

Article

# Predicting the Posture of High-Rise Building Machines Based on Multivariate Time Series Neural Network Models

Xi Pan <sup>1</sup>, Janguang Huang <sup>1,2,\*</sup> , Yiming Zhang <sup>2</sup> , Zibo Zuo <sup>1</sup> and Longlong Zhang <sup>1</sup>

<sup>1</sup> General Engineering Institute of Shanghai Construction Group, Shanghai Construction Group Co., Ltd., Shanghai 200080, China; panxi@scg.cn (X.P.); zuozb@scg.cn (Z.Z.); zhangll@scg.cn (L.Z.)

<sup>2</sup> School of Civil and Transportation Engineering, Hebei University of Technology, Xiping Road 5340, Tianjin 300401, China; yiming.zhang@hebut.edu.cn

\* Correspondence: 202111601004@stu.hebut.edu.cn

**Abstract:** High-rise building machines (HBMs) play a critical role in the successful construction of super-high skyscrapers, providing essential support and ensuring safety. The HBM's climbing system relies on a jacking mechanism consisting of several independent jacking cylinders. A reliable control system is imperative to maintain the smooth posture of the construction steel platform (SP) under the action of the jacking mechanism. Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), and Temporal Convolutional Network (TCN) are three multivariate time series (MTS) neural network models that are used in this study to predict the posture of HBMs. The models take pressure and stroke measurements from the jacking cylinders as inputs, and their outputs determine the levelness of the SP and the posture of the HBM at various climbing stages. The development and training of these neural networks are based on historical on-site data, with the predictions subjected to thorough comparative analysis. The proposed LSTM and GRU prediction models have similar performances in the prediction process of HBM posture, with medians  $R^2$  of 0.903 and 0.871, respectively. However, the median MAE of the GRU prediction model is more petite at 0.4, which exhibits stronger robustness. Additionally, sensitivity analysis showed that the change in the levelness of the position of the SP portion of the HBM exhibited high sensitivity to the stroke and pressure of the jacking cylinder, which clarified the position of the cylinder for adjusting the posture of the HBM. The results show that the MTS neural network-based prediction model can change the HBM posture and improve work stability by adjusting the jacking cylinder pressure value of the HBM.

**Keywords:** high-rise building machine (HBM); steel platform (SP); neural networks (NNs); multivariate time series (MTS); posture prediction



**Citation:** Pan, X.; Huang, J.; Zhang, Y.; Zuo, Z.; Zhang, L. Predicting the Posture of High-Rise Building Machines Based on Multivariate Time Series Neural Network Models.

*Sensors* **2024**, *24*, 1495. <https://doi.org/10.3390/s24051495>

Academic Editor: Filippo Ubertini

Received: 18 January 2024

Revised: 14 February 2024

Accepted: 23 February 2024

Published: 25 February 2024

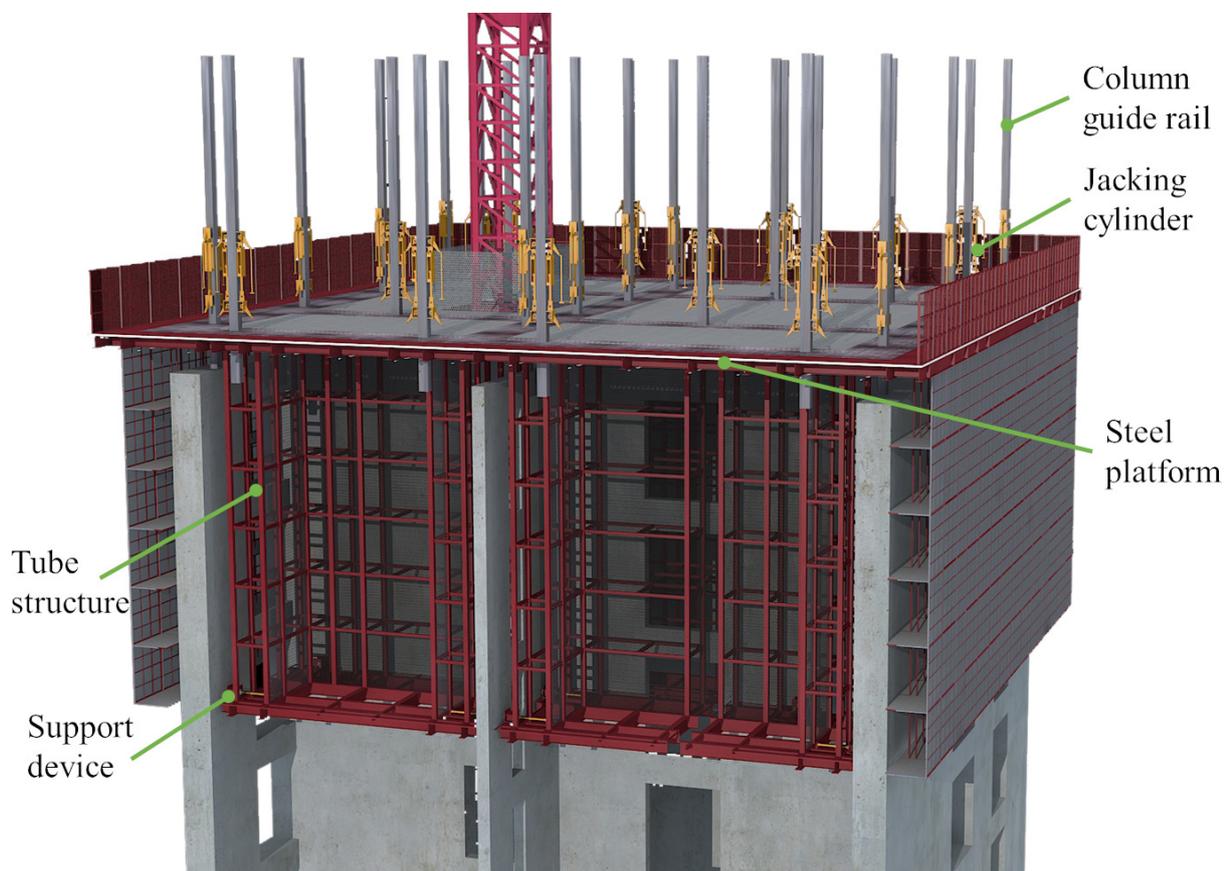


**Copyright:** © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

High-rise building machines (HBMs) are innovative self-elevating steel-structure-building machines designed for constructing super high-rise buildings [1–4]. They offer several advantages: high automation, enhanced safety, simplified construction organization, assured structural quality, and standardized processes [5–7]. These features contribute to accelerated construction progress. Figure 1 illustrates the three-dimensional structural details of the HBM [8–10]. The levelness of the steel platform (SP) represents the HBM's posture changes [11–13]. During the climbing process of the HBM, the telescopic support device retracts from the structurally reinforced concrete shear wall, while the climbing system supports the SP to ascend along the column guide rail. Once the SP reaches the desired position, the telescopic support device is extended into the reserved hole of the structurally reinforced concrete shear wall. However, due to uneven stacking loads on the SP, the levelness of the SP may fluctuate. If excessive levelness deviation is not promptly corrected, the HBM may undergo severe shape distortion, leading to collisions with obstacles and halting the climbing [14]. In the early stages of HBM usage, workers employed the hanging-line-pendant method and plumbing instruments to measure the HBM's verticality.

They used jacks to lift parts with low levelness, applying corrective forces to gradually restore the SP's levelness while continuing to climb [15,16]. Subsequently, climbing control operators monitored the HBM's posture information directly via signal and image displays on the center console. Although this approach improved efficiency and reduced measurement errors, it incurred significant costs. The detection equipment enhanced production efficiency, minimized operational errors, ensured detection repeatability, and generated extensive monitoring data [17–20]. However, these measures still failed to prevent levelness deviations during the climbing process, addressing the complexity of repairing such deviations and mitigating subjective errors in worker operations. Zuo et al. conducted a study on the determination of concrete strength during the construction of HBMs by proposing a remote real-time monitoring system [10]. Pan et al. collected the vibration frequency of HBMs and utilized a machine learning method to categorize and assess the four primary working states of this machinery [9]. They also proposed a method for monitoring and issuing warnings during the climbing process of an HBM [8]. However, these methods are primarily focused on monitoring and providing early warnings for the safe operation of HBMs. There needs to be more in-depth research on predicting the future working state of this machinery and dealing with potential dangers.



**Figure 1.** Schematic diagram of the HBM.

We have discovered that a change in posture of a HBM exhibits a multivariate time series (MTS)-related process. To effectively capture the intricate patterns and dependencies in time series data, neural networks (NNs) are a valuable tool [21–26]. NNs possess linear and nonlinear fitting capabilities, making them adept at handling such data. Notably, NN technology has shown promising results in prediction tasks [27–30]. It is important to note, though, that these neural networks don't do as well when they have to deal with tasks that use MTS class information and simulations that have strong back-and-forth correlations. Several articles from 2015 and 2020 have talked about recurrent neural

networks (RNNs) [31–35]. These are a type of time series neural network that have memory cells and feedback connections that make it easier for information to flow within the model and handle strong back-and-forth dependencies well. Using a recursive structure, RNNs retain contextual information from past observations and apply it to current or future predictions. However, RNNs encounter challenges when dealing with long time series, leading to gradient vanishing or exploding issues. Consequently, traditional RNNs excel at solving short-term dependencies. Advanced architectures such as the Long Short-Term Memory (LSTM) [36–38] and the Gated Recurrent Unit (GRU) [39–42] have been developed to address this. LSTM employs memory cells and gating mechanisms to filter out the noise and selectively capture long-term dependencies more efficiently. On the other hand, GRU makes the LSTM structure simpler by only using hidden states to send information. This cuts down on parameters and computational complexity while keeping the performance the same in some situations. Not only are recurrent neural networks used to process time series data, but so are Temporal Convolutional Neural Networks (TCNs) [43–46]. TCNs modify the conventional approach of using two-dimensional convolutional kernels for image processing by employing one-dimensional convolutional kernels to extract local patterns and features. TCNs possess a large enough sensory field to find patterns and regularities across a range of time scales by stacking multiple convolutional layers with pooling operations.

