water

MDPI

Article

Automatic Concrete Dam Deformation Prediction Model Based on TPE-STL-LSTM

Sihan Song [1,*], Qiujing Zhou [1], Tao Zhang [2] and Yintao Hu [1]

[1] China Institute of Water Resources and Hydropower Research, Beijing 100038, China; zhouqj@iwhr.com (Q.Z.); huyintao@edu.iwhr.com (Y.H.)
[2] Y.R. Wanjiazhai Water Multi-Purpose Dam Project Co., Ltd., Taiyuan 030002, China
* Correspondence: ssh95zz@163.com

Abstract: Concrete dam deformation prediction is important for assessing the safety of dams. A TPE-STL-LSTM deformation prediction model for concrete dams is established by introducing the TPE algorithm based on the decomposition–prediction model. Taking the Wanjiazhai gravity dam project as an example, a prediction model for the top deformation of 14 dam sections was established and the parameters were determined. The model was used for deformation prediction and compared with the measured deformation and similar methods to predict deformation for verification. The results show that the model has good prediction effect and matches well with the measured data; the accuracy is better than the Autoregressive Integrated Moving Average model and the Support Vector Machine model; and the model achieves the automatic determination of all parameters. The model can be used for dam engineering safety assessment, effectively improving the analysis accuracy and analysis efficiency.

Keywords: concrete dam; deformation properties; LSTM; prediction model; HPO

check for updates

Citation: Song, S.; Zhou, Q.; Zhang, T.; Hu, Y. Automatic Concrete Dam Deformation Prediction Model Based on TPE-STL-LSTM. *Water* **2023**, *15*, 2090. https://doi.org/10.3390/w15112090

Academic Editor: Paolo Mignosa

Received: 5 May 2023
Revised: 26 May 2023
Accepted: 29 May 2023
Published: 31 May 2023

Copyright: © 2023 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

Ensuring dam safety is a fundamental and critical aspect of engineering operation and management. Dam deformation is a key indicator that reflects the overall safety status of a dam and is used as the primary monitoring parameter both domestically and internationally [1]. Dam deformation is influenced by external factors such as water level, sedimentation, and environmental temperature, as well as internal factors such as material properties. It involves complex linear or nonlinear relationships. Traditional statistical models use factors such as water level, temperature, and creep for fitting and are widely used in engineering analysis. However, they have limitations in terms of multidimensional input, model adaptive learning, and fitting complex nonlinear relationships. Additionally, they generate significant selection bias because of the need to assume the distribution of data in advance.

To overcome these problems, in recent years, a large number of algorithm models based on mathematical methods such as neural networks have been proposed. Multi-layer neural networks have excellent nonlinear fitting capabilities, but multi-layer perceptrons and traditional recurrent neural networks cannot learn and remember long-term dependency information due to the vanishing gradient problem. Dam engineering deformation measurement data have obvious regularity, and it is necessary to consider the long-term dependency relationship between the data in the time dimension. The LSTM (Long Short-Term Memory) [2] proposed by Schnidhuber and Hochreiter is a variant of recurrent neural networks and has been widely used in time series analysis and natural language processing [3–5] due to its ability to extract and remember long-term dependency information. It has good application prospects in dam engineering deformation analysis. Ou Bin and others applied the LSTM neural network to the prediction of concrete dam deformation, effectively mining and learning the complex nonlinear relationships between dam

deformation and various environmental factors. By comparing with traditional methods such as stepwise regression and multivariate regression, it is shown that the deformation prediction model based on LSTM has superior performance [5–8].

Some parameters of neural networks need to be set manually before training and cannot be obtained during data learning. The reasonable determination of hyperparameters is the primary problem in optimizing neural networks, and automating the hyperparameter optimization process is key to deploying time series models in production environments. Current hyperparameter optimization algorithms mainly fall into two categories: Bayesian optimization and evolutionary algorithms [9]. Evolutionary algorithms, such as the Sparrow Search Algorithm (SSA) [10], Particle Swarm Optimization (PSO) [11], and Artificial Bee Colony Algorithm [12], have been widely applied by numerous researchers to optimize the hyperparameters of LSTM neural networks, achieving good optimization results. These algorithms offer broad applicability and higher robustness compared to traditional grid-search and random-search algorithms. The core of Bayesian optimization is constructing a prior probability model based on known sample points to estimate the posterior distribution. This method can make full use of existing information to determine the step size and direction of the next search [9]. Bayesian optimization has shown great potential in parameter estimation in various fields, such as combinatorial optimization, neural architecture searches, automatic machine learning, safety monitoring models, and environmental monitoring [13–16].

Time series data decomposition is an efficient algorithm widely used in the field of time series data prediction [17–19]. By decomposing measured data with temporal features and then predicting the decomposed residual, periodic, and trend components separately, the prediction accuracy can be further improved [20,21]. Li Bin and others combined the Seasonal Difference Autoregressive Integrated Moving Average (SARIMA) model and the Multivariate Linear Regression (MLR) model in their research on earth-rock dam displacement prediction. They predicted the periodic and trend components obtained after Hodrick–Prescott (HP) filtering [22]. Dong Yong and others first used Empirical Mode Decomposition (EMD) to decompose the original deformation monitoring data of a roller-compacted concrete dam and then further decomposed the high-frequency components using Ensemble Empirical Mode Decomposition (EEMD) to extract effective deformation information [23]. By separately modeling these components using LSTM, they effectively improved the prediction accuracy of a single model. Lin Chuan and others used the Complete Ensemble Empirical Mode Decomposition with Adaptive Noise (CEEMDAN) method to remove redundant data from the concrete dam deformation sequence, reducing the screening time and improving prediction accuracy [24]. The research work of these scholars shows that decomposing the original time series data using statistical methods can reduce the impact of random redundant data on a single model and, on the other hand, utilize the characteristics of different models to model different components [25], thereby further improving the prediction accuracy of the model.

In summary, neural networks have certain advantages over traditional models in adaptive learning and complex nonlinear relationship fitting, but the choice of hyperparameters significantly affects prediction accuracy. In this study, a forward analysis model based on prototype observation data is constructed. The LSTM neural network is used to learn the potential rules and patterns in the actual operation process of the dam, and periodic analysis, STL decomposition, and TPE hyperparameter optimization are introduced into the model to achieve automated modeling of numerous measurement points. The effectiveness of the model is verified through its practical application in the Wanjiazhai Dam project, and the prediction accuracy is improved compared to the single LSTM model and traditional Support Vector Machine (SVM) and Autoregressive Integrated Moving Average (ARIMA) models.

2. Principle of the TPE-STL-LSTM Model

2.1. LSTM

Recurrent Neural Networks (RNN) address the lack of sequential memory problem in Multi-Layer Perceptrons (MLP) by sharing model parameters through the structure of recurrent networks. However, as the sequence length increases, the gradients of the loss function can become very small when propagated back through many time steps (or layers). This leads to the problem of vanishing gradients [26,27].

As shown in Figure 1, Long Short-Term Memory (LSTM) [2] neural networks maintain and transmit a cell state across various layers, permeating the entire recurrent network architecture and employing specifically designed "gate" structures that determine the updating and modification of cell states through elementary linear operations, which optimizes the propagation of gradients in the backpropagation (BP) process. Consequently, they selectively rectify parameters returned by the error function during gradient descent to preserve long-term dependencies and discard irrelevant information, thereby mitigating, to a certain extent, the issues of gradient explosion and vanishing gradients. For each input time sequence, each layer of the LSTM neural network encompasses the following functions [28]. The symbols in Figure 1 are the same as in the formula:

$$\begin{aligned} i_t &= \sigma(W_{ii}x_t + b_{ii} + W_{hi}h_{t-1} + b_{hi}) \\ f_t &= \sigma\left(W_{if}x_t + b_{if} + W_{hf}h_{t-1} + b_{hf}\right) \\ g_t &= \tanh\left(W_{ig}x_t + b_{ig} + W_{hg}h_{t-1} + b_{hg}\right) \\ o_t &= \sigma(W_{io}x_t + b_{io} + W_{ho}h_{t-1} + b_{ho}) \\ c_t &= f_t \odot c_{t-1} + i_t \odot g_t \\ h_t &= o_t \odot \tanh(c_t) \end{aligned} \tag{1}$$

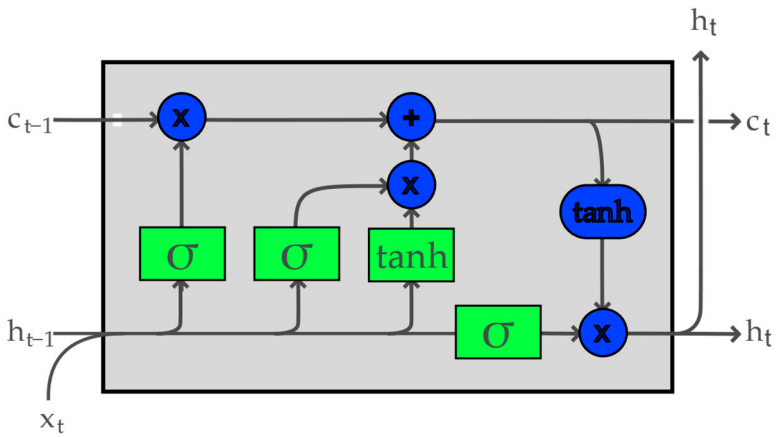

Figure 1. LSTM Cell.

