Desalination and Water Treatment
www.deswater.com

doi: 10.5004/dwt.2023.29771

303 (2023) 48–58
August

Development of a water quality prediction model using ensemble empirical mode decomposition and long short-term memory

Sukmin Yoon[a], Chi Hoon Park[b], No-Suk Park[c], Beomsu Baek[d], Youngsoon Kim[e,*]

[a]*Gyeongsang National University Office of Academy and Industry Collaboration and Engineering Research Institute, 501, Jinju-Daero, Jinju 52828, Republic of Korea*
[b]*Department of Energy Engineering, Future Convergence Technology Research Institute, Gyeongsang National University, 501, Jinju-Daero, Jinju 52828, Republic of Korea*
[c]*Department of Civil Engineering and Engineering Research Institute, Gyeongsang National University, 501, Jinju-Daero, Jinju 52828, Republic of Korea*
[d]*Department of Information and Statistics, Gyeongsang National University, 501, Jinju-Daero, Jinju 52828, Republic of Korea*
[e]*Department of Information and Statistics and Department of Bio & Medical Bigdata (BK21 Four Program), Gyeongsang National University, 501, Jinju-Daero, Jinju 52828, Republic of Korea, email: youngsoonkim@gnu.ac.kr*

Received 24 May 2023; Accepted 14 July 2023

ABSTRACT

Water distribution systems consistently supply high-quality water at suitable pressure and volume for human and industrial consumption. Meticulous water quality management is vital to these systems. South Korea, having established legal standards for water distribution in 1963, operates the National Auto Water Quality Monitoring System for real-time water quality monitoring and contamination warnings when levels exceed legal thresholds. The U.S. Environmental Protection Agency (EPA) points out that fixed thresholds can trigger an abundance of false-positive alarms, causing irregular hydraulic changes, and false-negative errors. This could potentially lead to a failure in detecting initial instances of pollution or micropollution that fall below the established threshold. To address this, our study developed an proactive contamination warning method for South Korea's monitoring system, utilizing long short-term memory (LSTM) for water quality prediction. We also employed ensemble empirical mode decomposition (EEMD) in feature engineering to enhance LSTM's prediction performance. Additionally, we devised an optimal water quality prediction model development methodology by comparing short- and long-term prediction performances. Our findings revealed that using EEMD for feature engineering improved the stability and reduced the prediction lag of LSTM, outperforming traditional methods. This refined approach offers a more reliable and efficient means of monitoring and managing water quality in distribution systems.

Keywords: Contamination warning; Ensemble empirical mode decomposition; Feature engineering; Long short-term memory; Water distribution system; Water quality

1. Introduction

A water distribution system (WDS) is a facility that continuously supplies high-quality water at an appropriate water pressure and quantity for human and industrial activities. Therefore, strict water quality (WQ) management is a key technical element of a WDS.

South Korea established legal management standards for WDSs in 1963 and has since been continuously monitoring and managing various properties that determine the WQ (e.g., pH, dissolved oxygen (DO), turbidity, and total

* Corresponding author.

organic carbon). Furthermore, the National Auto Water Quality Monitoring System (NAWQMS) has been established as a comprehensive system for real-time monitoring, protecting water resources, and issuing contamination warnings (CWs) in WDSs [1–3]. South Korea's NAWQMS provides the advantage of delivering real-time and continuous WQ measurement results by combining a telemonitoring system and supervisory control and data acquisition (SCADA). A CW is triggered when real-time WQ measurements exceed the established legal threshold. However, the U.S. Environmental Protection Agency (EPA) has identified a significant issue with this approach—a tendency for this threshold-based system to generate an abundance of false-positive alerts in the WDS, leading to unnecessary repairs and unexpected, sudden shifts in hydraulics; even more troubling is the risk of catastrophic false-negative errors. These errors can have serious implications as they can lead to the unnoticed beginnings of pollution or micro-contamination that sits below the established thresholds. Essentially, this flaw in the system could allow substantial water contamination to remain undetected, failing to set off the necessary CW [4,5], which is a critical issue that demands urgent attention and emphasizes the need for additional research to devise more effective solutions.

Numerous studies have been conducted based on the prediction models for WQ changes to compensate for the limitations of issuing CWs using the thresholds in WDSs. The EPA developed a linear prediction-correction filter (LPCF) model based on the autoregressive model—a traditional time-series model—and presented a WQ prediction methodology. Park et al. [6] applied the autoregressive integrated moving average (ARIMA) model to improve the LPCF model. Recently, the development of WQ prediction models using artificial neural networks (ANNs) has also been actively performed. Zhao et al. [7] and Salami et al. [8] used ANNs to predict the WQ of wastewater treatment plants and rivers.

Among the ANNs, recurrent neural networks (RNNs) are suitable for time-series prediction and can accurately predict the WQ. Wang et al. [9] conducted a study on the prediction of heavy metal content in rivers using long short-term memory (LSTM), which is a type of RNN. Liu et al. [10] applied LSTM for the prediction of DO in a WDS, and Liang et al. [11] used an LSTM for the prediction of chlorophyll a (Chl-a) in water streams.

Research on improving the performance of RNNs for WQ prediction has been largely conducted using two methods, that is, combining different ANNs and using feature engineering techniques. Yang et al. [12] and Barzegar et al. [13] combined a convolutional neural network and LSTM models to predict the pH and NH_3–N in mangrove areas and the DO and Chl-a in lakes. They reported that the prediction performance of the combined model was significantly improved compared with that of the LSTM model.

Feature engineering is a technique used for transforming the original data into appropriate features by utilizing the domain knowledge of the original data to improve the prediction performance of an ANN. In the case of an RNN, univariate or multivariate time-series data are used as original data; consequently, feature engineering is applied by decomposing or synthesizing the time series into multiple frequencies. Traditional time-series decomposition methodologies include the fast Fourier transform (FFT) and wavelet transform, which have limitations in managing abnormal time series or time series that exhibit nonlinear fluctuations. By contrast, empirical mode decomposition (EMD), which was proposed by Huang et al. [14], has the advantage of improved robustness compared with the FFT and wavelet transform as it decomposes nonlinear nonstationary time series into a finite number of intrinsic mode functions (IMFs). However, EMD has a disadvantage: when discontinuous signals, such as impact signals, are distributed in the original data, a modal mixing occurs in which the signals are not separated into appropriate IMFs but rather get mixed within multiple IMFs. To compensate for this shortcoming of a conventional EMD, Wu and Huang [15] proposed ensemble empirical mode decomposition (EEMD), in which Gaussian noise is added to the original data and EMD is repeated. In a study on WQ prediction using EEMD and LSTM, Zhang et al. [16] decomposed the observed DO in a river into the IMFs using EEMD and inputted each IMF to an independent LSTM model. In addition, Sha et al. [17] proposed converting the decomposed IMFs into two-dimensional images and inputting them into a single LSTM model.

