

Research Article

Analysis of Diaphragm Wall Deflection Induced by Excavation Based on Machine Learning

Huajing Zhao ¹, Wei Liu ¹, Hao Guan ¹ and Chunqing Fu ²

¹School of Urban Rail Transportation, Soochow University, Suzhou, Jiangsu 215131, China

²Beijing Uni.-Construction Group Co., Ltd., Beijing 100100, China

Correspondence should be addressed to Wei Liu; ggoulmmeng@suda.edu.cn

Received 2 December 2020; Revised 25 January 2021; Accepted 3 February 2021; Published 11 February 2021

Academic Editor: Wangcheng Zhang

Copyright © 2021 Huajing Zhao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

For the concrete diaphragm wall (CDW) supported excavation, excessive wall deflection may pose a potential risk to adjacent structures and utilities in urban areas. Therefore, it is of significance to predict the CDW deformation with high accuracy and efficiency. This paper investigates three machine learning algorithms, namely, back-propagation neural network (BPNN), long short-term memory (LSTM), and gated recurrent unit (GRU), to predict the excavation-induced CDW deflection. A database of field measurement collected from an excavation project in Suzhou, China, is used to verify the proposed models. The results show that GRU exhibits lower prediction errors and better robustness in 10-fold cross validation than BPNN and executes less computational time than LSTM. Therefore, GRU is the most suitable algorithm for CDW deflection prediction considering both effectiveness and efficiency, and the predicted results can provide reasonable assistance for safety monitoring and early warning strategies conducted on the construction site.

1. Introduction

In recent years, there has been a rapid development of metro construction in China. The metro stations are generally constructed using the cut and cover method, and concrete diaphragm wall (CDW) is one of the most widely used support techniques especially for deep excavation in saturated soils. The lateral deformation of CDW caused by excavation is a major concern to both engineers and researchers as it poses potential risks to the surrounding facilities and structures especially in urban areas [1–4]. Therefore, it is essential to implement accurate and prompt prediction on CDW deflection in engineering practice.

Traditionally, methods to predict the excavation-induced CDW deformation can be categorized into two groups: empirical formula and numerical simulation. The empirical formula based on historic projects is relatively simple in model and easy to conduct [5–8], but the predicted results usually tend to be broad and the formula is unable to represent the dynamic evolution of wall deflection. The elaborate

numerical simulation is theoretically more precise by considering the soil-structure interaction, but it is still difficult to take all the instinct and extinct factors into account, and a disparity of estimated results and field measurement frequently occurs [9, 10].

Alternatively, soft computing technique such as machine learning (ML) is fast becoming a widely accepted method for predictive models in geotechnical application, which is capable of capturing nonlinear characteristics with high dimensions and has demonstrated superior predictive ability [11]. Artificial neural network (ANN) is one of the most prevailing ML algorithms used in geotechnical engineering. Goh et al. [12] presented a back-propagation neural network (BPNN) model to provide initial estimates of maximum wall deflections for braced excavations in soft clay. Kung et al. [13] demonstrated that the wall deflection can be accurately predicted by ANN, using hypothetical cases from finite element simulation for model training and 12 excavation case histories in Taipei for model validation. Zhang et al. [14] employed ensemble learning methods (ELMs), including the eXtreme Gradient Boosting (XGBoost), and Random Forest

Regression (RFR) to predict the maximum lateral wall deformation.

Deep learning (DL) algorithms, considered as a subset or an evolution of ML, have deeper structure and can learn much more complex nonlinear features than conventional neural networks. For example, long short-term memory (LSTM) has achieved good practical application results in the dynamic and deep processing of massive, long-term, dependent data series [15, 16]. Qu et al. [17] established the concrete dam deformation prediction model based on LSTM. Li et al. [18] developed an LSTM model to predict the TBM performances including the total thrust and the cutter-head torque in a real-time manner. In recent years, gated recurrent unit (GRU) has been successfully applied to spatial-temporal data and has been quite popular in many fields. Khan et al. [19] adopted a GRU-based deep learning approach to predict hourly traffic volume. Li et al. [20] proposed a prediction model utilizing GRU algorithm for the electricity generation. As for the evaluation of retaining structure behaviors, few studies have considered the data-driven models using DL algorithms.

This paper proposes a dynamic prediction model for CDW deflection based on data-mining algorithms. The applicability and generality of three algorithms BPNN, LSTM, and GRU were studied and compared, among which BPNN represents the classic ML algorithm applicable for predictive problem, and LSTM and GRU represent more advanced DL algorithms specialized in sequential data. Accordingly, the algorithm with best performance is recommended as a useful solution for predicting excavation-induced CDW deflections, and the predicted results can be an early alert for field engineers. The remaining part of the paper proceeds as follows: in Section 2, BPNN, LSTM, and GRU prediction models are presented as well as the performance evaluation indicators. In Section 3, a real-life excavation project is adopted to testify the applicability of the proposed prediction models. In Section 4, the prediction results of three prediction models are compared and discussed. Finally, Section 5 summarized the conclusions obtained from the study with the highlights of major findings.

2. Methodology

2.1. Machine Learning Algorithm

2.1.1. BPNN. BPNN is a classic feedforward neural network consisting of an input layer, one or several hidden layers, and an output layer [21], as shown in Figure 1. Neurons in the input layer have the function of receiving and transmitting data. Hidden layer and output layer are composed of M-P neurons with activation functions.

The mathematical relationships of M-P neurons in hidden layer and the output layer are expressed by the two following equations, respectively:

$$b_h = f \left(\sum_{i=1}^d v_{ih} \cdot x_i + \gamma_h \right), \quad (1)$$

$$y_j = f \left(\sum_{h=1}^q w_{hj} \cdot b_h + \theta_j \right), \quad (2)$$

where b_h is the result of h^{th} hidden neuron, x_i is the i^{th} input value out of d inputs, y_j is the j^{th} output value, q is the total number of hidden neurons, f is the activation function of sigmoid, v_{hi} and w_{hi} are the weight terms, and γ_h and θ_j are the bias terms.