Previous studies have demonstrated the remarkable capabilities of neural networks in forecasting time series data. This study is organized as follows: Section 2 provides a detailed explanation of the characterization of the HBM through sensor-based monitoring during its climb. It also discusses the preprocessing techniques employed to handle this data effectively. Moving on to Section 3, an in-depth exploration of three multivariate time series-prediction models, namely the LSTM, GRU, and TCN models, is presented. The technical approach adopted to predict the HBM's posture in this study is also discussed. Section 4 delves into an insightful comparison between the prediction results of these three models and the actual values. It also includes a meticulous analysis of the potential sources of errors generated by the models. Finally, Sections 5 and 6 present the discussion and conclusions drawn from the research. It is shown that the suggested smart-prediction system is better than the old ways of manipulating HBMs and reduces the hysteresis of repairing the HBM's posture.

## 2. The Posture of the HBM

### 2.1. Data Sources

In this work, we observed and analyzed the HBM climbing process at the West Tower of the Shenzhen Xinghe Yabao Building project (356 m) in China. With an average climbing frequency of once every five days, we embarked on an exploration utilizing a dataset comprising 30 sets (days) of monitoring data. Each dataset, sampled at a rate of one sample per second, encompassed 168,048 monitoring data points. These data points comprised 68 sample characterizations, forming a foundation for our investigation. As showcased in Figure 2, our monitoring efforts focused on observing and recording the jacking stroke and pressure of 26 climbing jacking cylinders throughout the climbing process. We also monitored the levelness of 16 crucial monitoring points on the steel platform beam, as shown in Figure 3.

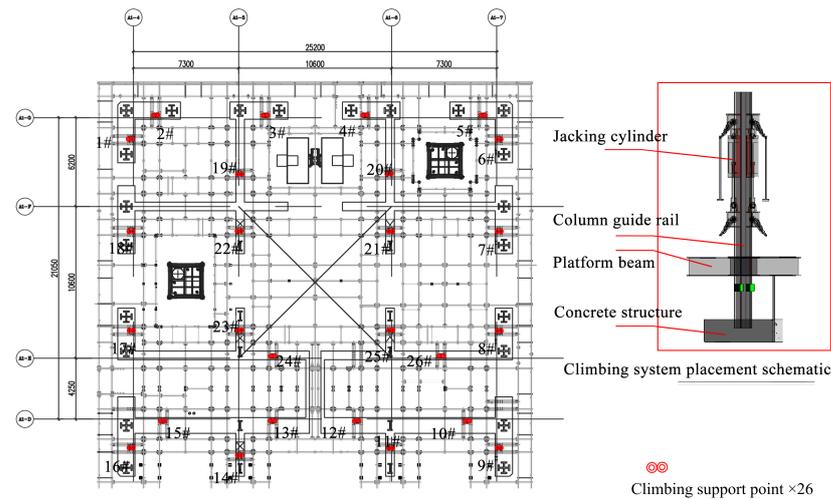


Figure 2. Schematic layout of jacking cylinders.

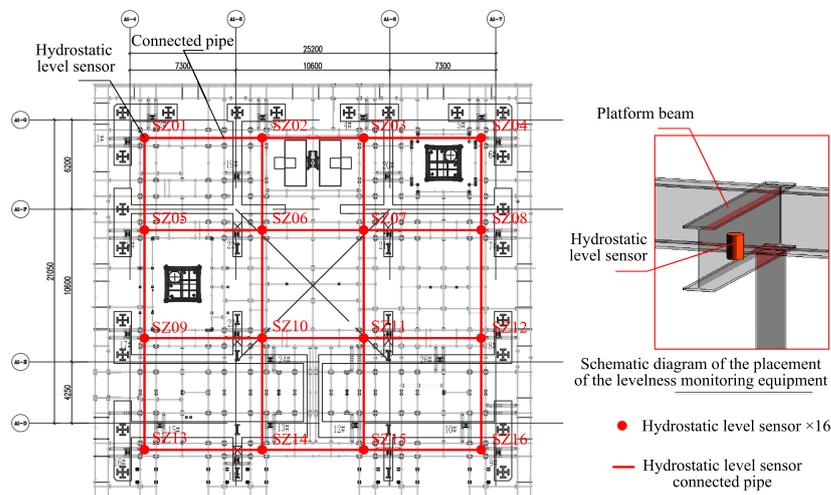
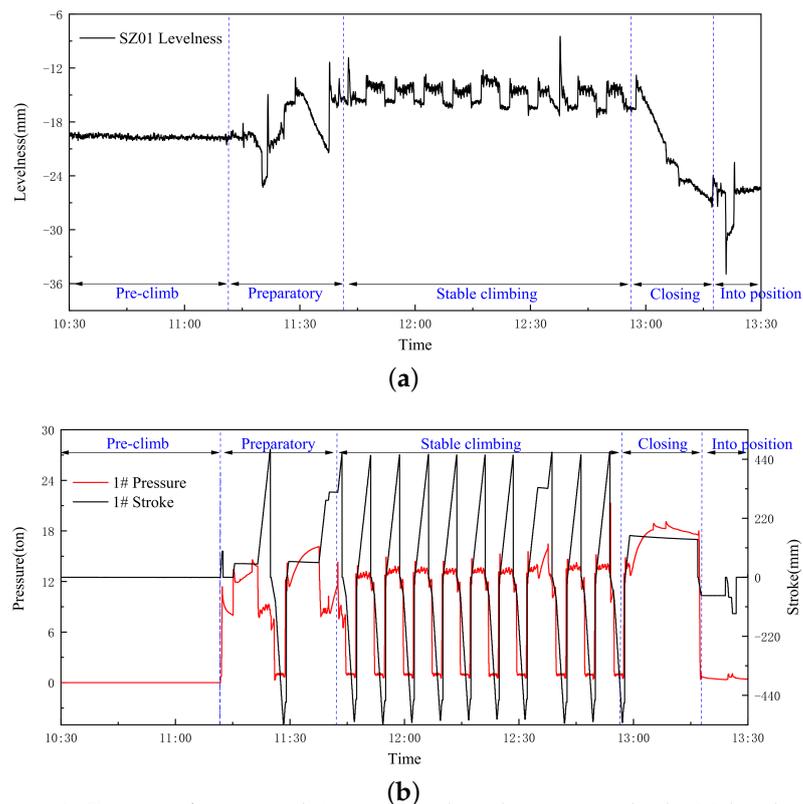


Figure 3. Schematic layout of levelness-monitoring equipment.

## 2.2. The Pre-Processing of Monitoring Data

Figure 4 depicts the once-climbing process of the HBM, presenting a narrative of the levelness of the SP, jacking cylinder jacking stroke, and jacking-cylinder pressure over time. This intricate climbing process can be divided into five distinct phases: the pre-climb stage, the preparation stage, the stable climbing stage, the closing of the climbing stage, and the climbing-into-position stage. Remarkably, the pre-climbing and climbing-into-position stages exhibited minimal changes in the posture of the HBM. On the other hand, the preparation stage and the closing of the climbing stage are influenced by equipment-operation errors arising from installing the jacking cylinders, such as support jacking cylinder position errors and jacking cylinder tilt. And, we want, as the climb progresses into the stabilized climbing stage, a gradual elimination of the levelness deviation of the SP to become evident. In addition, the initial levelness deviation observed in the stable climbing stage is not attributed to the climbing process itself. Consequently, our focus narrows to studying and analyzing the data solely from the stable climbing stage, where the SP's levelness is relatively stable and reliable. To mitigate the occurrence of data jumps during multiple stable climbs, we took measures to initialize the monitoring data for each levelness of the SP at the commencement of every stable climb, ensuring the accuracy and consistency of our analysis.



**Figure 4.** One set of measured time-range data during one climb: (a) levelness-sensor monitoring data, (b) jacking cylinder-monitoring instruments monitoring data.

To provide further insight into the study data, we conducted a brief analysis of the monitoring data collected during the stabilized climbing stage. The jacking cylinder stroke values ranged from  $-551.4$  to  $551.79$ , the jacking cylinder pressure values ranged from  $0$  to  $53.4$ , and the SP levelness values ranged from  $-39.12$  to  $40.48$ .

### 3. Methodology

Selecting an appropriate machine learning algorithm is crucial to extracting valuable insights from sensor-monitoring data during the HBM-climbing process. This algorithm should uncover the hidden data and establish a mapping relationship between the HBM climbing parameters and postures. In order to accurately predict postures in HBMs, the chosen model must effectively handle MTS features and automatically capture the nonlinear relationships present in the data. This study employs three MTS neural networks—LSTM, GRU, and TCN—to predict the postures during HBM climbing.