This study aimed to formulate a preemptive CW method for the real-time WQ observed at the NAWQMS in South Korea by employing a strategy of feature engineering based on EEMD and LSTM networks to minimize false-negative errors, that is, the critical challenge in the CW process. Through feature engineering, different features were established by reconstructing the IMFs decomposed via EEMD, with a synthesized component created from these IMFs. Considering the unique characteristics of each developed feature, an LSTM model was designed. This research sought to compare short-term and long-term predictive performances for WQ and ultimately aimed to propose a methodology for developing the most effective WQ prediction model.

2. Theoretical background

2.1. Ensemble empirical mode decomposition

EMD is a technique used for decomposing nonlinear and nonstationary time-series data into a finite number of IMFs, wherein the IMFs must satisfy two conditions: (1) the number of extrema in the entire time-series data must be equal to the number of zero crossings or differ by at least one; and (2) at any point, the mean value of the envelope defined by the local maxima and minima must be zero. If the raw time series is $x(t)$, we can obtain the upper envelope $u_1(t)$ and the lower envelope $l_1(t)$ by connecting all the maxima and minima, and the mean value $m_1(t)$ for these envelopes is given as:

$$m_1(t) = \frac{u_1(t) + l_1(t)}{2} \tag{1}$$

The $m_1(t)$ obtained from Eq. (1) is subtracted from the raw time series $x(t)$ to obtain the initial $h_1(t)$ as follows:

$$h_1(t) = x(t) - m_1(t) \tag{2}$$

If $h_1(t)$ obtained from Eq. (2) satisfies the two conditions for the IMFs, it becomes the first IMF $c_1(t)$; otherwise, the shifting process is repeated j times until $h_1(t)$ satisfies the IMF conditions as follows:

$$h_{1,j}\left(t\right)=h_1\left(t\right)+m_{1,j}\left(t\right) \tag{3}$$

Once the first IMF $c_1(t)$ is determined through the above process, it can be subtracted from the raw time series $x(t)$ to obtain the first residue $r_1(t)$ as follows:

$$r_1\left(t\right)=x\left(t\right)-c_1\left(t\right) \tag{4}$$

The above process is repeated for $r_1(t)$ obtained from Eq. (4) until $r_i(t)$ becomes a monotone function or the number of extrema is less than one or equal to one such that no more IMFs can be extracted. Finally, when n IMF $c_n(t)$ values are obtained, the raw time series $x(t)$ is equal to Eq. (5), which is the result of EMD.

$$x\left(t\right)=\sum_{i=1}^{n}c_n\left(t\right)+r_n\left(t\right) \tag{5}$$

However, as mentioned in the previous section, simply applying EMD to time-series decomposition leads to modal mixing, where the signal components of each IMF are mixed with each other. Therefore, Wu and Huang [15] proposed an EEMD technique in which white noise $w(t)$ is added to the raw time series $x(t)$ as follows:

$$X\left(t\right)=x\left(t\right)+w\left(t\right) \tag{6}$$

The EEMD technique can solve the problem of modal mixing by adding white noise $w(t)$ k times in the process, as given by Eq. (6), and repeating the EMD process to obtain an IMF by considering the ensemble average as follows:

$$c_i=\frac{1}{m}\sum_{m}^{1}c_{i,k}\quad\left(i=1,2,\ldots,n;k=1,2,\ldots,m\right) \tag{7}$$

2.2. Long short-term memory

Traditional statistical techniques such as ARIMA and deep-learning models, including RNNs, are commonly used to forecast the time series.

In particular, RNNs are characterized by a recurrent structure of interconnected units. In contrast to the conventional feedforward neural networks, an RNN can effectively process time-series data by using an internal hidden state to pass the output value of the hidden state at a certain timestep (t) to the hidden state at the next timestep (t + 1).

Eq. (8) indicates how the output value h_{t-1} of the hidden state at a certain timestep is passed to the hidden state at the next timestep.

$$h_t=\tan h\left(W_{\mathrm{hx}}x_t+W_{\mathrm{hh}}h_{t-1}+b_h\right) \tag{8}$$

where x_t is the input value at the current timestep t; h_{t-1} is the output value of the hidden state at the previous timestep t – 1; W_{hx} and W_{hh} are the weights multiplied by the input value at the current timestep and the output value of the hidden state at the previous timestep, respectively; b_h is the bias; and tanh is an activation function called the hyperbolic tangent.

However, the RNNs have the inherent problem of gradient vanishing, whereby the gradient of each timestep is multiplied as the sequence time of the time series increases and long-term information is lost.

Therefore, LSTM was proposed to improve the gradient vanishing problem of the RNN. LSTM introduces the concepts of cell state, input gate, forget gate, and output gate in RNNs, as shown in Fig. 1. Thus, it can alleviate the gradient vanishing problem by learning how much to forget and fully remember the information of the previous timestep.

3. Study area and procedures

3.1. Study area

The target area of this study was the G_water treatment plant (G_WTP) located in the Gyeonggi province, South Korea. The G_WTP uses sand filtration processes, and its water treatment capacity is 250,000 m^3/d. In each process of the G_WTP, the turbidity, residual chlorine, water temperature, pH, and electrical conductivity (EC) are measured in real time, and the data are automatically managed using a remote-control system and SCADA.

In this study, an LSTM was developed targeting the pH and EC among the five WQ indicators being measured at G_WTP. To improve the predictive performance of the LSTM, feature engineering using EEMD was applied. The measurement point for the pH and EC was the outlet of the filtration process of the G_WTP. The measurement interval for each WQ indicator was 1 min, and the data collection period and statistical characteristics are presented in Table 1.

3.2. Study procedures

In this study, EEMD-based feature engineering was performed to improve the performance of the LSTM model for pH and EC prediction. A flowchart of our methodology is shown in Fig. 2.

First, the original pH and EC data collected by the G_WPT were decomposed into n IMFs and residues by performing EEMD after adjusting the number of ensemble members and the amplitude of white noise.

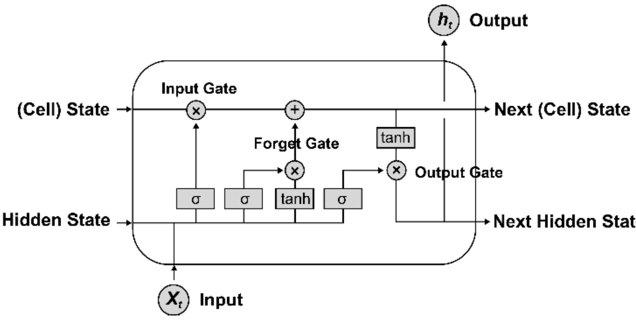

Fig. 1. Memory cell structure of the long short-term memory.