2.1.2. LSTM. LSTM is a deep neural network designed for data with sequence characteristics [22]. As shown in Figure 2, there are two states in an LSTM neuron: cell state $C^{(t)}$ containing long-term memory and hidden state $h^{(t)}$ containing short-term memory, where new information is selectively recorded using three “gate” modules. The “gate” modules effectively solve the gradient vanishing problem in long-term series. The gate signal ranges from 0 to 1, where the value 0 means abandoning all the input data and the value 1 indicates that all the new information is highly relevant and should be memorized.

The “forget gate” $f^{(t)}$ is used to calculate the forget ratio of the cell state at time t , expressed as

$$f^{(t)} = \sigma \left(W_f \left[h^{(t-1)}, x^{(t)} \right] + b_f \right). \quad (3)$$

The “input gate” $i^{(t)}$ determines the proportion of new information to be added to the cell state, expressed as

$$i^{(t)} = \sigma \left(W_i \cdot \left[h^{(t-1)}, x^{(t)} \right] + b_i \right), \quad (4)$$

$$\tilde{C}^{(t)} = \tanh \left(W_C \cdot \left[h^{(t-1)}, x^{(t)} \right] + b_C \right), \quad (5)$$

$$C^{(t)} = f^{(t)} \cdot C^{(t-1)} + i^{(t)} \tilde{C}^{(t)}. \quad (6)$$

The “output gate” $o^{(t)}$ calculates the output and updates the hidden state, expressed as

$$y^{(t)} = \sigma \left(W_o \cdot \left[h^{(t-1)}, x^{(t)} \right] + b_o \right), \quad (7)$$

$$h^{(t)} = o_t \cdot \tanh \left(C^t \right), \quad (8)$$

where $\tilde{C}^{(t)}$ is the candidate cell state of LSTM neuron at time t ; W and b are weight and bias terms; σ is the activation function of sigmoid; \tanh is the activation function of hyperbolic tangent.

2.1.3. GRU. GRU was proposed as a modification for LSTM by Cho et al. [23] in 2014, which was initially used for language models. Figure 3 is the typical structure of GRU neuron, where the “update” gate is designed to combine the functions of “forget” and “input” gates in the LSTM neuron.

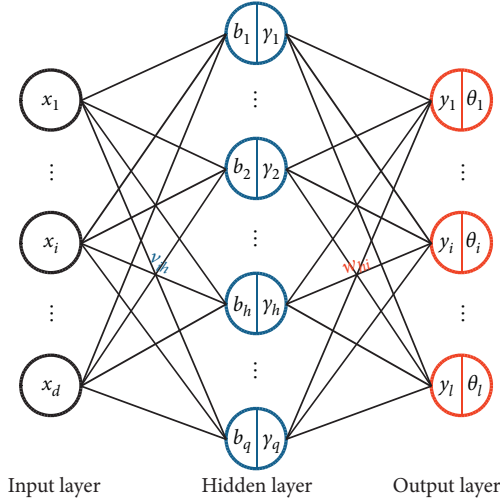


FIGURE 1: Schematic view of BPNN.

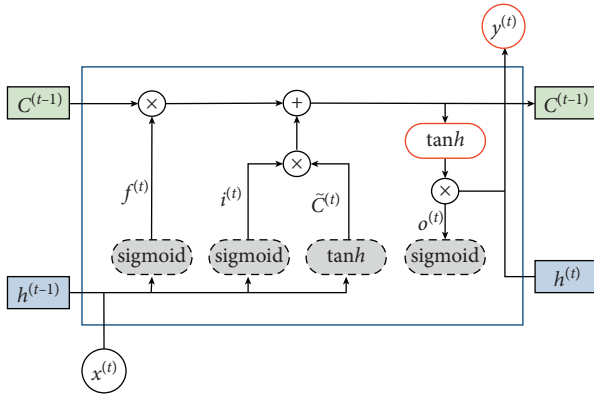


FIGURE 2: Schematic view of LSTM neuron.

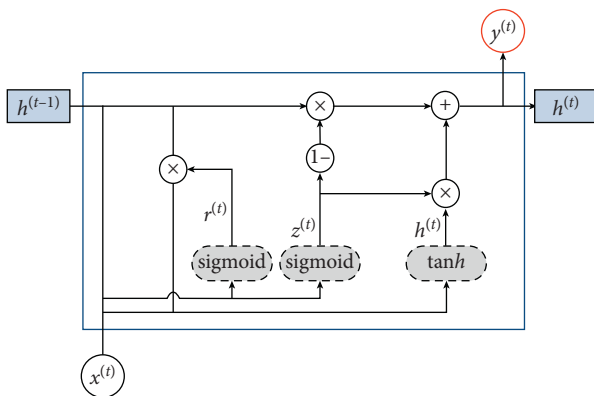


FIGURE 3: Schematic view of GRU neuron.

The GRU network structure is simplified and less parameters need to be trained; therefore, GRU can achieve high prediction accuracy but lower computational cost [24].

The first module is termed as “reset gate” $r^{(t)}$, which determines the proportion of last hidden state to be added to the new hidden state, governed by equation (3):

$$r^{(t)} = \sigma(W_r \cdot [h^{(t-1)}, x^{(t)}] + b_r), \quad (9)$$

where the superscript t denotes the time sequence, $x^{(t)}$ is the input at time t , $h^{(t-1)}$ is the hidden state of GRU neuron at time $t - 1$, W is the weight term, and σ is the activation function of sigmoid.

The second module is called “update gate” $z^{(t)}$, which is used to calculate the memorize ratio of the new input, expressed as equation (4):

$$z^{(t)} = \sigma(W_z \cdot [h^{(t-1)}, x^{(t)}] + b_z). \quad (10)$$

Then, the output of the GRU neuron is obtained via equations (5)–(7).

$$\begin{aligned} \tilde{h}^{(t)} &= \tanh(W_{\tilde{h}} \cdot [r^{(t-1)}, x^{(t)}]), \\ h^{(t)} &= (1 - z^{(t)}) \cdot h^{(t-1)} + z^{(t)} \tilde{h}^{(t)}, \\ y^{(t)} &= \sigma(W_o \cdot h^{(t)} + b_o), \end{aligned} \quad (11)$$

where $\tilde{h}^{(t)}$ is the candidate hidden state.