#### 3.1. MTS-Prediction Model

##### 3.1.1. LSTM Model

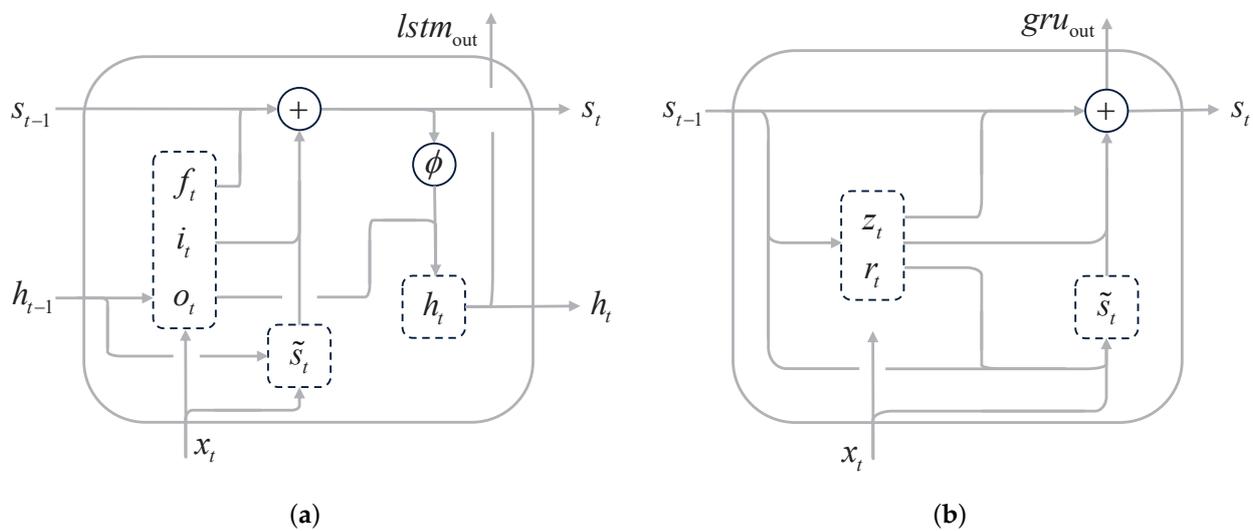
The LSTM model, introduced by Hochreiter and Schmidhuber (1997) [47], addresses the limitations of traditional RNNs by incorporating explicit memory-management mechanisms. By explicitly adding and subtracting information (see Equation (1)) from the LSTM state, the LSTM model ensures that each state cell remains constant over time. Moreover, gated cells and carefully designed memory cells enable the model to preserve its long-term memory while maintaining the relevance of the most recent state. This prevents information distortion, disappearance, and sensitivity explosion, which could occur in other models like Neural Turing Machines

$$S_t = S_{t-1} + \Delta S_t \quad (1)$$

A typical LSTM cell, as illustrated in Figure 5a, consists of three key gates: the forget gate, the input gate, and the output gate. These gates play a critical role in managing

the memory of the network by modulating the activation of the weighted sum function  $W(*)$ . As depicted in Equation (2) [47], the forget gate  $f_t$  decides which information should be retained or discarded. It utilizes a sigmoid function ( $\sigma(x) = \frac{1}{1+e^{-x}}$ ) to compute a vector (output) based on the previous cell state  $s_{t-1}$  and the current input information  $x_t$ . The purpose of this scaling is to rescale the values of each dimension of the data within the range of  $[0, 1]$ . Subsequently, the sigmoid function  $\sigma(x)$  determines the relevance of the current information and determines which information will contribute to the computation of the cell state  $s_t$ . The input gate  $i_t$  determines the information that should be stored in the cell state. It employs the sigmoid function  $\sigma(x)$  to compute its activation based on the previous cell state  $s_{t-1}$  and the current input information  $x_t$ . The candidate cell state  $\tilde{s}_t$  is computed using the hyperbolic tangent function ( $\phi(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$ ), which yields a value ranging from  $-1$  to  $+1$ . The outputs of these computations ( $s_{t-1}$  and  $\tilde{s}_t$ ) are combined to update the current internal cell state  $s_t$ . Lastly, the output gate  $o_t$  determines the final output  $lstm_{out}$  and the next hidden state  $h_t$ . The output gate  $o_t$  multiplies the previous hidden state  $h_{t-1}$  and the current input information  $x_t$  with the sigmoid function  $\sigma(x)$ -activation output. These calculations determine the information that the hidden layer  $h_t$  will carry.

$$\begin{cases} f_t = \sigma(W_f s_{t-1} + U_f x_t + b_f) \\ i_t = \sigma(W_i s_{t-1} + U_i x_t + b_i) \\ o_t = \sigma(W_o s_{t-1} + U_o x_t + b_o) \\ \tilde{s}_t = \phi(W(o_t \odot s_{t-1}) + U x_t + b) \\ s_t = f_t \odot s_{t-1} + i_t \odot \tilde{s}_t \\ h_t = o_t \phi(s_t) \end{cases} \quad (2)$$



**Figure 5.** Diagram of MTS prediction RNNs model: (a) schematic diagram of LSTM model, (b) schematic diagram of GRU model.

### 3.1.2. GRU Model

The GRU model (Cho et al., 2014 [39]) is a simplified variant of the RNNs architecture. Unlike the LSTM model, the GRU model consists of only two gates: the reset gate and the update gate. Figure 5b illustrates the structure of the GRU model. As shown in Equation (3) [39], in contrast to LSTM, GRU eliminates the need for a separate hidden state  $h$ . Instead, it directly replaces the input gates  $i_t$  with  $(1 - z_t)$ , where  $z_t$  represents the update gate. The reset gate  $r_t$  determines the combination of the input information  $x_t$  and

the previous cell state  $s_{t-1}$ . The update gate  $z_t$  controls the retention of the previous cell state  $s_{t-1}$ . The cell state  $s_t$  is then updated by combining all the computed information.

$$\begin{cases} r_t = \sigma(W_r s_{t-1} + U_r x_t + b_r) \\ z_t = \sigma(W_z s_{t-1} + U_z x_t + b_z) \\ \tilde{s}_t = \phi(W(r_t \odot s_{t-1}) + U x_t + b) \\ s_t = z_t \odot s_{t-1} + (1 - z_t) \odot \tilde{s}_t \end{cases} \quad (3)$$

### 3.1.3. The TCN Model

The TCN model (Bai et al., 2018 [43]) is a one-dimensional dilated causal-convolutions neural network (CNN) designed for solving time series problems. It incorporates the architecture of CNNs and introduces the concept of dilated causal convolutions. Figure 6 illustrates the main components of the TCN model. The TCN model utilizes a one-dimensional dilated causal convolution with a filter size of 3 and a residual network structure. The combination of causal convolution and dilated convolutions allows the convolutional layer to expand its receptive field while strictly adhering to temporal constraints. This larger receptive field enables the model to learn and extract historical information from multivariate time series data. The expansion convolution operation  $H(S)$  is employed in the sequence unit  $S$  to process a one-dimensional time series input  $X = (x_1, x_2, \dots, x_{t-1})$ , where  $f : \{0, 1, \dots, k - 1\}$  is the filter. The operation is defined as [43]

$$H(S) = (X^*_{df})(S) = \sum_{i=0}^{k-1} f(i) \cdot X_{S-d \cdot i} \quad (4)$$

where  $S$  represents the input sequence information,  $*$  is the convolution operator,  $k$  denotes the filter size,  $d = 2^v$  represents the dilation factor, where  $i$  is the levelness of the network, and  $S - d \cdot i$  indicates the localization of specific historical information.

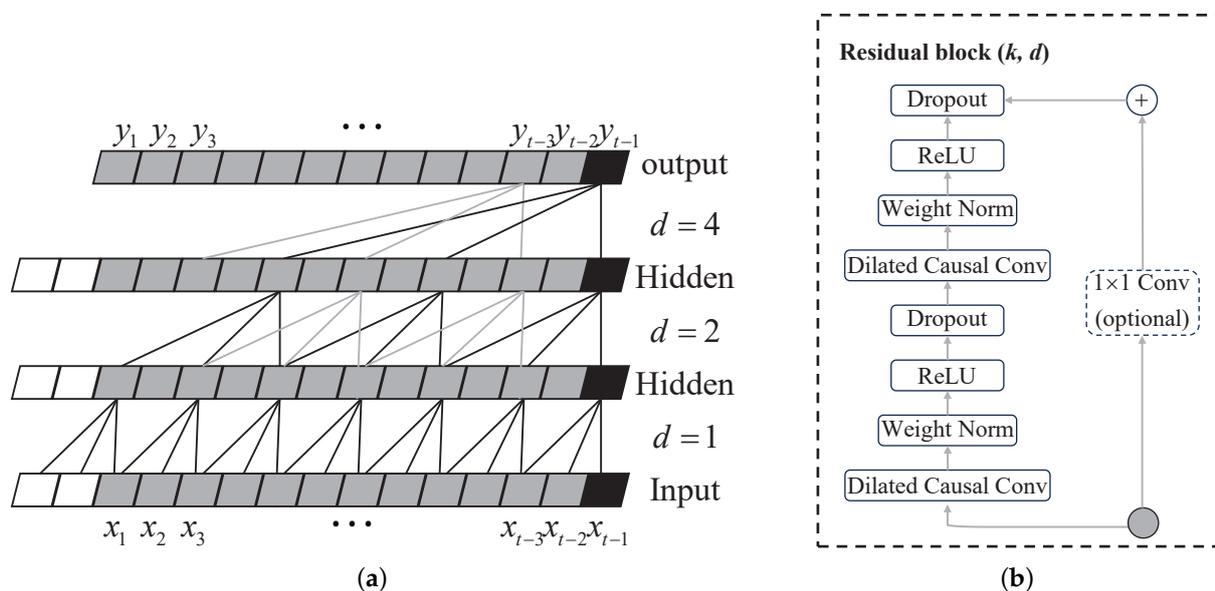


Figure 6. Schematic diagram of TCN model: (a) dilated causal convolution, (b) residual block network structure.

