Research Article

An Improved Long Short-Term Memory Model for Dam Displacement Prediction

Jun Zhang ¹, Xiyao Cao,² Jiemin Xie,^{1,3} and Pangao Kou¹

¹State Grid Hunan Electric Power Company Limited Research Institute, Changsha, Hunan, China

²State Grid Hunan Electric Power Company Limited, Changsha, Hunan, China

³Hunan Xiangdian Test & Research Institute, Changsha, Hunan, China

Correspondence should be addressed to Jun Zhang; sghnzj1988@163.com

Received 15 January 2019; Revised 27 March 2019; Accepted 11 April 2019; Published 24 April 2019

Academic Editor: Łukasz Jankowski

Copyright © 2019 Jun Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Displacement plays a vital role in dam safety monitoring data, which adequately responds to security risks such as the flood water pressure, extreme temperature, structure deterioration, and bottom bedrock damage. To make accurate predictions, former researchers established various models. However, these models' input variables cannot efficiently reflect the delays between the external environment and displacement. Therefore, a long short-term memory (LSTM) model is proposed to make full use of the historical data to reflect the delays. Furthermore, the LSTM model is improved to optimize the performance by making variables more physically reasonable. Finally, a real-world radial displacement dataset is used to compare the performance of LSTM models, multiple linear regression (MLR), multilayer perceptron (MLP) neural networks, support vector machine (SVM), and boosted regression tree (BRT). The results indicate that (1) the LSTM models can efficiently reflect the delays and make the variables selection more convenient and (2) the improved LSTM model achieves the best performance by optimizing the input form and network structure based on a clearer physical meaning.

1. Introduction

Health monitoring is an effective way to analyze various phenomena through the whole life cycle of the dam, such as displacement, leakage, and vibration. Among all the monitoring data, displacement plays an important role that is easy to measure and understand. Meanwhile, when the dam suffers some security risks such as the flood water pressure, extreme temperature, structure deterioration, and bottom bedrock damage, the variation of displacement can be captured in time. To master the variation rule of dam displacement, predict it accurately, and detect anomalies, researchers established models to reflect the linear or nonlinear relationship between dam displacement and external factors, such as the well-known hydrostatic-season-time (HST) model proposed by Beaujoint in 1967 [1].

In the past, the dam displacement prediction models can be classified as statistical model, deterministic model, hybrid model, and mixed model [2]. The statistical model is established only by observed data. Multiple linear regression

(MLR) and its advanced forms that belong to the statistical model are widely used so far due to its intuition and ease of operation [3–8]. The deterministic model sets up the dam, rock mass, fault, and other solid models by using several assumptions and simplifications such as the constitutive relation, antishear slippage, and boundary conditions. Then, the numerical method, usually finite element method, is used to solve and analyze the model. Complex and uncertainty process makes it hard for the deterministic model to predict the measured data of the dam [9, 10]. The hybrid model can be used to alleviate the problem by feedback adjustment according to the measured data [11, 12]. The mixed model is the mixture of the statistical model and the deterministic model. It deals with the water level factor by using the deterministic model to generate statistical data and the other factors such as the temperature and aging factor by the statistical method.

Recently, the machine learning algorithm has been accepted as a useful tool for modeling complex nonlinear systems and widely used for displacement prediction on dam

bodies. For example, Mata [13] compared neural network (NN) and MLR models for the horizontal displacement in a large Portuguese arch dam. The conclusion is that the NN model showed flexibility and proved to be more adequate for months with extreme temperature than the MLR model. Stojanovic [14] adopted and optimized artificial neural networks (ANN) for given conditions using genetic algorithms (GA) to establish the dam displacement model. The results of the analysis have shown that the ANN/GA model can give rather better accuracy compared to the MLR/GA model. Yu [15] applied the principal component analysis (PCA) method to realize data reduction, lower data redundancy, and reduce noise and false alarm rate in multivariate analysis of dam monitoring data. Su [16] used the support vector machine (SVM) to build a static model and real-time updated model to predict the displacement of an actual roller-compacted dam. Rankovic [17] adopted the support vector regression (SVR) and nonlinear autoregressive with exogenous inputs (NARX) model for the dam tangential displacement prediction. Kang [18] proposed an extreme learning machine- (ELM-) based health monitoring model for displacement prediction of Fengman gravity dams. The study shows that ELM can achieve better prediction performance for the horizontal displacements than backpropagation (BP) neural networks, MLR, and stepwise regression models. F. Salazar [19] used boosted regression trees (BRT), random forests (RF), NN, SVM, and multivariate adaptive regression splines (MARS) models for the displacement analysis on La Baells arch dam. By comparing the radial and tangential displacement and leakage data of dam, the BRT model obtained the best overall performance.

Due to the delays between the external environment and displacement, historical data such as air temperature needs to be used as input variables when the model is established [20]. Previous models adopt complex and diverse input variables to consider the delays, for example, moving average, trigonometric function while its delay function is also a trigonometric function, previous displacement used in the autoregressive (AR) model and its variants including autoregressive with exogenous inputs (ARX) model, NARX model, and singular spectrum analysis with autoregressive model (SSA-AR) [21]. It will bring some confusions, for instance, depending on prior knowledge, multicollinearity, and noncausality [22, 23]. Meanwhile, limited variables will lead to partial loss of the original information or long-term memory impairments during data processing. Therefore, it is formidable for previous models to design the appropriate input variables to deal with the arbitrary, unknown causal delays between the inputs and dam displacement.