2.1.4. Performance Analysis. Prediction error inevitably exists between output value y_i and real value \hat{y}_i for each training sample. The prediction error can be evaluated through loss functions, so as to update the parameters of neural network and assess the accuracy of prediction models. Three commonly used loss functions are summarized in Table 1.

2.2. Development of Prediction Models

2.2.1. Inputs and Outputs. Lateral displacements of CDW were measured by inclinometers embedded in the wall. The observation value collected from each inclinometer is denoted as x_i^t , indicating the deformation value of measuring point i on day t . In the process of construction, the inclinometers might be covered or interrupted, leading to the discontinuity of the recorded time series. In the data preprocess stage, the missing value can be filled by linear interpolation or other data augmentation methods.

The input layer and output layer of prediction model are listed in Table 2. The input information length N and output prediction step M will directly affect the training speed and prediction accuracy. Generally, the richer the input information, the higher the prediction accuracy and the longer the training time. However, the continuity of measured data might be restricted by the actual conditions of construction site. Long-term prediction can provide plenty of time for precaution and implementation of deformation control measures, but whether the prediction accuracy is acceptable needs to be discussed. Therefore, four prediction tasks are designed to verify the dynamic prediction ability of BPNN, LSTM, and GRU models for CDW deformation prediction. These tasks differ in prediction time spans, that is, short-term and long-term predictions, and in input information, that is, prediction

TABLE 1: Performance indicators for evaluation on prediction error.

Type of loss function	Calculation formula
Mean squared error	$MSE = 1/n \sum_{i=1}^n (y_i - \hat{y}_i)^2$
Mean absolute error	$MAE = 1/n \sum_{i=1}^n y_i - \hat{y}_i $
Mean absolute percentage error	$MAPE\% = 1/n \sum_{i=1}^n y_i - \hat{y}_i / \hat{y}_i$

TABLE 2: Input and output layers for the prediction models.

No.	Input layer	Output layer
1	$x_1^i, x_2^i, \dots, x_{N-1}^i, x_N^i$	x_{N+M}^i
2	$x_2^i, x_3^i, \dots, x_N^i, x_{N+1}^i$	x_{N+1+M}^i
...
$T-N+1$	$x_{t-N+1}^i, x_{t-N+2}^i, \dots, x_{t-1}^i, x_t^i$	x_{t+M}^i

using abundant information and prediction using limited information. The four tasks are listed in detail as follows:

- Task 1: 1-day deformation prediction using last 3-day monitored data
- Task 2: 7-day deformation prediction using last 3-day monitored data
- Task 3: 1-day deformation prediction using last 15-day monitored data
- Task 4: 7-day deformation prediction using last 15-day monitored data

2.2.2. Optimization. Training epoch, learning rate, and hidden neurons size were the three hyperparameters manually set before training. Training with small epochs may lead to insufficient learning and incomplete data characteristics extraction, termed as “under fitting,” but large epochs will lead to “overfitting,” where prediction performance on validation data is poor even when the prediction accuracy on training data has been improved to a great extent; small learning rate is time-consuming, but with large learning rate gradient may drop too fast and miss the best convergence point; less hidden neurons may limit the learning ability of the prediction model, whereas a large number of hidden neurons may lead to computational inefficiency.

The parameters of weight and bias in prediction models are obtained through autonomous learning of training samples. Since MAE loss and MAPE loss are not smooth at the error close to 0, MSE loss is the most commonly used loss function in training process. For BPNN, stochastic gradient descent (SGD) is applied to minimize the MSE loss of training samples and update the parameters. As for LSTM and GRU, deep network architecture will lead to difficulties in parameter optimization; therefore, adaptive moment estimation (ADAM) is employed. Compared with SGD algorithm, ADAM algorithm sets adaptive learning rate for different parameters and obtains more opportunities to reach the global optimization point by considering an additional gradient moment.

2.2.3. Generalization. The prediction model usually shows an excellent regression level on training sets, but it is more practically significant to achieve sound performance on datasets outside the training samples. This paper adopts K-fold cross validation (K-CV) method to evaluate the generalization ability of the prediction model. K-CV is a data partition technology, where the original database is randomly divided into K subdatasets, and K-1 subdatasets are used as training set with the remaining dataset used for testing [25]. The process is repeated K times so as to ensure that each sample is both trained and tested. Therefore, the randomness of database division can be eliminated, and the distribution characteristics of the original database are preserved to the maximum degree. The prediction performance of each model is determined by the average MSE of K validation sets, and the model with the lowest MSE is chosen to be the most appropriate one for CDW deformation prediction.

2.2.4. Model Development. Python (version 3.7.6) and PyTorch machine learning library are used to program CDW deformation prediction models. Figure 4 illustrates the prediction process: Firstly, preprocess the database and divide the database into training sets and testing sets based on 10-fold CV; secondly, apply BPNN, LSTM, and GRU to learn the training samples, and obtain the optimal network parameters; finally, compare the performances of BPNN, LSTM, and GRU using evaluation index, and find the most suitable prediction model accordingly.

3. Data Sources

3.1. Project Overview. Figure 5 shows the excavation layout of subway station in Suzhou metro line 5, China. The excavation area is divided into Area I and Area II by installation of 1 m thick CDWs. The layout of Area I is irregular and its length is 87 m. The width is 27 m at west and increases to 36 m at east. The layout of Area II is virtually rectangular with the width of 24 m and length of 104 m. As shown in Figure 5, the inclinometers were installed along the periphery of the excavation zone. There were 9 inclinometers installed in the CDWs with a spacing of roughly 20 m in Area I, while 14 inclinometers were installed in the CDWs in Area II, and the monitoring spacing was condensed to 11 m in the north of Area II where buildings are adjacent to the CDWs with a minimum distance of 1.7 m leading to an extremely sensitive construction environment.