Table 1
Descriptive statistics of water quality data

Water quality	Observation dates	Number of data samples	Minimum	Median	Maximum	Mean	Standard deviation
pH	Jan/01/2019, 00:00–Jul/30/2021, 23:00	22,608	6.04	7.21	7.88	7.19	0.22
Electrical conductivity (μS/cm)	Jan/01/2019, 00:00–Jul/30/2021, 23:00	22,608	87.21	187.68	304.69	192.34	36.9

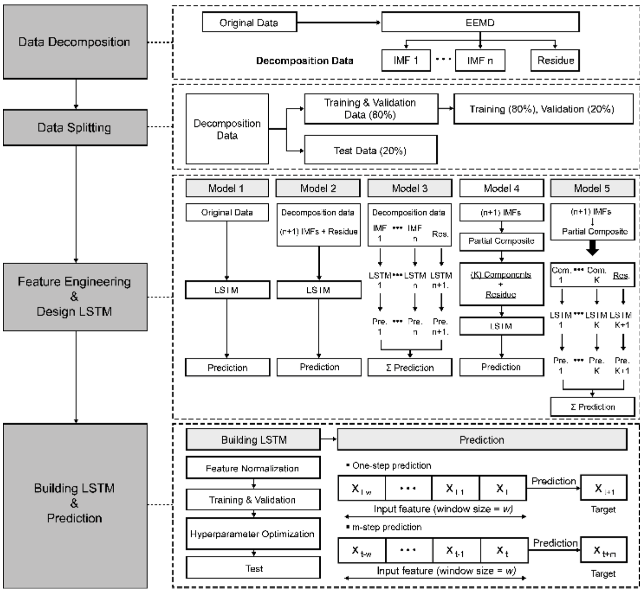

Fig. 2. Research flowchart for feature engineering and building the long short-term memory model.

Second, the original data of each WQ, along with the IMFs and residues decomposed via EEMD, were utilized as input features of the LSTM model. Data splitting was performed for training, validation, and testing, as shown in Fig. 2. The observation period for the pH and EC was from 1 January 2019, 00:00, to 30 July 2021, 23:00 (Table 1), and the measurement interval was 1 min; thus, the total number of data samples was 22,608. From these data, 18,288 samples (1 January 2019, 00:00–31 January 2021, 23:00, 80% of the total) were used for training and validation, and 4,320 samples (1 February 2021, 00:00–30 July 2021, 23:00, 20% of the total) were used for testing.

Third, considering the batch, the feature input into each LSTM consisted of the tensors expressed by the following equation:

$$X \in R^{w,v,b} \tag{9}$$

where X is a tensor for the feature of LSTM; w represents the window size (or sequence time); v represents the number of features; and b represents the batch size.

To evaluate the effectiveness of feature engineering using EEMD, four types of input features were developed in addition to the original data for each WQ. The characteristics of each feature and the development process of the LSTM are as follows:

- Model 1: the original WQ data were composed of features, which were input into a single LSTM. As each WQ was a one-dimensional (1D) time series, the input feature was a tensor with dimensions of $w \times 1 \times b$ ($X \in R^{w,1,b}$).
- Model 2: the n IMFs and residues decomposed using EEMD constituted one feature and were input into a single LSTM. As the n IMFs and residues were both 1D time series, the input feature was a tensor with dimensions of $w \times (n + 1) \times b$ ($X \in R^{w,n+1,b}$).
- Model 3: the n IMFs and residues decomposed via EEMD were organized into independent features and input into each LSTM. As the IMFs and residues input into each LSTM were 1D time series, each input feature was a tensor with dimensions of $w \times 1 \times b$ ($X \in R^{w,1,b}$). Thus, the final WQ prediction was calculated by summing the predictions from (n + 1) independent LSTM models.
- Model 4: the average period of n IMFs and the correlation coefficient for the original data were calculated. Subsequently, according to the average period and correlation coefficient, the IMFs were categorized into K groups and summed to create new components. In model 4, the K components and residuals were organized into one feature and input into a single LSTM model. As the K components and residuals were 1D time-series data, the input feature was a tensor with dimensions of $w \times (K + 1) \times b$ ($X \in R^{w,n+1,b}$).
- Model 5: each of the K components and residues was composed of independent features and input into an individual LSTM. As the components and residues input into each LSTM were 1D time series, the input feature was a tensor with dimensions of $w \times 1 \times b$ ($X \in R^{w,1,b}$). The final WQ prediction was then calculated by summing the predictions from (K + 1) independent LSTM models.

Finally, for training and validating the LSTM model, feature scaling was performed using min–max normalization as follows:

$$\hat{x} = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \tag{10}$$

where $\hat{x}$ represents normalized data; $x_{\max}$ represents the maximum value of the data; and $x_{\min}$ represents the minimum value.

In addition, the hyperparameters of the LSTM were optimized through the training and validation processes. The mean squared error (MSE), expressed in (11), was used as the loss function for fitting the LSTM.

Time-series prediction using LSTM can be categorized into one-step prediction, that is, predicting the next timestep (t + 1) from the current timestep (t), and multistep prediction, that is, predicting the time after m hours (t + m). In this study, we utilized a sliding window methodology for both single-step and multi-step forecasts, enabling a detailed examination of the influence of feature engineering on the LSTM model's short-term and long-term predictive performances. The performance of each model was evaluated using four test criteria: the root-mean-square error (RMSE), mean absolute error (MAE), mean absolute percentage error (MAPE), and Pearson's correlation coefficient (CC), which are expressed as follows:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(T_i^o - T_i^p\right)^2 \tag{11}$$

$$\text{RMSE} = \sqrt{\text{MSE}} \tag{12}$$

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}\left|T_i^o - T_i^p\right| \tag{13}$$

$$\text{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{T_i^o - T_i^p}{T_i^o}\right| \times 100 \tag{14}$$

$$\text{CC} = \frac{\sum_{i=1}^{n}\left(T_i^o - \overline{T_i^o}\right)\left(T_i^p - \overline{T_i^p}\right)}{\sqrt{\sum_{i=1}^{n}\left(T_i^o - \overline{T_i^o}\right)^2}\sqrt{\sum_{i=1}^{n}\left(T_i^p - \overline{T_i^p}\right)^2}} \tag{15}$$

where T_i^o and T_i^p represent the original and predicted WQ data at time i, respectively; $\overline{T_i^o}$ and $\overline{T_i^p}$ represent the mean values of the original and predicted WQ data at time i, respectively; and n represents the number of data sample points.