Additionally, the TCN model incorporates a residual network structure, which helps address issues like gradient vanishing or explosion and model degradation during deep neural network training. Typically, the TCN model consists of two layers of residual modules. Each module comprises dilated causal convolution, weight normalization, ReLU-activation function ( $f(x) = \max(0, x)$ , where  $x$  is the input), and dropout. The previous layer's output serves as the input to the dilated causal convolution of the next layer, ensuring that both the inputs  $x_i$  and outputs  $x_{i+1}$  of the module have the exact dimensions. An additional  $1 \times 1$  convolution is applied to restore the original number of channels.

$$x_{i+1} = f(X + \mathcal{F}(X)) \quad (5)$$

where  $\mathcal{F}(\ast)$  refers to a series of transformations leading from one residual module to another.

### 3.2. The Architecture for the Posture-Prediction Model of HBMs

#### 3.2.1. Data Standardization and Set Partitioning

The data obtained from different sensors often have varying size ranges, resulting in sample data in the dataset having different scales. We perform standardized preprocessing on the raw data to address this issue and prevent model bias towards certain features due to the magnitude differences during training. In this experiment, we apply min–max normalization to the sensor data samples, resulting in a value range of  $[-1, 1]$ . After normalization, the scaled values of the sensor data samples  $X_{\text{norm}}$  or  $Y_{\text{norm}}$  (where  $X$  represents the active input dataset and  $Y$  represents the passive output dataset) can be expressed as follows:

$$X_{\text{norm}} = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \quad (6)$$

where  $X_{\max}$  denotes the maximum value of the sample dataset, and  $X_{\min}$  denotes the minimum value of the sample dataset.

The dataset is partitioned into a training set (80%), a validation set (10%), and a test set (10%). The training set is utilized to train the neural network model. The validation set is used to assess the performance of the untrained model, serving as a means to prevent overfitting and underfitting. Finally, the test set evaluates the optimal model's accuracy, precision, and recall performance metrics.

#### 3.2.2. The Evaluation of Predictive Performance

In this study, the mean absolute error (MAE) and coefficient of determination for Goodness of Fit ( $R^2$ ) is used as an evaluation metric for the prediction model:

$$\begin{aligned} \text{MAE} &= \frac{1}{m} \sum_{i=1}^m |y_i - \hat{y}_i| \\ R^2 &= 1 - \frac{\sum_{i=1}^m (y_i - \hat{y}_i)^2}{\sum_{i=1}^m (y_i - \bar{y}_i)^2} \end{aligned} \quad (7)$$

where  $m$  represents the total number of moments,  $y_i$  denotes the actual value at moment  $i$ ,  $\hat{y}_i$  refers to the predicted value of the model output at moment  $i$ , and  $\bar{y}_i$  denotes the average of all actual values.

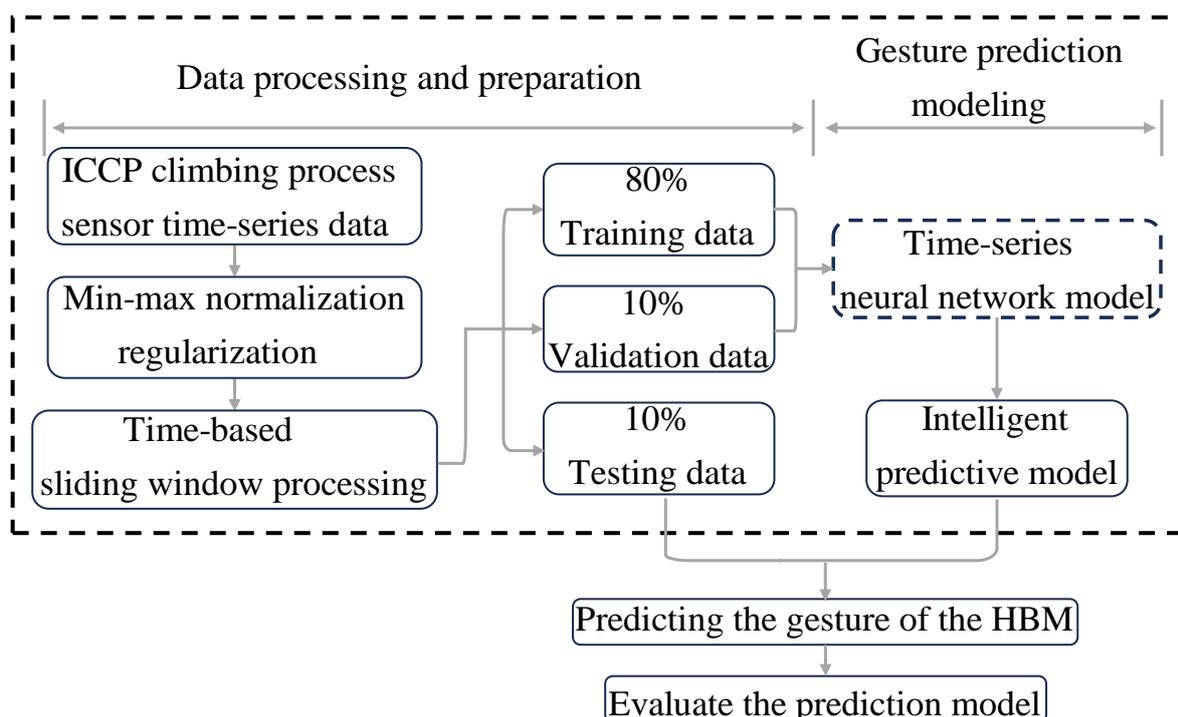
The MAE can better reflect the reality of prediction-value errors. A more petite MAE indicates a better prediction effect. On the other hand,  $R^2$  ranges between 0 and 1, where a value closer to 1 indicates a better fit of the model to the data, suggesting a more vital predictive capability.

### 3.2.3. MTS-Prediction Architecture

The flowchart in Figure 7 illustrates the technical route employed in this study to predict the HBM postures. Initially, the sensor-monitoring data of the HBM is preprocessed. Subsequently, a MTS neural network model is trained using the training set data, while the validation-set data is utilized for model selection and fine-tuning. This experiment employs the root mean square error (RMSE) as the loss function:

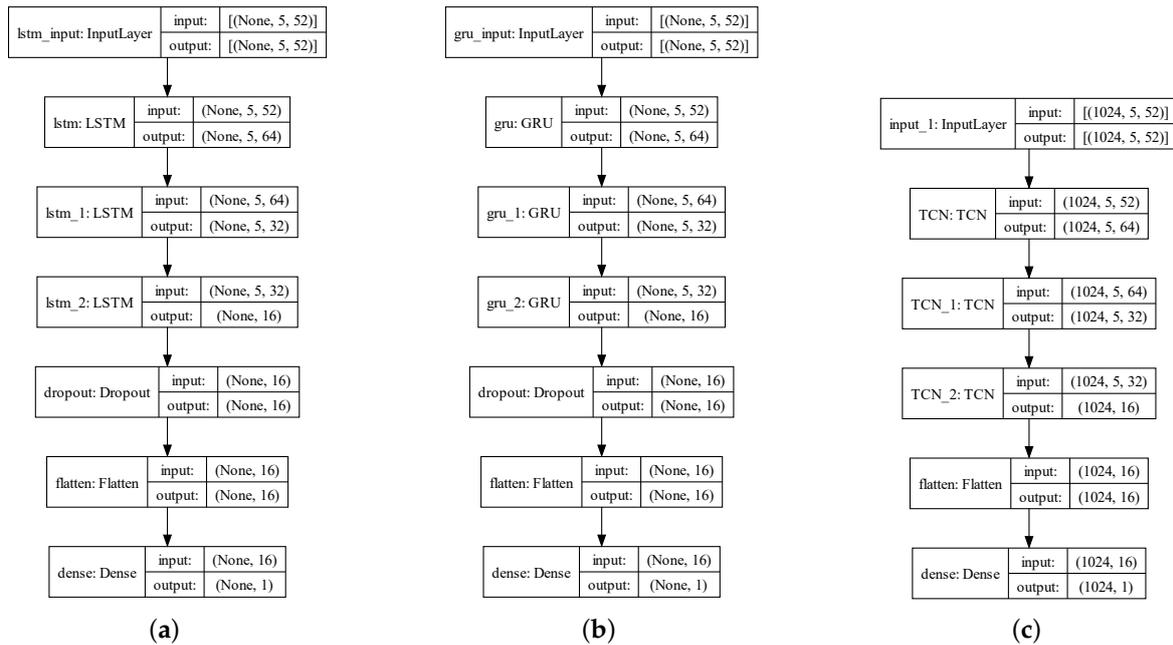
$$\text{RMSE} = \sqrt{\frac{1}{m} \sum_{i=1}^m (y_i - \hat{y}_i)^2} \quad (8)$$

In addition, the Adam optimizer was used to update the network parameters for model training and model output.



