by considering the data upstream of the river as well. In addition, we need to develop a deep learning algorithm designed to correlate not only precipitation, but also humidity, upstream dam operation, etc. like the PNP algorithm.

Fig. 23 Results of models using PNP and LSTM algorithms



Author contributions GB and YB conceived and designed the conceptualization, GB performed the computer simulation, and Y.B. analyzed the data and algorithms; GB wrote the paper. All authors have given approval to the final version of the manuscript.

Funding This research was supported by Chonnam National University(Smart Plant Reliability Center) grant funded by the Ministry of Education.(2020R1A6C101B197) and "Regional Innovation Strategy (RIS)" through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(MOE) (Grant number: 2022RIS-002).

Data availability Enquiries about data availability should be directed to the authors.

Declarations

Conflict of Interest The authors certify that there is no conflict of interest with any individual or organization for the present work.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Abidin HZ, Djaja R, Darmawan D, Hadi S, Akbar A, Rajiyowiryono H, Subarya C (2001) Land subsidence of Jakarta (Indonesia) and its geodetic monitoring system. *Nat Hazards* 23(2):365–387
- Asgari Taghanaki S, Abhishek K, Cohen JP, Cohen-Adad J, Hamarneh G (2021) Deep semantic segmentation of natural and medical images: a review. *Artif Intell Rev* 54(1):137–178

- Bai S, Kolter JZ, Koltun V (2018) An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. arXiv preprint [arXiv:1803.01271](https://arxiv.org/abs/1803.01271)
- Bordes A, Glorot X, Weston J, Bengio Y (2012) Joint learning of words and meaning representations for open-text semantic parsing. In: Artificial intelligence and statistics. PMLR, 127–135
- Borovykh A, Bohte S, Oosterlee CW (2017) Conditional time series forecasting with convolutional neural networks. arXiv preprint [arXiv:1703.04691](https://arxiv.org/abs/1703.04691)
- Chung J, Gulcehre C, Cho K, Bengio Y (2014) Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint [arXiv:1412.3555](https://arxiv.org/abs/1412.3555).
- Ciregan D, Meier U, Schmidhuber J (2012) Multi-column deep neural networks for image classification. In: 2012 IEEE conference on computer vision and pattern recognition. 3642–3649.
- Cireşan DC, Meier U, Schmidhuber J (2012) Transfer learning for Latin and Chinese characters with deep neural networks. In: The 2012 international joint conference on neural networks (IJCNN). IEEE, 1–6
- Debbarma S, Choudhury P (2020) River flow prediction with memory-based artificial neural networks: a case study of the Dholai river basin. *Int J Adv Intell Paradig* 15(1):51–62
- Dodangeh E, Choubin B, Eigdir AN, Nabipour N, Panahi M, Shamshirband S, Mosavi A (2020) Integrated machine learning methods with resampling algorithms for flood susceptibility prediction. *Sci Total Environ* 705:135983
- Elman JL (1990) Finding structure in time. *Cogn Sci* 14(2):179–211
- Fathian F, Mehdizadeh S, Sales AK, Safari MJS (2019) Hybrid models to improve the monthly river flow prediction: integrating artificial intelligence and non-linear time series models. *J Hydrol* 575:1200–1213
- Field CB, Barros V, Stocker TF, Dahe Q (2012) Managing the risks of extreme events and disasters to advance climate change adaptation: special report of the intergovernmental panel on climate change. Cambridge University Press, New York
- Fowler AM, Hennessy KJ (1995) Potential impacts of global warming on the frequency and magnitude of heavy precipitation. *Nat Hazards* 11(3):283–303
- Gharaei RH, Sharify R, Nezamabadi-Pour H (2022) An efficient outlier detection method based on distance ratio of k-nearest neighbors. In: 2022 9th Iranian Joint Congress on Fuzzy and Intelligent Systems (CFIS), IEEE, 1–5
- Ghimire S, Yaseen ZM, Farooque AA, Deo RC, Zhang J, Tao X (2021) Streamflow prediction using an integrated methodology based on convolutional neural network and long short-term memory networks. *Sci Rep* 11(1):1–26
- Hadsell R, Erkan A, Sermanet P, Scoffier M, Muller U, LeCun Y (2008) Deep belief net learning in a long-range vision system for autonomous off-road driving. In 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE, 628–633
- Hendrycks D, Mazeika M, Dietterich T (2018) Deep anomaly detection with outlier exposure. arXiv preprint [arXiv:1812.04606](https://arxiv.org/abs/1812.04606)
- Hochreiter S, Schmidhuber J (1997) Long short-term memory. *Neural Comput* 9(8):1735–1780
- Hoegh-Guldberg O, Jacob D, Bindi M, Brown S, Camilloni I, Diedhiou A, Zougmore RB (2018) Impacts of 1.5 °C global warming on natural and human systems. *Glob Warm* 1.5 °C.
- Huang X, Li Y, Tian Z, Ye Q, Ke Q, Fan D, Liu J (2021) Evaluation of short-term streamflow prediction methods in urban river basins. *Phys Chem Earth Parts A/B/C* 123:103027
- Hubel DH, Wiesel TN (1959) Receptive fields of single neurones in the cat's striate cortex. *J Physiol* 148(3):574
- Hubel DH, Wiesel TN (1962) Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *J Physiol* 160(1):106
- Jaganathan K, Panagiotopoulou SK, McRae JF, Darbandi SF, Knowles D, Li YI, Farh KKH (2019) Predicting splicing from primary sequence with deep learning. *Cell* 176(3):535–548
- Kim D, Lee J, Kim J, Lee M, Wang W, Kim HS (2022) Comparative analysis of long short-term memory and storage function model for flood water level forecasting of Bokha stream in NamHan River, Korea. *J Hydrol* 606:127415
- Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. *Adv Neural Inform Process Syst* 25:1–9
- Lauer S, Sanderson MR, Manning DT, Suter JF, Hrozencik RA, Guerrero B, Golden B (2018) Values and groundwater management in the Ogallala Aquifer region. *J Soil Water Conserv* 73(5):593–600
- Le XH, Ho HV, Lee G, Jung S (2019) Application of long short-term memory (LSTM) neural network for flood forecasting. *Water* 11(7):1387
- Li Y, Yu R, Shahabi C, Liu Y (2017) Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. arXiv preprint [arXiv:1707.01926](https://arxiv.org/abs/1707.01926).
- Lim B, Arik SÖ, Loeff N, Pfister T (2021) Temporal fusion transformers for interpretable multi-horizon time series forecasting. *Int J Forecast* 37(4):1748–1764
- Malyan SK, Singh R, Rawat M, Kumar M, Pugazhendhi A, Kumar A, Kumar SS (2019) An overview of carcinogenic pollutants in groundwater of India. *Biocatal Agric Biotechnol* 21:101288
- Mehedi MAA, Khosravi M, Yazdan MMS, Shabani H (2022) Exploring temporal dynamics of river discharge using univariate long short-term memory (LSTM) recurrent neural network at east branch of Delaware River. *Hydrology* 9(11):202
- Mikolov T, Sutskever I, Chen K, Corrado GS, Dean J (2013) Distributed representations of words and phrases and their compositionality. *Adv Neural Inform Process Syst* 26:1–9
- Mudassir M, Bennbaia S, Unal D, Hammoudeh M (2020) Time-series forecasting of Bitcoin prices using high-dimensional features: a machine learning approach. *Neural Comput Appl* 1–15.
- Mudelsee M (2019) Trend analysis of climate time series: a review of methods. *Earth Sci Rev* 190:310–322
- Musarat MA, Alaloul WS, Rabbani MBA, Ali M, Altaf M, Fediuk R, Farooq W (2021) Kabul river flow prediction using automated ARIMA forecasting: a machine learning approach. *Sustainability* 13(19):10720
- Myhre G, Alterskjær K, Stjern CW, Hodnebrog Ø, Marelle L, Samset BH, Stohl A (2019) Frequency of extreme precipitation increases extensively with event rareness under global warming. *Sci. Rep.* 9(1):1–10
- Oord AVD, Dieleman S, Zen H, Simonyan K, Vinyals O, Graves A, Kavukcuoglu K (2016) Wavenet: a generative model for raw audio. arXiv preprint [arXiv:1609.03499](https://arxiv.org/abs/1609.03499).
- Palmer TN, Räisänen J (2002) Quantifying the risk of extreme seasonal precipitation events in a changing climate. *Nature* 415(6871):512–514
- Pang G, Shen C, Cao L, Hengel AVD (2021) Deep learning for anomaly detection: a review. *ACM Comput Surv (CSUR)* 54(2):1–38
- Papalexioiu SM, Montanari A (2019) Global and regional increase of precipitation extremes under global warming. *Water Resour Res* 55(6):4901–4914