4. Results

4.1. Feature engineering using EEMD

In this study, the original data of the pH and EC collected from the G_WTP were utilized as input features of the LSTM. For further feature development, EEMD was performed with the number of ensembles set to k = 1,000 and the amplitude of white noise $w(t)$ set to 0.2 times the standard deviation of the original data. All the original data were decomposed into IMFs and residues.

Fig. 3a and b show the EEMD results for the pH and EC, respectively. The original data were decomposed into nine IMFs and residues. Among the decomposed IMFs, IMF 1 represents the highest frequency signal, whereas IMF 9 represents the lowest frequency signal.

Each IMF and residue decomposed using EEMD was arranged as a tensor and utilized as an input feature for the LSTM. Additionally, to develop additional features using the partial synthesis of the IMFs, we calculated the average period for each IMF as well as the CC between each IMF and the original data. The results are presented in Table 2.

An analysis of the characteristics of the pH-decomposed IMFs revealed that IMFs 1–3 were the components that represented short-term periods of ≤24 h, and the CCs for the original data were ≤0.10. Whereas IMFs 4–7 represented the periods ranging from 88 h (3.7 d) to 1,025 h (42.7 d), and the correlation coefficients for the original

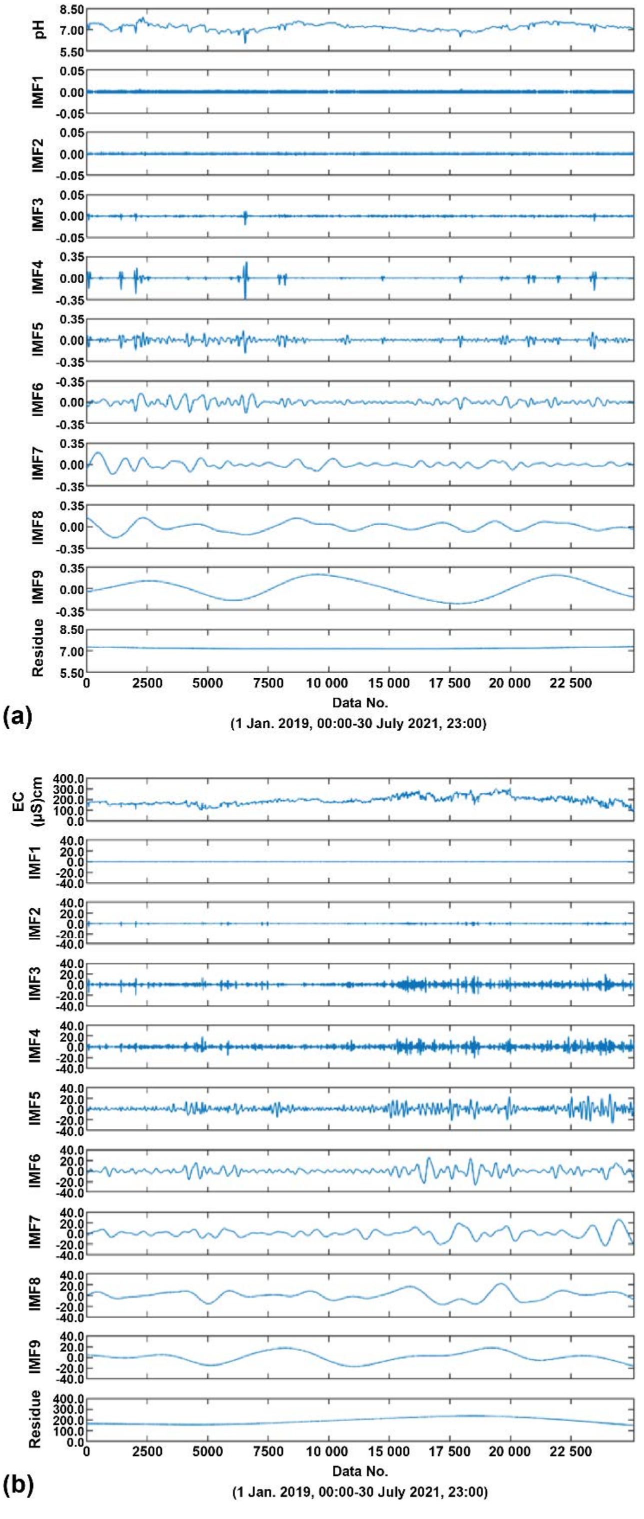

Fig. 3. Original water quality time series and components decomposed via ensemble empirical mode decomposition: (a) pH and (b) electrical conductivity.

Table 2
Decomposition results for the pH obtained using ensemble empirical mode decomposition

Water quality	Signal	Minimum	Median	Maximum	Mean period (h)	CC
pH	IMF 1	–0.005	0.000	0.007	3	0.01
	IMF 2	–0.005	0.000	0.004	9	0.01
	IMF 3	–0.021	0.000	0.011	22	0.09
	IMF 4	–0.394	0.000	0.259	88	0.22
	IMF 5	–0.214	0.001	0.152	159	0.30
	IMF 6	–0.176	0.001	0.141	402	0.34
	IMF 7	–0.155	–0.002	0.192	1,025	0.37
	IMF 8	–0.178	–0.004	0.148	2,382	0.60
	IMF 9	–0.244	0.017	0.230	7,350	0.80
	Residue	7.143	7.162	7.295	–	–
Electrical conductivity	IMF 1	–0.543	0.000	0.298	9	0.04
	IMF 2	–7.090	0.000	4.826	15	0.07
	IMF 3	–19.819	0.000	19.758	25	0.12
	IMF 4	–21.036	0.000	18.907	59	0.18
	IMF 5	–27.234	0.001	27.346	159	0.26
	IMF 6	–25.361	0.001	25.425	450	0.27
	IMF 7	–23.032	–0.002	25.972	900	0.31
	IMF 8	–16.872	–0.004	21.952	2,260	0.23
	IMF 9	–16.396	0.017	18.427	5,130	0.48
	Residue	150.438	7.162	237.996	–	–

data were 0.22–0.37. Furthermore, IMFs 8 and 9 represented long-term periods of 2,382 h (99.2 d) and 7,350 h (306.3 d), and their correlation coefficients for the original data were 0.60 and 0.80, respectively.

As with the pH, the EC was decomposed into nine IMFs, and the average periods of the IMFs were similar. IMFs 1–3 had average periods of ≤25 h, and the correlation coefficients for the original data were ≤0.12. The average periods of IMFs 4–7 ranged from 59 h (2.4 d) to 900 h (37.4 d), and the correlation coefficients for the original data ranged from 0.18–0.31. IMFs 8 and 9 had average periods of 2,260 h (94.16 d) and 5,130 h (213.8 d), and their correlation coefficients for the original data were 0.23 and 0.48, respectively.

