Article

Prediction of River Stage Using Multistep-Ahead Machine Learning Techniques for a Tidal River of Taiwan

Wen-Dar Guo ^{1,*}, Wei-Bo Chen ¹ , Sen-Hai Yeh ¹, Chih-Hsin Chang ¹ and Hongey Chen ^{1,2}

¹ National Science and Technology Center for Disaster Reduction, New Taipei City 23143, Taiwan; wbchen@ncdr.nat.gov.tw (W.-B.C.); shyeh@ncdr.nat.gov.tw (S.-H.Y.); chang.c.h@ncdr.nat.gov.tw (C.-H.C.); hchen@ntu.edu.tw (H.C.)

² Department of Geosciences, National Taiwan University, Taipei City 10617, Taiwan

* Correspondence: wdguo@ncdr.nat.gov.tw; Tel.: +886-2-81958600

Abstract: Time-series prediction of a river stage during typhoons or storms is essential for flood control or flood disaster prevention. Data-driven models using machine learning (ML) techniques have become an attractive and effective approach to modeling and analyzing river stage dynamics. However, relatively new ML techniques, such as the light gradient boosting machine regression (LGBMR), have rarely been applied to predict the river stage in a tidal river. In this study, data-driven ML models were developed under a multistep-ahead prediction framework and evaluated for river stage modeling. Four ML techniques, namely support vector regression (SVR), random forest regression (RFR), multilayer perceptron regression (MLPR), and LGBMR, were employed to establish data-driven ML models with Bayesian optimization. The models were applied to simulate river stage hydrographs of the tidal reach of the Lan-Yang River Basin in Northeastern Taiwan. Historical measurements of rainfall, river stages, and tidal levels were collected from 2004 to 2017 and used for training and validation of the four models. Four scenarios were used to investigate the effect of the combinations of input variables on river stage predictions. The results indicated that (1) the tidal level at a previous stage significantly affected the prediction results; (2) the LGBMR model achieves more favorable prediction performance than the SVR, RFR, and MLPR models; and (3) the LGBMR model could efficiently and accurately predict the 1–6-h river stage in the tidal river. This study provides an extensive and insightful comparison of four data-driven ML models for river stage forecasting that can be helpful for model selection and flood mitigation.

Keywords: river stage; data driven; machine learning; light gradient boosting; multistep ahead; Bayesian optimization

Citation: Guo, W.-D.; Chen, W.-B.; Yeh, S.-H.; Chang, C.-H.; Chen, H. Prediction of River Stage Using Multistep-Ahead Machine Learning Techniques for a Tidal River of Taiwan. *Water* **2021**, *13*, 920. <https://doi.org/10.3390/w13070920>

Academic Editor: Achim A. Beylich

Received: 5 March 2021

Accepted: 23 March 2021

Published: 27 March 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Accurate river stage forecasting is a crucial component in the flood early warning system and plays a key role in flood disaster mitigation. Taiwan had an average of four to five typhoons per year over the past 10 years [1]. Typhoon-induced floods can frequently cause considerable social and economic losses. For example, Typhoon Morakot hit Taiwan in 2009, resulting in a torrential rainfall of 2748 mm in only 72 h [2]. Such an extreme rainfall caused compound hazards, such as floods, river overflows, landslides, river embankment failures, and driftwood accumulation. Typhoon Morakot caused approximately 680 casualties and approximately NT\$90 billion in expenses for direct damage [3]. Since Typhoon Morakot, more intensive investigations, analyses, and developments for disaster prevention have been conducted to better understand disaster risk assessment. The flood warning system is a vital mitigation technique during natural disasters that can be used by river managers to make decisions before the arrival of a typhoon. Therefore, studies on accurate and reliable river stage forecasting are required to reduce the impact of flood disasters.

Two main approaches are used to establish flood prediction models. The first approach involves the use of a flood dynamic process to perform mathematical modeling. This

approach produces physics-based models, such as the Hydrologic Engineering Centers River Analysis System [4], the SOBEK model developed by Deltares [5], and Watershed Systems of 1D Stream-River Network, 2D Overland Regime, and 3D Subsurface Media [6]. A physics-based model requires cross-sectional bed elevation data or digital elevation model data for establishing the simulation domain; therefore, simulation results obtained using a physics-based model are highly dependent on the quality of topographic survey data [7,8]. In addition, because the parameters in a physics-based model may affect the simulation results, the parameters must be calibrated. An alternative approach is the use of a data-driven model, which is based on the collection and analysis of data [9,10]. Currently, machine learning (ML) techniques, such as artificial neural networks (ANNs), K-nearest neighbors (KNNs), support vector regression (SVR), random forest regression (RFR), and multilayer perceptron regression (MLPR), are some of the most widely used approaches for data-driven models. Compared with a physics-based model, a data-driven model does not require bed elevation data and prevents numerical instability without any additional treatment.

In the last decade, data-driven models based on ML techniques have been proposed and extensively used for hydrology and flood-related predictions, including those related to rainfall runoff, reservoir inflow, river stage, urban inundation, and water quality simulation. For instance, Maity et al. [11] employed SVR to predict the monthly streamflow in the Mahanadi River in India. Chen et al. [12,13] applied ANNs to predict the typhoon-induced storm surge tide and estuarine water stage and compared their results with those obtained using 2D and 3D hydrodynamic models. Lin et al. [14] adopted SVR and the K-means clustering algorithm to develop a regional-inundation forecasting model. In their model, three main processes were adopted: classification, point forecasting, and spatial expansion. Wu et al. [15] proposed an improved streamflow forecasting model based on SVR by using a self-organizing map (SOM) and demonstrated that the proposed model could accurately forecast the hourly streamflow. Furthermore, Hosseini and Mahjouri [16] combined SVR with ANNs for daily rainfall-runoff modeling. They reported that the prediction accuracy was higher when integrated SVR was used than when conventional ANNs were used. Jhong et al. [17] proposed a two-stage approach based on SVR for urban inundation forecasting. Their approach could provide accurate flood maps with lead times of 1–6 h during typhoons. Applying SVR and the genetic algorithm, Seo et al. [18] performed daily river stage modeling in the Chogang Watershed, South Korea. Jhong et al. [19] combined back-propagation networks (BPNs) and SOMs to propose a hybrid neural network model for typhoon flood forecasting. Muñoz et al. [20] proposed a stepwise methodology for rainfall-runoff forecasting in an Andean mountain. In their proposed methodology, RFR was applied for short-term forecasting with different lead times of 4, 8, 12, 18, and 24 h. Wu et al. [21] applied SVR to forecast flash floods in the small catchment Anhe in China. Kim and Han [22] employed SVR and SOMs for predicting inundation maps in the Gangnam District, Seoul, South Korea. Nguyen and Chen [23] developed a probabilistic forecasting model based on SVR, KNN, and a fuzzy inference model, and the developed model was applied to forecast floods with a lead time of 1–3 h. Chen et al. [24] used ANNs to model the dissolved oxygen concentration in a reservoir in Taiwan.

The aforementioned studies have indicated that a data-driven model with ML techniques can effectively learn the nonlinear relationship between input and output variables without requiring explicit knowledge regarding the physical process. In flood forecasting using ML techniques, several factors can affect the prediction accuracy, including the combinations of input vectors, employed parameters, and different ML techniques. Hence, several attempts have been made using different strategies to improve the accuracy. For example, Lin et al. [25] employed SVR and BPNs to forecast hourly reservoir inflows. Their results indicated that SVR was more accurate than BPNs. Nguyen et al. [26] applied the least absolute shrinkage and selection operator, RFR, and SVR to forecast the daily time series of water levels at the Thakhek station of Mekong River. Li et al. [27] compared the performance of RFR, SVR, ANNs, and a linear model for forecasting lake water levels. In

addition, they investigated the effects of previous water levels at different time lags on the forecasting accuracy. They reported that RFR exhibited the most satisfactory performance among the tested models. Furthermore, the combination of input vectors involving the discharge in the previous four days and the average water level in the previous week was robust and accurate for daily forecasting. Panagoulia et al. [28] investigated the nonlinear relationship between river flow and input variables selected using ANNs. Yang et al. [29] employed RFR, SVR, and ANNs to forecast monthly reservoir inflow and found that RFR exhibited the most satisfactory performance. In addition, their results indicated that the optimal input variables were precipitation in the previous three days and river flow in the previous four days. Pini et al. [30] evaluated three ML techniques (ANNs, RFR, and SVR) to forecast stream inflow in Lake Como, Italy. Their results indicated that the streamflow prediction accuracy was higher when ANNs were used than when SVR and RFR were used. Ebrahimi and Shourian [31] employed the particle swarm optimization algorithm to develop a dynamic KNN model for predicting daily river flow in the Gheslagh reservoir in Iran. Compared with the classic KNN, ANNs, RFR, and SVR, their proposed model had a higher prediction accuracy. Maspo et al. [32] systematically reviewed the flood prediction evaluation performance of existing ML techniques. They also identified notable input parameters that can serve as guidelines for flood forecasting.