In this formulation, h_t denotes the hidden state at time t, c_t signifies the cell state at time t, x_t represents the input time series data at time t, and h_{t-1} corresponds to the hidden state from time t or the initial state. The input gate, forget gate, cell state, and output gate are respectively represented by i_t, f_t, g_t, and o_t. The input gate updates the cell state, the forget gate determines whether to discard or retain information from the cell state, and the output gate controls the input received by subsequent neurons. This gate structure ensures the preservation of latent information and patterns in long time measurement sequences.

2.2. TPE Hyperparameter Optimization

In Bayesian optimization, the Bayesian theorem is used to update the prior probability distribution of unknown parameters based on observed data. This method is particularly useful for optimizing objective functions that are computationally expensive

or time-consuming to evaluate [29]. The method consists of two important components: a probabilistic surrogate model and an acquisition function. The surrogate model approximates the black-box objective function by tracking past evaluation results, and the acquisition function is used to predict the potential locations of optimal points given the current known data.

The Tree-structured Parzen Estimator (TPE) [30] is a hyperparameter optimization algorithm proposed by Bergstra based on the Bayesian rule that employs Gaussian mixture models to construct probability surrogate models. The surrogate model comprises two parts, with the formula as follows:

$$p(x \mid y) = \begin{cases} \ell(x) & y < y^* \\ g(x) & y \geq y^* \end{cases} \tag{2}$$

In this expression, y^* denotes the best observed objective function value to date. Separate probability density functions are maintained for observations smaller and larger than this value. The next sampling point is evaluated based on the expected improvement (EI), with the aim of obtaining a superior parameter combination compared to the existing observations.

In this study, the Optuna [31] framework proposed by Takuya Akiba, Shotaro Sano, and others is employed for hyperparameter optimization. This framework boasts several characteristics that render it advantageous in practical applications:

1. Lightweight: Optuna's design is concise and easy to install and use, allowing for effortless deployment and application in various computing environments.
2. Modular design: Optuna adopts a modular approach, enabling users to freely combine different optimization algorithms, search space definitions, and evaluation methods, making it adaptable to a wide range of scenarios.
3. High extensibility: Optuna's design permits the addition of custom optimization algorithms and evaluation methods to meet the requirements of specific problems, ensuring continuity and customization for subsequent research.
4. Database-based parallelism: Capitalizing on TPE's characteristics, Optuna implements a parallel mode that maintains surrogate models through a central database. In multi-node computing environments, this accelerates the hyperparameter search process, enabling the discovery of better parameter combinations within a limited timeframe.

2.3. STL Decomposition

Seasonal and Trend decomposition using Loess (STL) [32], a robust and versatile time series decomposition method based on local weighted regression, is primarily designed for non-stationary time series. Its implementation consists of two parts: an inner loop and an outer loop. The inner loop extracts the trend and seasonal components of the time series, while the outer loop adjusts the weight parameters of the robust local weighted regression during the inner loop process. Through this double recursion, non-stationary time series are ultimately decomposed into three additive components: trend, seasonal, and residual.

In STL decomposition applications, a periodic parameter must be provided. To enhance decomposition accuracy and automate the modeling process, the autocorrelation function and Fourier transform are employed to determine the periodic parameter of the original monitoring data.

The Autocorrelation Function (ACF) characterizes the similarity of a time series at different time intervals, with its value ranging from -1 to 1. A value closer to 1 indicates a positive correlation between the time series data in the two intervals. The autocorrelation function value reaches its maximum at integer multiples of the actual time series period, making it a common tool for identifying repetitive patterns in time-domain signals. The formula for the autocorrelation function is as follows. In the equation, k is the periodicity

parameter, n is the total length of the sequence, t represents the cumulative variable, Z_t is the t-th data point, and $\overleftarrow{Z}$ represents the mean of all the data:

$$ACF(k) = \Sigma_{t=k+1}^{n} \frac{\left(Z_t - \overleftarrow{Z}\right)\left(Z_{t-k} - \overleftarrow{Z}\right)}{\Sigma_{t=1}^{n}\left(Z_t - \overleftarrow{Z}\right)^2} \quad (3)$$

In this study, the Fourier transform is employed to convert time series data signals from the time domain to the frequency domain, with the maximum amplitude sine wave selected as the optimal period (T) for the time series. The autocorrelation coefficients of several periods with relatively high weights are calculated to ensure the reliability of the periodic parameter selection.

2.4. Modeling Process

Figure 2 illustrates the model construction process coupling TPE, STL, and LSTM, as well as their execution logic within the automated monitoring system. The red-framed section in the figure represents the modeling service workflow, while the blue section below illustrates the dam displacement prediction service workflow in a production environment. If the prediction accuracy does not meet the specified requirements, the model will be rebuilt and stored in the model database.

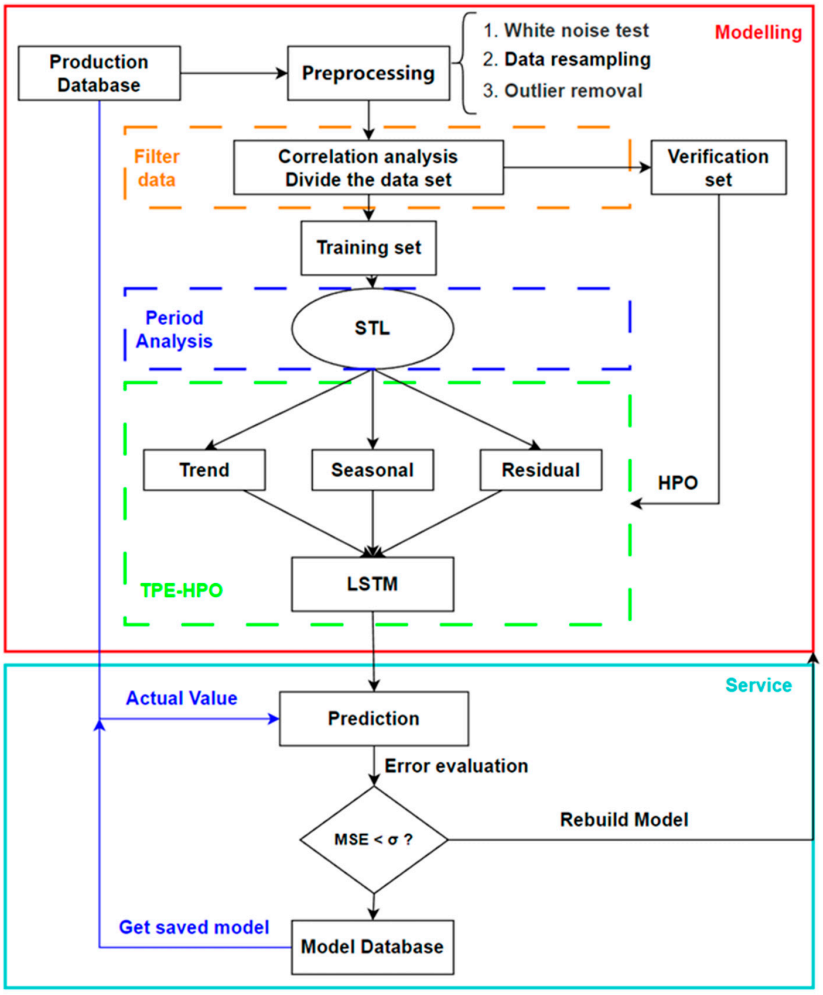

Figure 2. Modeling Process.

As shown in Figure 2, the modeling process comprises four core components: preprocess, filter data, period analysis, and TPE-HPO (Hyperparameter optimization).