Figure 6 shows the typical cross-sectional profiles of excavation Area I and Area II. In Figure 6(a), the subsurface soils in Area I consist of 3.3 m thick ① miscellaneous fill, 7.8 m ② silty clay, 11.2 m ③ silty sand, 8.5 m thick ④ silty clay, and 4.9 m thick ⑤ silty sand underlain with ⑥ silty clay. The groundwater table is at 2.0 m below the ground surface. The excavation depth is 24.2 m and the bottom is in ④. The average length of CDWs is about 49 m and the

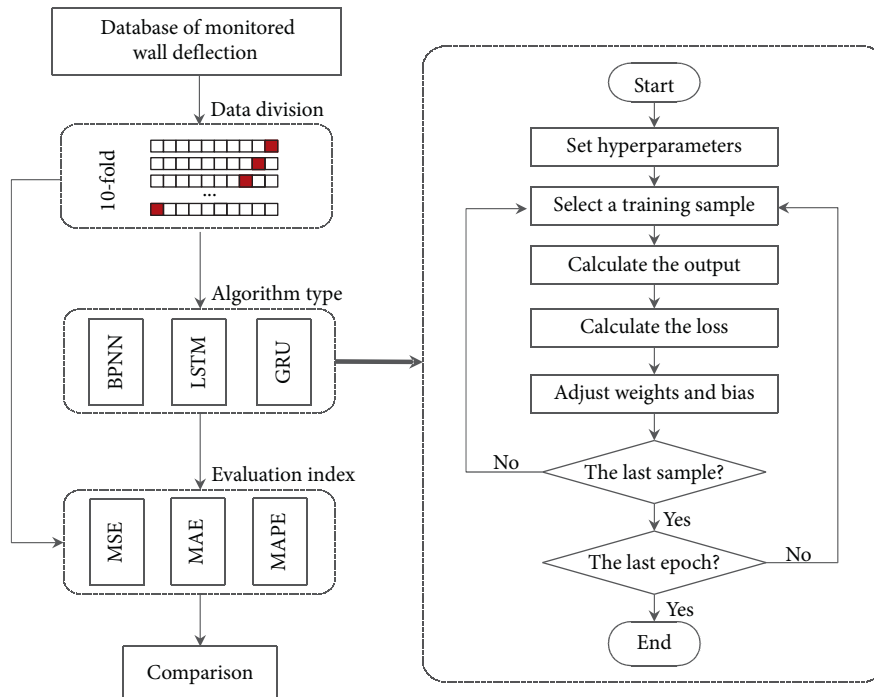


FIGURE 4: Workflow of prediction models.

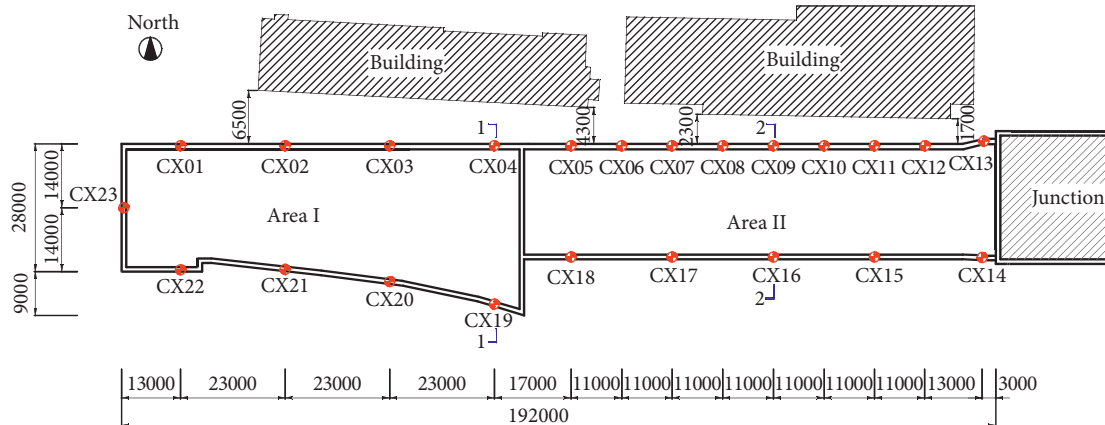


FIGURE 5: Plane view and instrumentation layout of the excavation project (unit: mm).

embedding ratio is roughly 1.0. In Figure 6(b), the similar geological conditions for excavation in Area II are found. The excavation Area II has a depth of 24 m and is supported by 45 m long CDWs with an embedding ratio of 0.875. The parameters of the subsurface strata are listed in Table 3 as provided by the geotechnical data report (GDR) of the project. During the excavation, six levels of struts containing three concrete struts and three steel struts were installed. The procedure of excavation is detailed in Table 4.

3.2. *In Situ Monitoring.* The monitoring report was recorded daily from August 8th, 2018, to July 9th, 2019 (a number of 336 pieces of monitored data for each inclinometer). The size of the CDW deformation database after missing data imputation is 7728. Figure 7 plots the time curves of maximum lateral displacement of CDW during

excavation. This plot shows that the development of wall deflection at each inclinometer exhibits a similar pattern: in the initial stage of construction, due to small excavation depth and support of the first reinforced concrete strut, the deflection is relatively small; as the excavation proceeds, the deflections continuously increase, and substantial deflection occurs in the exposure time without supports, accounting for more than 60% of the total deformation; after pouring the bottom slab, the wall deflection is stabilized as a result of the synergistic effect of the whole supporting system.

4. Results

4.1. *10-Fold CV.* The influence of data input size N on prediction accuracy can be validated by comparing the results of task 1 and task 3 and the results of task 2 and task 4.