**Figure 7.** Flowchart of the posture-prediction model framework for the HBM-climbing process.

Figure 8 illustrates this study's three MTS neural network-model architectures. Notably, there are similarities between the LSTM and GRU model architectures. The LSTM, GRU, and TCN functions are utilized in the hidden layers of these neural network-model architectures. The weight initialization techniques used for all three functions are the commonly used Xavier initialization, including the initialization of input gate, forget gate, output gate, and candidate memory cell weights in the LSTM function, the initialization of reset gate, update gate, and candidate hidden state weights in the GRU function, and the initialization of one-dimensional convolution kernel, residual block, and fully connected layer weights in the TCN function. All three models in this experiment use 3 hidden layers to make the prediction results are comparable and set the same learning rate  $\eta = 0.01$ . In order to prevent overfitting, the Dropout function ( $1 - p = 0.2$ ) is employed as a regularization technique. It is worth mentioning that both the LSTM and GRU models implement the Dropout function after the hidden layers, while the TCN model incorporates the Dropout function within each TCN function. Finally, a fully connected layer is employed to achieve a dimensionality reduction in the output.



**Figure 8.** MTS neural network-model architecture: (a) LSTM model architecture, (b) GRU model architecture, (c) TCN model architecture.

## 4. Results

### 4.1. Sensitivity Analysis

This study selected 26 jacking cylinder strokes and 26 jacking cylinder pressures as the active inputs  $X$  for the multivariate analysis of HBM-climb posture-prediction. The levelness of 16 monitoring points on the SP during the HBM climb was chosen as the passive output  $Y$ .

Figure 9a,b illustrates the Pearson correlation coefficients  $\rho_{X,Y}$  (see Equation (9)) between the jacking cylinder stroke and the SP levelness, and between the jacking cylinder pressure and the SP levelness, separately. The Pearson correlation coefficient provides insight into the relationship between the jacking cylinder stroke, jacking cylinder pressure, and the levelness of each position on the SP. Notably, the displacements of the SP at SZ01, SZ03, SZ04, SZ07, SZ08, SZ09, SZ12, and SZ14 show a more pronounced response to the activity of the jacking cylinder.

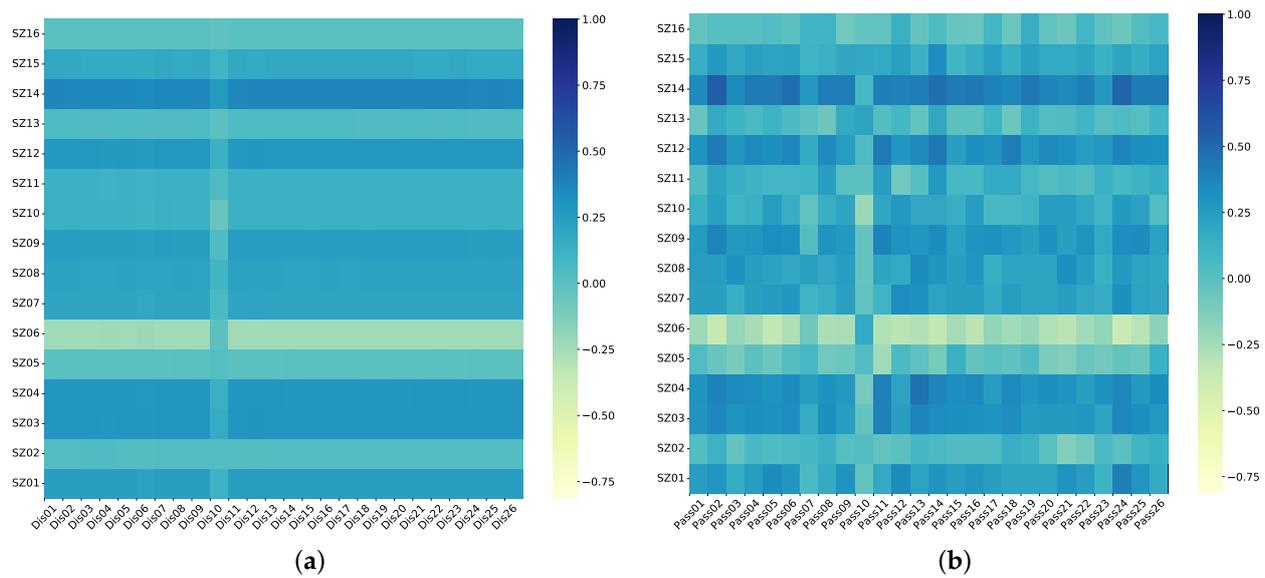
$$\rho_{X,Y} = \text{corr}(X, Y) = \frac{\text{cov}(X, Y)}{\sigma_X \sigma_Y} = \frac{E[(X - \mu_X)(Y - \mu_Y)]}{\sigma_X \sigma_Y} \quad (9)$$

where  $\text{cov}(*, *)$  is the covariance and  $\sigma$  is the standard deviation.

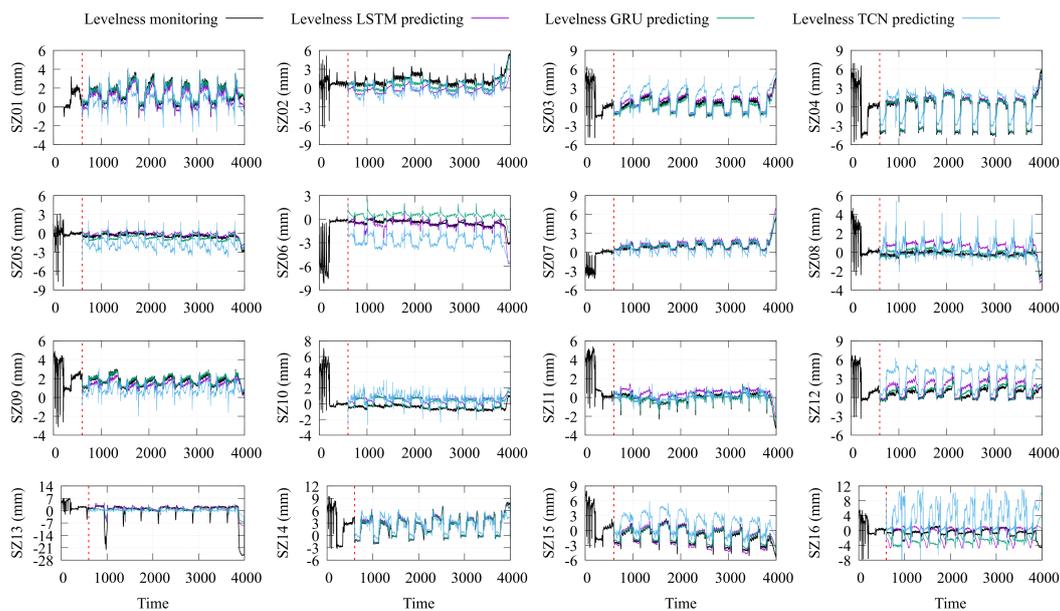
### 4.2. Model Comparison and Evaluation

In this study, a posture-prediction model for HBM is constructed using the LSTM neural network, GRU neural network, and TCN neural network. These models were trained and validated using separate datasets. After 200 epochs with a batch size of 1024, the models generated an intelligent prediction model for the 16-point levelness of the SP in HBM.

The intelligent prediction model takes historical time series values of the monitored stroke and pressure of the jacking cylinder as input, and the predicted time series values of steel platform levelness as output. A test dataset was used to validate the final HBM posture-prediction model. The prediction results of the LSTM, GRU, and TCN models were compared with the actual levelness of the 16 monitoring points on the construction platform, as shown in Figure 10.



**Figure 9.** Sensitivity analysis of sensor-monitoring information during climbing of HBM: (a) Sensitivity between jacking cylinder stroke and SP levelness, (b) sensitivity between jacking cylinder pressures and SP levelness.



**Figure 10.** Predicted SP levelness using LSTM, GRU, and TCN models.

Based on the fluctuating patterns and peaks of the overall levelness, the LSTM and GRU models exhibited a certain levelness of reliability in predicting the HBM postures. However, the TCN model demonstrated relatively more significant prediction errors. An in-depth analysis was conducted to evaluate the accuracy and predictive ability of the three intelligent-prediction models. The goodness-of-fit correlation coefficient ( $R^2$ ) and the mean absolute error (MAE) (see Equation (7)) were used as performance indicators, as detailed in Table 1. The TCN model consistently exhibited MAE values generally exceeding 1.0, indicating a substantial deviation between the predicted values and the actual monitoring values. Similarly, the  $R^2$  values for the TCN model were predominantly below 0.6, suggesting a lack of compatibility. For the LSTM model, MAE values surpassing 1.0 were observed at monitoring points SZ13, SZ15, and SZ16, with a significant error of 7.305 at SZ13. The corresponding  $R^2$  values were less than 0.8 at SZ06, SZ10, SZ13,

and SZ16, indicating poor fitting at these monitor points. The GRU model only exhibited a MAE value greater than 1.0 at monitoring point SZ13, reaching 1.128. Moreover, the  $R^2$  values at SZ05, SZ06, and SZ16 fell below 0.8, indicating a weaker fit at these monitoring points. Notably, the  $R^2$  value at SZ16 was a mere 0.268, suggesting an almost complete lack of fitting ability at this point.