With the development and improvement of ML techniques, data-driven models are rapidly becoming a key approach for flood mitigation. More recently, a few advanced ML techniques have been proposed. For instance, Chen and Guestrin [33] proposed the extreme gradient boosting (XGBoost) algorithm based on the framework of the gradient boosting decision tree (GBDT) method. Because the XGBoost model is applied in a learning system, it uses a level-wise method to construct a decision tree, resulting in its favorable performance in several fields [34–38]. However, the XGBoost algorithm may exert a negative effect on big data treatment and requires more time during the learning process [39,40]. To reduce the high computational cost, the light gradient boosting machine regression (LGBMR) for time-series forecasting was proposed by Microsoft Research [41]. LGBMR is an ensemble ML technique that uses the new GBDT framework to handle big data with high accuracy. The LGBMR model is a relatively new ML technique that has demonstrated favorable performance in various fields, such as wind turbine operation [39], blood glucose prediction [40], human activity recognition [42], and particulate matter concentration prediction [43]. LGBMR has several advantages; for example, it has high computational efficiency, can prevent the overfitting problem, can make accurate global predictions, and can solve both classification and regression problems. Although some studies have used LGBMR to solve various time-series regression-type problems, few studies have used it for river stage forecasting. Hence, this study applied LGBMR to forecast floods and compared its performance with that of other ML techniques.

The present study developed four data-driven ML models (SVR, RFR, MLPR, and LGBMR models) for direct multistep forecasting; among these models, the LGBMR model is relatively new and has rarely been applied for the prediction of river floods. To determine the relationship between time-series input and output variables, hourly hydrological data measured from 2004 to 2017 at the Lan-Yang River were collected and divided into training and testing datasets. An accurate flood forecasting model should consider significant factors, such as rainfall, river stage, and discharge. However, few studies have considered the effects of the status of the previous tidal stage while forecasting river floods. Hence, to improve the accuracy of flood forecasting in a tidal river, hydrological records, such as rainfall, water level, and tidal stage data, for the previous 1–6 h were used as input vectors for training the constructed model. To achieve optimal inputs, the effects of the different combinations of input variables on the prediction results were examined in this study. On the basis of optimal inputs, optimal parameters were determined through Bayesian optimization and through the use of 10 cross-validation sets in the training phase. After the establishment of the four models, the test dataset was used to predict the river stage

with a lead time of 1–6 h. According to the evaluation criteria, the forecasting performance of the four models was evaluated and compared for both the training and test results.

The primary contributions of this study are summarized as follows:

1. This study contributes to improving forecasting performance by revealing the optimal combinations of input variables, such as rainfall, water level, and tidal stage.
2. This is the first study to propose a direct multistep forecasting model based on LGBMR with Bayesian optimization for flood forecasting with a lead time of 1–6 h.
3. The present study comprehensively assessed and compared the performance of four models (SVR, RFR, MLPR, and LGBMR) for forecasting the water level in a tidal river.

2. Methodology

2.1. Data-Driven Model for River Stage Forecasting

The main process in data-driven modeling is called “the learning stage,” in which the relationship between a system’s input and output variables is constructed [9]:

$$y = f(\mathbf{x}) \quad (1)$$

with available data

$$[(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \dots, (\mathbf{x}_n, y_n)] = \{\mathbf{x}_i, y_i\}_{i=1}^n \quad (2)$$

in which $\mathbf{x}$ is the input vector, y is the desired output, n is the number of data, and f is the regression function.

The general representative approaches for time-series forecasting include direct and recursive multistep forecasting [44]. Compared with the recursive approach, the direct approach is simpler and easier to employ. In addition, the direct approach does not produce any significant prediction errors during the forecasting process. Therefore, the direct approach is employed in Equation (1) for river stage modeling, yielding the equation on which the data-driven model is based:

$$\hat{H}_{t+\Delta t} = f(R_t, R_{t-1}, \dots, R_{t-L}, H_t, H_{t-1}, \dots, H_{t-L}, S_t, S_{t-1}, \dots, S_{t-L}) \quad (3)$$

where t is the current time, Δt is the lead time, $\hat{H}_{t+\Delta t}$ is the forecasted river stage at time $t + \Delta t$, L denotes the lag length of the input variables, R_{t-L} is the antecedent rainfall at time $t - L$, H_{t-L} is the antecedent river stage at time $t - L$, and S_{t-L} is the antecedent tidal level at time $t - L$. Following the approach adopted by Wang et al. [45], the lag length was set as 6 h in the present study; this lag length takes into consideration the concentration time of a watershed. To investigate the lead time, the lead time commonly applied in hydrology modeling of 1–6 h was used in this study [45–47].

In Equation (3), four ML techniques were applied to construct the data-driven models. ML techniques can be used to solve classification or regression problems. This study focused on forecasting water levels, which is a nonlinear regression problem. The regression algorithms of four ML techniques are presented in Sections 2.2–2.5. Figure 1 displays a conceptual diagram of the four ML techniques.

Figure 1. The conceptual diagram for (a) support vector regression (SVR), (b) random forest regression (RFR), (c) multilayer perceptron regression (MLPR), and (d) light gradient boosting machine regression (LGBMR).

2.2. SVR

As indicated in Equation (3), a suitable ML technique for constructing the regression function f is required. The SVR approach proposed by Drucker et al. [48] was employed herein for nonlinear regression. The regression function of SVR can be expressed as follows [46,49]:

$$f^{\text{SVR}}(\mathbf{x}) = \mathbf{w}^T \cdot \phi(\mathbf{x}) + b \quad (4)$$

where $\mathbf{w}$ is the weight vector, ϕ is the nonlinear mapping function, and b is the bias term. According to the fundamental concept of structural risk minimization to prevent overfitting, Equation (4) can be further expressed as follows:

$$\min_{\mathbf{w}, b, \xi, \xi^*} \frac{1}{2} \|\mathbf{w}^2\| + C \sum_{i=1}^n (\xi_i + \xi_i^*) \quad (5)$$

$$\text{subject to } \begin{cases} y_i - [\mathbf{w}^T \cdot \phi(\mathbf{x}_i) + b] \leq \epsilon + \xi_i \\ [\mathbf{w}^T \cdot \phi(\mathbf{x}_i) + b] - y_i \leq \epsilon + \xi_i^* \\ \xi_i \geq 0, \xi_i^* \geq 0, i = 1, 2, \dots, n \end{cases} \quad (6)$$

where C denotes the cost parameter or penalty parameter, ξ and ξ^* are nonnegative slack variables, and ϵ is the parameter of the insensitive loss function. On the basis of Lagrange multipliers, the optimization problem of SVR can be written as a dual pattern [50]:

$$f^{\text{SVR}}(\mathbf{x}) = \sum_{i=1}^n (\alpha_i - \alpha_i^*) K(\mathbf{x}_i, \mathbf{x}) + b \quad (7)$$

in which α and α^* are Lagrange multipliers and K is the kernel function. In this study, a commonly used radial basis function was employed to estimate the kernel function. Detailed descriptions of the SVR methodology can be found in the literature [51,52].

2.3. RFR

The RFR approach proposed by Breiman [53] is a tree-based ensemble ML technique based on the combination of bagging (bootstrap aggregation) and the random subspace method. In the training process, the binary recursive partitioning of classification and the regression tree is used to build each decision tree. Once a forest has been constructed, predictions from each tree in the forest are combined as the final result. The advantages of RFR are its simplicity and the low number of required parameters. The RFR algorithm, as shown in Figure 1b, is summarized as follows [20,26,27,54]:

1. On the basis of the bootstrap method, a subset of samples is randomly produced with replacements from the original dataset.
2. These bootstrap samples are employed to construct regression trees. The optimal split criterion is used to split each node of the regression trees into two descendant nodes. The process on each descendant node is continued recursively until a termination criterion is fulfilled.
3. Each regression tree provides a predicted result. Once all of the regression trees have reached their maximum size, the final prediction is determined as the average of the results from all of the regression trees:

$$f^{\text{RFR}}(\mathbf{x}) = \frac{1}{tr} \sum_{tr=1}^{N_{tree}} \hat{h}_{tr}(\mathbf{x}) \quad (8)$$

in which tr is the number of trees, N_{tree} is the maximum size of the trees, and $\hat{h}_{tr}$ denotes the prediction of each regression tree. Detailed descriptions of RFR have been provided in previous studies [55,56].

