Journal of Rock Mechanics and Geotechnical Engineering 16 (2024) 5179–5192

Contents lists available at ScienceDirect

Journal of Rock Mechanics and Geotechnical Engineering

journal homepage: www.jrmge.cn

Full Length Article

A virtual calibration chamber for cone penetration test based on deep-learning approaches

Mingpeng Liu [a], Enci Sun [b], Ningning Zhang [a], Fengwen Lai [c, *], Raul Fuentes [a]

[a] Institute of Geomechanics and Underground Technology, RWTH Aachen University, Aachen, 52074, Germany
[b] School of Qilu Transportation, Shandong University, Jinan, 250061, China
[c] College of Civil Engineering, Fuzhou University, Fuzhou, 350116, China

ARTICLE INFO

Article history:
Received 10 August 2023
Received in revised form
2 October 2024
Accepted 21 October 2024
Available online 31 October 2024

Keywords:
Cone penetration test (CPT)
Virtual calibration chamber
Bayesian optimization (BO)
Multilayer perceptron (MLP)
Long short-term memory (LSTM) network

ABSTRACT

The interpretation of the cone penetration test (CPT) still relies largely on empirical correlations that have been predominantly developed in resource-intensive and time-consuming calibration chambers. This paper presents a CPT virtual calibration chamber using deep learning (DL) approaches, which allow for the consideration of depth-dependent cone resistance profiles through the implementation of two proposed strategies: (1) depth-resistance mapping using a multilayer perceptron (MLP) and (2) sequence-to-sequence training using a long short-term memory (LSTM) neural network. Two DL models are developed to predict cone resistance profiles (q_c) under various soil states and testing conditions, where Bayesian optimization (BO) is adopted to identify the optimal hyperparameters. Subsequently, the BO-MLP and BO-LSTM networks are trained using the available data from published datasets. The results show that the models with BO can effectively improve the prediction accuracy and efficiency of neural networks compared to those without BO. The two training strategies yielded comparable results in the testing set, and both can be used to reproduce the whole cone resistance profile. An extended comparison and validation of the prediction results are carried out against numerical results obtained from a coupled Eulerian-Lagrangian (CEL) model, demonstrating a high degree of agreement between the DL and CEL models. Ultimately, to demonstrate the usability of this new virtual calibration chamber, the predicted q_c is used to enhance the preceding correlations with the relative density (D_r) of the sand. The improved correlation with superior generalization has an R^2 of 82% when considering all data, and 89.6% when examining the pure experimental data.

1. Introduction

The cone penetration test (CPT) is one of the most common and popular in situ test tools for site characterization (Liyanapathirana, 2009). One notable advantage of the CPT is to rapidly obtain continuous and reproducible soil testing records (i.e. cone resistance q_c and sleeve friction f_s), with minimal disturbance compared to laboratory element testing (Lunne et al., 2002). The cone resistance profile obtained from the CPT has been widely used to interpret soil properties. However, many CPT-based interpretations of soil parameters still greatly rely on empirical correlations (Yu et al., 1996; Butlanska et al., 2014). The majority of these correlations are presented from calibration chamber tests, where the soil state and properties can be well-controlled (Chen et al., 2020). However, calibration chamber testing is resource-intensive and time-consuming. As a result, the available CPT data from calibration chamber tests remains limited, which in turn renders some of the presented empirical correlations applicable only to specific soil types and conditions (e.g. overconsolidation ratio) (Pournaghiazar et al., 2013).