1. Preprocess: Preprocess the raw monitoring data obtained from the monitoring database, including white noise detection, data resampling, and outlier removal.
2. Filter data: Select input data for the model through correlation analysis, partitioning them into training and validation sets. The training set is used for model training, while the validation set supports TPE algorithm parameter optimization.
3. Period analysis: Employing discrete Fourier transform, convert the input data to the frequency domain, and select the period of the sine wave with the highest amplitude as the periodic parameter for the time series data. Then, decompose the data using STL, yielding trend (T_t), seasonal (S_t), and residual (R_t) components.
4. TPE-HPO: Establish corresponding LSTM neural network models (N_t, N_s, and N_r) for the trend, seasonal, and residual components, and use the TPE algorithm to optimize the hyperparameter values of the models, selecting the best model. The hyperparameter optimization is represented by the flowchart in Figure 3, with train loop representing each epoch, and HPO loop denoting each trial. The train loop contains pruning operations, comparing model fitting accuracy within the same training iteration. If the accuracy is insufficient, the training terminates early, discarding unfruitful training attempts and improving HPO efficiency. TPE optimization determines the hyperparameter values for the next train loop using the TPE algorithm.

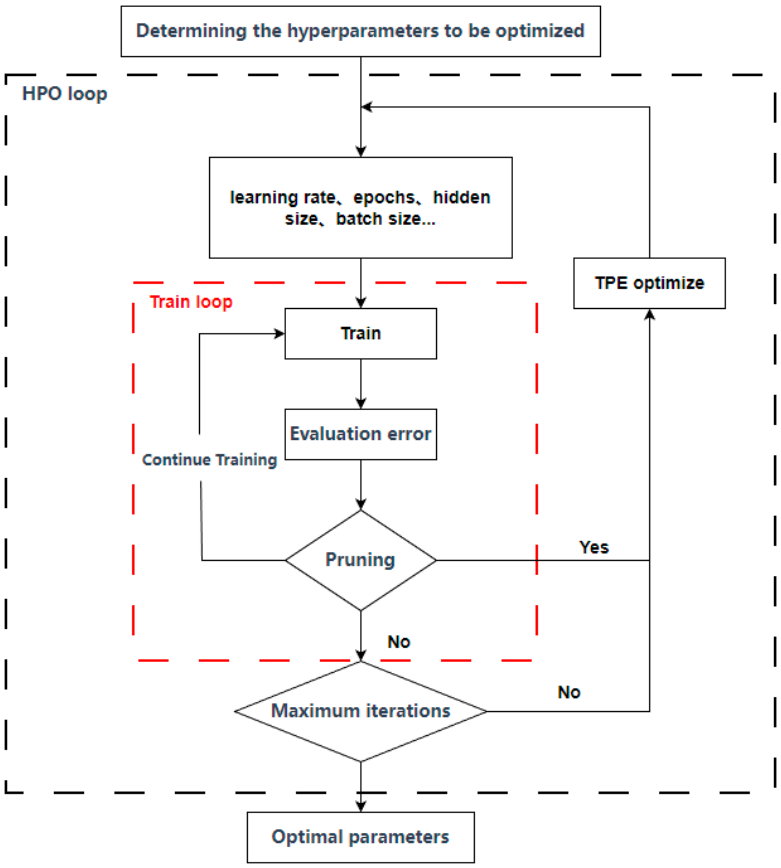

Figure 3. TPE hyperparameter optimization process.

5. Service: Obtain the predicted displacement value by summing the predicted results of the trend, seasonal, and residual components. If the model optimized by the TPE algorithm is validated using the latest monitoring data, it will be stored in the model database.

3. Engineering Case Application

The Wanjiazhai project is one of the key initiatives within the pilot implementation plan for digital-twin river basins constructed by China's Ministry of Water Resources, with dam deformation prediction and analysis being a crucial aspect. In this study, the Wanjiazhai concrete gravity dam serves as the case project, and an automated TPE-STL-LSTM dam deformation prediction model is established, predicting dam deformation using data from 14 dam sections.

3.1. Project Overview

This Grade I large-scale project primarily comprises a river-blocking dam, a powerhouse behind the dam, and other structures. The river-blocking dam is a semi-integral concrete gravity dam with a crest length of 443 m and a maximum dam height of 105 m, and it is divided into 22 sections, as shown in Figure 4. The dam crest displacement is jointly observed by sightlines and vacuum laser alignment systems and mutually verified, while dam body deflections are monitored by four sets of plumb and reverse plumb lines distributed across the bank slope and riverbed sections. Horizontal and vertical displacement control networks are established near the dam site to monitor rock mass displacements in the vicinity and verify displacement changes in the dam's horizontal and vertical working points.

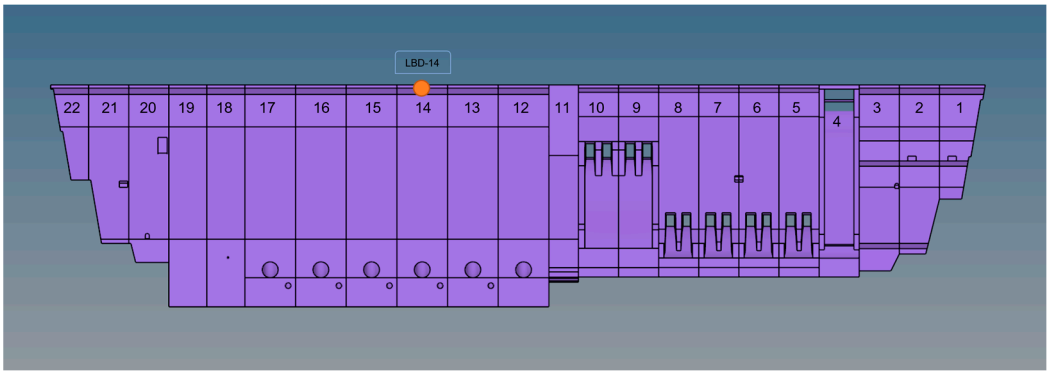

Figure 4. Engineering layout plan.

Section 14 of the dam serves as the powerhouse section and is a typical observation section. Various monitoring instruments are comprehensively and centrally arranged on the observation cross-section of this section. This study is based on the actual observed measurements of section 14 for modeling, prediction, and analysis. Displacement data are obtained from the LBD-14 laser alignment system measuring point at the top of the dam. A series of daily measured data from 1 January 2018 to 31 December 2021, spanning four years, are used as the original modeling samples. After outlier filtering and data resampling, 1462 sets of measured data are obtained, and they are divided into training and validation sets at a ratio of 4:1. The training set is used to determine the weights between neurons in the neural network, while the validation set is used to test the model's prediction performance and generalization capabilities.

As seen in the measured data curve in Figures 5 and 6, the water level during the dam's operation phase is significantly affected by the dispatching scheme, and the crest displacement along the river and pressure of dam foundation are strongly correlated with the reservoir water level. The reservoir water level rises and is lowered to the lowest point of the year in September due to sediment discharge scheduling, causing the crest deformation towards upstream to reach its maximum value for the year. Figure 6a,b represent the monitored strain values on the surface and internally, respectively, and both exhibit a high correlation with temperature. The measured data conform to the general deformation pattern of concrete gravity dams and can be used to test model performance.