2.4. MLPR

MLPR, which belongs to the feed-forward neural network, includes three layers: input, hidden, and output layers (Figure 1c). The neural network in MLPR consists of neurons, biases assigned to neurons, connections among neurons, and weights connecting neurons. Mathematically, the regression function of MLPR can be expressed as follows [57,58]:

$$f^{\text{MLPR}}(\mathbf{x}) = c_r + \sum_q u_{qr} \cdot a_q(\mathbf{x}) \quad (9)$$

where c_r denotes the bias of the r -th output neuron, u_{qr} is the weight connecting the q -th neuron in the hidden layer to the r -th neuron in the output layer, and $a_q(\mathbf{x})$ represents the activation function of the hidden neuron, which can be expressed in terms of F :

$$a_q(\mathbf{x}) = F \left(d_q + \sum_p v_{pq} \cdot x_p \right) \quad (10)$$

in which d_q is the bias of the q -th hidden neuron, x_p is the input variable, and v_{pq} is the weight connecting the p -th neuron in the input layer to the q -th neuron in the hidden layer. Several types of activation functions can be employed, including linear, sigmoid, hyperbolic tangent (tanh), and rectified linear unit (ReLU) functions. In the training process of MLPR, the back-propagation algorithm is used for adjusting the weights connecting neurons to minimize errors [19,59]. Details regarding the theory of MLPR have been provided in previous studies [60–62].

2.5. LGBMR

LGBMR uses four main algorithms to improve computational efficiency and prevent overfitting: gradient-based one-side sampling (GOSS), exclusive feature bundling (EFB), a histogram-based algorithm, and a leaf-wise growth algorithm [41,63]. As shown in Figure 1d, the leaf-wise growth algorithm allows the identification of the leaf node with the largest split gain, enabling the management of big data while preventing overfitting. In addition, LGBMR adopts the histogram-based decision tree algorithm to divide continuous floating-point features into variety intervals for reduce the computational power required for prediction. Moreover, GOSS and EFB are used to reduce the number of samples for accelerating the training process of LGBMR.

The objective function of LGBMR can be written as follows:

$$\text{Obj}^t = \sum_{i=1}^n l(y_i, \hat{y}_i^t) + \sum_{i=1}^t \Omega(f_i) \quad (11)$$

where l is the loss function, Ω is the regularization term of a decision tree f_i at the t time iteration, y_i is the true (objective) value, and $\hat{y}_i$ is the predicted value. On the basis of the boosting algorithm, Equation (11) can be further expressed as follows:

$$\text{Obj}^t = \sum_{i=1}^n l[y_i, \hat{y}_i^{(t-1)} + f_t(x_i)] + \sum_{i=1}^t \Omega(f_i) \quad (12)$$

where $\hat{y}_i^{t-1}$ is the predicted value at the $t - 1$ step model and $f_t(x_i)$ denotes the new predicted value at the t -th step. To solve the objective function, the Newton method is employed to simplify Equation (12) into the following equation:

$$\text{Obj}^t = \sum_{i=1}^n \left[g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \sum_{i=1}^t \Omega(f_i) \quad (13)$$

where g_i and h_i are, respectively, the first and second derivatives of the loss function, which can be expressed as follows:

$$g_i = \hat{y}_i^{(t-1)} l[y_i, \hat{y}_i^{(t-1)}], \quad h_i = \frac{2}{\hat{y}_i^{(t-1)}} l[y_i, \hat{y}_i^{(t-1)}] \quad (14)$$

Samples in the regression trees are related to leaf nodes. The final value of loss can be determined from the accumulation of the loss values of the leaf nodes. Thus, with the use of I_j to represent the sample of leaf j , Equation (13) can be rewritten as

$$\text{Obj}^t = \sum_{j=1}^T \left[\left(\sum_{i \in I_j} g_i \right) w_j + \frac{1}{2} \left(\sum_{i \in I_j} h_i + \lambda \right) w_j^2 \right] \quad (15)$$

where T is the total number of regression trees, w is the weight of the lead node, and λ is the regularization parameter. To conclude, the optimal objective function can be solved through minimization of the quadratic function. Detailed descriptions of LGBMR have been provided in previous studies [39,41,63].

2.6. Bayesian Optimization and Cross-Validation

The prediction performance of data-driven models can be considerably affected by the selection of parameters. To use the training data set for construction of the models, training parameters (hyperparameters) must be selected. In general, several methods can be adopted for the tuning of hyperparameters, including grid search, random search, and the use of genetic algorithms [64]. Grid and random search are traditional parameter optimization methods in ML; however, they require brute-force search or certain experi-

ence [65]. In addition, genetic algorithms can be applied to search for optimal parameters; however, they require several control variables and have a poor local search capacity [66].

Another favorable choice is Bayesian optimization, which has been demonstrated to be more efficient than genetic algorithms in practice. Bayesian optimization is primarily based on the concept of employing a Gaussian process to establish and optimize a substitute function with consideration of the previous evaluation results of the objective function. In this study, Bayesian optimization was employed and combined with 10-fold cross-validation to enhance the prediction accuracy. Bayesian optimization was implemented for model training according to the following steps [39,40]:

1. The objective function was defined, and the interval range of the parameters was set.
2. During the training process of the four models, the indicator of the mean square error was used to evaluate the result of each parameter combination.
3. The optimal combination of parameters could be determined through 10-fold cross-validation.
4. With these optimal parameters, the model was finally constructed, and the test dataset was used for river stage prediction.

2.7. Performance Evaluation Criteria

To quantitatively evaluate the performance of the four models, the following four evaluation criteria were employed [67–71]:

1. Nash–Sutcliffe efficiency (NSE)

$$NSE = 1 - \frac{\sum_{i=1}^n (H_i^{\text{mea}} - H_i^{\text{pre}})^2}{\sum_{i=1}^n (H_i^{\text{mea}} - \bar{H}^{\text{mea}})^2} \quad (16)$$

2. Coefficient of determination (R^2)

$$R^2 = \left[\frac{\sum_{i=1}^n (H_i^{\text{mea}} - \bar{H}^{\text{mea}})(H_i^{\text{pre}} - \bar{H}^{\text{pre}})}{\sqrt{\sum_{i=1}^n (H_i^{\text{mea}} - \bar{H}^{\text{mea}})^2 \sum_{i=1}^n (H_i^{\text{pre}} - \bar{H}^{\text{pre}})^2}} \right]^2 \quad (17)$$

3. Mean absolute error (MAE)

$$MAE = \frac{\sum_{i=1}^n |H_i^{\text{mea}} - H_i^{\text{pre}}|}{n} \quad (18)$$

4. Root mean square error (RMSE)

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (H_i^{\text{mea}} - H_i^{\text{pre}})^2}{n}} \quad (19)$$

where H_i^{mea} and H_i^{pre} are, respectively, the measured and predicted river stages and $\bar{H}^{\text{mea}}$ and $\bar{H}^{\text{pre}}$ are, respectively, the mean of the measured and predicted river stages.

The NSE value may range from $-\infty$ to 1. The closer the value of NSE to 1, the better the prediction ability during modeling. A negative NSE value indicates considerably poor prediction performance. An R^2 value ranging from 0 to 1 is used to represent the relationship between measured and predicted river stages. An R^2 value of 0 indicates no relation, while the value of 1 denotes the predicted values equal to the measured. MAE and RMSE are indicators showing the number of errors obtained by a model, and the optimal value of MAE and RMSE is zero.

To further evaluate the capability of the four models for river stage modeling, two additional error indexes were employed:

1. Peak water-level error (PWE)

$$\text{PWE} = H_p^{\text{pre}} - H_p^{\text{mea}} \quad (20)$$

2. Error of time-to-peak water level (ETP)

$$\text{ETP} = T_p^{\text{pre}} - T_p^{\text{mea}} \quad (21)$$

where H_p^{mea} and H_p^{pre} are, respectively, the measured and predicted peak river stages and T_p^{mea} and T_p^{pre} denote the measured and predicted time to the peak river stage, respectively. The closer the values of PWE and ETP are to 0, the higher the accuracy of the model is.

2.8. Flowchart for Training and Validation

Figure 2 displays a flowchart of the process of establishing and evaluating the four models, which involved three main steps: preprocessing, ML, and validation. In the preprocessing step, time-series data were collected, including rainfall, water level, and tidal stage data. Subsequently, the collected data were divided into training and test datasets. The training datasets were employed to establish the data-driven models using ML. Then, the test datasets were utilized to evaluate the performance of the constructed (i.e., trained) models for the purpose of validation. To prevent the obtention of inaccurate prediction results, this study employed the commonly used min–max normalization method to normalize the collected data into the range of (−1, 1). After preprocessing, the optimal combinations of input vectors were investigated, and Bayesian optimization with 10–fold cross-validation was employed to enhance the training results. After the establishment of the four models, the test datasets were used as inputs and the river stage was accordingly forecasted. To assess whether the four models produced accurate and reliable results, several indicators were used to evaluate the performance of the proposed models.

Figure 2. Primary flowchart for establishing data-driven models as well as predicting river stage.

3. Study Area and Data

The Lan-Yang River basin, which has a watershed area of 978 km², is located in the northeast part of Taiwan (Figure 3). The main river reach of the Lan-Yang River has a length of 73 km and an average slope of 1/55. The Yi-Lan River, with a length of 17.25 km, is a tributary of Lan-Yang River. Figure 3 shows the locations of hydrological stations, namely four rainfall gauging stations, three river stage gauging stations, and one tidal gauging station. The four models were constructed to forecast river stages at the Lanyan,

Simon, and Kavalan stations with a lead time of 1–6 h. The Kavalan station is located at the Yi-Lan River and is situated near the estuary. The river stage at the Kavalan station may be affected by the upstream discharge and tidal level. Therefore, this study considered the tidal effect while predicting the river stage at the Kavalan station.