To address the aforementioned scarcity of CPT data in calibration chambers, analytical modeling (e.g. cavity expansion theory (Cudmani and Osinov, 2001; Mo et al., 2017)) and numerical simulations (e.g. coupled Eulerian-Lagrangian (CEL) method (Susila and Hryciw, 2003; Wang et al., 2015) and discrete element method (DEM) (Zhang et al., 2019)) have been proposed to estimate q_c profiles for various types of soils. However, the complicated formulations lead to a gap between the analytical solutions and their practical applications. The numerical solutions are notorious

* Corresponding author.
E-mail address: laifengwen@fzu.edu.cn (F. Lai).
Peer review under responsibility of Institute of Rock and Soil Mechanics, Chinese Academy of Sciences.

https://doi.org/10.1016/j.jrmge.2024.10.004

for being challenging due to the high computational costs and the complexity of the calibration process for constitutive models (Fan et al., 2018; Yu et al., 2022), although the DEM has been previously presented as a solution for virtual calibration chambers.

Over the past decades, deep learning (DL) approaches have emerged as a compelling alternative, offering a robust capacity to leverage existing data for predictive modeling without making any prior assumptions (Zhang, 2019; Lai et al., 2023). Extensive attempts have been made towards the application of DL-based models in practice (Phoon and Zhang, 2023), e.g. prediction of TBM parameters (Gao et al., 2019), crack detection in tunnels (Huang et al., 2018; Protopapadakis et al., 2019), landslide susceptibility (Bui et al., 2020; Nhu et al., 2020), evaluation of soil strength (Yousefpour et al., 2021a; Jas and Dodagoudar, 2023), and bridge scour forecast (Yousefpour et al., 2021b; Yousefpour and Correa, 2023). However, there is little research investigating the prediction of CPT profiles, which are largely dependent on soil type and depth (overburden stress), based on DL approaches. To bridge this gap, a nonlinear mapping that can accommodate sequence loading scenarios may be more appropriate. Two potential DL options are the multilayer perceptron (MLP) and the long short-term memory (LSTM) neural networks (Guan et al., 2023). For example, Zhang et al. (2020a) and Guan and Yang (2023) demonstrated the efficacy of the LSTM model in reproducing the constitutive responses of sands under both monotonic and cyclic loading. Wang and Sun (2018) employed a multiscale framework in conjunction with an LSTM to capture the hydro-mechanical coupling effects of porous media. Guan et al. (2023) and Wu et al. (2023) employed an MLP to describe the constitutive behaviors of soils and incorporated the MLP into the finite element method. Their results demonstrated that both MLP and LSTM have great potential in predicting nonlinear-mapping datasets. Therefore, it can be hypothesized that MLP and LSTM neural networks are capable of reproducing and predicting cone resistance profiles in calibration chambers.

The selection of hyperparameters is a critical task for the construction of DL-based models. The current methods for optimizing hyperparameters, such as grid search or random search (Yousefpour and Correa, 2023), typically necessitate substantial computational resources. To address this limitation, some researchers (Yazici and Taskin, 2023) have used Bayesian optimization (BO) to identify the optimal hyperparameters for various networks. The BO can utilize previously observed historical information to update prior knowledge and rapidly identify global optimal solutions in smaller iterations with greater efficiency (Liu et al., 2024). In addition, the BO has a reduced risk of overfitting for DL models compared to some metaheuristic algorithms (Yazici and Taskin, 2023). For these reasons, this paper combines the MLP and LSTM neural networks with BO (i.e. BO-MLP and BO-LSTM) to rapidly predict the CPT profiles in calibration chamber tests and to discuss their applications in soil interpretation based on the predicted CPT data.

The objective of this paper is to develop accurate and computationally efficient virtual calibration chambers for the generation of q_c profiles of the CPT in sand through the utilization of DL approaches. First, an overview of the MLP, LSTM, and BO algorithms is provided to illustrate the fundamental principles of Bayesian-optimized neural networks. Subsequently, the BO-MLP and BO-LSTM models are developed and trained by feeding observed data, and their performance is then evaluated using the testing dataset. The developed models are further compared and validated against the solutions from the CEL method. Ultimately, an example is provided to illustrate how the developed models are used to correlate cone resistance with soil properties and to enhance the empirical equations.