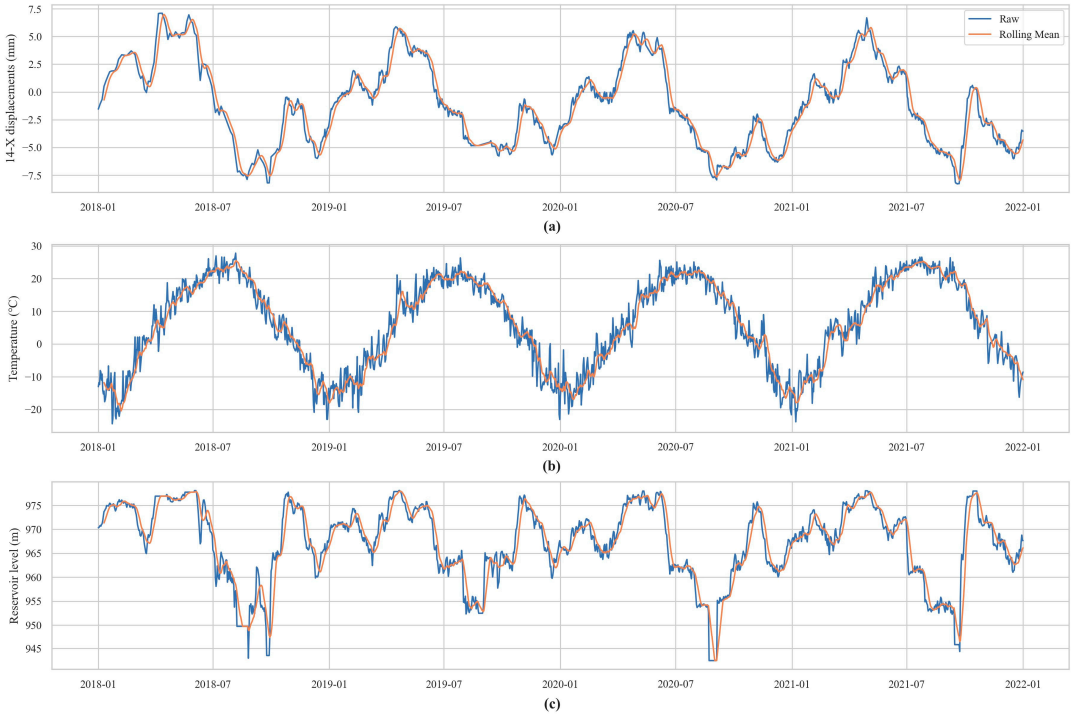

Figure 5. Measured data process line chart: Measured values are shown in blue, and moving averages are shown in orange. (**a**) Horizontal displacement of dam section #14. (**b**) The ambient temperature of the project site. (**c**) Water level of reservoir.

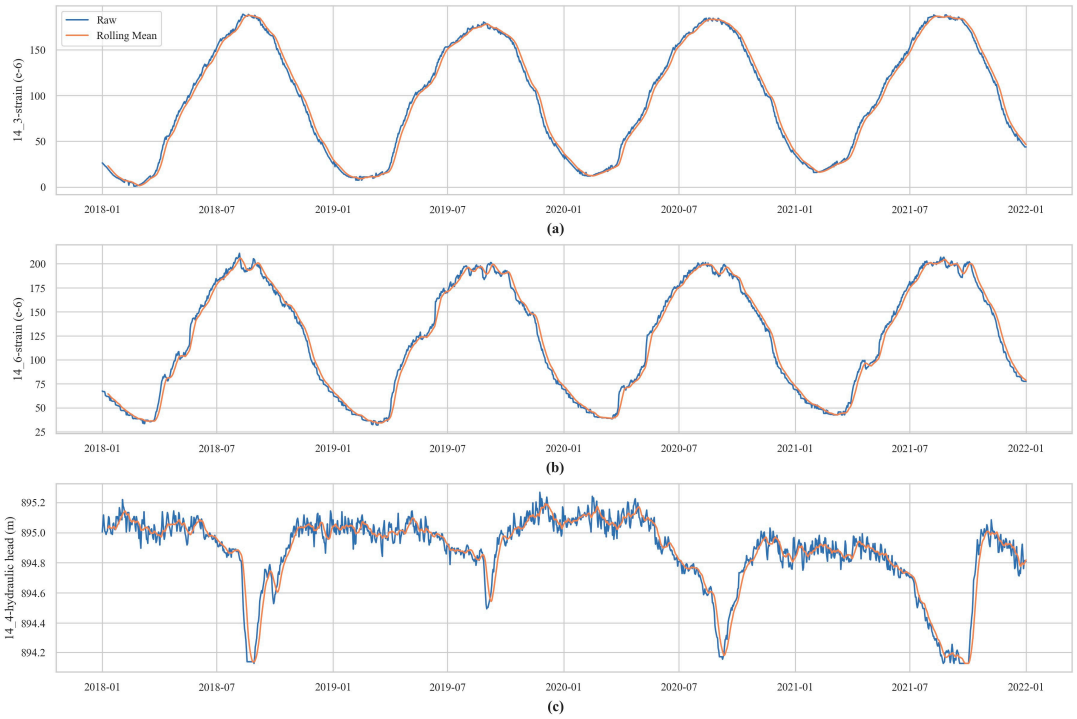

Figure 6. Measured data process line chart: Measured values are shown in blue, and moving averages are shown in orange. (**a**) Strain monitoring values at measuring point 14_3. (**b**) Strain monitoring values at measuring point 14_6. (**c**) The pressure-monitoring value of the dam foundation.

3.2. Correlation Analysis

Dam deformation is influenced by various environmental factors and exhibits a complex nonlinear relationship with them. The abundance of measured data from automated monitoring facilitates models that fully learn various features related to dam deformation, while also introducing a large volume of irrelevant data. Utilizing all measured data for model training would result in excessive computation and prolonged training times. By conducting correlation analysis, some irrelevant information can be eliminated and the validity of monitoring data can be verified. Employing measured data with a high correlation to dam displacement as the original training sample avoids ineffective inputs, extracts main features, and considers computational efficiency.

In the correlation heatmap, prefixes indicate the dam section where the instrument is located, X represents the along-stream (horizontal) direction, and Y denotes the vertical direction. As shown in Figure 7, reservoir water levels exhibit a distinct correlation with the along-stream displacement of the dam crest in section 14. Some internal monitoring instruments are significantly affected by temperature, displaying a strong positive correlation. This is mutually corroborated by the qualitative analysis of deformation and water level curves in the previous section, demonstrating the effectiveness of correlation analysis. In addition to environmental factors such as water level and temperature, this study also utilizes highly correlated strains from the same dam section as the original data for establishing a multi-dimensional LSTM model.

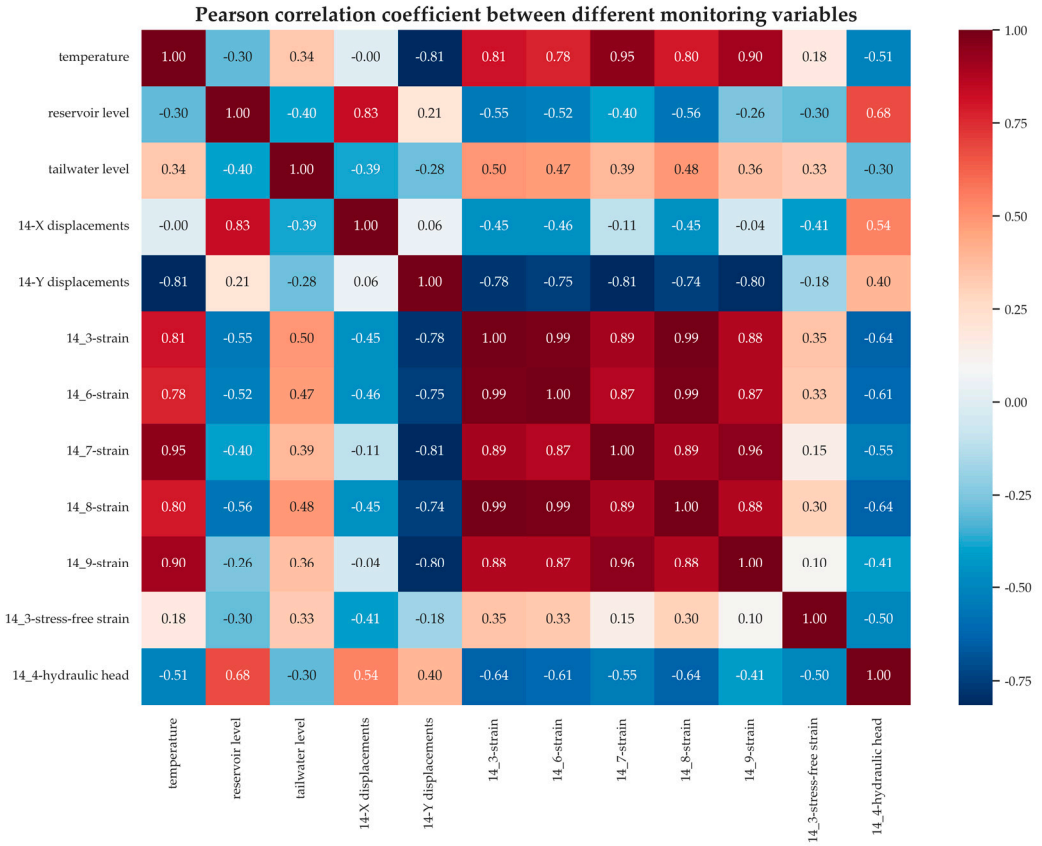

Figure 7. Heat map for monitoring volume correlation.