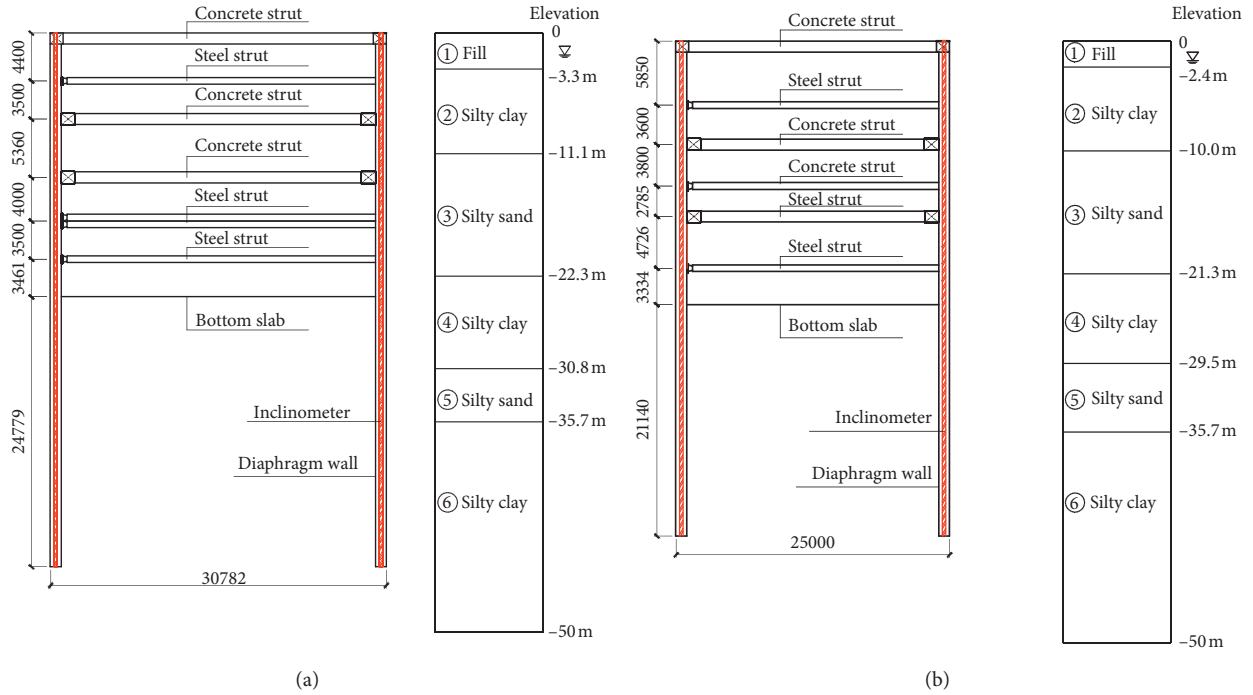


FIGURE 6: Profile views of excavation in different areas (unit: mm). (a) Cross section 1-1. (b) Cross section 2-2.

TABLE 3: Soil strata and parameters.

Soil type	γ_{sat} (kN/m ³)	SPT (<i>N</i> -value)	<i>E</i> (MPa)	<i>k</i> (cm/sec)	<i>s_u</i> (kPa)	φ' (°)
① Miscellaneous fill	18.5	3	3.5	5.0×10^{-5}	—	27
② Silty clay	18.5	9	4.1	1.4×10^{-5}	50	26
③ Silty sand	19.1	11	9.6	6.5×10^{-4}	—	30
④ Silty clay	18.4	11	3.6	1.4×10^{-5}	80	28
⑤ Silty sand	20.0	15	10.1	3.0×10^{-3}	—	32
⑥ Silty clay	19.5	18	4.0	1.4×10^{-5}	180	30

TABLE 4: Construction procedures in Area I.

Stage	Construction details	
	Area I	Area II
1	Excavate to -0.1 m Install concrete struts (800 × 1000) at -0.5 m	Excavate to -0.1 m Install concrete struts (800 × 1000) at -0.5 m
2	Excavate to -4.9 m Install steel struts (Φ609/ <i>t</i> = 16) at -4.4 m	Excavate to -6.35 m Install steel struts (Φ609/ <i>t</i> = 16) at -5.85 m
3	Excavate to -8.4 m Install concrete struts (1000 × 1000) at -7.9 m	Excavate to -9.95 m Install concrete struts (1000 × 1000) at -9.45 m
4	Excavate to -13.76 m Install concrete struts (1200 × 1000) at -13.26 m	Excavate to -13.75 m Install steel struts (Φ609/ <i>t</i> = 16 mm) at -13.25 m
5	Excavate to -17.76 m Install double steel struts (Φ609/ <i>t</i> = 16) at -17.26 m	Excavate to -16.54 m Install concrete struts (1200 × 1000) at -16.04 m
6	Excavate to -21.26 m Install steel struts (Φ609/ <i>t</i> = 16) at -20.76 m	Excavate to -21.26 m Install steel struts (Φ609/ <i>t</i> = 16 mm) at -20.76 m
7	Excavate to -25.70 m Pour bottom slab	Excavate to -25.60 m Pour bottom slab

As shown in Table 5, when the input size is fixed to 3 days, the MSEs of BPNN are the largest and fluctuate fiercely in 10 subsets, which indicates poor generalization ability of BPNN

on new data sample. In contrast, the performances of LSTM and GRU are relatively stable. When the input data increases to 15 days, for BPNN, the variation of MSEs among 10