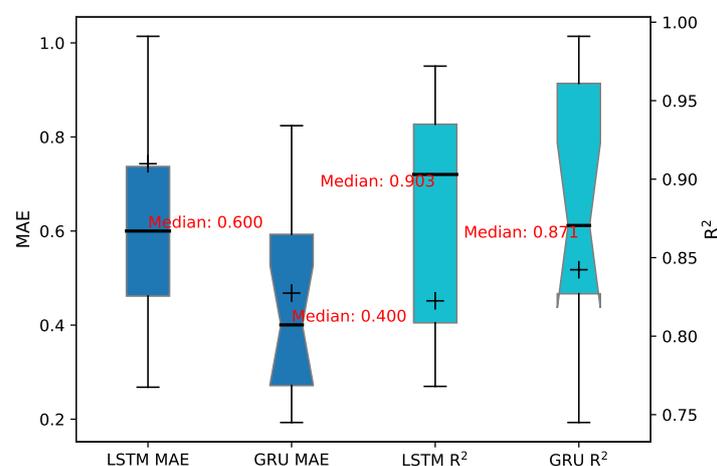
**Table 1.** MAE and  $R^2$  of prediction using LSTM, GRU, and TCN models.

	LSTM MAE	LSTM $R^2$	GRU MAE	GRU $R^2$	TCN MAE	TCN $R^2$
SZ01	0.599	0.927	0.274	0.870	1.784	0.263
SZ02	0.484	0.813	0.345	0.860	1.536	0.034
SZ03	0.601	0.944	0.593	0.961	1.294	0.612
SZ04	0.581	0.963	0.593	0.961	1.510	0.731
SZ05	0.795	0.821	0.456	0.745	1.069	0.267
SZ06	0.522	0.795	0.625	0.585	1.995	−0.934
SZ07	0.395	0.932	0.593	0.961	1.445	0.065
SZ08	0.364	0.903	0.224	0.815	0.680	0.671
SZ09	0.268	0.972	0.193	0.854	1.269	0.296
SZ10	0.718	0.768	0.304	0.871	1.266	0.243
SZ11	0.316	0.903	0.261	0.959	0.855	0.641
SZ12	0.626	0.970	0.298	0.831	3.435	0.117
SZ13	7.305	0.462	1.128	0.988	9.923	0.071
SZ14	0.656	0.916	0.515	0.957	2.072	0.350
SZ15	1.014	0.871	0.264	0.991	2.026	0.613
SZ16	1.975	−0.207	0.824	0.268	6.316	−8.734

Observing the prediction results of the LSTM and GRU models, although not all the predicted values exactly matched the real levelness, the levelness of the building-steel platforms at points SZ01, SZ03, SZ04, SZ07, SZ08, SZ09, SZ12, SZ14, and SZ15 showed a better prediction effect, with similar fluctuation patterns and peaks in the predicted and actual monitoring values. This further validates the results of the sensitivity analysis between multiple explanatory variables and multiple observed variables in Section 4.1.

Figure 11 also demonstrates that the GRU model consistently had smaller MAE values, indicating higher accuracy in predicting the HBM posture. On the other hand, the LSTM model showcased  $R^2$  values closer to 1 across all instances, indicating a better fit to the HBM posture.

However, monitoring points SZ10, SZ13, and SZ16 exhibited poor predictive performance across all three intelligent-prediction models, indicating a lack of sensitivity to jacking cylinder data and susceptibility to other influencing factors. The unsatisfactory prediction outcomes of the TCN model imply an inability to capture the significance of long-term features, requiring further enhancements such as introducing larger convolution kernels to capture the long-term trends of HBM postures.



**Figure 11.** MAE and  $R^2$  under LSTM and GRU model prediction.

## 5. Discussion

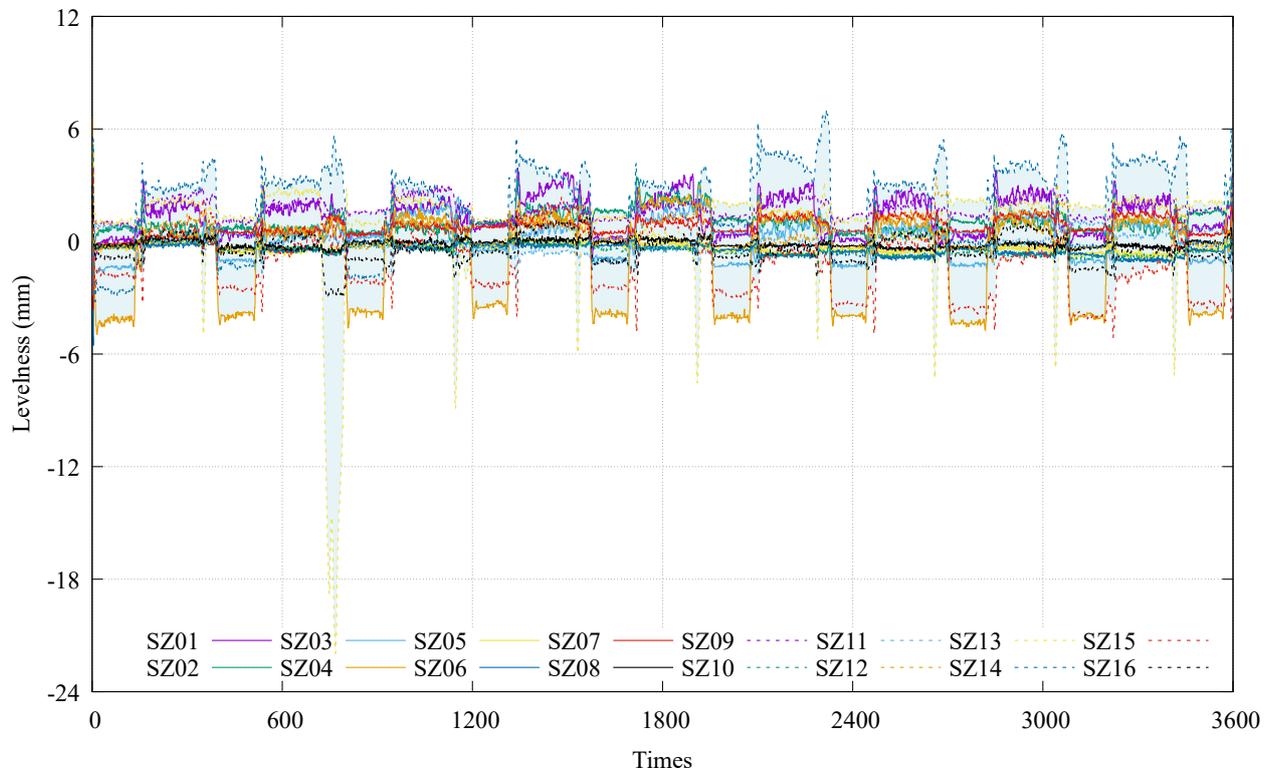
In this research, we focused on utilizing various neural network models, including LSTM, GRU, and TCN, to predict changes in the postures of the HBM based on historical data on the levelness of SPs, stroke, and the pressure of climbing jacking cylinders. We employed a multivariate time series-related model approach to address the challenge of multiple output prediction. We conducted a thorough analysis by comparing and evaluating the prediction outcomes of these time series neural network methods. We selected the optimal solution based on calculating the mean square error, which effectively reduced model randomness and improved the correlation of levelness at each monitoring point of the HBM. By training the multivariate time series neural network, we successfully obtained a model capable of predicting the postures of the HBM. Leveraging the historical monitored data during the climbing process, our model enabled accurate predictions of subsequent data. Thus, the necessary adjustments were made to each of the jacking cylinders, even when the load distribution of the SP remained unknown.

During the climbing of the HBM, it is imperative to minimize the variation in the SP's postures. Drawing upon the comprehensive analysis presented earlier, it becomes clear that the GRU model exhibits superior accuracy in predicting the hand postures of the HBM. Consequently, the HBM operator can leverage the intelligent-prediction model with the GRU neural network to anticipate future changes in HBM postures during the climbing process. Moreover, it enables a continuous adjustment of the HBM's postures in response to the predicted outcomes. Figure 12 illustrates the temporal data obtained from 16 levelness sensors during the climbing of the HBM. As the HBM climbs, the readings from the 16 sensors consistently demonstrate changes in the measured levelness. Furthermore, the levelness deviations across the 16 positions on the SP undergo dynamic changes in conjunction with the climbing process.