3.3. Periodic Analysis and STL Decomposition

In STL decomposition, if there is an error in determining the cycle parameter, the decomposition result will deviate abnormally. The weight of the seasonal component will decrease, and the trend or residual components will contain cyclical factors, rendering the decomposition work meaningless and reducing prediction accuracy. By using the

discrete Fourier transform, the along-river displacement time series data are converted to the frequency domain, resulting in the decomposition curve shown in Figure 8. The horizontal axis represents the cycle, and the vertical axis represents the amplitude of the corresponding cycle. A larger amplitude indicates that the cycle has the highest proportion in the original measured data, and the cycle with the largest amplitude is selected as the optimal cycle.

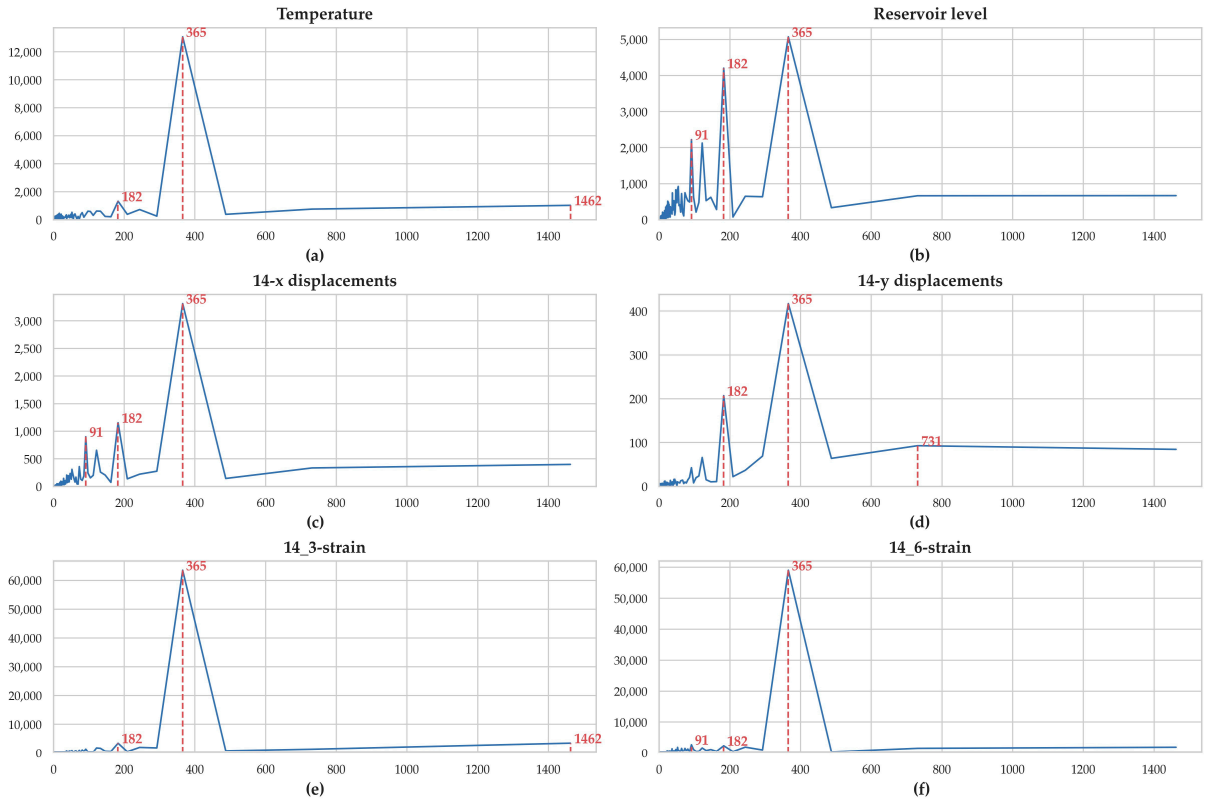

Figure 8. Amplitude spectra obtained by Fourier decomposition.

According to the spectral analysis, the autocorrelation coefficients for each optimal cycle are calculated using the autocorrelation function. Some calculation results of the measured sequences are shown in Table 1, where vertical displacement is taken from the 14-Y absolute displacement, and horizontal displacement is taken from the 14-X absolute displacement. The optimal cycles are concentrated around 365 and 182 days. The ACF value of temperature with a 182-day cycle is −0.7763, while the ACF value with a 365-day cycle is 0.6813. This result deviates from the optimal cycle obtained by Fourier decomposition, but considering practical experience, the decomposition cycle is still set to 365. Other data exhibit high consistency between the cycles characterized by discrete Fourier decomposition and the autocorrelation function, indicating that using the discrete Fourier decomposition method for cycle analysis is reliable.

Table 1. Periodic analysis results for each typical item.

Parameter	Amplitude	Period/Days	ACF
Temperature	1318.47	182	−0.7763
	13,071.87	365	0.6813
	1024.38	1462	0.0011

Table 1. *Cont.*

Parameter	Amplitude	Period/Days	ACF
Reservoir Level	2221.47	91	−0.2075
	4203.37	182	−0.1055
	5072.38	365	0.5725
Tailwater Level	231.49	292	0.0836
	464.30	365	0.2149
	325.88	1462	−0.0001
14-X Displacement	899.81	91	0.0159
	1153.71	182	−0.5282
	3312.85	365	0.6499
14-Y Displacement	207.05	182	−0.4947
	417.00	365	0.639
	92.91	731	0.4115

Based on the cycle parameters obtained from cycle analysis, the measured values are decomposed using the STL method. Taking horizontal displacement data as an example, cycle parameters of 91, 182, and 365 days are chosen, and the decomposition results are shown in Figure 9. The cycle parameters significantly affect the STL decomposition results. As the cycle parameters approach the optimal cycle obtained from cycle analysis, the periodic fluctuations in the trend component are gradually attributed to the seasonal component, and the regularity of the residual component decreases accordingly. This demonstrates that the STL algorithm performs better at the optimal cycle than at other parameter values. The decomposition results show that the horizontal displacement of the concrete dam is mainly influenced by periodic environmental factors such as water level and temperature. The trend component of the displacement, which is affected by time-dependent factors and concrete creep, has stabilized, which is consistent with the performance characteristics of the dam's stable operation for more than twenty years.

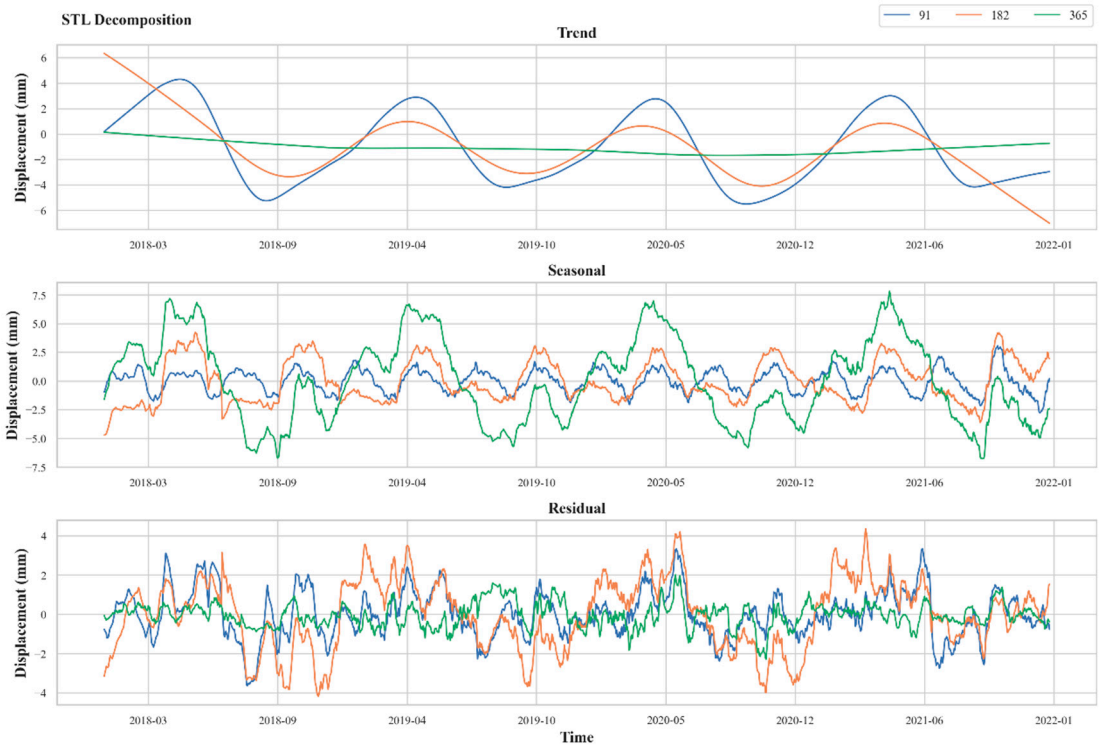

Figure 9. Decomposition results of different cycles.