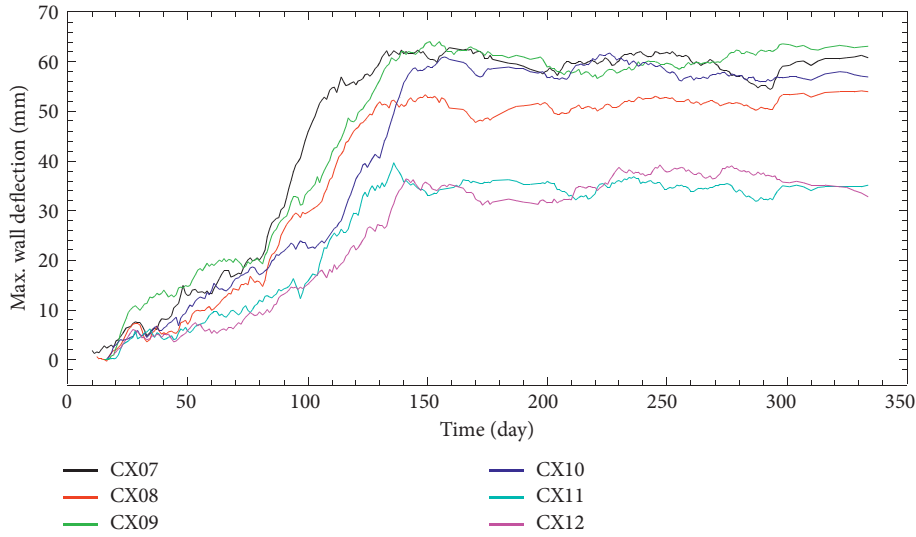


FIGURE 7: Time curves of the maximum diaphragm wall deflection.

TABLE 5: MSEs of different algorithms on 10-fold cross validation (unit: mm^2).

Subset no.	Task 1			Task 3		
	BPNN	LSTM	GRU	BPNN	LSTM	GRU
<i>Short-term prediction tasks</i>						
1	3.47	2.21	0.91	1.39	1.22	0.38
2	12.49	2.36	1.22	1.86	2.07	0.64
3	7.14	3.79	0.78	10.46	6.70	0.39
4	4.11	1.24	0.53	0.18	0.38	0.30
5	9.22	1.86	0.61	0.65	0.76	0.37
6	5.41	1.35	0.46	32.95	20.14	0.73
7	17.25	8.56	1.76	0.27	0.27	1.56
8	125.36	4.86	1.07	0.41	0.19	0.69
9	624.14	5.93	1.27	13.84	5.00	1.62
10	63.04	10.09	2.50	0.27	0.26	1.46
Avg.	87.16	4.23	1.11	6.23	3.70	0.81
Subset no.	Task 2			Task 4		
	BPNN	LSTM	GRU	BPNN	LSTM	GRU
<i>Long-term prediction tasks</i>						
1	181.38	19.47	3.50	8.29	5.51	2.21
2	44.18	14.67	6.99	6.61	4.85	2.36
3	75.09	14.38	6.94	7.78	6.34	3.79
4	44.07	4.42	2.78	5.01	5.05	1.24
5	101.18	2.68	3.53	4.67	3.44	1.86
6	7.04	41.55	2.69	8.05	18.82	1.35
7	402.82	2.23	5.55	16.89	6.60	8.56
8	185.21	3.67	9.34	17.76	1.34	4.86
9	98.77	24.87	9.24	29.57	12.02	5.93
10	194.29	4.17	6.62	23.31	3.95	10.09
Avg.	133.40	13.21	5.72	12.79	6.79	4.23

subsets tends to be smooth; for LSTM, MSEs see a drop of 12.5% and 48.6% in short-term and long-term tasks; as for GRU, the decreases are 27.0% and 26.0%, respectively. Therefore, considering larger number of historical data can improve the model stability as well as the prediction accuracy.

The performance between short-term predictions ($M=1$) and long-term predictions ($M=7$) can be demonstrated by comparing the results of task 1 and task 2 and the

results of task 3 and task 4. As shown in Table 5, when the input information length is fixed to 15 days, long-term predictions using BPNN achieve an average MSE of 12.79 mm^2 , which is the double of short-term prediction error. Average MSE of LSTM increases from 3.70 to 6.79 mm^2 , and that of GRU increases from 0.81 to 4.23 mm^2 when prediction step extends from 1 day to 7 days. Therefore, it can be inferred that the increase of prediction step has an adverse effect on prediction accuracy, and GRU

TABLE 6: Training times of different algorithms (unit: s).

Task	BPNN	LSTM	GRU
1	44.51	128.63	117.36
2	32.81	125.97	111.28
3	67.75	314.39	295.97
4	33.60	332.01	287.16