Figure 3. The map of study area displaying the locations of hydrological stations.

In this study, data regarding hourly rainfall, river stage, and tidal level at each station were collected for two types of events, namely typhoons and storms. Table 1 lists data collected from June 2004 to October 2017 as well as the maximum water level at the three stations. The total collected data of 37 events were considered for both Lanyan and Simon stations. Because the Kavalan station had limited available data and some missing data, the collected data of 20 events were used. As shown in Table 1, the maximum values of recorded river stages at the Lanyan, Simon, and Kavalan stations were 8.06, 7.54, and 3.11 m, respectively. To further examine the hydrology statistics, Table 2 lists the characteristic based on collected data from 2004 to 2017. For the training data sets of Lanyan station, the result indicates that the water level ranged from 1.80 m to 7.40 m, and the mean water level is 3.42 m.

To construct and assess the four models, the total collected data were separated into training and test datasets. Figure 4 shows the measured river stage data used for the Lanyan, Simon, and Kavalan stations. Significant increases and decreases in the river stage usually occur from May to October. In addition, the river stages at the Lanyan and Simon stations were unaffected by the tidal current, whereas the river stage at the Kavalan station was significantly affected by the tidal current. Figure 4 displays the employed datasets, in which 70% (Event No. 1–22) was used for training and the remaining 30% (Event No. 23–37) was utilized for testing.

Table 1. The information of the collected hydrology data in the study area.

No.	Events	Date	Duration (h)	Maximum Water Level (m)		
				Lanyan	Simon	Kavalan
1	Typhoon Mindulle	30 June 2004	162	2.87	4.94	-
2	Typhoon Aere	23 August 2004	258	5.77	6.72	-
3	Storm	28 May 2005	138	3.20	5.26	-
4	Typhoon Haitang	16 July 2005	210	7.11	5.92	-
5	Typhoon Matsa	3 August 2005	378	4.80	6.05	-
6	Typhoon Talim	29 August 2005	138	6.56	5.78	-
7	Typhoon Damrey	21 September 2005	186	4.92	6.00	-
8	Typhoon Longwang	30 September 2005	138	6.21	5.36	-
9	Typhoon Chanchu	15 May 2006	186	3.82	5.20	0.98
10	Storm	9 June 2006	162	3.90	5.03	1.06
11	Typhoon Saomai	7 August 2006	186	4.11	5.04	1.39
12	Typhoon Shanshan	7 September 2006	306	5.20	5.38	1.30
13	Storm	5 June 2007	234	4.32	5.14	1.07
14	Typhoon Wutip	6 August 2007	138	4.47	5.07	1.28
15	Typhoon Sepat	15 August 2007	186	6.73	5.83	2.12
16	Typhoon Wipha	17 September 2007	330	4.42	5.64	1.23
17	Typhoon Fung-Wong	27 July 2008	114	6.39	5.50	1.87
18	Typhoon Sinlaku	12 September 2008	210	6.14	7.54	2.33
19	Typhoon Jangmi	27 September 2008	90	7.40	6.60	3.11
20	Typhoon Morakot	4 August 2009	210	5.75	5.33	1.96
21	Typhoon Parma	3 October 2009	162	7.21	6.51	-
22	Typhoon Fanapi	17 September 2010	258	4.88	5.36	1.79
23	Typhoon Megi	27 October 2010	330	6.62	7.24	-
24	Typhoon Nanmadol	3 September 2011	258	3.60	5.34	1.13
25	Storm	7 October 2011	186	6.98	5.76	1.64
26	Typhoon Saola	7 August 2012	234	8.06	7.08	-
27	Typhoon Soulik	17 July 2013	138	3.70	6.08	1.86
28	Typhoon Trami	3 September 2013	354	3.51	5.20	1.43
29	Typhoon Matmo	27 July 2014	138	5.20	5.56	-
30	Storm	28 September 2014	186	3.70	5.96	-
31	Typhoon Soudelor	11 August 2015	114	7.11	7.01	-
32	Typhoon Dujan	29 September 2015	54	6.11	6.15	-
33	Storm	17 May 2016	114	3.77	4.68	0.97
34	Typhoon Nepartak	16 July 2016	234	3.77	4.87	0.93
35	Typhoon Megi	1 October 2016	138	7.51	5.86	-
36	Typhoon Nesat	3 August 2017	162	5.22	4.71	1.70
37	Storm	19 October 2017	402	5.83	6.20	-

Table 2. The hydrological statistics of the collected data in the study area.

Characteristic	Training Data Sets			Test Data Sets		
	Lanyan	Simon	Kavalan	Lanyan	Simon	Kavalan
Maximum Water Level (m)	7.40	7.70	3.16	8.06	7.28	1.87
Minimum Water Level (m)	1.80	4.10	−0.30	1.98	3.96	−0.41
Mean Water Level (m)	3.42	4.94	0.68	3.39	4.80	0.39

Figure 4. The measured hourly river stage data at (a) Lanyan, (b) Simon, and (c) Kavalan stations.

4. Results and Discussion

As indicated in the flowchart illustrated in Figure 2, four ML techniques were applied to construct the data-driven model for river stage prediction. The models were trained and validated using datasets of collected measurements. The forecasting results of model training and validation are presented and discussed in this section. Six evaluation criteria were used to examine the model performance during the training and testing processes. Furthermore, the performance of the four models was compared to examine their applicability for river stage forecasting. All models were developed in a Python 3.7 environment running on an Intel Core i5 and 3.0-GHZ CPU with 8.0-GB RAM.

4.1. Analysis for Combinations of Input Variables

Before model training and validation, the commonly used SVR was employed to determine the optimal combination of input variables. The target output was the prediction of the river stage at the Kavalan station with a lead time of 1 h (i.e., $t + 1$). As shown in Table 3, four combination scenarios were evaluated. For the first combination of input variables, namely C1, only the antecedent tidal stage from t to $t - 6$ was adopted as the input. For the second combination of input variables, named C2, the antecedent hourly rainfall and tidal stage were utilized. The combination of antecedent rainfall, river stage,

and tidal level data from t to $t - 6$ was used as the third combination, namely C3. In the final combination, namely C4, only antecedent rainfall data at each station were considered.

Table 3. Four combination scenarios for conducting optimal inputs.

Combinations of Inputs	Input Vectors			Output Variables
	Rainfall	River Stage	Tidal Level	
C1	-	-	$S_t \dots S_{t-6}$	$\hat{H}_{t+1}$
C2	$R_t \dots R_{t-6}$	-	$S_t \dots S_{t-6}$	$\hat{H}_{t+1}$
C3	$R_t \dots R_{t-6}$	$H_t \dots H_{t-6}$	$S_t \dots S_{t-6}$	$\hat{H}_{t+1}$
C4	$R_t \dots R_{t-6}$	-	-	$\hat{H}_{t+1}$

The four input combinations were subsequently used to construct the data-driven SVR model for river stage prediction. Figure 5 displays the model training results, presented as a scatter plot of the measured and forecasted river stages obtained using the four combinations of input variables. The scatter points of SVR with C3 were closer to the 45° line ($y = x$, black dotted line) than those of SVR with C1, C2, and C4. In addition, Figure 5 shows that the scatter points of SVR with C4 were dispersed from the 45° line for the low river stage (approximately lower than 1.0 m). Furthermore, to assess the performance of the four combinations of input variables, four evaluation criteria, namely R^2 , MAE, RMSE, and NSE were adopted. Table 4 shows the performance of river stage forecasting at a lead time of 1 h when SVR was used with the four input combinations. As shown in Figure 5 and Table 4, C3 achieved higher accuracy during model training than did C1, C2, and C4 for both high and low river stages.

Figure 5. Scatter plot of measured and forecasted river stages in SVR model training with four combinations of inputs.

After the quantitative analysis of model training, model testing was performed using the four combinations of input variables. Figure 6 shows simulated results with four close-up views displaying the four flood events. These close-up views of the simulated water levels reveal the differences in river stage prediction results among the four combinations. The simulated river stage hydrographs obtained using SVR with C3 agrees very closely with the measured river stage hydrographs. However, a large discrepancy between the simulated and measured river stage was observed when SVR was used with C1, indicating

that considering only the tidal input does not correctly indicate the peak water level and time-to-peak water level.

Table 4. Training performances of river stage forecasting using SVR with four input combinations.

Combinations of Inputs	R ²	MAE (m)	RMSE (m)	NSE
C1	0.587	0.171	0.263	0.579
C2	0.711	0.142	0.219	0.708
C3	0.980	0.040	0.056	0.981
C4	0.359	0.256	0.325	0.359

Figure 6. The comparisons of the simulated river stage with the measured one at Kavalan station using SVR with four input combinations for (a) all test dataset and (b–e) four selected events.