In summary, we validate the periodic parameter using the Autocorrelation Function (ACF) coefficients and compare the decomposition results for different periodic parameters. The periodic analysis is performed using the Fourier method, and the period with the maximum amplitude is selected as the optimal period. We then use the optimal period as a parameter for the STL decomposition, resulting in a trend component that exhibits the least periodic characteristics, aligning with the modeling approach and requirements.

3.4. Hyperparameter Optimization

Following data preprocessing, periodicity analysis, and time series decomposition steps, a series of data including trend, periodic, and residual components is obtained. To eliminate the impact of differing data scales on training performance and enhance model convergence speed, these data need to be standardized. The multi-dimensional LSTM neural network is defined and implemented using Pytorch, with an input layer composed of a multi-dimensional tensor. The input data include ten dimensions, containing information on environmental factors, water head, and strain within the same dam section. The batch size is set to 30, the loss function is the mean squared error function, and the widely used Adaptive Moment Estimation (Adam) method is employed for optimization.

To compare and validate the advantages of the decomposition–prediction model over a single model, this study divides the model hyperparameter optimization into four parts: one part being a multi-dimensional LSTM model directly predicting the original sample data; and the other three being multi-dimensional LSTM models predicting trend, periodic, and residual components obtained from STL decomposition separately. All four parts use TPE algorithm for automated hyperparameter optimization. The hyperparameters to be optimized include training epochs, LSTM layers, hidden layers, and learning rate. Table 2 presents the optimal parameter values obtained from the optimization of the four parts, where Hidden_Size represents the feature scale of the LSTM hidden layer, Learning_Rate represents the learning rate, Num_Epochs represents the training epochs, and Num_Layers represents the stacked layers of LSTM. Hidden_Size determines the number or scale of neurons in the LSTM hidden layer. Through the hidden layer, LSTM can remember and utilize previous information to influence current outputs. Learning_Rate controls the step size or rate at which the model updates weights and biases during each iteration. Num_Epochs represents the number of times the model traverses the entire training dataset during the training process.

Table 2. Optimal parameter.

	Hidden_Size	Learning_Rate	Num_Epochs	Num_Layers
LSTM	71	0.00014	91	1
STL_Trend	10	0.00153	432	1
STL_Seasonal	189	0.00005	222	1
STL_Resid	59	0.00075	468	1

Hyperparameter importance analysis is an essential task that helps with understanding the stability and accuracy of the model, and it can guide similar time series modeling tasks to better optimize the hyperparameter search space and improve search efficiency. In this study, we explore the impact of common hyperparameters such as learning rate, network depth, hidden layer size, and training epochs on model establishment by tracking training history.

Optuna offers optional fANOVA and Mean Decrease Impurity methods for assessing hyperparameter importance [31]. We choose the latter, which uses a random forest regression model to predict the objective values of complete trials based on their parameter configurations and computes feature importance using MDI. This approach exhibits robustness for high-dimensional, nonlinear datasets.

The relationships among these hyperparameters are complex, and Optuna's hyperparameter optimization framework can conveniently implement database-based parallelism

based on the TPE algorithm, greatly enhancing the efficiency of exploring the hyperparameter space. As a result, we can investigate the patterns among hyperparameters in many experimental samples. Figure 10 displays hyperparameter importance, showing the importance of each hyperparameter for the residual, periodic, and trend components after STL decomposition and for undecomposed original data when modeling directly.

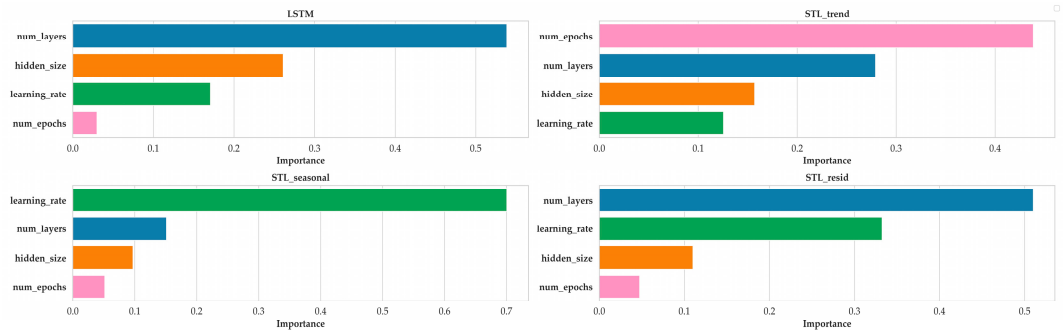

Figure 10. Hyperparameter importance.

Both original data and residual components exhibit high sensitivity to num_layers, which may be related to the high randomness and complex nonlinear relationships in the data. The expressive capability of neural networks is closely associated with their width and depth. Trend components capture the long-term tendencies of time series, and these tendencies typically require more iterations during training to be adequately learned and fitted. For periodic components, because the periodic data themselves possess apparent periodicity and regularity, a smaller learning rate may cause slower model convergence and inferior fitting accuracy with the same training epochs. In summary, the understanding of hyperparameter importance is closely related to the inherent characteristics of time series. By adjusting the step size and exploring the parameter search space, we can incorporate prior knowledge into the hyperparameter optimization process, thereby improving its efficiency and accuracy.

4. Discussion

The hyperparameters obtained from the hyperparameter optimization are used to train the model and predict the horizontal displacement of the dam crest at section 14. To demonstrate the superiority and applicability of this model for dam deformation prediction more intuitively, the prediction results are compared with those of SVM and ARIMA models. The SVM model demonstrates excellent generalization and robustness in situations with small samples, high dimensionality, and nonlinearity [33], making it widely applied in dam deformation prediction. The ARIMA model decomposes time series data into three parts, namely autoregressive, difference, and moving average, thereby describing the autocorrelation and non-stationarity of the data. It can effectively handle both seasonal and non-seasonal trends in time series data. In this model, SVM is supported by the famous Scikit-learn [34] open source project, while ARIMA is supported by the open source project pmdarima, similar to the well-known auto.arima [35] in R.

Table 3 presents the monthly average values of the prediction results, actual measurements, and residuals of the four models. Figure 11 shows the residual area plots of each model. It can be seen from the graphs that the residuals of all four models exhibit positive and negative fluctuations, indicating that the predictions fluctuate above and below the actual measurements. However, the TPE-STL-LSTM model's residual distribution is more even compared to the other three models. As time goes on, the positive and negative fluctuations in SVM and ARIMA residuals become larger. This verifies that the LSTM model can mine and learn the relationships between pieces of long-sequence information, thereby improving prediction accuracy.

Table 3. Comparison of predicted values and predicted residuals among models (all values in mm).

	Real	Predicted Value				Monthly Average Residual			
		TPE-STL-LSTM	TPE_LSTM	SVM	ARIMA	TPE-STL-LSTM	TPE_LSTM	SVM	ARIMA
2021/4	5.36	5.37	4.57	5.01	5.06	0.00	0.79	0.36	0.31
2021/5	3.64	3.53	3.45	3.59	3.79	0.11	0.19	0.04	−0.15
2021/6	1.61	1.54	1.74	1.62	1.64	0.07	−0.13	−0.01	−0.03
2021/7	−2.12	−2.12	−1.89	−1.88	−1.84	0.00	−0.23	−0.24	−0.28
2021/8	−5.07	−5.02	−4.95	−5.10	−4.88	−0.05	−0.12	0.03	−0.19
2021/9	−6.43	−6.26	−6.43	−6.40	−6.37	−0.18	0.00	−0.04	−0.07
2021/10	−1.07	−1.26	−1.01	−1.74	−1.19	0.19	−0.06	0.67	0.12
2021/11	−3.60	−3.59	−3.69	−3.50	−3.41	−0.01	0.08	−0.11	−0.19
2021/12	−5.05	−5.00	−5.27	−5.04	−4.97	−0.05	0.22	−0.01	−0.09

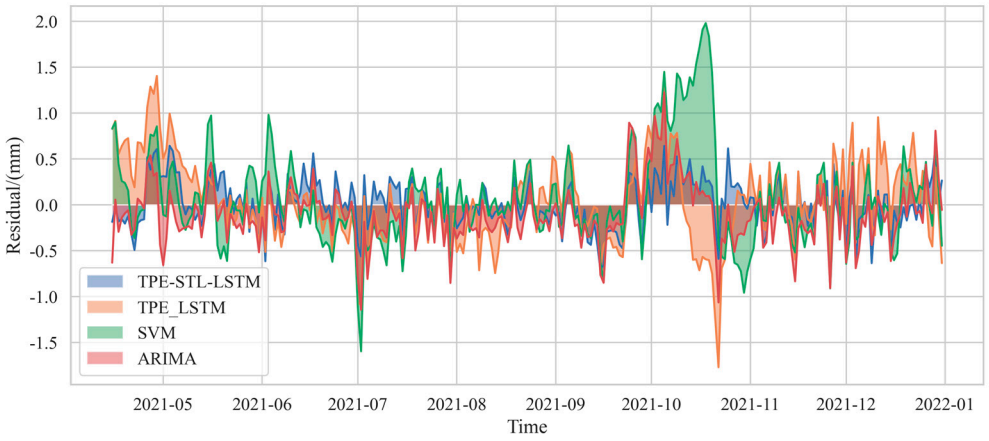

Figure 11. Residuals Area Charts.