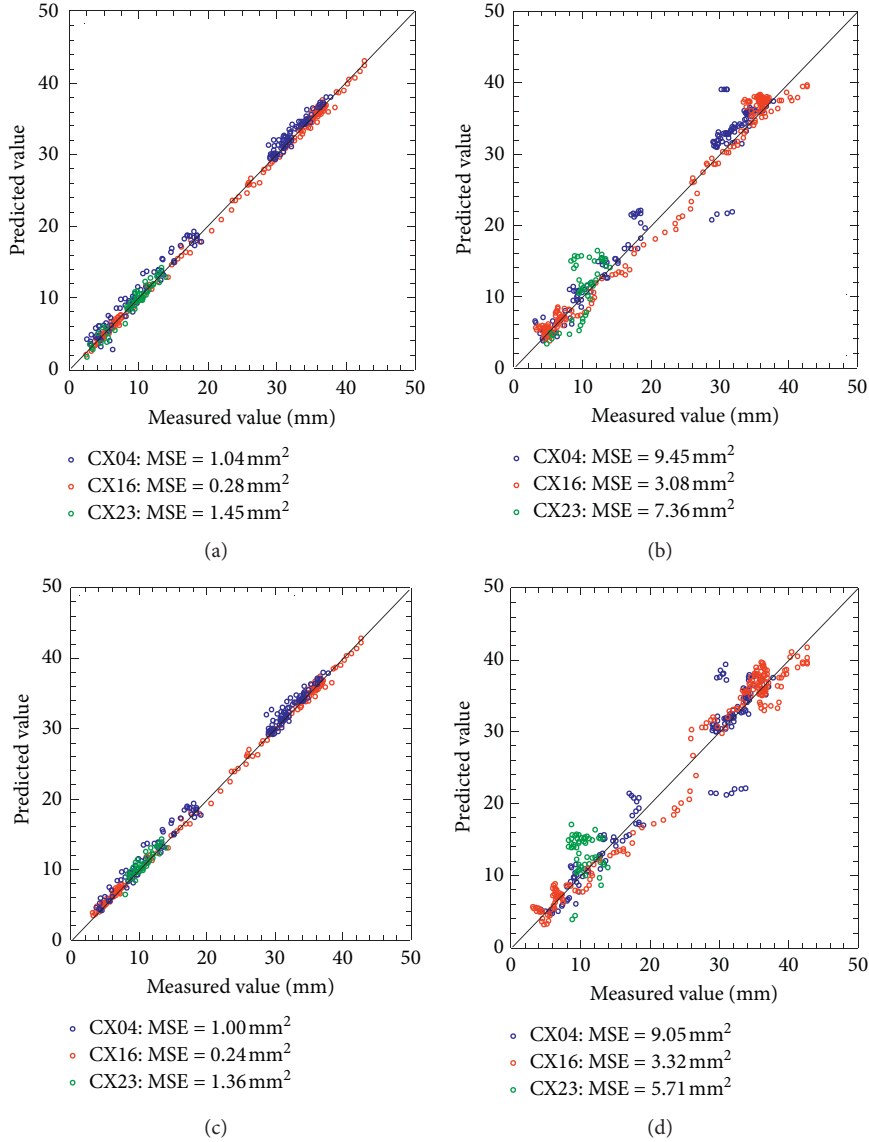


FIGURE 8: Comparisons between the predicted and the measured deflections of new monitoring points. (a) Task 1. (b) Task 2. (c) Task 3. (d) Task 4.

achieves highest precision even in long-term prediction tasks because it can extract the correlation of input data in time sequence.

Table 6 shows the training times of each algorithm on different tasks. BPNN trains the model more efficiently than the other two DL algorithms because it has a simple network structure and is relatively easy to train, but prediction accuracy and robustness resulting from BPNN are inferior. GRU trains the model slightly faster than LSTM, and the

superiority will be amplified when applied to a larger-scale database. Therefore, to achieve balance between efficiency and effectiveness, GRU is considered to be the most suitable algorithm for CDW deflection prediction.

4.2. New Monitored Points. In order to further verify the generalization ability of the deformation prediction model, the optimal model obtained from 10-fold CV is selected to

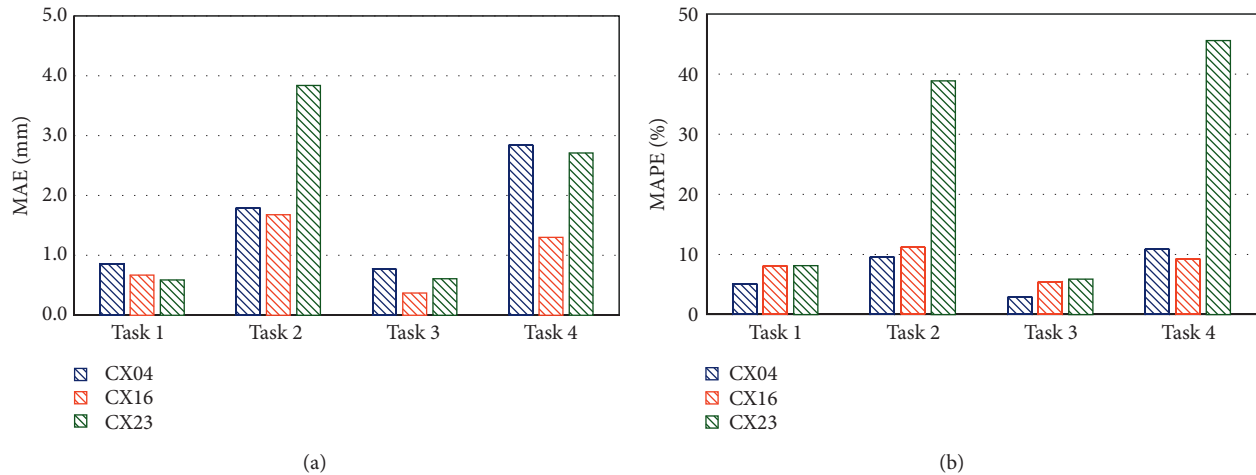


FIGURE 9: Prediction errors in new monitoring points CX04, CX16, and CX23. (a) MAE. (b) MAPE.

make prediction in three new measuring points excluded in the original database, which are CX04, CX16, and CX23, respectively. As shown in Figure 5, CX04 is noteworthy as it is at the section with large excavation width in Area I. CX16 located in the center of the southern wall in Area II and CX23 located in the center of the western wall in Area I were chosen for verification because the deflections in the middle of excavation zone are generally much larger than those near the centers.

The prediction model has an ideal performance on CX04 and CX16. It can be seen from Figures 8 and 9 that the predicted deflections fit the measurement perfectly in short-term predictions (task 1 and task 3). Long-term predictions (task 2 and task 4) display higher dispersion degree, but MAEs are under 3 mm, which remain to be acceptable. The prediction results on CX23 deviate from the measured deflections, that is, MAPE reaching 45.6% in task 4. CX23 is in the transversal wall of excavation Area I, where the geometry and support system are quite different from all the other inclination points in the longitudinal walls; besides the measured data are limited due to the low monitoring frequency at the specific point. All the above reasons lead to the poor prediction performance on CX23; thereby, it should be noted that the reliability of the trained model cannot be guaranteed without similarity and quantity of data used in training.

5. Conclusions

This study established a dynamic prediction method for excavation-induced CDW deformation based on classic algorithm BPNN and DL algorithms LSTM and GRU, which can automatically extract the temporal correlation of monitored data. Four prediction tasks are designed to verify the influence of data input size and prediction time spans. A database of in situ measurement collected from a real-life excavation project is considered to evaluate the applicability of the proposed method. The results show the following:

- (1) Considering more historical deformation data not only improves the stability of the model but also reduces the prediction error. The increase of prediction steps reduces precision, but DL algorithms maintain satisfactory performance even in long-term prediction tasks.
- (2) DL algorithms outperform BPNN in all prediction tasks with a substantial improvement in accuracy and exhibit less variation and strong robustness in 10-fold CV.
- (3) BPNN trains the model much faster but yields unusable prediction results. GRU achieves good balance between effectiveness and efficiency. Therefore, GRU is considered to be the most suitable algorithm for dynamic prediction of CDW deformation.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest regarding the publication of this paper.