2. Fundamentals of DL approaches

2.1. MLP

The MLP is one of the most popular deep neural networks for modeling and predicting complex nonlinear responses and processes (Kohestani and Hassanlourad, 2016). Fig. 1 shows a typical MLP structure consisting of an input layer, one or multiple hidden layers, and an output layer. A set of neurons are arranged in each layer and connected through weights and bias. The input data are first presented through the input layer and then pass through the hidden layers to eventually predict values in the output layer. This process can be mathematically expressed by considering a feed-forward propagation process that uses input $\boldsymbol{x}$ to estimate the output $\boldsymbol{y}$:

$$\boldsymbol{h}_1 = F(\boldsymbol{W}_1\boldsymbol{x} + \boldsymbol{b}_1) \tag{1a}$$

$$\boldsymbol{h}_2 = F(\boldsymbol{W}_2\boldsymbol{h}_1 + \boldsymbol{b}_2) \tag{1b}$$

$$\boldsymbol{y} = \boldsymbol{W}_3\boldsymbol{h}_2 + \boldsymbol{b}_3 \tag{1c}$$

Eqs. (1a) and (1b) represent two hidden layers, where $\boldsymbol{h}_1$ and $\boldsymbol{h}_2$ signify the results of the first and second hidden layers, respectively; $\boldsymbol{W}_1$, $\boldsymbol{W}_2$, and $\boldsymbol{W}_3$ represent the weight matrices of the corresponding layer; $\boldsymbol{b}_1$, $\boldsymbol{b}_2$, and $\boldsymbol{b}_3$ are the bias vectors of the corresponding layer; and F represents the activation function. The result of each neuron is included in $\boldsymbol{h}_1$ and $\boldsymbol{h}_2$, where $\boldsymbol{h}_1 = [h_{11}, h_{12}, \ldots, h_{1i}, \ldots, h_{1N_{n1}}]$, and $\boldsymbol{h}_2 = [h_{21}, h_{22}, \ldots, h_{2i}, \ldots, h_{2N_{n2}}]$, in which N_{n1} and N_{n2} are the number of neurons in the first and second hidden layers, respectively.

The most commonly used activation functions include sigmoid, hyperbolic tangent (tanh), rectified linear unit (Relu), and leaky Relu activation functions (Zhang et al., 2021). The weight matrices and the bias vectors are randomly initialized and then updated throughout the training process. The training of MLP can be considered to comprise two phases: forward calculation and backward propagation. During the forward process, the value of each hidden neuron is calculated by summing the values of input neurons multiplied by the corresponding connection weights. The error between the output and the real values can be calculated and then minimized by the backward algorithm, which updates the connection weights. Further details on MLP can be found in Dai and MacBeth (1997).

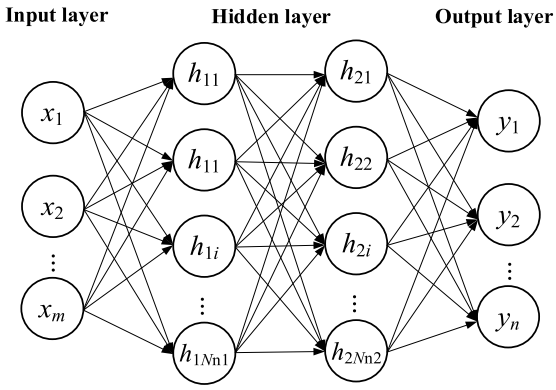

Fig. 1. Structure of a general MLP.

2.2. LSTM neural network

LSTM is a representative type of recurrent neural networks (RNNs), which has been widely used to predict sequence-based problems. The outputs of an RNN depend on the inputs of the network, including not only the current time step but also previous time steps. This presents the ability to predict future information related to previous inputs, thereby enabling its application in modeling sequential problems. By introducing a memory cell in place of the neurons, the LSTM can overcome the shortcoming of gradient vanishing or exploding in the back-propagation algorithm for conventional RNNs (Hochreiter and Schmidhuber, 1997; Guan and Yang, 2023).