Herein, a novel input feature was developed by partially synthesizing the IMFs of the pH and EC from the average period and correlation coefficient of each IMF presented in Table 2. The results are shown in Fig. 4.

Fig. 4a shows the partial synthesis of IMFs of the pH decomposed via EEMD and the input features developed using the residue. Component 1 was generated by synthesizing IMFs 1–3, which represented short-term periods of <1 day. Component 2 was generated by synthesizing IMFs 4–7, which represented periods of <3 months. Components 3–5 were generated from IMFs 8 and 9 and the residue, respectively, which represented long-term periods of ≥3 months. Fig. 4b shows the partial synthesis of IMFs of the EC decomposed via EEMD and the composition of input features developed using the residue. Each component was generated using the same method that was used for the pH.

4.2. Development of LSTM and validation results

Hyperparameter tuning, such as determining the size of the hidden layer and the learning rate, is required to develop the LSTM model. Herein, a single hidden layer was used to simplify the LSTM model. The hyperparameters are presented in Table 3.

As described previously, five features were constructed via the partial synthesis of IMFs, residues, and IMFs decomposed using EEMD along with the original data of each WQ, and these were input into an independent LSTM model. The training and validation data (18,288 samples from 1 January 2019, 00:00 to 31 January 2021, 23:00) classified by the input features were used to optimize the hyperparameters of the LSTM model presented in Table 3. The window size, that is, the sequence time of the input feature, had an optimal value of 24, the average number of neurons in the hidden layer had an optimal value of 100, and the learning rate and batch size had optimal values of 0.01 and 64, respectively. Subsequently, the MSE was applied as the loss function for fitting each LSTM model.

Fig. 5a shows the validation results of models 1–5 developed for predicting the pH at each prediction step, wherein the RMSE was used instead of the MSE. In model 1, the original pH data were organized into a tensor with dimensions of 24 × 1 × 64 ($w \times v \times b$) and input into a single LSTM model. Through validation, the RMSE was calculated to be 0.003–0.042 for prediction steps of 1–48 h. In model 2, the 10 signals decomposed via EEMD were organized into a tensor with dimensions of 24 × 10 × 64 and input into a single LSTM model; the RMSE for each prediction step was calculated to be 0.010–0.071. In model 3, the nine IMFs and residues decomposed via EEMD were each organized into a tensor with dimensions of 24 × 1 × 64 and fed into 10 independent LSTM models; the RMSE for each prediction step was calculated to be 0.004–0.020. In model 4, the five components generated by the partial synthesis of IMFs were organized into a tensor with dimensions of 24 × 5 × 64 and fed into a single LSTM model; the RMSE for each prediction step was calculated to be 0.004–0.044. In model 5, the

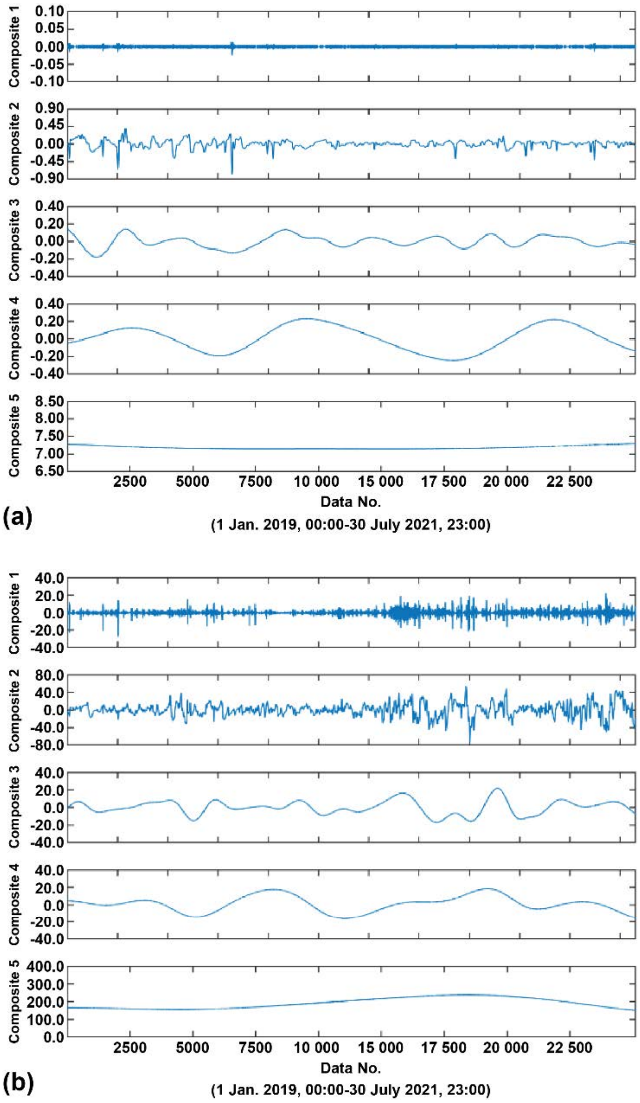

Fig. 4. Results of the partial synthesis using intrinsic mode functions: (a) pH and (b) electrical conductivity.

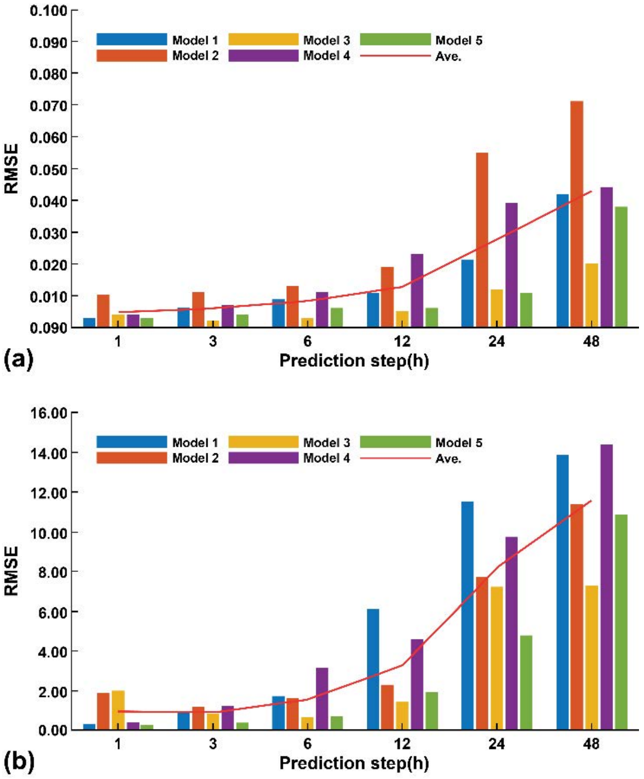

Fig. 5. Validation results of the long short-term memory models at each prediction step: (a) pH and (b) electrical conductivity.