The recurrent neural network (RNN) evolves from the Hopfield network of internal feedback connection which can deal with the delays [24, 25]. It is a kind of ANN which allows nodes to be connected into a directed cyclic [26]. The traditional RNN model is subjected to gradient vanishing or blowing up after passing many time steps. The long short-term memory (LSTM) proposed by Hochreiter [27] has successfully solved the problem by using a method called "constant error carousel (CEC)". Meanwhile, stored information and already correct outputs can be protected

against perturbation by using the proposed gate units. Gers [28] improved the LSTM by using the forget gate to know how to segment the input streams into training subsequences without prior knowledge which finally become the most typical LSTM model. Owing to these advantages of LSTM, the historical data of the dam can be fully utilized.

In this study, the LSTM model is applied for dam radial displacement to make accurate prediction through making full use of the historical data such as water level and temperature to reflect the delays. Meanwhile, an improved LSTM model is proposed to optimize the performance by making the variables more physically reasonable. It divides the variables into two parts, the delay variables and no-delay variables, which go through different kinds of hidden nodes including LSTM memory blocks and the normal ones. Results are also compared with that of the MLR, multilayer perceptron (MLP), SVM, and BRT models by using a real-world dataset of a concrete arch dam. The rest of the paper is organized as follows. In Section 2, LSTM model and a pretreatment method for the dam health monitoring (DHM) data are introduced. Then an improved LSTM model is present in detail. In Section 3, the performance of the LSTM, MLR, MLP, SVM, and BRT models is demonstrated. In Section 4, conclusions are provided.

2. LSTM Model

2.1. Model Establishment. The RNN model and LSTM memory block are shown in Figure 1.

LSTM model contains subnetwork called memory block, which is used to replace the hidden layer node in the RNN as shown in Figure 1(a). One memory block consists of a memory cell, a forget gate, input and output squashing units, input and output gates, and input and output gating. Components of the memory block are demonstrated as follows [29].

The input squashing unit uses nonlinear transformation to deal with the memory block output $\mathbf{h}_{t-1}$ at time step $t-1$ and the input $\mathbf{x}_t$ at time step t . They are run through a tanh activation function and then generate the input squashing $\mathbf{g}_t$.

The input gate uses a sigmoid function to transfer the weighted sum of $\mathbf{x}_t$ and $\mathbf{h}_{t-1}$ and then generate $\mathbf{i}_t$. The input gating is the Hadamard product of $\mathbf{i}_t$ and $\mathbf{g}_t$. Thus, the information flow of the input squashing unit cuts off and goes through when the input gate outputs are 0 and 1, respectively. Multiplicative gates are distinctive features of the LSTM model.

The memory cell with a self-connected recurrent edge is the core of the memory block. It is updated by a constant linear function. The self-connected recurrent edge with constant weight 1 is to ensure that the gradient can pass across many time steps without vanishing or exploding. The state of the memory cell is called the internal state $\mathbf{s}$. In vector notation, $\mathbf{s}_t = \mathbf{s}_{t-1} + \mathbf{i}_t \odot \mathbf{g}_t$, where $\odot$ denotes the Hadamard product.

The forget gate uses a sigmoid function to transfer the weighted sum of $\mathbf{x}_t$ and $\mathbf{h}_{t-1}$ and then generate $\mathbf{f}_t$. The internal state uses $\mathbf{f}_t$ to flush the contents. This design is especially

FIGURE 1: (a) RNN; (b) LSTM memory block.

useful in continuously running networks. Thus $s_t = s_{t-1} \odot f_t + i_t \odot g_t$.

The output gate uses a sigmoid function to transfer the weighted sum of x_t and h_{t-1} and then generate o_t .

The output squashing unit transfers the internal state s_t using a tanh function and then generate $\tanh(s_t)$.

The memory block output h_t is the Hadamard product of o_t and the input squashing $\tanh(s_t)$.

2.2. Learning Algorithm of LSTM. The LSTM can contain many hidden layers, and each hidden layer can contain many extended memory blocks, and each memory block can contain one or more memory cells. Here we mainly consider the situation that one memory block only has one memory cell. The backpropagation outside the memory block used the algorithm of backpropagation through time (BPTT) [30]. For the sake of clarity, only the backpropagation inside the memory block at time step t is given in Figure 2.

As can be seen, W^{xg} denotes the weight matrix between x_t and g_t . W^{hg} represents the weight matrix between h_{t-1} and g_t . W^{xi} represents the weight matrix x_t with i_t . W^{hi} represents the weight matrix between h_{t-1} and i_t . W^{xf} represents the weight matrix between x_t and f_t . W^{hf} represents the weight between h_{t-1} and f_t . W^{xo} represents the weight matrix between x_t and o_t . W^{ho} represents the weight matrix between h_{t-1} and o_t . b_g , b_i , b_f , and b_o denote the bias of g_t , i_t , f_t , and o_t , respectively. There are several parameters outside of the memory block: the output layer weight matrix W and bias b , the loss function L , the sample size N , and the last time step k . It is noteworthy that each time step shares the same weight and bias matrices. Therefore, the time steps can be selected long enough to meet demand.