To further quantify model prediction accuracy, the evaluation indicators of the TPE-LSTM, TPE-STL-LSTM, SVM, and ARIMA models are calculated, as shown in Table 4. The TPE-STL-LSTM model performs better on the validation set than the TPE-LSTM model, with a 45.5% reduction in MAPE, a 38.1% reduction in MAE, a 64.3% reduction in MSE, and a 40.2% reduction in RMSE. This indicates that by using the STL decomposition to distinguish different features of the time series data and then allowing the neural network model to learn the data, the model can effectively mine the periodic and trend changes in the data and improve prediction accuracy. Compared to traditional SVM and ARIMA models, the TPE-STL-LSTM model reduces MAPE by 67% and 64%, MAE by 42% and 17%, and MSE by 73% and 66%, respectively. As shown in Figure 12, all four models can make good predictions of the data trends, with the TPE-STL-LSTM model still demonstrating decent predictive performance for measurements with larger or smaller absolute values.

Table 4. Evaluation of predicted results (all values in mm).

Criteria	TPE_LSTM	TPE_STL_LSTM	SVM	ARIMA	Trend	Seasonal	Residual
MAPE	0.2321	0.1265	0.3880	0.3609	0.0223	0.1744	1.2330
MAE	0.3617	0.2237	0.3920	0.2712	0.0223	0.1821	0.0992
MSE	0.2173	0.0776	0.2906	0.2348	0.0007	0.0530	0.0158
RMSE	0.4661	0.2785	0.5391	0.4846	0.0272	0.2302	0.1255

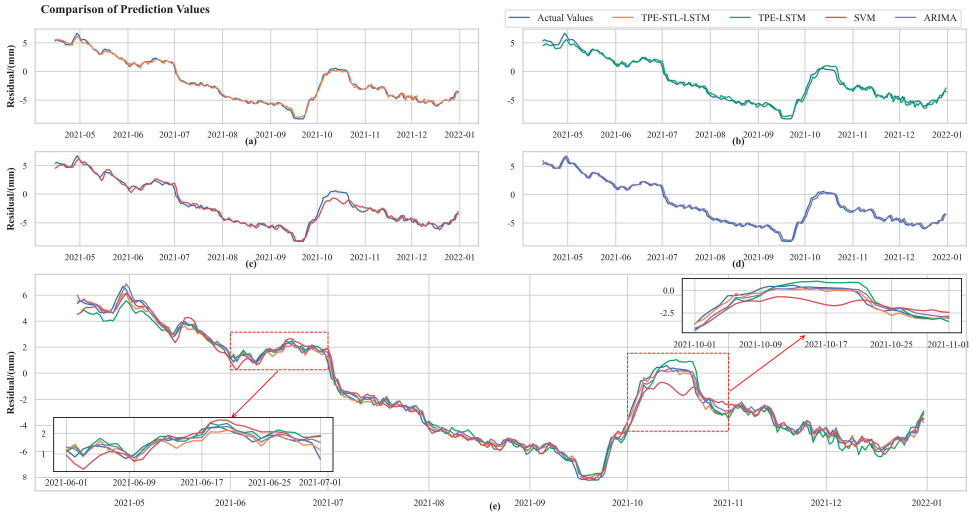

Figure 12. Comparison of different model forecasts: (**a**) TPE-STL-LSTM model; (**b**) TPE-LSTM model; (**c**) SVM model; (**d**) ARIMA model; (**e**) Comparison of all predicted values.

As a classical statistical analysis method, multiple regression models have the advantages of concise mathematical expressions, good interpretability, and low data requirements, making them widely used in hydrological engineering data analysis. However, multiple regression models have clear limitations in handling nonlinear relationships and high-dimensional data. To ensure the completeness of the analysis, this study applies four different multiple regression models, namely Linear Regression, Second-Order Polynomial Regression, Lasso Regression, and Ridge Regression, provided by the open-source project scikit-learn [34], to make predictions on a common dataset.

These models are all used to establish the relationships between multiple independent variables and a dependent variable. As shown in Figure 13, the predicted mean squared errors (MSE) of the four models are around 2.0. However, it should be noted that multiple regression models may fail to capture the complex relationships between historical dependencies and high-dimensional features. This limitation restricts the predictive ability of the models and may result in significant prediction errors.

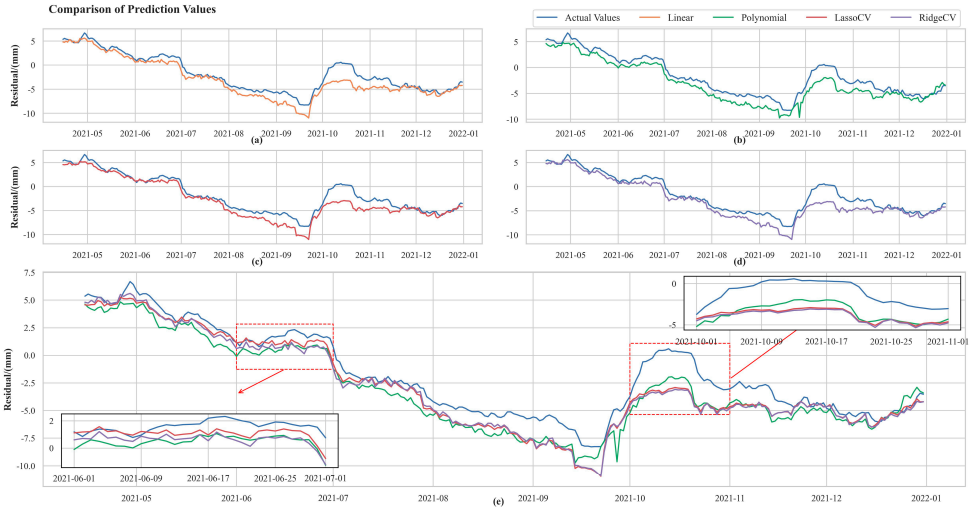

Figure 13. Comparison of different model forecasts based on regression: (**a**) Linear Regression Model; (**b**) Second-Order Polynomial Regression Model; (**c**) Lasso Regression Model; (**d**) Ridge Regression Model; (**e**) Comparison of all predicted values.

Neural network models, unlike traditional statistical models, are data-driven black-box models. However, due to their powerful predictive capabilities and automatic feature learning (which eliminates the need for manual feature extraction), they have been increasingly used by researchers in recent years for time series prediction and safety status determination in dam engineering. In this context, the autoregressive model based on historical time series data showed good performance. A series of hyperparameter optimization procedures have made it possible to embed this model into the production system. However, as a black-box model, its interpretability severely limits its development in safety monitoring. Nevertheless, the opacity of the model can be reduced through methods such as pre-training models, data decomposition, and rational network structure design.

5. Conclusions

In this study, we constructed a TPE-STL-LSTM model and applied it to the Wanjiazhai Dam project. The following conclusions were drawn:

1. The Wanjiazhai project deformation prediction results are satisfactory, with monthly average residuals of less than 0.3 mm, meeting the engineering evaluation accuracy requirements and demonstrating the model's practicality.
2. Compared to the SVM and ARIMA models' prediction performance on the same dataset, the TPE-STL-LSTM model has higher accuracy, enabling more precise predictions. Compared to the TPE-LSTM and TPE-STL-LSTM models' prediction performance, The STL decomposition in the TPE-STL-LSTM model contributed to the model's interpretability by providing insights into the specific components.
3. The model achieves automated determination of all parameters, avoiding errors caused by manual parameter setting, effectively improving the automation level of modeling, and providing technical support for the construction of digital-twin water conservancy projects.
4. We have noticed that many excellent prediction models have been proposed and applied to the engineering field based on empirical data. However, there has been limited discussion and exploration on how to integrate these prediction models into production environments and ensure their stable and efficient operation. In our future work, we plan to discuss and evaluate some of the highly cited models, such as CNN-LSTM, HST (Hydrostatic Seasonal Time), and neural network models with attention mechanisms. We will assess their training time, number of parameters, stability in handling abnormal data, and other factors to explore their potential for practical implementation in real production environments. Furthermore, an area worthy of further research and improvement is how a multivariate prediction model based on historically measured data learns the deformation characteristics of different sections in space and utilizes them for safety warnings.