To further investigate the model validation, four flood events at the Kavalan station were selected to evaluate the river stage prediction capability of SVR with different input combinations. Table 5 lists simulated results, including those for ETP and PWE. SVR with

C3 demonstrated the most favorable performance, with the lowest average absolute PWE of 0.17 m, whereas SVR with C1 and C4 achieved almost the same largest average absolute PWE of 0.37. Furthermore, SVR with C3 was found to have the lowest ETP among all of the combinations. Thus, SVR with C1 and C4 demonstrated poor performance, whereas SVR with C3 exhibited the most favorable performance. In summary, these results indicate that considering three input variables (i.e., rainfall, river stage, and tidal level) in the previous 6 h can significantly improve the accuracy of river stage prediction in a tidal river.

Table 5. Model test performances of river stage forecasting using SVR with four input combinations.

Events	C1		C2		C3		C4	
	ETP (h)	PWE (m)	ETP (h)	PWE (m)	ETP (h)	PWE (m)	ETP (h)	PWE (m)
No. 25 (Storm)	−6	−0.61	−3	0.29	0	−0.05	3	0.32
No. 27 (Typhoon Soulik)	3	−0.18	2	0.62	0	0.17	0	0.40
No. 33 (Storm)	−1	−0.23	−1	0.36	−1	0.16	−1	0.54
No. 36 (Typhoon Nesat)	4	0.48	1	0.15	1	0.30	1	−0.23

4.2. Analysis of Model Training Results

After the optimal combination of input variables was determined, the four models were trained. The optimal combination, namely C3, was used as the input to establish the river stage prediction model for each station. In ML, hyperparameters play a vital role in model performance. During model training, Bayesian optimization together with a 10-fold cross-validation was employed to determine the optimal parameters. For SVR, penalty and kernel parameters were selected as the hyperparameters [46]. According to Choi et al. [54], the two main parameters to be determined in an RFR model are N_{split} and N_{tree} , where N_{split} is the minimum number of samples required to split. For MLPR, a single hidden layer is used because of its relatively wide application. Hence, the number of neurons in the hidden layer N_{neu} must be determined. As hyperparameters in the LGBMR model, the parameters N_{dep} and N_{leaves} respectively represent the maximum tree depth and maximum number of tree leaves. Table 6 summarizes the optimal parameters for the three stations obtained using the proposed models. The result shows different optimal parameters for different stations and lead times. In the MLPR model, the activation function of tanh was preferred for Lanyan and Simon stations. For the Kavalan station, the optimal activation function was ReLU.

Table 6. The hyperparameters results of four models for 1–6-h lead times at three stations.

Stations	Lead Times (h)	SVR		RFR		MLPR		LGBMR		
		C	γ	N_{split}	N_{tree}	N_{neu}	a_q	N_{dep}	N_{tree}	N_{leaves}
Lanyan	1	50.0	0.005	7	215	6	tanh	5	881	6
	3	49.6	0.02	3	296	6	tanh	4	900	98
	6	42.9	0.005	3	298	6	tanh	16	895	6
Simon	1	44.8	0.005	6	300	6	tanh	19	106	5
	3	34.0	0.007	17	144	6	tanh	3	214	25
	6	24.7	0.005	36	295	6	tanh	3	107	6
Kavalan	1	50.0	0.005	3	298	12	ReLU	4	895	100
	3	44.3	0.007	4	298	12	ReLU	7	105	97
	6	27.7	0.011	3	297	12	ReLU	20	900	100

The four models were trained using the optimal parameters listed in Table 6. Figure 7 presents the training results obtained using four models at each station for lead times of 1, 3, and 6 h. For the 1-h lead time, the forecasted points nearly reached the 45° reference line for all stations. When the lead time increased to 3 and 6 h, the forecasted points moved away from the reference line. The results indicated that the forecasted points at the Simon station with a high river stage condition (>7 m), as determined through SVR, RFR, and LGBMR, were lower than the 45° line. Furthermore, SVR, RFR, and LGBMR achieved a similar underestimated trend at the Kavalan station with a high river stage condition (>2.5 m). Overall, the points forecasted using LGBMR were closer to the 45° reference line than those forecasted using SVR, RFR, and MLPR, indicating that LGBMR demonstrated the most favorable training performance among the four models except for a few peak values. Table 7 lists the training results of the four models for 1–6-h lead times at the three stations obtained using the four evaluation indexes. All models demonstrated favorable performance in terms of R^2 , which was over 0.72 for 1–6-h lead times at the three stations. Both the RMSE and MAE values for RFR and LGBMR models were slightly lower than those for the SVR and MLPR models. In addition, both the RFR and LGBMR models yielded higher NSE values than did the SVR and MLPR models.

Figure 7. The Scatter plots of measured river stage against forecasted river stage using four models for 1-h lead time at (a) Simon (b) Lanyan (c) Kavalan stations, for 3-h lead time at (d) Simon (e) Lanyan (f) Kavalan stations, and for 6-h lead time at (g) Simon (h) Lanyan (i) Kavalan stations.

Table 7. The training results of four models for 1–6-h lead times at three stations.

Stations	Models	R ²						MAE (m)						RMSE (m)						NSE					
		1-h	2-h	3-h	4-h	5-h	6-h	1-h	2-h	3-h	4-h	5-h	6-h	1-h	2-h	3-h	4-h	5-h	6-h	1-h	2-h	3-h	4-h	5-h	6-h
Lanyan	SVR	0.98	0.98	0.98	0.94	0.92	0.90	0.03	0.05	0.06	0.08	0.11	0.12	0.08	0.12	0.13	0.18	0.23	0.25	0.99	0.98	0.97	0.94	0.91	0.89
	RFR	0.98	0.98	0.98	0.98	0.98	0.98	0.02	0.02	0.03	0.03	0.04	0.04	0.04	0.05	0.06	0.07	0.08	0.09	0.99	0.99	0.99	0.99	0.98	0.98
	MLPR	0.98	0.98	0.96	0.94	0.90	0.90	0.03	0.05	0.08	0.08	0.12	0.12	0.08	0.11	0.17	0.18	0.24	0.24	0.99	0.98	0.95	0.95	0.90	0.89
	LGBMR	0.98	0.98	0.98	0.98	0.96	0.96	0.01	0.02	0.02	0.06	0.07	0.09	0.03	0.03	0.04	0.10	0.12	0.16	0.99	0.99	0.99	0.98	0.97	0.96
Simon	SVR	0.98	0.98	0.96	0.92	0.90	0.85	0.02	0.03	0.04	0.05	0.05	0.06	0.04	0.05	0.07	0.09	0.11	0.13	0.99	0.98	0.95	0.92	0.89	0.85
	RFR	0.98	0.98	0.98	0.96	0.94	0.92	0.01	0.01	0.02	0.03	0.04	0.04	0.02	0.03	0.05	0.06	0.08	0.10	0.99	0.99	0.98	0.97	0.94	0.91
	MLPR	0.98	0.98	0.94	0.92	0.88	0.85	0.01	0.02	0.03	0.04	0.05	0.06	0.03	0.05	0.07	0.09	0.11	0.13	0.99	0.98	0.96	0.92	0.89	0.84
	LGBMR	0.98	0.98	0.98	0.96	0.96	0.88	0.01	0.02	0.02	0.03	0.03	0.06	0.03	0.04	0.04	0.05	0.06	0.11	0.98	0.98	0.98	0.97	0.96	0.88
Kavalan	SVR	0.98	0.96	0.92	0.90	0.85	0.77	0.04	0.06	0.08	0.10	0.12	0.14	0.06	0.08	0.11	0.13	0.16	0.19	0.98	0.96	0.92	0.89	0.84	0.77
	RFR	0.98	0.98	0.98	0.98	0.98	0.96	0.01	0.03	0.04	0.05	0.05	0.06	0.02	0.04	0.05	0.06	0.07	0.08	0.99	0.99	0.98	0.97	0.97	0.96
	MLPR	0.98	0.96	0.90	0.83	0.83	0.72	0.03	0.06	0.09	0.12	0.12	0.16	0.05	0.09	0.13	0.16	0.16	0.22	0.99	0.96	0.90	0.83	0.81	0.72
	LGBMR	0.98	0.98	0.98	0.96	0.96	0.96	0.01	0.03	0.04	0.05	0.05	0.05	0.02	0.05	0.05	0.06	0.07	0.07	0.99	0.98	0.98	0.97	0.97	0.97

4.3. Results of Model Validation

On the basis of the four models trained in Section 4.2, the collected test dataset was employed to drive four models for river stage forecasting with 1–6-h ahead lead times. Figure 8 compares the measured river stage with the forecasted river stage for Typhoon Megi with 1–6-h lead times at the Lanyan station. As the lead time increased, the difference in forecasted results among the four models became more significant. The results (Figure 8) revealed that MLPR yielded overestimated values of peak river stages except for the 3-h lead time. A comparison between the measured and forecasted river stages for Typhoon Saola at the Simon station is presented in Figure 9. The results reveal that the four models could efficiently forecast the river stage at a 1-h lead time. A higher ETP (i.e., phase lag) between the measured and forecasted results was observed for 3–6-h lead times. Figure 10 shows the results of a comparison of the four models for Typhoon Soulik with 1–6-h lead times at the Kavalan station. The river stage hydrographs forecasted using the four models agreed with the measured river stage hydrographs for a 1-h lead time. The difference among the model validation results considerably increased for a lead time of 2 to 6 h. The four models also overestimated the peak river stages.