2.3. Pretreatment of Dam Health Monitoring (DHM) Data for LSTM. Unlike the traditional models, LSTM inputs the variable values of different time into the network sequentially, thus adding a sequence-related dimension named time step mentioned above. Primarily, it uses input variable values

of different time to fit displacement. The input for LSTM needs to be processed into three-dimensional data X_{ijv} . The subscript i of X_{ijv} denotes the i -th displacement sample. The subscript j of X_{ijv} means the j -th time step. The last time step k corresponds to the observation time of the displacement sample, and k also represents the length of input subsequence. The time interval of the time step does not need to be consistent with the displacement sampling interval. Since the displacement may be related to the historical daily temperature, the time interval of the time step should be set as one day to meet the requirement. The subscript v of X_{ijv} represents the v -th input variable. For input variables selection, the LSTM model can directly use 3 variables of water level, temperature, and aging factor time in the raw subsequence form due to its memory and nonlinear advantages. Figure 3 manifests the LSTM calculating process of dam displacement. The network structure involves three layers: input layer, hidden layer, and output layer. The input layer is fed by the input X_{ijv} . There is one memory block in the hidden layer. The output layer applies a sigmoid transformation to the memory block output $h_i(t_k)$ at the last time step and yields a response to compute the model output.

2.4. Improved LSTM Model for Dam Displacement. As far as we know, not all input variables need to take into account the delays, such as the aging factor influencing the irreversible deformation of the dam and the water level used in HST model. However, the water level in the past has a particular influence on the temperature distribution and water pressure of the dam and the surrounding rock, which potentially affects the displacement of the dam. Therefore, it is reasonable to consider the delay of the water level, for example, the moving averaged form used by Kang [18]. Meanwhile, the redundancy of the input subsequence will increase the complexity and training time of the model which seems unnecessary.

An improved model is proposed to solve the above problems. The input variables are divided into the delay variables and no-delay variables. The delay variables participate in

FIGURE 2: Backpropagation algorithm flow of LSTM inside the memory block at time step t . FIGURE 3: LSTM calculating process of dam displacement with one hidden layer and one memory block. The water level H , temperature T , and aging factor time θ in the raw subsequence form are used as input subsequences; t_j represents the j -th time step.

LSTM model (see Figure 3) calculation and generate the memory block output $\mathbf{h}_t$, while the no-delay variables merge with $\mathbf{h}_t$ to form new input. Then, a feedforward neural network is established for the new input. One more hidden layer 2 is added to enhance the nonlinear expression ability of the network. So the no-delay variables can directly use aging factor time θ at the last time step without complex transformation and the form of subsequence. The improved LSTM model also can save training time because, for no-delay

variables, there is no need for recurrent calculation inside the memory block. However, it increases the time consumption due to the additional hidden layer 2. Therefore, the neurons' number and the transfer function of hidden layer 2 should be appropriately set to control the time consumption. Figure 4 shows the improved LSTM model calculating process.

The backpropagation algorithm in Figure 2 also needs to be modified to accommodate this improved model. The loss function only considers the output at the last time step and

FIGURE 4: The improved LSTM model calculating process of dam displacement with two hidden layers. m , memory blocks and n , no-delay variables.

can be expressed as

$$L_N = \frac{1}{2} \sum_{l=1}^N (\mathbf{o}_k^l - \mathbf{y}_k^l)^2 \quad (1)$$

Step 1 and step 2 are the same as Figure 2. However, step 3 should be expressed as

$$\sigma(\mathbf{u}_{k,1}^l) = \sigma([\mathbf{W}\mathbf{h}_k^l \quad \mathbf{W}^n \mathbf{r}^l] + [\mathbf{b} \quad \mathbf{b}^n]) \quad (2)$$

where $\mathbf{r}^l$ represents the matrix of no-delay variables; $\mathbf{W}^n$ and $\mathbf{b}^n$ represent the weight and bias matrix of no-delay variables, respectively; $\mathbf{u}_{k,1}^l$ denotes the output of hidden layer 1 which is not activated.

Because of the additional hidden layer 2, the next two steps are expressed as

$$\mathbf{o}_k^l = \sigma(\mathbf{u}_{k,2}^l) = \sigma(\mathbf{W}^{out} \sigma(\mathbf{u}_{k,1}^l) + \mathbf{b}^{out}) \quad (3)$$

$$\delta_k^l = \sigma(\mathbf{o}_k^l - \mathbf{y}_k^l) \odot \sigma'(\mathbf{u}_{k,2}^l) \quad (4)$$

where $\mathbf{W}^{out}$ and $\mathbf{b}^{out}$ represent the weight and bias matrix of the output layer, respectively; $\mathbf{u}_{k,2}^l$ denotes the output of hidden layer 2 which is not activated.