Author Contributions: Conceptualization, S.S.; Methodology, Q.Z.; Software, S.S.; Validation, Q.Z., T.Z. and Y.H.; Resources, T.Z.; Writing—original draft, S.S.; Writing—review & editing, Q.Z.; Visualization, S.S. and Y.H.; Supervision, Q.Z.; Project administration, Q.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Due to confidentiality reasons, the relevant data cannot be provided. Please understand, this is to ensure the security of information and the privacy protection of participants. Thank you for your understanding and respect.

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

References

1. Wu, Z. *Theory and Experimental Techniques for Dam Safety Monitoring*; China Water & Power Press: Beijing, China, 2009; ISBN 978-7-5084-6855-6.
2. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef]

3. Shi, X.; Chen, Z.; Wang, H.; Yeung, D.-Y.; Wong, W.; Woo, W. Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting. *Adv. Neural Inf. Process. Syst.* **2015**, *28*. [CrossRef]
4. Li, R.; Xiao, R. Sentiment evolution prediction based on improved Wolf Pack Algorithm optimized LSTM network. *Complex Syst. Complex. Sci.* **2023**, *5*, 1–13.
5. Yang, D.; Gu, C.; Zhu, Y.; Dai, B.; Zhang, K.; Zhang, Z.; Li, B. A Concrete Dam Deformation Prediction Method Based on LSTM with Attention Mechanism. *IEEE Access* **2020**, *8*, 185177–185186. [CrossRef]
6. Ou, B.; Wu, B.; Yuan, J.; Li, S. Deformation prediction model of concrete dam based on LSTM. *Adv. Water Resour. Hydropower Technol.* **2022**, *42*, 21–26.
7. Liu, Q.; Li, N. Application research of deep learning in dam deformation prediction. *Geomat. Spat. Inf.* **2020**, *43*, 201–203+207+210.
8. Wang, X.; Yin, H.; He, M. Potential conflict prediction of airport runway activities based on LSTM. *J. Beijing Univ. Aeronaut. Astronaut.* **2022**, *6*, 1–15. [CrossRef]
9. He, X.; Zhao, K.; Chu, X. AutoML: A Survey of the State-of-the-Art. *Knowl.-Based Syst.* **2021**, *212*, 106622. [CrossRef]
10. Kong, Q.; Qiao, H.; Fei, X.; Wu, Z.; Ge, J. Deformation prediction model of concrete arch dam based on SSA-LSTM. *Northwest Hydropower* **2022**, *3*, 81–86.
11. Gao, J.; Jia, Z.; Wang, X.; Xing, H. Degradation trend prediction of proton exchange membrane fuel cell based on PSO-LSTM. *J. Jilin Univ.* **2022**, *52*, 2192–2202. [CrossRef]
12. Li, P.; Su, H.; Guo, Z.; Qian, Q. Deformation monitoring model of dams based on artificial bee colony algorithm and Elman neural network. *Water Resour. Hydropower Eng.* **2017**, *48*, 104–108. [CrossRef]
13. Jones, D.R.; Schonlau, M. Efficient Global Optimization of Expensive Black-Box Functions. *J. Glob. Optim.* **1998**, *13*, 455. [CrossRef]
14. Shahriari, B.; Swersky, K.; Wang, Z.; Adams, R.P.; de Freitas, N. Taking the Human Out of the Loop: A Review of Bayesian Optimization. *Proc. IEEE* **2016**, *104*, 148–175. [CrossRef]
15. Gao, Z.; Bao, T.; Li, Y.; Wang, Y. Deformation prediction model of dams based on Bayesian-optimized LightGBM. *J. Yangtze River Sci. Res. Inst.* **2021**, *38*, 46–50+57.
16. Cheng, L.; Chen, J.; Ma, C.; Yang, J.; Xu, X.; Yuan, S. Multipoint Deformation Safety Monitoring Model for Concrete Arch Dams Based on Bayesian Model Selection and Averaging. *Struct. Control. Health Monit.* **2023**, *2023*, e5042882. [CrossRef]
17. Zhang, W.; Zhang, Y.; Gu, X.; Wu, C.; Han, L. Application of LSTM and Prophet Algorithm in Slope Displacement Prediction. In *Application of Soft Computing, Machine Learning, Deep Learning and Optimizations in Geoengineering and Geoscience*; Zhang, W., Zhang, Y., Gu, X., Wu, C., Han, L., Eds.; Springer: Singapore, 2022; pp. 73–92. ISBN 9789811668357.
18. Zhu, J.; Liu, J.; Wu, P.; Chen, H.; Zhou, L. A Novel Decomposition-Ensemble Approach to Crude Oil Price Forecasting with Evolution Clustering and Combined Model. *Int. J. Mach. Learn. Cyber.* **2019**, *10*, 3349–3362. [CrossRef]
19. Zhang, J.; Heng, Y. Deformation prediction model of concrete dams based on VMD-PE-CNN. *Water Resour. Hydropower Eng.* **2022**, 1–11.
20. Li, H. Short-Term Passenger Flow Prediction of Urban Rail Transit Based on Time Series Decomposition and LSTM Neural Network. Master's Thesis, Beijing Jiaotong University, Beijing, China, 2020.
21. Zhou, Q. Application Research of CNN and LSTM in Short-Term Stock Price Prediction of Cyclical Stocks. Master's Thesis, Zhejiang University, Hangzhou, China, 2022.
22. Li, B.; Hu, D.; Yang, J.; Cheng, L. Displacement prediction of earth-rock dams based on MLR-SARIMA model. *J. Eng. Sci. Technol.* **2019**, *51*, 108–114. [CrossRef]
23. Dong, Y.; Liu, X.; Li, Y.; Jia, Y. Deformation prediction model of dams based on EMD-EEMD-LSTM. *Water Power* **2022**, *48*, 1–6.
24. Lin, C.; Wang, X.; Su, Y.; Zhang, T.; Chen, Z. Deformation prediction of concrete dams using combined clustering methods and deep learning. *J. Hydroelectr. Eng.* **2022**, *41*, 1–20.
25. Hu, A.; Bao, T.; Yang, C.; Zhang, J. Combined prediction model of dam deformation based on LSTM-Arima and its application. *J. Yangtze River Sci. Res. Inst.* **2020**, *37*, 64–68+75.
26. Sherstinsky, A. Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) Network. *Phys. D Nonlinear Phenom.* **2020**, *404*, 132306. [CrossRef]
27. Zaremba, W.; Sutskever, I.; Vinyals, O. Recurrent Neural Network Regularization. *arXiv* **2015**, arXiv:1409.2329.
28. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. *Adv. Neural Inf. Process. Syst.* **2019**, *32*, 12.
29. Duda, R.O.; Hart, P.E.; Stork, D.G. *Pattern Classification*, 2nd ed.; Wiley: New York, NY, USA, 2001; ISBN 978-0-471-05669-0.
30. Bergstra, J.S.; Bardenet, R.; Bengio, Y.; Kégl, B. Algorithms for Hyper-Parameter Optimization. *Adv. Neural Inf. Process. Syst.* **2011**, *24*, 9.
31. Akiba, T.; Sano, S.; Yanase, T.; Ohta, T.; Koyama, M. Optuna: A Next-Generation Hyperparameter Optimization Framework. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Anchorage, AK, USA, 4–8 August 2019; Association for Computing Machinery: New York, NY, USA, 2019; pp. 2623–2631.
32. Cleveland, R.B.; Cleveland, W.S.; McRae, J.E.; Terpenning, I. STL: A Seasonal-Trend Decomposition Procedure Based on Loess (with Discussion). *J. Off. Stat.* **1990**, *6*, 3–73.
33. Vapnik, V. *The Nature of Statistical Learning Theory*, 2nd ed.; Springer: New York, NY, USA, 1999; ISBN 978-0-387-98780-4.

34. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-Learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
35. Smith, T.G. pmdarima: ARIMA Estimators for Python. 2017. Available online: http://alkaline-ml.com/pmdarima/ (accessed on 4 May 2023).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.