Figure 8. The comparison of measured and forecasted river stages at Lanyan station for Typhoon Megi with (a) 1-, (b) 2-, (c) 3-, (d) 4-, (e) 5-, and (f) 6-h lead times.

Figure 9. The comparison of measured and forecasted river stages at Simon station for Typhoon Saola with (a) 1-, (b) 2-, (c) 3-, (d) 4-, (e) 5-, and (f) 6-h lead times.

4.4. Performance Evaluation of River Stage Forecasting

This section presents the evaluation of river stage forecasting performance using two evaluation metrics, namely ETP and PWE. Table 8 summarizes the results of 1-, 3-, and 6-h lead times using ETP and PWE for the three stations. As listed in Table 8, the maximum absolute values of PWE determined using SVR, RFR, MLPR, and LGBMR were respectively 0.89, 1.42, 1.33, and 0.89 m at the Lanyan station and 0.63, 0.77, 0.50, and 0.43 m at the Simon station. Figure 11 presents boxplots of the absolute values of PWE from four selected events for 1-, 3-, and 6-h lead times. As displayed in Figure 11, the box results obtained using the RFR and MLPR models had broader distributions for 3- and 6-h lead times than the distributions obtained using the SVR and LGBMR models. Moreover, the median of the absolute values of PWE obtained using the LGBMR model was slightly closer to zero than the medians obtained using the SVR, RFR, and MLPR models. Although RFR exhibited satisfactory training performance (see Section 4.2), it exhibited poor performance in the model validation. According to the results of the range and length of the box and whiskers, the LGBMR model demonstrated more favorable performance with an average PWE value of 0.22 m. Furthermore, to compare the performance of the models, the ETP results listed in Table 8 were averaged by all events and stations; the results are displayed in Figure 12.

The average value of ETP obtained using LGBMR was close to 2 h for 1–6-h lead times. The results of the comprehensive assessment revealed that the LGBMR model exhibited more favorable performance in forecasting both the peak river stage and time-to-peak river stage.

Figure 10. The comparison of measured and forecasted river stages at Kavalan station for Typhoon Soulik with (a) 1-, (b) 2-, (c) 3-, (d) 4-, (e) 5-, and (f) 6-h lead times.

Table 8. The model validation results displaying lead times of 1, 3, and 6 h at three stations.

Stations	Events	SVR						RFR						MLPR						LGBMR					
		ETP (h)			PWE (m)			ETP (h)			PWE (m)			ETP (h)			PWE (m)			ETP (h)			PWE (m)		
		1-h	3-h	6-h	1-h	3-h	6-h	1-h	3-h	6-h	1-h	3-h	6-h	1-h	3-h	6-h	1-h	3-h	6-h	1-h	3-h	6-h	1-h	3-h	6-h
Lanyan	No. 23 (Typhoon Megi)	1	1	5	0.19	0.11	0.41	1	4	3	0.09	0.20	−0.21	1	3	7	0.29	0.60	1.33	1	0	3	0.02	0.17	0.89
	No. 26 (Typhoon Saola)	0	2	5	0.02	−0.12	−0.06	−1	1	3	−0.95	−1.03	−1.42	1	2	6	0.42	0.94	0.08	1	1	3	−0.53	−0.41	−0.56
	No. 31 (Typhoon Soudelor)	0	0	7	0.33	−0.17	0.89	1	2	6	−0.22	−0.38	−1.13	0	1	3	0.66	1.30	1.28	−1	1	4	0.26	0.42	0.29
	No. 35 (Typhoon Megi)	1	0	3	−0.04	−0.42	0.31	0	3	3	−0.35	−0.87	−1.15	1	1	3	0.45	0.80	0.53	0	2	3	−0.31	−0.30	−0.40
Simon	No. 23 (Typhoon Megi)	1	3	5	−0.09	−0.24	−0.28	0	3	5	−0.03	−0.38	−0.72	−1	1	4	−0.02	0.05	−0.19	−2	−1	2	0.01	0.04	−0.43
	No. 26 (Typhoon Saola)	0	1	4	0.03	0.14	−0.29	1	2	4	−0.04	−0.36	−0.56	0	2	4	0.16	0.13	0.09	0	1	3	0.06	−0.16	−0.36
	No. 31 (Typhoon Soudelor)	2	3	5	−0.19	−0.27	−0.63	2	3	6	−0.03	−0.26	−0.77	2	3	6	0.26	0.29	−0.35	0	1	5	0.17	0.02	0.08
	No. 37 (Storm)	1	2	5	0.10	0.10	−0.43	2	1	4	−0.04	0.30	−0.36	1	2	5	0.11	0.19	−0.50	−1	0	3	−0.01	0.35	0.02
Kavalan	No. 25 (Storm)	−1	2	9	0.02	0.11	0.14	0	4	9	−0.05	0.32	0.83	1	5	9	−0.02	0.69	0.61	1	2	9	−0.01	0.12	0.50
	No. 27 (Typhoon Soulík)	0	2	4	0.16	0.23	0.19	1	3	4	0.10	0.09	0.06	0	3	4	0.20	0.56	0.31	1	2	4	0.14	0.10	0.09
	No. 33 (Storm)	−2	0	2	0.16	0.80	0.61	0	4	4	0.01	0.10	0.09	−3	0	2	0.05	0.41	0.25	0	0	4	0.02	0.05	0.18
	No. 36 (Typhoon Nesat)	1	3	5	0.32	0.99	0.38	1	4	6	0.01	−0.04	−0.06	0	3	5	0.35	0.97	0.39	0	2	6	0.11	0.20	0.14

Figure 11. The boxplot in terms of absolute values of peak water-level error (PWE) by four models for (a) 1-, (b) 3-, and (c) 6-h lead times.

Figure 12. The average absolute values of error of time-to-peak water level (ETP) by four models for 1-, 3-, and 6-h lead times.

A quantitative comparison of the CPU time is presented in Table 9, including for the training and validation stages. The LGBMR model required the shortest CPU time, whereas the MLPR model required the longest CPU time in the training stage. In the model validation process, predictions made by each model during 1–6-h lead times were completed within 0.5 s. This finding implies that all four models could satisfy the operational requirements of flood forecasting computational time. Nevertheless, LGBMR would be a more favorable choice for practical river stage simulation because of its high efficiency and accuracy.

Table 9. The consumed CPU time by four models in training and validation processes.

Process	Models	Consumed CPU Time (sec)
Training (10-fold cross-validation) (1–6 h lead times)	SVR	192
	RFR	1080
	MLPR	1873
	LGBMR	156
Validation (1–6 h lead times)	SVR	0.38
	RFR	0.06
	MLPR	0.01
	LGBMR	0.03

5. Conclusions

This study presents a multistep-ahead framework involving Bayesian optimization to construct data-driven prediction models based on four ML techniques, namely SVR, RFR, MLPR, and LGBMR, for river stage forecasting. The models were successfully applied to predict the evolution of river stages in a tidal river, namely the Lan-Yang River Basin in Taiwan. The application of LGBMR in river stage simulation is a relatively new approach. Nearly 14 years of hydrology data were collected and employed to train data-driven models through the application of Bayesian optimization with a 10-fold cross-validation. The constructed models were then applied to forecast the hourly future river stage up to 6 h ahead at three stations. The performance of the four models was also compared on the basis of six evaluation criteria. The results revealed that the LGBMR model produced the most accurate river stage hydrograph among the four models.

The primary findings and contributions of this study are as follows:

1. The optimal combination of input variables was determined using the SVR model with four designed combinations. The results indicate that the combination of rainfall, river stage, and tidal level as the input variables can improve the river stage prediction accuracy in a tidal reach.
2. The results of the quantitative analysis of model training and validation were used to compare the forecasting performance of the four models. The results demonstrated that the LGBMR model produced more satisfactory river stage forecasting at a lead time of up to 6 h. The average PWE and ETP values obtained using LGBMR were 0.22 m and 2 h, respectively, indicating an acceptable accuracy in river stage forecasting.

The high efficiency and accuracy of the LGBMR model make it a robust approach for river stage prediction. To develop a real-time flood forecasting system, the previous values of input variables, such as rainfall and tidal level, can be extended to forecast values using physics-based models. Therefore, future studies should focus on integrating a physics-based model with a data-driven model by correcting errors for improving flood forecasting accuracy. Meanwhile, more data-driven techniques, such as deep learning, should be adopted to enhance comparative research in the future.