The remaining steps are expressed as follows:

$$\begin{aligned} \delta_{o,k}^l &= (\mathbf{W}^{out} \mathbf{W})^T \delta_k^l \odot \sigma'(\mathbf{u}_{k,1}^l) \odot \tanh(\mathbf{s}_k^l) \\ &\odot \sigma'(\mathbf{u}_{o,k}^l) \end{aligned} \quad (5)$$

$$\begin{aligned} \delta_{f,k}^l &= (\mathbf{W}^{out} \mathbf{W})^T \delta_k^l \odot \sigma'(\mathbf{u}_{k,1}^l) \odot \tanh'(\mathbf{s}_k^l) \\ &\odot \sigma'(\mathbf{u}_{o,k}^l) \odot \mathbf{s}_{k-1}^l \odot \sigma(\mathbf{u}_{o,k}^l) \end{aligned} \quad (6)$$

$$\begin{aligned} \delta_{i,k}^l &= (\mathbf{W}^{out} \mathbf{W})^T \delta_k^l \odot \sigma'(\mathbf{u}_{k,1}^l) \odot \tanh'(\mathbf{s}_k^l) \\ &\odot \sigma'(\mathbf{u}_{i,k}^l) \odot \mathbf{g}_k^l \odot \sigma(\mathbf{u}_{o,k}^l) \end{aligned} \quad (7)$$

$$\begin{aligned} \delta_{g,k}^l &= (\mathbf{W}^{out} \mathbf{W})^T \delta_k^l \odot \sigma'(\mathbf{u}_{k,1}^l) \odot \tanh'(\mathbf{s}_k^l) \\ &\odot \tanh'(\mathbf{u}_{g,k}^l) \odot \mathbf{i}_k^l \odot \sigma(\mathbf{u}_{o,k}^l) \end{aligned} \quad (8)$$

$$\frac{\partial L_N}{\partial \mathbf{W}^{out}} = \sum_{l=1}^N \delta_k^l (\sigma(\mathbf{u}_{k,1}^l))^T \quad (9)$$

$$\frac{\partial L_N}{\partial \mathbf{b}^{out}} = \sum_{l=1}^N \delta_k^l$$

FIGURE 5: Dongjiang Dam in China: (a) upstream; (b) downstream.

$$\frac{\partial L_N}{\partial \mathbf{W}} = \sum_{l=1}^N (\mathbf{W}^{out})^T \delta_k^l \odot \sigma'(\mathbf{u}_{k,1}^l) (\mathbf{h}_k^l)^T,$$

$$\frac{\partial L_N}{\partial \mathbf{W}^n} = \sum_{l=1}^N (\mathbf{W}^{out})^T \delta_k^l \odot \sigma'(\mathbf{u}_{k,1}^l) (\mathbf{r}^l)^T$$

(10)

$$\frac{\partial L_N}{\partial \mathbf{b}} = \sum_{l=1}^N (\mathbf{W}^{out})^T \delta_k^l \odot \sigma'(\mathbf{u}_{k,1}^l),$$

$$\frac{\partial L_N}{\partial \mathbf{b}^n} = \sum_{l=1}^N (\mathbf{W}^{out})^T \delta_k^l \odot \sigma'(\mathbf{u}_{k,1}^l)$$

$$\frac{\partial L_N}{\partial \mathbf{W}^{xt}} = \sum_{l=1}^N \delta_{\tau,k}^l (\mathbf{x}_k^l)^T$$

$$\frac{\partial L_N}{\partial \mathbf{W}^{ht}} = \sum_{l=1}^N \delta_{\tau,k}^l (\mathbf{h}_{k-1}^l)^T, \quad \tau \in \{g, i, f, o\}$$

(11)

The last step is to update the weights and biases.

3. Case Study

In this section, the performance of the LSTM model and its improved model for the radial displacement prediction of the Dongjiang arch dam is demonstrated. The MLR, MLP, SVM, and BRT models are used as contrast models. All the algorithms are implemented in Python 3.6 environment. The modeling process using MLR, MLP, SVM, and BRT is carried out using an application program interface (API) scikit-learn (v0.19.0). LSTM algorithm is implemented using TensorFlow (v1.8.1) developed by Google. All these algorithms are used in their original version and are performed via a laptop computer with detailed technical parameters shown as one CPU of Intel core i5-4200U@1.60 Hz and 2 core processors, one RAM of 8GB, and 64-bit system type. Before running these models, data were normalized into the range [0, 1].

3.1. Dongjiang Arch Dam. Dongjiang arch dam was constructed between 1978 and 1980. It is a double-curvature arch dam consisting of 29 blocks. It is provided with 4 horizontal inspection galleries and the base gallery located near the foundation rock. It has a crest elevation of 294 m, height of 157 m, crest length of 438 m, crest thickness of 7m, and base thickness of 35 m. Figure 5 shows the dam.

3.2. Dataset and Model Variables. The horizontal displacements are obtained with the pendulum method as shown in Figure 6. The pendulums consist of the direct pendulum and inverted pendulum. They are used to measure the relative displacement from the reference point at the dam crest and rock foundation, respectively. By assuming that the displacement at the rock base is zero, the displacement of each point can be calculated.

L5 is the label of the pendulum at the crown cantilever situated in the 15th block. L5H291R represents the radial displacement of L5 at 291m elevation. The available time series data of the L5H291R are used for modeling. These data cover 14 years (from 11/6/2000 to 29/5/2014) with 341 displacement samples) and consist of daily reservoir water level and air temperature as shown in Figures 7 and 8. Finally, the dataset splits into training dataset and validation dataset by using proportion 0.8, which contains 272 training samples and 69 validation samples.