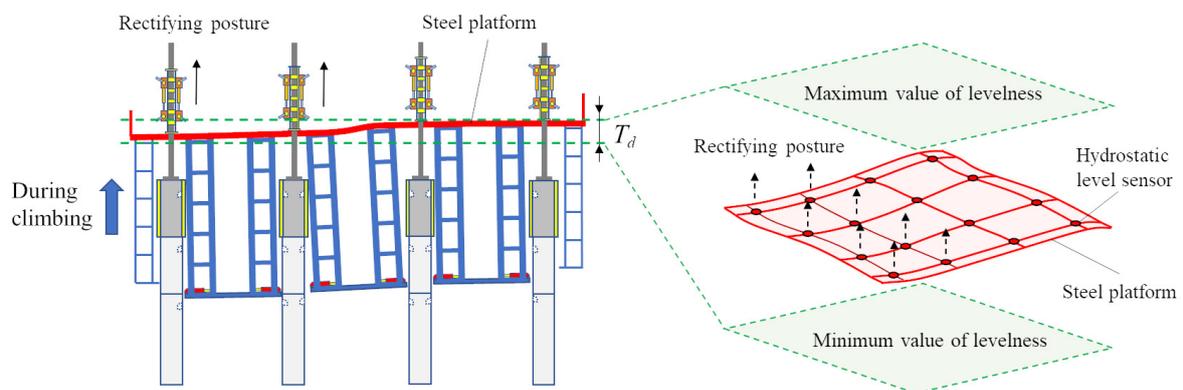
During the intricate process of climbing the HBM, the stroke of the jacking cylinder directly controls the HBM's posture while significantly impacting its overall posture dynamics. So, it is clearly suboptimal to rely on adjusting the stroke of the jacking cylinder to rectify the HBM's posture. In order to further leverage the proposed model to offer guidance on the HBM's climbing state, we engage in a dynamic adjustment of the jacking pressure values across the jacking cylinders of the various positions. This approach enables us to dynamically rectify the HBM's postures, thereby minimizing the levelness deviations among different positions on the SP. Figure 9b showcases the results of a Pearson correlation analysis conducted between the levelness of the SP and the pressure exerted by the jacking cylinder. Remarkably, there is always the highest correlation discernible between the fluctuations observed in the levelness at each position on the SP and the pressure values of one jacking cylinder. Consequently, when encountering instances where the levelness at a particular position is suboptimal, we adjust the corresponding jacking cylinder's pressure value, enhancing the levelness, diminishing the deviation in levelness across the SP, and rectifying the HBM's posture.

In summary, in situations where significant deviations in the HBM's postures manifest, we can minimize the levelness deviations on the SP by augmenting the pressure value of the jacking cylinder, which has a positive correlation with the low-levelness position point on the SP. This approach allows us to rectify the HBM's postures and attenuate the levelness deviations. By harnessing the predictive capabilities of the proposed model, we can anticipate the forthcoming changes in the HBM's posture. This integration of the model into the HBM's intelligent system empowers the HBM to adjust its climbing parameters autonomously, ensuring the smooth and secure operation of the HBM. Figure 13 visually illustrates the process of adjusting the hand postures of the HBM during its climbing. Throughout the actual climbing process, it is imperative to minimize the levelness deviations among each monitoring position on the SP. Furthermore, it is crucial to maintain the maximum and minimum values of levelness deviations within a threshold ( $T_d$ ) to mitigate the risk of the tube structure colliding with the reinforced concrete main structure of the building. Such collisions can lead to instability in the climbing process, posing a

potential hazard to the HBM. The proposed model not only accurately predicts the climbing postures of the HBM but also provides specific adjustment recommendations to ensure the safe operation of the HBM and mitigate construction risks.



**Figure 12.** Use of hydrostatic levelness sensors to monitor the time course of the levelness curves at different positions of the steel platform during the climbing process.



**Figure 13.** Schematic diagram of the posture adjustment during HBM climbing process.

## 6. Conclusions

In this paper, we propose a multivariate time series neural network prediction system to control the HBM and maintain a smooth posture during construction operations. The system utilizes three multivariate time series neural network models: LSTM, GRU, and TCN. The pressure and stroke of the jacking cylinder serve as inputs, while the monitored values of sensor levelness at 16 different positions on the SP are used as outputs. The main objective is to address the issue of unstable posture in the HBM caused by significant levelness deviations resulting from uneven stacking on the SP during the climbing process. By predicting future posture changes based on the climb data from the historical working stage of the HBM, we aim to proactively control the levelness deviations of the SP within a

threshold value and prevent instability in the HBM's posture during the climbing process. The results of the study are as follows:

1. For the same neural network architecture and dataset size, the prediction system that uses the GRU neural network model does a better job of guessing how the HBM posture will change while it climbs. Among the multiple levelness sensors installed on the HBM, only a subset of them demonstrate a strong correlation with the jacking parameters of the jacking mechanism. By adjusting the pressure value of the jacking cylinders, the posture of the HBM can be conveniently corrected. Therefore, we propose employing the GRU neural network-prediction system to anticipate the posture changes of the HBM. Additionally, by adjusting the jacking cylinder pressure value, it is possible to maintain levelness within the threshold value;
2. The validation of the measured data demonstrates that the proposed prediction models can accurately determine the levelness deviations and posture of the HBM's steel platform by solely utilizing the working data from the jacking cylinders. This capability allows for real-time warnings, indicating that these networks can make significant contributions to the safe and efficient operation of the HBM. Moreover, this modified method can also be extended to monitor the operational status of other engineering equipment, such as hydraulic climbing molds, sliding molds, and integral lifting scaffolds. The widespread adoption and implementation of this method could improve the construction levelness of high-rise buildings;
3. However, it is important to note that the model was trained and tested solely based on data from a single HBM, and its applicability to other HBMs or construction platforms has yet to be verified. Additionally, the model primarily considers the pressure and stroke of the jacking cylinder as inputs without accounting for the potential influence of other environmental factors, such as weather conditions. Finally, while the GRU provided the best prediction in this study, it should not be assumed that the GRU is the optimal choice in all scenarios. Therefore, future studies should improve the preprocessing and cleaning of the data and validate the generalization ability of these models. More characteristic factors should be considered under a wider range of equipment and conditions for different types of HBM operational data. This approach will lead to more comprehensive and high-performance predictive models;
4. The developed models offer real-time predictions to site managers and operators, allowing them to understand the HBM's status during the climbing process promptly. This timely understanding enables them to make the necessary adjustments in accordance with HBM management requirements and standard specification terms, thus ensuring a safer and more efficient climbing process.

**Author Contributions:** Conceptualization, X.P. and J.H.; methodology, J.H.; formal analysis, J.H.; resources, X.P. and L.Z.; data curation, L.Z.; writing—original draft preparation, X.P. and J.H.; writing—review and editing, Y.Z. and Z.Z.; visualization, L.Z.; supervision, Z.Z.; design of the experimental procedure, J.H.; project administration, J.H.; funding acquisition, X.P. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the National Key R&D Program of China (Grant No. 2022YFC3802200) and the Shanghai Enterprise Innovation Development and Capacity Enhancement Project (Grant No. 2022007).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The data presented in this study are available upon request from the corresponding author. The data are private since the data would reveal the operation of the integral climbing construction platform.

**Acknowledgments:** At this point, we would like to thank the editors and the reviewers for their valuable time in reviewing this study. Any suggestions for improvement you put forward will play an important role in promoting the improvement and perfection of this paper. Thanks again, from all the authors.

**Conflicts of Interest:** Author Xi Pan, Zibo Zuo, Longlong Zhang were employed by the company Shanghai Construction Group Co., Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

### Abbreviations

The following abbreviations are used in this manuscript:

HBM	High-rise building machine
SP	Steel platform
NNs	Neural networks
RNN	Recurrent neural network
LSTM	Long short-term memory
GRU	Gated recurrent unit
TCN	Temporal convolutional neural network
MTS	Multivariate time series
MAE	Mean absolute error
RMSE	Root mean square error
R <sup>2</sup>	R-squared coefficient

### References

1. Wakisaka, T.; Furuya, N.; Inoue, Y.; Shiokawa, T. Automated construction system for high-rise reinforced concrete buildings. *Autom. Constr.* **2000**, *9*, 229–250. [\[CrossRef\]](#)
2. Ikeda, Y.; Harada, T. Application of the automated building construction system using the conventional construction method together. In Proceedings of the 23rd International Symposium on Automation and Robotics in Construction, Tokyo, Japan, 3–5 October 2006; pp. 722–727. [\[CrossRef\]](#)
3. Bock, T.; Linner, T. *Site Automation: Automated/Robotic On-Site Factories*, 1st ed.; Cambridge University Press: Cambridge, MA, USA, 2016.
4. Zayed, T.; Mohamed, E. A case productivity model for automatic climbing system. *Engineering* **2014**, *21*, 33–50. [\[CrossRef\]](#)
5. Liu, X.; Hu, Y.; Chen, D.; Wang, L. Safety Control of Hydraulic Self-climbing Formwork in South Tower Construction of Taizhou Bridge. *Procedia Eng.* **2012**, *45*, 248–252. [\[CrossRef\]](#)
6. Kannan, M.R.; Santhi, M.H. Constructability Assessment of Climbing Formwork Systems Using Building Information Modeling. *Procedia Eng.* **2013**, *64*, 1129–1138. [\[CrossRef\]](#)
7. Gong, J.; Fang, T.; Zuo, J. A Review of Key Technologies Development of Super High-Rise Building Construction in China. *Adv. Civ. Eng.* **2022**, *2022*, 5438917. [\[CrossRef\]](#)
8. Pan, X.; Zuo, Z.; Zhang, L.; Zhao, T. Research on Dynamic Monitoring and Early Warning of the High-Rise Building Machine during the Climbing Stage. *Adv. Civ. Eng.* **2023**, *2023*, 9326791. [\[CrossRef\]](#)
9. Pan, X.; Zhao, T.; Li, X.; Zuo, Z.; Zong, G.; Zhang, L. Automatic Identification of the Working State of High-Rise Building Machine Based on Machine Learning. *Appl. Sci.* **2023**, *13*, 11411. [\[CrossRef\]](#)
10. Zuo, Z.; Huang, Y.; Pan, X.; Zhan, Y.; Zhang, L.; Li, X.; Zhu, M.; Zhang, L.; De Corte, W. Experimental research on remote real-time monitoring of concrete strength for highrise building machine during construction. *Measurement* **2021**, *178*, 109430. [\[CrossRef\]](#)
11. Golafshani, E.M.; Talatahari, S. Predicting the climbing rate of slip formwork systems using linear biogeography-based programming. *Appl. Soft Comput.* **2018**, *70*, 263–278. [\[CrossRef\]](#)
12. Li, S.; Yu, Z.; Meng, Z.; Han, G.; Huang, F.; Zhang, D.; Zhang, Y.; Zhu, W.; Wei, D. Study on construction technology of hydraulic climbing formwork for super high-rise building under aluminum formwork system. *Iop Conf. Ser. Earth Environ. Sci.* **2021**, *769*, 032062. [\[CrossRef\]](#)
13. Kim, M.J.; Kim, T.; Lim, H.; Cho, H.; Kang, K.I. Automated Layout Planning of Climbing Formwork System Using Genetic Algorithm. In Proceedings of the 33rd International Symposium on Automation and Robotics in Construction (ISARC), Auburn, AL, USA, 18–21 July 2016; pp. 770–777. [\[CrossRef\]](#)
14. Yao, G.; Guo, H.; Yang, Y.; Xiang, C.; Soltys, R. Dynamic Characteristics and Time-History Analysis of Hydraulic Climbing Formwork for Seismic Motions. *Adv. Civ. Eng.* **2021**, *2021*, 2139153. [\[CrossRef\]](#)
15. Dong, J.; Liu, H.; Lei, M.; Fang, Z.; Guo, L. Safety and stability analysis of variable cross-section disc-buckle type steel pipe high support system. *Int. J. Press. Vessel. Pip.* **2022**, *200*, 104831. [\[CrossRef\]](#)