Table 3
Hyperparameters for long short-term memory development

Hyperparameter	Long short-term memory Model				
	Model 1	Model 2	Model 3	Model 4	Model 5
Number of hidden layers	1				
Number of neurons	10–150				
Gate activation	Sigmoid				
Recurrent activation	Tanh				
Learning rate	0.1–0.001				
Batch size	32–256				
Number of epochs	100–500				
Window size	6, 12, 24, 48, 60, 168				
Optimizer	Adam				
Loss function	MSE				

five components generated by partial synthesis were each organized into a tensor with dimensions of 24 × 1 × 64 and fed into five independent LSTM models; the RMSE for each prediction step was calculated to be 0.003–0.038.

Fig. 5b shows the validation results of each model developed for EC prediction for each prediction step, wherein the MSE was recalculated as the RMSE. The EC data were decomposed into nine IMFs and residues using EEMD, similar to the pH, and the average period of the IMFs was similar to that for the pH. Thus, the input features and LSTM configuration for the EC were identical to those for the pH. Subsequently, the validation of models 1–5 was performed at each prediction step, and the RMSE was calculated as 0.309–13.865, 1.849–11.409, 1.968–7.306, 0.388–14.388, and 0.277–10.807 for models 1–5, respectively.

4.3. Test results

To evaluate the prediction performance of models 1–5 for each WQ fitted through training and validation, a test was performed. The test data were applied to 4,320 samples from the period of 01 February 2021, 00:00–30 July 2021, 23:00 among the original data of each WQ indicator.

In addition, the four test criteria defined by Eqs. (15)–(18) were used to evaluate and compare the prediction performance of the LSTM developed for different input features. In general, the RMSE, MAE, and MAPE are used to measure the difference between the predicted and measured values. These values being close to 0 indicates a small difference between the predicted and measured values, thus suggesting that the model has a high prediction accuracy. CC represents a linear relationship between the predicted and measured values; CC close to 0 indicates a meaningless correlation between the predicted and measured values. CCs close to –1 and 1 indicate strong positive and negative correlations, respectively.

The test results for models 1–5 developed for predicting the pH are presented in Table 4. In the prediction step of 1 h, which involved predicting the next timestep (t + 1) from the current timestep (t), the CCs of models 1–5 were all >0.90. This indicates that the predicted values had strong positive correlations with the measured values. For models 1, 3, and 5, their RMSE was between 0.003 and 0.004, MAE was between 0.002 and 0.003, and MAPE was between 0.03 and 0.05, thus exhibiting similar results. In comparison, for models 2 and 4, the RMSE (0.011, 0.006), MAE (0.011, 0.005), and MAPE (14.8, 7.5) were relatively large.

The m-step prediction, which involves predicting m hours from the current timestep t, was performed with prediction steps of 3, 6, 12, 24, and 48 h for convenience. The results revealed that, as the prediction step increased, the RMSE, MAE, and MAPE increased, whereas the CC decreased for models 1 and 5. For the prediction step of 48 h, the RMSE (0.030–0.155), MAE (0.022–0.148), and MAPE (0.29–2.06) of each model were maximized, whereas the CC was reduced to 0.85–0.98.

Overall, models 3 and 5 exhibited better prediction performance than the other models for all the prediction steps. Additionally, the changes in the test criteria as the prediction step increased were the smallest for these models. Overall, models 2 and 4 exhibited a lower prediction performance than model 1.

Table 5 presents the test results for models 1–5 developed for EC prediction. For the prediction step of 1 h, the CCs of models 1–5 were all >0.90, thus indicating a strong positive correlation with the measured values. For models 1, 4, and 5, the RMSE, MAE, and MAPE ranged from 0.207–0.277, 0.149–0.265, and 0.09–0.14, respectively, thus exhibiting similar results. However, for models 2 and 3, the RMSE (1.229, 1.968), MAE (1.182, 0.707), and MAPE (0.63, 0.39) were relatively large. For prediction steps of 3–48 h, as the prediction step increased, the RMSE, MAE, and MAPE of each model increased, whereas the CC decreased, similar to the results for the pH. For the prediction step of 48 h, the RMSE, MAE, and MAPE were maximized, whereas the CC was minimized for all the models.

A comparison of the models with different prediction steps revealed that models 3 and 5 had relatively good prediction performance and exhibited minor changes in the test criteria, except for the prediction steps of 1 and 48 h. By contrast, models 1, 2, and 4 exhibited relatively

Table 4
RMSE, MAE, MAPE, and CC calculation results in the case of pH data for each Long short-term memory model and prediction step

Prediction step (h)	Model	RMSE	MAE	MAPE	CC
1	1	0.003	0.002	0.03	1.00
	2	0.011	0.011	0.15	1.00
	3	0.004	0.003	0.05	1.00
	4	0.006	0.005	0.07	1.00
	5	0.003	0.002	0.03	1.00
3	1	0.006	0.005	0.06	1.00
	2	0.039	0.037	0.50	1.00
	3	0.002	0.002	0.02	1.00
	4	0.009	0.006	0.09	1.00
	5	0.003	0.002	0.03	1.00
6	1	0.031	0.029	0.39	1.00
	2	0.029	0.028	0.38	1.00
	3	0.003	0.002	0.03	1.00
	4	0.013	0.009	0.12	1.00
	5	0.008	0.007	0.10	1.00
12	1	0.015	0.011	0.15	1.00
	2	0.030	0.016	0.23	0.98
	3	0.007	0.006	0.08	1.00
	4	0.029	0.016	0.22	0.98
	5	0.006	0.004	0.06	1.00
24	1	0.035	0.023	0.31	0.97
	2	0.072	0.069	0.93	0.99
	3	0.019	0.014	0.20	0.99
	4	0.067	0.050	0.68	0.93
	5	0.021	0.013	0.17	0.99
48	1	0.076	0.048	0.66	0.87
	2	0.155	0.148	2.06	0.94
	3	0.030	0.022	0.29	0.98
	4	0.081	0.049	0.66	0.85
	5	0.064	0.041	0.56	0.92

Table 5
RMSE, MAE, MAPE, and CC calculation results in the case of electrical conductivity data for each Long short-term memory model and prediction step