Acknowledgments

This study was financially supported by the National Natural Science Foundation of China (no. 51978430).

References

- [1] W. Liu, P. Shi, G. Cai, and P. Gan, "A three-dimensional mechanism for global stability of slurry trench in frictional soils," *European Journal of Environmental and Civil Engineering*, vol. 2, pp. 1–26, 2019.

- [2] W. Liu, P. Shi, G. Cai et al., "Seepage on local stability of slurry trench in deep excavation of diaphragm wall construction," *Computers and Geotechnics*, vol. 129, Article ID 103878, 2021.
- [3] M. L. Lings, C. W. W. Ng, and D. F. T. Nash, "The lateral pressure of wet concrete in diaphragm wall panels cast under bentonite," *Proceedings of the Institution of Civil Engineers—Geotechnical Engineering*, vol. 107, no. 3, pp. 163–172, 1994.
- [4] A. Z. Mohamed, *Effect of diaphragm wall construction on adjacent deep foundation*, Ph.D. thesis, Technische Universitat Bergakademie, Freiberg, Germany, 2017.
- [5] R. B. Peck, "Deep excavations and tunneling in soft ground," in *Proceedings of the 7th International Conference on Soil Mechanics and Foundation Engineering*, pp. 225–290, Mexico City, Mexico, 1969.
- [6] G. W. Clough and T. D. O'Rourke, *Construction Induced Movements of in Situ Walls*, pp. 439–470, ASCE, Reston, VA, USA, 1990.
- [7] M. Long, "Database for retaining wall and ground movements due to deep excavations," *Journal of Geotechnical and Geoenvironmental Engineering*, vol. 127, no. 3, pp. 203–224, 2001.
- [8] C. Moormann, "Analysis of wall and ground movements due to deep excavations in soft soil based on a new worldwide database," *Soils and Foundations*, vol. 44, no. 1, pp. 87–98, 2004.
- [9] A. T. C. Goh, F. Zhang, W. Zhang, Y. Zhang, and H. Liu, "A simple estimation model for 3D braced excavation wall deflection," *Computers and Geotechnics*, vol. 83, no. 3, pp. 106–113, 2017.
- [10] R. N. Hwang, T. Y. Lee, C. R. Chou et al., "Evaluation of performance of diaphragm walls by wall deflection paths," *Journal of Geoenvironmental Engineering*, vol. 7, no. 1, pp. 001–012, 2012.
- [11] R. Ray, D. Kumar, P. Samui, L. B. Roy, A. T. C. Goh, and W. Zhang, "Application of soft computing techniques for shallow foundation reliability in geotechnical engineering," *Geoscience Frontiers*, vol. 12, no. 1, pp. 375–383, 2021.
- [12] A. T. C. Goh, K. S. Wong, and B. B. Broms, "Estimation of lateral wall movements in braced excavations using neural networks," *Canadian Geotechnical Journal*, vol. 32, no. 6, pp. 1059–1064, 1995.
- [13] G. T. C. Kung, E. C. L. Hsiao, M. Schuster, and C. H. Juang, "A neural network approach to estimating deflection of diaphragm walls caused by excavation in clays," *Computers and Geotechnics*, vol. 34, no. 5, pp. 385–396, 2007.
- [14] R. Zhang, C. Wu, A. T. C. Goh, T. Böhlke, and W. Zhang, "Estimation of diaphragm wall deflections for deep braced excavation in anisotropic clays using ensemble learning," *Geoscience Frontiers*, vol. 12, no. 1, pp. 365–373, 2021.
- [15] B. B. Yang, K. L. Yin, and J. Du, "Dynamic prediction model of landslide displacement based on time series and long and short time memory networks," *Chinese Journal of Rock Mechanics and Engineering*, vol. 37, no. 10, pp. 2334–2343, 2018.
- [16] Y. X. Liu, Q. X. Fan, Y. Z. Shang et al., "Short-term water level prediction method for hydro-power stations based on LSTM neural network," *Advances in Science and Technology of Water Resources*, vol. 39, no. 2, pp. 56–60+78, 2019.
- [17] X. Qu, J. Yang, and M. Chang, "A deep learning model for concrete dam deformation prediction based on RS-LSTM," *Journal of Sensors*, vol. 2019, pp. 1–14, 2019.
- [18] J. Li, P. Li, D. Guo, X. Li et al., "Advanced prediction of tunnel boring machine performance based on big data," *Geoscience Frontier*, vol. 12, no. 1, pp. 331–338, 2020.
- [19] Z. Khan, S. M. Khan, K. Dey et al., "Development and evaluation of recurrent neural network-based models for hourly traffic volume and annual average daily traffic prediction," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2673, no. 7, pp. 4889–503, 2019.
- [20] W. Li, T. Logenthiran, and W. L. Woo, "Multi-GRU prediction system for electricity generation's planning and operation," *IET Generation, Transmission & Distribution*, vol. 13, no. 9, pp. 1630–1637, 2019.
- [21] S. K. Das, "Artificial neural networks in geotechnical engineering: modeling and application issues," *Metaheuristics in Water, Geotechnical and Transport Engineering*, pp. 231–270, 2013.
- [22] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: continual prediction with LSTM," *Neural Computation*, vol. 12, no. 10, pp. 2451–2471, 2000.
- [23] K. Cho, B. van Merriënboer, D. Bahdanau et al., "On the properties of neural machine translation: encoder-decoder approaches," in *Proceedings of the SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, Doha, Qatar, October 2014.
- [24] R. Fu, Z. Zhang, and L. Li, "Using LSTM and GRU neural network methods for traffic flow prediction," in *Proceedings of the 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC)*, pp. 324–328, Wuhan, China, November 2016.
- [25] M. Stone, "Cross-validatory choice and assessment of statistical predictions," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 36, no. 2, pp. 111–133, 1974.