For input variables selection, MLR, MLP, SVM, and BRT adopt 10 variables according to the HST model involving the water level, temperature, and aging factor; $\mathbf{X} = [H, H^2, H^3, H^4, \sin(2\pi t/365), \cos(2\pi t/365), \sin^2(2\pi t/365), \sin(2\pi t/365) * \cos(2\pi t/365), \theta, e^{-\theta}]$, where H stands for reservoir water level; the trigonometric function is used to simulate the temperature field with quasi-stable state for the dam running for many years; t represents the number of days between the beginning of the year (January 1) and the date of observation; aging factor time θ stands for the number of days since the initial sample date. The air temperature (see Figure 8) is finally not used in these four models because the variables selection of moving averaged method [18, 19] is diverse and the results perform worse than the HST model for the dataset by our tests.

FIGURE 6: Horizontal displacements obtained by L5 pendulums.

FIGURE 7: Daily reservoir water level curve from 11/6/2000 to 29/5/2014 (H). FIGURE 8: Daily air temperature curve from 11/6/2000 to 29/5/2014 (T_0).

The LSTM model uses 3 variables of water level H , air temperature T , and aging factor time θ in the raw subsequence form as discussed in Section 2.3. The improved LSTM model uses the delay variables water level H and air temperature T in the raw subsequence form and the no-delay variable aging factor time θ at the last time step. In this study, the unimproved model is named LSTM and the improved model is named ILSTM.

follows:

$$MSE = \frac{1}{N} \sum_{i=1}^N (yp_i - y_i)^2 \quad (12)$$

$$MAE = \frac{1}{N} \sum_{i=1}^N |yp_i - y_i| \quad (13)$$

$$R = \frac{\sum_{i=1}^N (yp_i - \bar{y} \bar{p}_i) (y_i - \bar{y}_i)}{\sqrt{\sum_{i=1}^N (yp_i - \bar{y} \bar{p}_i)^2 \sum_{i=1}^N (y_i - \bar{y}_i)^2}} \quad (14)$$

3.3. Model Evaluation Index. The evaluation indexes of the models adopt the mean square error (MSE), the mean absolute error (MAE), and the correlation coefficient (R) as

TABLE 1: Calculated regression coefficients of the MLR model.

Variables of input	MLR Coefficients	Variables of input	MLR Coefficients
H	-870.272	$\sin^2(2\pi t/365)$	0.011
H^2	2638.398	$\sin(2\pi t/365) * \cos(2\pi t/365)$	-0.018
H^3	-2673.127	θ	0.027
H^4	905.647	$e^{-\theta}$	-0.013
$\sin(2\pi t/365)$	0.688	const	-0.279
$\cos(2\pi t/365)$	0.346		

where $\bar{y}p$ and $\bar{y}\bar{p}$ are simulated values and mean simulated values from the model; y and $\bar{y}$ are the measured values and the mean measured values.

3.4. Model Performance Analysis. The MLR model parameters are listed in Table 1. It can be seen from Table 1 that the coefficients of H , H^2 , H^3 , and H^4 are significantly greater than the other coefficients and there exists serious linear correlation. It can be deduced that when the linear correlation equilibrium relationship is broken, such as extreme water level, the predicted value may be considerably biased.

A MLP, SVM, and BRT comparison for dam behavior modeling was performed in detail by F. Salazar [19]. For the sake of clarity, we mainly focus on the parameters selection of these models in this work. For the case of MLP, the structure with one hidden layer with 4, 5, 6, 7, 8, and 9 neurons, the numbers of training epochs 100, 300, 500, and 700, the regularization parameters 0.01, 0.001, and 0.0001, and the hyperbolic tangent, sigmoid, rectified linear unit (Relu) [31] transfer function are chosen. In particular, the solver is limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) which converges faster and with better solutions on small datasets [32]. For SVM model, the “cost” parameter $C = 10, 100, 300, 500$, and 700, the ϵ -insensitive error function parameter $\epsilon = 0.1$, the radial basis kernel function parameter $\gamma = 1, 0.1, 0.01, 0.001$, and 0.0001 are considered. For BRT model, the learning rate $\nu = 0.001, 0.0001$, the estimators with 1000, 2000, 3000, 4000, and 5000 maximum trees, and the trees depths 1, 2, and 3 are applied for candidates. The best parameters are selected by the method of 5-fold cross-validation to prevent overfitting. Finally, the MLP with one hidden layer and 6 neurons, 500 training epochs, regularization parameters 0.01, and the Relu transfer function are selected. The SVM model with $C = 500$ and $\gamma = 0.01$ and the BRT model with learning rate $\nu = 0.001$, 4000 estimators, and the tree depth 2 are the best.