Prediction step (h)	Model	RMSE	MAE	MAPE	CC
1	1	0.207	0.149	0.09	1.00
	2	1.229	1.182	0.63	1.00
	3	1.968	0.707	0.39	1.00
	4	0.276	0.200	0.12	1.00
	5	0.277	0.265	0.14	1.00
3	1	1.196	0.901	0.51	1.00
	2	1.012	0.599	0.38	1.00
	3	0.859	0.666	0.37	1.00
	4	0.862	0.658	0.38	1.00
	5	0.512	0.387	0.22	1.00
6	1	2.700	1.965	1.11	1.00
	2	1.881	1.297	0.77	1.00
	3	0.709	0.549	0.31	1.00
	4	3.377	2.722	1.55	1.00
	5	1.183	0.922	0.51	1.00
12	1	9.475	7.101	4.03	0.95
	2	2.554	1.805	1.03	1.00
	3	2.213	1.502	0.88	1.00
	4	5.317	4.012	2.25	0.98
	5	2.522	1.916	1.10	1.00
24	1	15.038	11.145	6.32	0.87
	2	11.124	7.647	4.33	0.95
	3	10.431	6.254	3.45	0.93
	4	12.706	9.502	5.30	0.91
	5	9.387	7.153	4.03	0.95
48	1	20.549	15.353	8.61	0.74
	2	11.637	8.643	4.71	0.94
	3	11.032	8.181	4.61	0.93
	4	17.346	13.789	7.69	0.82
	5	17.047	13.294	7.53	0.83

large changes in the test criteria as the prediction step changed.

The test results for the pH and EC revealed that models 3 and 5, which decomposed the original data via EEMD and input each signal into an independent LSTM to obtain the final prediction value, demonstrated relatively good prediction performance. No significant degradation was observed in the prediction performance with an increase in the prediction step. Furthermore, models 2 and 4, in which each signal decomposed via EEMD was organized into a single tensor and input into a single LSTM model, did not exhibit a significant improvement in prediction performance compared with model 1, in which the original data were input into a single LSTM model.

To further analyze the improvement in the prediction performance of the LSTM in models 3 and 5, the actual test data and the predicted values from each model were compared as shown in Figs. 6 and 7.

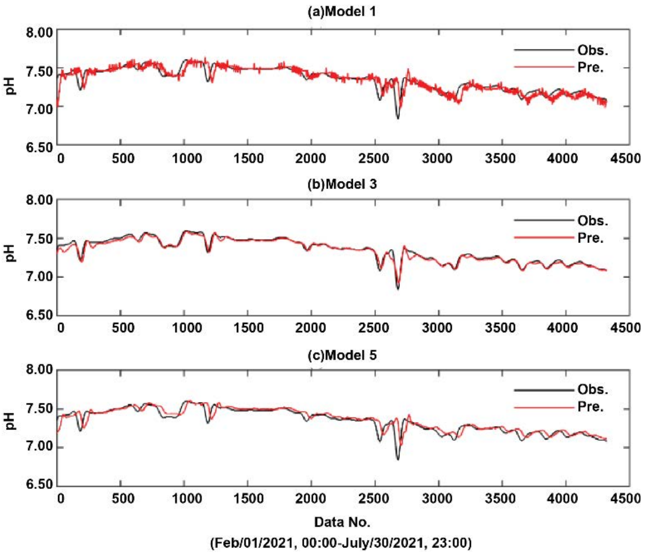

Fig. 6. Simulation results for 48-h-ahead forecasting using ensemble empirical mode decomposition and long short-term memory for the pH test data: (a) model 1, (b) model 3 and (c) model 5.

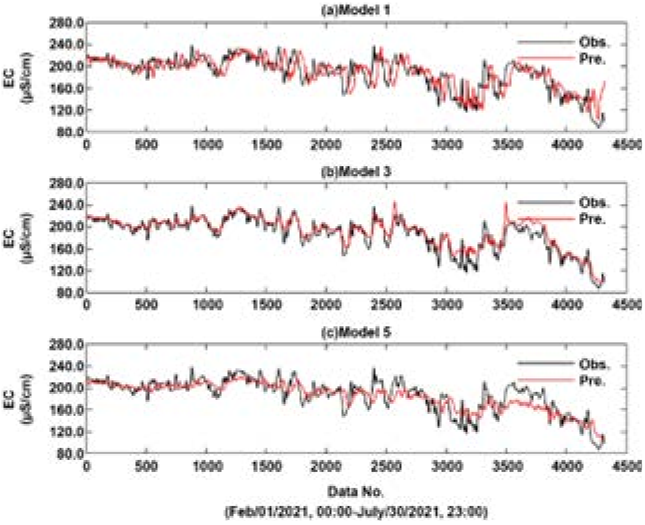

Fig. 7. Simulation results for 48-h-ahead forecasting using ensemble empirical mode decomposition and long short-term memory for the electrical conductivity test data: (a) model 1, (b) model 3 and (c) model 5.

Fig. 6 shows the measured values of the pH test data for the prediction step of 48 h along with the predicted values of models 1, 3, and 5. As shown in Fig. 6a, the predicted values of model 1 captured the overall fluctuation of the measured values well but exhibited strong high-frequency variability. Furthermore, as reported by Zhang et al. [16], Kratzert et al. [18], and Xiang et al. [19], a distinct delay was observed between the measured and predicted values for model 1, in which the original data were fed into a single LSTM model. By contrast, model 3 (Fig. 6b), in which the IMFs and residues decomposed via EEMD were input into each independent LSTM, did not exhibit large high-frequency fluctuations in the predicted values owing to cancel-out, and the delay between the measured and predicted values was resolved. Model 5 (Fig. 6c), in which the components generated by the partial synthesis of IMFs were fed into each independent LSTM, did not exhibit large high-frequency fluctuations in the predicted values owing to cancel-out, similar to model 3. However, the problem of delay in the predicted values remained.

Fig. 7 shows the measured values of the EC test data for the prediction step of 48 h, along with the predicted values of models 1, 3, and 5. Similar to the pH test results, model 1 exhibited larger fluctuations in the high-frequency components compared with models 3 and 5 along with a delay in the predicted value (Fig. 7a). Model 3 simulated the fluctuation of the measured values well (Fig. 7b), whereas model 5 exhibited large differences between the measured and predicted values.