Author Contributions: Conceptualization, W.-D.G. and W.-B.C.; methodology, validation and formal analysis, W.-D.G.; investigation and data curation, S.-H.Y. and W.-B.C.; writing—original draft preparation, W.-D.G., writing—review and editing, W.-D.G., W.-B.C., and C.-H.C.; supervision, H.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data sharing not applicable.

Acknowledgments: The authors would like to thank the Water Resources Agency, Ministry of Economic Affairs, Taiwan for providing the measured hydrology data. The rainfall and tidal datasets provided by the Central Weather Bureau, Taiwan, are also acknowledged.

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

References

1. Water Resources Agency (WRA). *Hydrological Yearbook of Taiwan*; Total Report 00-H-30-47, Ministry of Economic Affairs; Water Resources Agency: Taipei, Taiwan, 2019. Available online: <https://gweb.wra.gov.tw/wrhygis/> (accessed on 31 December 2020).
2. Hsu, W.K.; Huang, P.C.; Chang, C.C.; Chen, C.W.; Hung, D.M.; Chiang, W.L. An integrated flood risk assessment model for property insurance industry in Taiwan. *Nat. Hazards* **2011**, *58*, 1295–1309. [\[CrossRef\]](#)
3. Li, H.C.; Hsieh, L.S.; Chen, L.C.; Lin, L.Y.; Li, W.S. Disaster investigation and analysis of Typhoon Morakot. *J. Chin. Inst.* **2014**, *37*, 558–569. [\[CrossRef\]](#)
4. U.S. Army Corps of Engineers. *HEC-RAS. River Analysis System*; Hydraulic Reference Manual; Hydrologic Engineering Center: Davis, CA, USA, 2020; Available online: <https://www.hec.usace.army.mil/software/hecras/documentation.aspx> (accessed on 31 December 2020).
5. Deltares. *SOBEK. Hydrodynamics, Rainfall Runoff and Real Time Control*; User Manual. Deltares: Delft, The Netherlands, 2019. Available online: https://content.oss.deltares.nl/delft3d/manuals/SOBEK_User_Manual.pdf (accessed on 31 December 2020).
6. Liu, P.C.; Shih, D.S.; Chou, C.Y.; Chen, C.H.; Wang, Y.C. Development of a parallel computing watershed model for flood forecasts. *Procedia Eng.* **2016**, *154*, 1043–1049. [\[CrossRef\]](#)
7. Liu, W.C.; Chen, W.B.; Hsu, M.H.; Fu, J.C. Dynamic routing modeling for flash flood forecast in river system. *Nat. Hazards* **2010**, *52*, 519–537. [\[CrossRef\]](#)
8. Chen, W.B.; Liu, W.C. Modeling the influence of river cross-section data on a river stage using a two-dimensional/three-dimensional hydrodynamic model. *Water* **2017**, *9*, 203. [\[CrossRef\]](#)
9. Solomatine, D.P.; Ostfeld, A. Data-driven modelling: Some past experiences and new approaches. *J. Hydroinform.* **2008**, *10*, 3–22. [\[CrossRef\]](#)
10. Mosavi, A.; Ozturk, P.; Chau, K.W. Flood prediction using machine learning models: Literature review. *Water* **2018**, *10*, 1536. [\[CrossRef\]](#)
11. Maity, R.; Bhagwat, P.P.; Bhatnagar, A. Potential of support vector regression for prediction of monthly streamflow using endogenous property. *Hydrol. Process.* **2010**, *24*, 917–923. [\[CrossRef\]](#)
12. Chen, W.B.; Liu, W.C.; Hsu, M.H. Predicting typhoon-induced storm surge tide with a two-dimensional hydrodynamic model and artificial neural network model. *Nat. Hazards Earth Syst. Sci.* **2012**, *12*, 3799–3809. [\[CrossRef\]](#)
13. Chen, W.B.; Liu, W.C.; Hsu, M.H. Comparison of ANN approach with 2D and 3D hydrodynamic models for simulating estuary water stage. *Adv. Eng. Softw.* **2012**, *45*, 69–79. [\[CrossRef\]](#)
14. Lin, G.F.; Lin, H.Y.; Chou, Y.C. Development of a real-time regional-inundation forecasting model for the inundation warning system. *J. Hydroinform.* **2013**, *15*, 1391–1407. [\[CrossRef\]](#)
15. Wu, M.C.; Lin, G.F.; Lin, H.Y. Improving the forecasts of extreme streamflow by support vector regression with the data extracted by self-organizing map. *Hydrol. Process.* **2014**, *28*, 386–397. [\[CrossRef\]](#)
16. Hosseini, S.M.; Mahjouri, N. Integrating support vector regression and a geomorphologic artificial neural network for daily rainfall-runoff modeling. *Appl. Soft Comput. J.* **2016**, *38*, 329–345. [\[CrossRef\]](#)
17. Jhong, B.C.; Wang, J.H.; Lin, G.F. An integrated two-stage support vector machine approach to forecast inundation maps during typhoons. *J. Hydrol.* **2017**, *547*, 236–252. [\[CrossRef\]](#)
18. Seo, Y.; Choi, Y.; Choi, J. River stage modeling by combining maximal overlap discrete wavelet transform, support vector machines and genetic algorithm. *Water* **2017**, *9*, 525.
19. Jhong, Y.D.; Chen, C.S.; Lin, H.P.; Chen, S.T. Physical hybrid neural network model to forecast typhoon floods. *Water* **2018**, *10*, 632. [\[CrossRef\]](#)
20. Muñoz, P.; Orellana-Alvear, J.; Willems, P.; Céleri, R. Flash-flood forecasting in an Andean mountain catchment-development of a step-wise methodology based on the random forest algorithm. *Water* **2018**, *10*, 1519. [\[CrossRef\]](#)
21. Wu, J.; Liu, H.; Wei, G.; Song, T.; Zhang, C.; Zhou, H. Flash flood forecasting using support vector regression model in a small mountainous catchment. *Water* **2019**, *11*, 1327. [\[CrossRef\]](#)
22. Kim, H.I.; Han, K.Y. Inundation map prediction with rainfall return period and machine learning. *Water* **2020**, *12*, 1552. [\[CrossRef\]](#)
23. Nguyen, D.T.; Chen, S.T. Real-time probabilistic flood forecasting using multiple machine learning methods. *Water* **2020**, *12*, 787. [\[CrossRef\]](#)
24. Chen, W.B.; Liu, W.C.; Hsu, M.H. Artificial neural network modeling of dissolved oxygen in reservoir. *Environ. Monit. Assess.* **2014**, *186*, 1203–1217. [\[CrossRef\]](#) [\[PubMed\]](#)
25. Lin, G.F.; Chen, G.R.; Huang, P.Y. Effective typhoon characteristics and their effects on hourly reservoir inflow forecasting. *Adv. Water Resour.* **2010**, *33*, 887–898. [\[CrossRef\]](#)
26. Nguyen, T.T.; Huu, Q.N.; Li, M.J. Forecasting time series water levels on Mekong River using machine learning models. In Proceedings of the 7th International Conference on Knowledge and Systems Engineering, Ho Chi Minh City, Vietnam, 8–10 October 2015; pp. 292–297.
27. Li, B.; Yang, G.; Wan, R.; Dai, X.; Zhang, Y. Comparison of random forests and other statistical methods for the prediction of lake water level: A case study of the Poyang Lake in China. *Hydrol. Res.* **2016**, *47*, 69–83. [\[CrossRef\]](#)