For the LSTM models (LSTM, ILSTM), the hyperparameters include the training epochs and convergence tolerance, the number of the hidden layers and nodes in each hidden layer, the kind of nodes in each hidden layer, transfer function, learning rate and its updater, and the input subsequence length. The parameters have many combination modes, and it is hard to design without any reference before in the dam displacement prediction field. In this work, these parameters are obtained by the trial-and-error method including the 5-fold cross-validation. Finally, the training epochs are set as 3000 and the convergence tolerance is 0.001. Hidden layer

1 with 3 memory blocks is designed for the LSTM and ILSTM models’ network structures. Hidden layer 2 with 4 neurons is applied for the ILSTM model. Hidden layer 2 is not used in the LSTM model because we found that there is no facilitation or even vulnerability to overfitting. The Relu transfer function is used in the hidden layer 2 and the linear transfer function is used in the output layer to replace the sigmoid function in Figures 2–4 by multiple tries. The initial learning rate is set as 0.01 and the Adam updater which designs an independent adaptive learning rate for different parameters by calculating the first-order moment estimation and the second-order moment estimation of the gradient is used. It is very robust and usually converges quickly and gives pretty good performance and the hyperparameters $\beta_1=0.9$, $\beta_2=0.999$, and $\epsilon=10e^{-8}$ [33].

The weight and bias matrices of LSTM and ILSTM models are initialized by a normal distribution with a mean of 0 and a variance of 1. Five runs are conducted for considering the impact of different initial weights and biases.

With the parameters above, the effects of the input subsequence length to the LSTM models’ performance are analyzed and the results are shown in Figure 9. As can be seen, the training MSE and validation MSE are relatively small when the length is greater than 60 (delay days) and have the smallest validation MSE at length 135. It can infer that short length is susceptible to information insufficiency and the input data within 45 days before the observed displacement date cannot fit the displacement well. Meanwhile, it has not much to improve accuracy when the length is long enough. Thus, the input subsequence length (delay days) is set as 135. The evaluation indexes of LSTM and ILSTM models are listed in Tables 2 and 3, respectively.

The performance of the six models on both the training and the validation datasets is shown in Table 4.

It can be seen from Table 4 that the training and validation accuracy of MLR model seems satisfactory without significant prediction deviation. MLP model’s training and validation accuracy are relatively high, which provide more support that the nonlinear relationship exists between the input with ten variables and output displacement. SVM model’s training accuracy is the worst, while the validation accuracy is better than the MLR and BRT models and the training accuracy. The training accuracy of BRT is the best, while the prediction accuracy presents a significant deviation. This phenomenon is shown in Figure 13 in the validation phase (2011/08/28~29/5/2014). Though the cross-validation can prevent overfitting such as the MLP and SVM models,

TABLE 2: Evaluation indexes in the LSTM model of L5H291R (135 delay days).

Run number	MAE		MSE		R		Training time, s
	Training	Validation	Training	Validation	Training	Validation	
1	1.106	1.120	1.996	2.073	0.987	0.989	339.63
2	1.034	1.593	1.612	4.902	0.990	0.977	337.36
3	1.119	1.105	1.930	2.116	0.988	0.988	336.45
4	1.296	1.124	2.529	1.860	0.987	0.989	338.44
5	0.859	1.548	1.325	3.189	0.992	0.984	328.54
Mean	1.083	1.298	1.878	2.828	0.989	0.986	336.08

MAE = mean absolute error; MSE = mean square error; R = the correlation coefficient.

TABLE 3: Evaluation indexes in the ILSTM model of L5H291R (135 delay days).

Run number	MAE		MSE		R		Training time, s
	Training	Validation	Training	Validation	Training	Validation	
1	1.086	1.078	1.859	1.942	0.988	0.990	341.55
2	1.204	1.206	2.258	2.358	0.987	0.987	341.64
3	1.046	1.194	1.665	2.329	0.990	0.989	338.41
4	0.899	0.994	1.383	1.606	0.992	0.991	338.71
5	1.204	1.245	2.292	2.284	0.987	0.988	340.83
Mean	1.088	1.143	1.891	2.104	0.989	0.989	340.23

MAE = mean absolute error; MSE = mean square error; R = the correlation coefficient.

TABLE 4: Evaluation indexes in the MLR, MLP, BRT, LSTM, and ILSTM models of L5H291R.

Model	MAE		MSE		R		Training time, s
	Training	Validation	Training	Validation	Training	Validation	
MLR	1.434	1.603	3.042	3.870	0.981	0.981	0.001
MLP	1.176	1.243	2.172	2.516	0.986	0.986	0.018
SVM	1.465	1.454	3.217	3.288	0.980	0.982	0.003
BRT	0.249	1.601	0.106	3.684	0.999	0.979	1.49
LSTM	1.083	1.298	1.878	2.828	0.989	0.986	336.08
ILSTM	1.088	1.143	1.891	2.104	0.989	0.989	340.23

MAE = mean absolute error; MSE = mean square error; R = the correlation coefficient; MLR = multiple linear regression; MLP = multilayer perceptron; SVM = support vector machine; BRT = boosted regression tree. The best results are highlighted in *italic*.

FIGURE 9: Effects of the input subsequence length of the LSTM models. Five runs have been conducted under each length condition. The region is enveloped by the maximum and minimum values. The curve in the region is connected by the mean value. MSE, mean square error.

FIGURE 10: Radial displacement and residual error measured and modeled by MLR.