- Quang D, Chen Y, Xie X (2015) DANN: a deep learning approach for annotating the pathogenicity of genetic variants. *Bioinformatics* 31(5):761–763
- Ren J, Xu L (2015) On vectorization of deep convolutional neural networks for vision tasks. *Proc AAAI Conf Artif Intell* 29(1):1840–1846
- Rosenblatt F (1958) The perceptron: a probabilistic model for information storage and organization in the brain. *Psychol Rev* 65(6):386
- Salinas D, Flunkert V, Gasthaus J, Januschowski T (2020) DeepAR: probabilistic forecasting with autoregressive recurrent networks. *Int J Forecast* 36(3):1181–1191
- Schuster M, Paliwal KK (1997) Bidirectional recurrent neural networks. *IEEE Trans Signal Process* 45(11):2673–2681
- Senent-Aparicio J, Jimeno-Sáez P, Bueno-Crespo A, Pérez-Sánchez J, Pulido-Velázquez D (2019) Coupling machine-learning techniques with SWAT model for instantaneous peak flow prediction. *Biosys Eng* 177:67–77
- Shankar PV, Kulkarni H, Krishnan S (2011) India's groundwater challenge and the way forward. *Econ Polit Week* 37–45
- Tunyasuvunakool K, Adler J, Wu Z, Green T, Zielinski M, Židek A, Hassabis D (2021) Highly accurate protein structure prediction for the human proteome. *Nature* 596(7873):590–596
- Wang H, Cimen E, Singh N, Buckler E (2020) Deep learning for plant genomics and crop improvement. *Curr Opin Plant Biol* 54:34–41

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.