5. Conclusion

In this study, we developed a proactive CW plan for the WQ observed in real time at NAWQMS in South Korea. LSTM was used to predict the pH and EC observed in real time at the G_WTP. To improve the prediction performance of LSTM, feature engineering using EEMD was applied. The original data of the pH and EC were 1D time series and were decomposed into IMFs and residues through EEMD. Subsequently, four additional features were developed using EEMD, along with the features composed of the original data. The LSTM models were developed according to the configuration of each feature. The main results of this study are as follows:

- In model 1, the original 1D data were organized into features using a conventional method, and they were input into a single LSTM model. The final analysis using the test data indicated that, for the prediction step of 1 h, which involved predicting the next timestep (t + 1) from the current timestep (t), the prediction performance of model 1 did not differ significantly from that of models 2–5 to which feature engineering was applied. However, the overall prediction performance deteriorated and fluctuated significantly as the prediction step increased.
- In models 2–5, the original data were decomposed into nine IMFs and residues for feature engineering using EEMD. Model 2 organized each decomposed signal into a single tensor and fed it into a single LSTM model. Model 3 organized each decomposed signal into independent features and fed them into individual LSTM models. By contrast model 4 organized the five components generated by the partial synthesis of IMFs into a single tensor, which was fed into a single LSTM model. In the final analysis using the test data, models 2 and 4 with a single LSTM model did not exhibit significant improvements in the prediction performance compared with model 1. As the prediction step increased, the prediction performance deteriorated and fluctuated significantly. By contrast, models 3 and 5, in which each feature generated through EEMD was input into an independent LSTM model, exhibited higher prediction performance compared with the other models for all the prediction steps. The prediction performance did not change significantly as the prediction step increased.
- To further analyze the improvement in the prediction performance for models 3 and 5, their predicted values were compared with the measured values of the test data, along with the predicted values of model 1, for the prediction step of 48 h. The predicted values of model 1 simulated the fluctuations of the measured values well but exhibited a strong high-frequency variability. This resulted in a prediction delay, which has been reported as a drawback of conventional LSTM models. However, for models 3 and 5, strong high-frequency fluctuations did not occur in the predicted values owing to cancel-out, which occurred during the process of summing the predicted results of independent LSTM models to obtain the final predicted value. Furthermore, model 3, which was composed of more LSTM models than model 5, was analyzed to mitigate the problem of prediction delay.

Based on the results of this study, we conclude that LSTM is useful for the prediction of the observed WQ in WDSs. Furthermore, the application of EEMD-based feature engineering is expected to improve the stable prediction performance and reduce the prediction delay compared with the conventional method of inputting 1D WQ data to a single LSTM model. However, the process of developing independent LSTM models for IMFs and residues decomposed from EEMDs increases the computational cost. Therefore, in the future, we aim to develop a methodology to minimize the computational cost for EEMD-based feature engineering and further improve the LSTM model by applying the partial synthesis of IMFs attempted in model 5 in various ways.

Acknowledgement

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2020R1I1A1A01070718) and the (2021 Post-Doc. Fellowship Program) of Gyeongsang National University.

References

[1] Korea Ministry of Government Legislation Home Page, 2021. Available at: www.law.go.kr

[2] Korea Environment Corporation. Available at: www.keco.or.kr

[3] S.S. Park, N.-S. Park, S.S. Kim, G. Jo, S.M. Yoon, Outlier detection of water quality data using ensemble empirical mode decomposition, J. Korean Soc. Environ. Eng., 43 (2021) 160–170.

[4] U.S. EPA, Online Water Quality Monitoring Primer for Water Quality Surveillance and Response Systems (EPA 817-B-15–002A), United States Environmental Protection Agency, 2015a. Available at: https://www.epa.gov/sites/default/files/2015-06/documents/online_water_quality_monitoring_primer.pdf

[5] U.S. EPA, Summary of Implementation Approaches and Lessons Learned From the Water Security Initiative Contamination Warning System Pilots (EPA 817-R-15–002), United States Environmental Protection Agency, 2015b. Available at: https://www.epa.gov/sites/default/files/2015-12/documents/wsi_pilot_summary_report_102715.pdf

[6] N.-S. Park, S.-S. Kim, I.S. Seo, S.M. Yoon, Application of LPCF model based on ARIMA model to prediction of water quality change in water supply system, Desal. Water Treat., 212 (2021) 8–16.

[7] Y. Zhao, L. Guo, J. Liang, M. Zhang, Seasonal artificial neural network model for water quality prediction via a clustering analysis method in a wastewater treatment plant of China, Desal. Water Treat., 57 (2016) 3452–3465.

[8] E.S. Salami, M. Salari, M. Ehteshami, N.T. Bidokhti, H. Ghadimi, Application of artificial neural networks and mathematical modeling for the prediction of water quality variables (case study: southwest of Iran), Desal. Water Treat., 57 (2016) 27073–27084.

[9] S. Wang, T. Lou, C. Zhang, J. Hao, Y. Zhan, L. Ping, Prediction of heavy metal content in multivariate chaotic time series based on LSTM, Desal. Water Treat., 197 (2020) 249–260.

[10] P. Liu, J. Wang, A.K. Sangaiah, Y. Xie, X. Yin, Analysis and prediction of water quality using LSTM deep neural networks in IoT environment, Sustainability, 11 (2019), doi: 10.3390/su11072058.

[11] Z. Liang, R. Zou, X. Chen, T. Ren, H. Su, Y. Liu, Simulate the forecast capacity of a complicated water quality model using the long short-term memory approach, J. Hydrol., 581 (2020) 124432, doi: 10.1016/j.jhydrol.2019.124432.

[12] Y. Yang, Q. Xiong, C. Wu, Q. Zou, Y. Yu, H. Yi, M. Gao, A study on water quality prediction by a hybrid CNN-LSTM model with attention mechanism, Environ. Sci. Pollut. Res., 28 (2021) 55129–55139.

[13] R. Barzegar, M.T. Aalami, J. Adamowski, Short-term water quality variable prediction using a hybrid CNN–LSTM deep learning model, Stochastic Environ. Res. Risk Assess., 34 (2020) 415–433.

[14] N.E. Huang, Z. Shen, S.R. Long, M.C. Wu, H.H. Shih, Q. Zheng, N.C. Yen, C.C. Tung, H.H. Liu, The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis, Proc. R. Soc. London, Ser. A, 454 (1998) 903–995.

[15] Z. Wu, N.E. Huang, Ensemble empirical mode decomposition: a noise-assisted data analysis method, Adv. Adapt. Data Anal., 1 (2009) 1–41.

[16] D. Zhang, R. Chang, H. Wang, Y. Wang, H. Wang, S. Chen, Predicting Water Quality Based on EEMD and LSTM Networks, Proc. 2021 33rd Chin. Control Decis. Conf. (CCDC), Kunming, China, 2021, pp. 2372–2377.

[17] J. Sha, X. Li, M. Zhang, Z.-L. Wan, Comparison of forecasting models for real-time monitoring of water quality parameters based on hybrid deep learning neural networks, Water, 13 (2021), doi: 10.3390/w13111547.

[18] F. Kratzert, D. Klotz, C. Brenner, K. Schulz, M. Herrnegger, Rainfall-runoff modelling using long short-term memory (LSTM) networks, Hydrol. Earth Syst. Sci., 22 (2018) 6005–6022.

[19] Z. Xiang, J. Yan, I. Demir, A rainfall-runoff model with LSTM-based sequence-to-sequence learning, Water Resour. Res., 56, (2020), doi: 10.1029/2019WR025326.