FIGURE 11: Radial displacement and residual error measured and modeled by MLP.

it seems that the method does not work well in the BRT model. Figures 10–13 show that these models' results with ten variables have a slightly rising trend at the validation phase.

The accuracy of the LSTM model is better than the MLR and SVM models and almost the same as MLP which has relatively higher training accuracy and lower validation accuracy. Figure 14 shows that LSTM model has a slightly

downtrend trend in the validation phase. ILSTM has the best accuracy at the validation phase, and its training accuracy is only lower than the BRT model. It also has similar accuracy between the training and validation phases without any rising trend or downtrend (see Figure 15) as well as the overfitting problem. Therefore, synthesizing these models' performance, ILSTM model is the best choice.

FIGURE 12: Radial displacement and residual error measured and modeled by SVM.

FIGURE 13: Radial displacement and residual error measured and modeled by BRT.

Interestingly, the models' errors except BRT have great negative deviation at 2007/08/19 mainly because of a sudden increase in water level (see Figure 7). Therefore, these models may be not suitable for displacement prediction with a rapid jump of water level in the short term. Although the change can be simulated in the training phase by using the BRT

model, the big deviation in the validation phase compared to the training phase leads to potential instability. In general, the sudden increase or drop of water level requires special attention because this phenomenon may be connected with the flood, leakage channels, and so forth. Therefore, the prediction accuracy in this situation is not necessary.

FIGURE 14: Radial displacement and residual error measured and modeled by LSTM.

FIGURE 15: Radial displacement and residual error measured and modeled by ILSTM.

The training time as listed in Tables 2, 3, and 4 means that the model uses the parameters to train the 272 training samples at one time, excluding the time consumed by the 5-fold cross-validation. As we can see from Tables 2–4, LSTM models need more than 300 seconds to train once, which is one of the main problems of the deep learning model. The parallel processing capabilities of GPUs can accelerate the

LSTM training and inference processes to solve the problem [34].

4. Discussion

The LSTM models can make full use of the historical data to predict the dam displacement. Their variables in the

form of subsequence can reflect the delays more convenient and practical instead of the complex and diverse variables selection. The improved LSTM model splits the input data into two sets: variables delayed through LSTM memory blocks and variables instantaneously fed through the network without memory blocks, making the variables more physically reasonable. The improved LSTM model is trained and validated on Dongjiang arch dam displacement data, and the results achieve the best performance compared to the results obtained with the typical LSTM, as well as MLP, MLR, SVM, and BRT models.

LSTM models also have some deficiencies compared with the previous models. The sequence form of the input increases the difficulty to calculate the impact of each variable. Meanwhile, the hyperparameters need to be adjusted, and their parameter combination modes for optimization is hard when the calculation speed is not keeping up.

In the future, LSTM models will be further studied for those monitoring data depending on the historical data of the external environment and attempting to replace those models of less accuracy or demanding of complex variables selection.

Data Availability

https://figshare.com/articles/data_for_dam/7588457.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors would like to thank Deng Zhenhua, Li Changsen, and Liu Zixing of Dongjiang power plant for providing them with detailed and reliable data.