16. Chandransu, T.; Rasmussen, K. Investigation of geometric imperfections and joint stiffness of support scaffold systems. *J. Constr. Steel Res.* **2011**, *67*, 576–584. [[CrossRef](#)]
17. Shen, Y.; Xu, M.; Lin, Y.; Cui, C.; Shi, X.; Liu, Y. Safety Risk Management of Prefabricated Building Construction Based on Ontology Technology in the BIM Environment. *Buildings* **2022**, *12*, 765. [[CrossRef](#)]
18. Pham, C.P.; Nguyen, P.T.; Phan, P.T.; Nguyen, Q.L.; Le, L.P.; Duong, M.T. Risk Factors Affecting Equipment Management in Construction Firms. *J. Asian Financ. Econ. Bus.* **2020**, *7*, 347–356. [[CrossRef](#)]
19. He, Z.; Gao, M.; Liang, T.; Lu, Y.; Lai, X.; Pan, F. Tornado-affected safety assessment of tower cranes outer-attached to super high-rise buildings in construction. *J. Build. Eng.* **2022**, *51*, 104320. [[CrossRef](#)]
20. Jiang, L.; Zhao, T.; Zhang, W.; Hu, J. System Hazard Analysis of Tower Crane in Different Phases on Construction Site. *Adv. Civ. Eng.* **2021**, *2021*, 1–16. [[CrossRef](#)]
21. Audibert, J.; Michiardi, P.; Guyard, F.; Marti, S.; Zuluaga, M.A. Do deep neural networks contribute to multivariate time series anomaly detection? *Pattern Recognit.* **2022**, *132*, 108945. [[CrossRef](#)]
22. Park, D.; Hoshi, Y.; Kemp, C.C. A Multimodal Anomaly Detector for Robot-Assisted Feeding Using an LSTM-Based Variational Autoencoder. *IEEE Robot. Autom. Lett.* **2018**, *3*, 1544–1551. [[CrossRef](#)]
23. Li, G.; Jung, J.J. Deep learning for anomaly detection in multivariate time series: Approaches, applications, and challenges. *Inf. Fusion* **2023**, *91*, 93–102. [[CrossRef](#)]
24. Nguyen, H.; Tran, K.; Thomassey, S.; Hamad, M. Forecasting and Anomaly Detection approaches using LSTM and LSTM Autoencoder techniques with the applications in supply chain management. *Int. J. Inf. Manag.* **2021**, *57*, 102282. [[CrossRef](#)]
25. Qin, H.; Yan, M.; Ji, H. Application of Controller Area Network (CAN) bus anomaly detection based on time series prediction. *Veh. Commun.* **2021**, *27*, 100291. [[CrossRef](#)]
26. Chen, H.; Liu, H.; Chu, X.; Liu, Q.; Xue, D. Anomaly detection and critical SCADA parameters identification for wind turbines based on LSTM-AE neural network. *Renew. Energy* **2021**, *172*, 829–840. [[CrossRef](#)]
27. Ranjan, K.; Tripathy, D.; Prusty, B.R.; Jena, D. An improved sliding window prediction-based outlier detection and correction for volatile time-series. *Int. J. Numer. Model. Electron. Netw. Devices Fields* **2020**, *34*, e2816. [[CrossRef](#)]
28. Chen, Z.; Chen, D.; Zhang, X.; Yuan, Z.; Cheng, X. Learning Graph Structures With Transformer for Multivariate Time-Series Anomaly Detection in IoT. *IEEE Internet Things J.* **2022**, *9*, 9179–9189. [[CrossRef](#)]
29. Kapoor, B.; Nagpal, B.; Jain, P.K.; Abraham, A.; Gabralla, L.A. Epileptic Seizure Prediction Based on Hybrid Seek Optimization Tuned Ensemble Classifier Using EEG Signals. *Sensors* **2023**, *23*, 423. [[CrossRef](#)]
30. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the 31st Annual Conference on Neural Information Processing Systems (NIPS), Long Beach, CA, USA, 4–9 December 2017.
31. Schmidhuber, J. Deep learning in neural networks: An overview. *Neural Netw.* **2015**, *61*, 85–117. [[CrossRef](#)]
32. Liu, W.; Wang, Z.; Liu, X.; Zeng, N.; Liu, Y.; Alsaadi, F. A survey of deep neural network architectures and their applications. *Neurocomputing* **2016**, *234*. [[CrossRef](#)]
33. Wu, Y.c.; Feng, J.w. Development and Application of Artificial Neural Network. *Wirel. Pers. Commun.* **2018**, *102*. [[CrossRef](#)]
34. Choi, K.; Yi, J.; Park, C.; Yoon, S. Deep Learning for Anomaly Detection in Time-Series Data: Review, Analysis, and Guidelines. *IEEE Access* **2021**, *9*, 120043–120065. [[CrossRef](#)]
35. Rollo, F.; Bachechi, C.; Po, L. Anomaly Detection and Repairing for Improving Air Quality Monitoring. *Sensors* **2023**, *23*, 640. [[CrossRef](#)]
36. Ding, N.; Ma, H.; Gao, H.; Ma, Y.; Tan, G. Real-time anomaly detection based on long short-Term memory and Gaussian Mixture Model. *Comput. Electr. Eng.* **2019**, *79*, 106458. [[CrossRef](#)]
37. Greff, K.; Srivastava, R.; Koutník, J.; Steunebrink, B.; Schmidhuber, J. LSTM: A search space odyssey. *IEEE Trans. Neural Netw. Learn. Syst.* **2015**, *28*, 2222–2232. [[CrossRef](#)] [[PubMed](#)]
38. Vos, K.; Peng, Z.; Jenkins, C.; Shahriar, M.R.; Borghesani, P.; Wang, W. Vibration-based anomaly detection using LSTM/SVM approaches. *Mech. Syst. Signal Process.* **2022**, *169*, 108752. [[CrossRef](#)]
39. Cho, K.; Merriënboer, B.; Bahdanau, D.; Bengio, Y. On the Properties of Neural Machine Translation: Encoder-Decoder Approaches. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2014. [[CrossRef](#)]
40. Henry, A.; Gautam, S.; Khanna, S.; Rabie, K.; Shongwe, T.; Bhattacharya, P.; Sharma, B.; Chowdhury, S. Composition of Hybrid Deep Learning Model and Feature Optimization for Intrusion Detection System. *Sensors* **2023**, *23*, 890. [[CrossRef](#)]
41. Chang, W.; Sun, D.; Du, Q. Intelligent Sensors for POI Recommendation Model Using Deep Learning in Location-Based Social Network Big Data. *Sensors* **2023**, *23*, 850. [[CrossRef](#)]
42. Yang, B.; Lu, P.; Du, C.; Cao, F. A GRU network framework towards fault-tolerant control for flight vehicles based on a gain-scheduled approach. *Aerosp. Sci. Technol.* **2024**, *146*, 108954. [[CrossRef](#)]
43. Bai, S.; Kolter, J.Z.; Koltun, V. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv* **2018**, arXiv:1803.01271.
44. Yin, C.; Zhang, S.; Wang, J.; Xiong, N.N. Anomaly Detection Based on Convolutional Recurrent Autoencoder for IoT Time Series. *IEEE Trans. Syst. Man. Cybern. Syst.* **2022**, *52*, 112–122. [[CrossRef](#)]

45. Wang, S.; Li, B.; Li, G.; Yao, B.; Wu, J. Short-term wind power prediction based on multidimensional data cleaning and feature reconfiguration. *Appl. Energy* **2021**, *292*, 116851. [[CrossRef](#)]
46. Mohan, A.; Venkatesan, M.; Prabhavathy, P.; Jayakrishnan, A. Temporal convolutional network based rice crop yield prediction using multispectral satellite data. *Infrared Phys. Technol.* **2023**, *135*, 104960. [[CrossRef](#)]
47. Hochreiter, S.; Schmidhuber, J. Long Short-term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [[CrossRef](#)] [[PubMed](#)]

**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.