28. Panagoulia, D.; Tsekouras, G.J.; Kousiouris, G. A multi-stage methodology for selecting input variables in ANN forecasting of river flows. *Glob. Nest J.* **2017**, *19*, 49–57.
29. Yang, T.; Asanjan, A.A.; Welles, E.; Gao, X.; Sorooshian, S.; Liu, X. Developing reservoir monthly inflow forecasts using artificial intelligence and climate phenomenon information. *Water Resour. Res.* **2017**, *53*, 2786–2812. [\[CrossRef\]](#)
30. Pini, M.; Scalvini, A.; Liaqat, M.U.; Ranzi, R.; Serina, I.; Mehmood, T. Evaluation of machine learning techniques for inflow prediction in Lake Como, Italy. *Procedia Comput. Sci.* **2020**, *176*, 918–927. [\[CrossRef\]](#)
31. Ebrahimi, E.; Shourian, M. River flow prediction using dynamic method for selecting and prioritizing k-nearest neighbors based on data features. *J. Hydrol. Eng.* **2020**, *25*, 04020010. [\[CrossRef\]](#)
32. Maspo, N.A.; Bin Harun, A.N.; Goto, M.; Cheros, F.; Haron, N.A.; Mohd Nawi, M.N. Evaluation of Machine Learning approach in flood prediction scenarios and its input parameters: A systematic review. *IOP Conf. Ser. Earth Environ. Sci.* **2020**, *479*, 012038. [\[CrossRef\]](#)
33. Chen, T.; Guestrin, C. Xgboost: A scalable tree boosting system. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 785–794.
34. Fan, J.; Yue, W.; Wu, L.; Zhang, F.; Cai, H.; Wang, X.; Lu, X.; Xiang, Y. Evaluation of SVM, ELM and four tree-based ensemble models for predicting daily reference evapotranspiration using limited meteorological data in different climates of China. *Agric. For. Meteorol.* **2018**, *263*, 225–241. [\[CrossRef\]](#)
35. Jin, Q.; Fan, X.; Liu, J.; Xue, Z.; Jian, H. Using eXtreme gradient BOOSTing to predict changes in tropical cyclone intensity over the Western North Pacific. *Atmosphere* **2019**, *10*, 341. [\[CrossRef\]](#)
36. Tama, B.A.; Rhee, K.H. An in-depth experimental study of anomaly detection using gradient boosted machine. *Neural Comput. Appl.* **2019**, *31*, 955–965. [\[CrossRef\]](#)
37. Sun, R.; Wang, G.Y.; Zhang, W.Y.; Hsu, L.T.; Ochieng, W.Y. A gradient boosting decision tree based GPS signal reception classification algorithm. *Appl. Soft Comput.* **2020**, *86*, 105942. [\[CrossRef\]](#)
38. Lucas, A.; Pegios, K.; Kotsakis, E.; Clarke, D. Price forecasting for the balancing energy market using machine-learning regression. *Energies* **2020**, *13*, 5420. [\[CrossRef\]](#)
39. Tang, M.; Zhao, Q.; Ding, S.X.; Wu, H.; Li, L.; Long, W.; Huang, B. An improved lightGBM algorithm for online fault detection of wind turbine gearboxes. *Energies* **2020**, *13*, 807. [\[CrossRef\]](#)
40. Wang, Y.; Wang, T.E. Application of improved LightGBM model in blood glucose prediction. *Appl. Sci.* **2020**, *10*, 3227. [\[CrossRef\]](#)
41. Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; Liu, T.Y. LightGBM: A highly efficient gradient boosting decision tree. In Proceedings of the 31st Annual Conference on Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 3146–3154.
42. Gao, X.; Luo, H.; Wang, Q.; Zhao, F.; Ye, L.; Zhang, Y. A human activity recognition algorithm based on stacking denoising autoencoder and lightGBM. *Sensors* **2019**, *19*, 947. [\[CrossRef\]](#)
43. Qadeer, K.; Jeon, M. Prediction of PM10 concentration in South Korea using gradient tree boosting models. In Proceedings of the 3rd International Conference on Vision, Image and Signal Processing, Vancouver, BC, Canada, 26–28 August 2019; pp. 1–6.
44. Bontempi, G.; Ben Taieb, S.; Le Borgne, Y.A. Machine learning strategies for time series forecasting. *Lect. Notes Bus. Inf. Process.* **2013**, *138*, 62–77.
45. Wang, J.H.; Lin, G.F.; Chang, M.J.; Huang, I.H.; Chen, Y.R. Real-time water-level forecasting using dilated causal convolutional neural networks. *Water Resour. Manag.* **2019**, *33*, 3759–3780. [\[CrossRef\]](#)
46. Yu, P.S.; Chen, S.T.; Chang, I.F. Support vector regression for real-time flood stage forecasting. *J. Hydrol.* **2006**, *328*, 704–716. [\[CrossRef\]](#)
47. Kao, I.F.; Zhou, Y.; Chang, L.C.; Chang, F.J. Exploring a Long Short-Term Memory based Encoder-Decoder framework for multi-step-ahead flood forecasting. *J. Hydrol.* **2020**, *583*, 124631. [\[CrossRef\]](#)
48. Drucker, H.; Burges, C.J.C.; Kaufman, L.; Smola, A.; Vapnik, V. Support vector regression machines. *Adv. Neural Inform. Process. Syst.* **1997**, *9*, 155–161.
49. Liong, S.Y.; Chandrasekaran, S. Flood stage forecasting with support vector machines. *J. Am. Water Resour. Assoc.* **2007**, *38*, 173–186. [\[CrossRef\]](#)
50. Wu, C.L.; Chau, K.W.; Li, Y.S. River stage prediction based on a distributed support vector regression. *J. Hydrol.* **2008**, *358*, 96–111. [\[CrossRef\]](#)
51. Gunn, S.R. *Support Vector Machines for Classification and Regression*; Technical Report; University of Southampton: Southampton, UK, 1998.
52. Chang, C.C.; Lin, C.J. LIBSVM: A library for support vector machines. *ACM Trans. Intern. Syst. Technol.* **2001**, *2*, 1–27. [\[CrossRef\]](#)
53. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [\[CrossRef\]](#)
54. Choi, C.; Kim, J.; Han, H.; Han, D.; Kim, H.S. Development of water level prediction models using machine learning in wetlands: A case study of Upo Wetland in South Korea. *Water* **2020**, *12*, 93. [\[CrossRef\]](#)
55. Boulesteix, A.L.; Janitza, S.; Kruppa, J.; König, I.R. Overview of random forest methodology and practical guidance with emphasis on computational biology and bioinformatics. *Wiley Interdisc. Rev. Data Min. Knowl. Discov.* **2012**, *2*, 493–507. [\[CrossRef\]](#)
56. Biau, G.; Scornet, E. A random forest guided tour. *Test* **2016**, *25*, 197–227. [\[CrossRef\]](#)
57. Khan, M.S.; Coulibaly, P. Application of support vector machine in lake water level prediction. *J. Hydrol. Eng.* **2006**, *11*, 199–205. [\[CrossRef\]](#)

-
58. Chen, C.; He, W.; Zhou, H.; Xue, Y.; Zhu, M. A comparative study among machine learning and numerical models for simulating groundwater dynamics in the Heihe River Basin, northwestern China. *Sci. Rep.* **2020**, *10*, 3904. [[CrossRef](#)]
 59. Rumelhart, D.E.; Hinton, G.E.; Williams, R.J. Learning representations by back-propagating errors. *Nature* **1986**, *323*, 533–536. [[CrossRef](#)]
 60. Haykin, S. *Neural Networks: A Comprehensive Foundation*; MacMillan: New York, NY, USA, 1994.
 61. Hagan, M.T.; Demuth, H.B.; Beale, M.H. *Neural Network Design*; PWS Publishing: Boston, MA, USA, 1996.
 62. Govindaraju, R.S.; Rao, A.R. Artificial neural networks in hydrology. I: Preliminary concepts. *J. Hydrol. Eng.* **2000**, *5*, 115–123.
 63. Ju, Y.; Sun, G.; Chen, Q.; Zhang, M.; Zhu, H.; Rehman, M.U. A model combining convolutional neural network and lightgbm algorithm for ultra-short-term wind power forecasting. *IEEE Access* **2019**, *7*, 28309–28318. [[CrossRef](#)]
 64. Kopsiaftis, G.; Protopapadakis, E.; Voulodimos, A.; Doulamis, N.; Mantoglou, A. Gaussian process regression tuned by Bayesian optimization for seawater intrusion prediction. *Comput. Intell. Neurosci.* **2019**, *2019*, 2859429. [[CrossRef](#)]
 65. Snoek, J.; Larochelle, H.; Adams, R.P. Practical Bayesian optimization of machine learning algorithms. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 2960–2968.
 66. Su, B.; Wang, Y. Genetic algorithm based feature selection and parameter optimization for support vector regression applied to semantic textual similarity. *J. Shanghai Jiaotong Univ.* **2015**, *20*, 143–148. [[CrossRef](#)]
 67. Patel, S.S.; Ramachandran, P. A comparison of machine learning techniques for modeling river flow time series: The case of upper Cauvery River Basin. *Water Resour. Manag.* **2015**, *29*, 589–602. [[CrossRef](#)]
 68. Lin, G.F.; Chou, Y.C.; Wu, M.C. Typhoon flood forecasting using integrated two-stage support vector machine approach. *J. Hydrol.* **2013**, *486*, 334–342. [[CrossRef](#)]
 69. Le, X.H.; Ho, H.V.; Lee, G.; Jung, S. Application of long short-term memory (LSTM) neural network for flood forecasting. *Water* **2019**, *11*, 1387. [[CrossRef](#)]
 70. Liu, M.; Huang, Y.; Li, Z.; Tong, B.; Liu, Z.; Sun, M.; Jiang, F.; Zhang, H. The applicability of LSTM-KNN model for real-time flood forecasting in different climate zones in China. *Water* **2020**, *12*, 440. [[CrossRef](#)]
 71. Van, S.P.; Le, H.M.; Thanh, D.V.; Dang, T.D.; Loc, H.H.; Anh, D.T. Deep learning convolutional neural network in rainfall-runoff modelling. *J. Hydroinform.* **2020**, *22*, 541–561. [[CrossRef](#)]