References

- [1] N. G. Beaujoint, "Les méthodes de surveillance des barrages au service de la production hydraulique d'Electricité de France, problèmes anciens et solutions nouvelles," in *Proceedings of the IXth International Congress on Large Dams*, pp. 529–550, 1967.
- [2] "Methods of analysis for the prediction and the verification of dam behavior," Swiss Committee on Dams, ICOLD, 2003.
- [3] A. de Sortis and P. Paoliani, "Statistical analysis and structural identification in concrete dam monitoring," *Engineering Structures*, vol. 29, no. 1, pp. 110–120, 2007.
- [4] D. Hu, Z. Zhou, Y. Li, and X. Wu, "Dam safety analysis based on stepwise regression model," *Advanced Materials Research*, vol. 204–210, pp. 2158–2161, 2011.
- [5] B. Stojanovic, M. Milivojevic, M. Ivanovic, N. Milivojevic, and D. Divac, "Adaptive system for dam behavior modeling based on linear regression and genetic algorithms," *Advances in Engineering Software*, vol. 65, pp. 182–190, 2013.
- [6] G. Prakash, A. Sadhu, S. Narasimhan, and J.-M. Brehe, "Initial service life data towards structural health monitoring of a concrete arch dam," *Structural Control and Health Monitoring*, vol. 25, no. 1, 2018.
- [7] J. Mata, A. Tavares de Castro, and J. Sá da Costa, "Constructing statistical models for arch dam deformation," *Structural Control and Health Monitoring*, vol. 21, no. 3, pp. 423–427, 2014.
- [8] S. Gamse and M. Oberguggenberger, "Assessment of long-term coordinate time series using hydrostatic-season-time model for rock-fill embankment dam," *Structural Control and Health Monitoring*, vol. 24, no. 1, Article ID e1859, 2017.
- [9] R. Vanatwerp, "Engineering and design: deformation monitoring and control surveying," *Engineer manual US Army corps of engineering EM:1110-1111*, 1994.
- [10] K.-T. T. Bui, D. T. Bui, J. Zou, C. Van Doan, and I. Revhau, "A novel hybrid artificial intelligent approach based on neural fuzzy inference model and particle swarm optimization for horizontal displacement modeling of hydropower dam," *Neural Computing and Applications*, vol. 29, pp. 1–12, 2016.
- [11] J. Zhang, X. Cao, and W. Tang, "Inversion and stress analysis of mechanical parameters of arch dam of dongjiang river," *DEStech Transactions on Engineering and Technology Research*, 2016.
- [12] H. Mirzabozorg, M. A. Hariri-Ardebili, M. Heshmati, and S. M. Seyed-Kolbadi, "Structural safety evaluation of Karun III Dam and calibration of its finite element model using instrumentation and site observation," *Case Studies in Structural Engineering*, vol. 1, no. 1, pp. 6–12, 2014.
- [13] J. Mata, "Interpretation of concrete dam behaviour with artificial neural network and multiple linear regression models," *Engineering Structures*, vol. 33, no. 3, pp. 903–910, 2011.
- [14] B. Stojanovic, M. Milivojevic, N. Milivojevic, and D. Antonijevic, "A self-tuning system for dam behavior modeling based on evolving artificial neural networks," *Advances in Engineering Software*, vol. 97, pp. 85–95, 2016.
- [15] H. Yu, Z. R. Wu, T. F. Bao, and L. Zhang, "Multivariate analysis in dam monitoring data with PCA," *Science China Technological Sciences*, vol. 53, no. 4, pp. 1088–1097, 2010.
- [16] H. Su, Z. Chen, and Z. Wen, "Performance improvement method of support vector machine-based model monitoring dam safety," *Structural Control and Health Monitoring*, vol. 23, no. 2, pp. 252–266, 2016.
- [17] V. Ranković, N. Grujović, D. Divac, and N. Milivojević, "Development of support vector regression identification model for prediction of dam structural behaviour," *Structural Safety*, vol. 48, pp. 33–39, 2014.
- [18] F. Kang, J. Liu, J. Li, and S. Li, "Concrete dam deformation prediction model for health monitoring based on extreme learning machine," *Structural Control and Health Monitoring*, vol. 24, no. 10, 2017.
- [19] F. Salazar, M. A. Toledo, E. Oñate, and R. Morán, "An empirical comparison of machine learning techniques for dam behaviour modelling," *Structural Safety*, vol. 56, pp. 9–17, 2015.
- [20] G. Lombardi, F. Amberg, and G. R. Darbre, "Algorithm for the prediction of functional delays in the behaviour of concrete dams," *International Journal on Hydropower and Dams*, vol. 15, no. 3, pp. 111–116, 2008.
- [21] C. Loh, C. Chen, and T. Hsu, "Application of advanced statistical methods for extracting long-term trends in static monitoring data from an arch dam," *Structural Health Monitoring: An International Journal*, vol. 10, no. 6, pp. 587–601, 2011.
- [22] F. Salazar, M. Á. Toledo, J. M. González, and E. Oñate, "Early detection of anomalies in dam performance: A methodology based on boosted regression trees," *Structural Control and Health Monitoring*, vol. 24, no. 11, 2017.

- [23] C. Shao, C. Gu, M. Yang, Y. Xu, and H. Su, "A novel model of dam displacement based on panel data," *Structural Control and Health Monitoring*, vol. 25, no. 1, 2018.
- [24] P. Pławiak and K. Rzecki, "Approximation of phenol concentration using computational intelligence methods based on signals from the metal-oxide sensor array," *IEEE Sensors Journal*, vol. 15, no. 3, pp. 1770–1783, 2015.
- [25] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," in *Spin Glass Theory and Beyond: An Introduction to the Replica Method and Its Applications*, pp. 411–415, 1987.
- [26] G. Kuhn, R. L. Watrous, and B. Ladendorf, "Connected recognition with a recurrent network," *Speech Communication*, vol. 9, no. 1, pp. 41–48, 1990.
- [27] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [28] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: Continual prediction with LSTM," in *Proceedings of the 1999 the 9th International Conference on 'Artificial Neural Networks (ICANN99)*, pp. 850–855, September 1999.
- [29] Z. C. Lipton, J. Berkowitz, and C. Elkan, "A critical review of recurrent neural networks for sequence learning," *Computer Science: Machine Learning*, 2015.
- [30] P. J. Werbos, "Backpropagation through time: what it does and how to do it," *Proceedings of the IEEE*, vol. 78, no. 10, pp. 1550–1560, 1990.
- [31] J. Appleyard, T. Kocisky, and P. Blunsom, "Optimizing performance of recurrent neural networks on gpus," *Computer Science: Machine Learning*, 2016.
- [32] V. Nair and G. E. Hinton, "Rectified linear units improve Restricted Boltzmann machines," in *Proceedings of the 27th International Conference on Machine Learning (ICML '10)*, pp. 807–814, Haifa, Israel, June 2010.
- [33] R. H. Byrd, P. Lu, J. Nocedal, and C. Y. Zhu, "A limited memory algorithm for bound constrained optimization," *SIAM Journal on Scientific Computing*, vol. 16, no. 5, pp. 1190–1208, 1995.
- [34] D. P. Kingma and J. B. Adam, "A method for stochastic optimization," *International conference on learning representations*, 2015.

Submit your manuscripts at
www.hindawi.com