The typical structure of an LSTM memory cell is illustrated in Fig. 2 (Guan and Yang, 2023). The memory cell has three gates (the forget gate $\boldsymbol{f}^t$, the input gate $\boldsymbol{i}^t$, and the output $\boldsymbol{o}^t$) that regulate the information flow. Firstly, the output at the previous time step $\boldsymbol{h}^{t-1}$ in a memory cell and the input at the current time step $\boldsymbol{x}^t$ are both used to calculate the forget gate $\boldsymbol{f}^t$, input gate $\boldsymbol{i}^t$, output gate $\boldsymbol{o}^t$, and the storage cell $\tilde{\boldsymbol{c}}^t$, as shown in Eqs. (2a)–(2d). Then, the forget gate $\boldsymbol{f}^t$ acts on the memory cell state at the previous time step $\boldsymbol{c}^t$, while the input gate $\boldsymbol{i}^t$ acts on the storage cell at the current time step $\tilde{\boldsymbol{c}}^t$, as shown in Eq. (2e). The $\boldsymbol{f}^t$ and $\boldsymbol{i}^t$ together determine whether information should be discarded or stored, and update the current memory cell state $\boldsymbol{c}^t$. Ultimately, the output $\boldsymbol{o}^t$ decides the final results of $\boldsymbol{h}^t$, as shown in Eq. (2f). The specific formulas are provided below:

$$\boldsymbol{i}^t = \sigma\left(\boldsymbol{W}_{\mathrm{i}}\boldsymbol{x}^t + \boldsymbol{U}_{\mathrm{i}}\boldsymbol{h}^{t-1} + \boldsymbol{b}_{\mathrm{i}}\right) \tag{2a}$$

$$\boldsymbol{f}^t = \sigma\left(\boldsymbol{W}_{\mathrm{f}}\boldsymbol{x}^t + \boldsymbol{U}_{\mathrm{f}}\boldsymbol{h}^{t-1} + \boldsymbol{b}_{\mathrm{f}}\right) \tag{2b}$$

$$\boldsymbol{o}^t = \sigma\left(\boldsymbol{W}_{\mathrm{o}}\boldsymbol{x}^t + \boldsymbol{U}_{\mathrm{o}}\boldsymbol{h}^{t-1} + \boldsymbol{b}_{\mathrm{o}}\right) \tag{2c}$$

$$\tilde{\boldsymbol{c}}^t = \tanh\left(\boldsymbol{W}_{\mathrm{c}}\boldsymbol{x}^t + \boldsymbol{U}_{\mathrm{c}}\boldsymbol{h}^{t-1} + \boldsymbol{b}_{\mathrm{c}}\right) \tag{2d}$$

$$\boldsymbol{c}^t = \boldsymbol{f}^t \otimes \boldsymbol{c}^{t-1} + \boldsymbol{i}^t \otimes \tilde{\boldsymbol{c}}^t \tag{2e}$$

$$\boldsymbol{h}^t = \boldsymbol{o}^t \otimes \tanh \boldsymbol{c}^t \tag{2f}$$

where $\boldsymbol{W}_{\mathrm{i}}$, $\boldsymbol{W}_{\mathrm{f}}$, $\boldsymbol{W}_{\mathrm{o}}$, and $\boldsymbol{W}_{\mathrm{c}}$ represent the weight matrices corresponding to the inputs within different gates; $\boldsymbol{U}_{\mathrm{i}}$, $\boldsymbol{U}_{\mathrm{f}}$, $\boldsymbol{U}_{\mathrm{o}}$, and $\boldsymbol{U}_{\mathrm{c}}$ represent the weight matrices corresponding to the outputs at the previous time step with different gates; $\boldsymbol{b}_{\mathrm{i}}$, $\boldsymbol{b}_{\mathrm{f}}$, $\boldsymbol{b}_{\mathrm{o}}$, and $\boldsymbol{b}_{\mathrm{c}}$ denote the bias matrices of each gate; σ is the sigmoid activation function; and $\otimes$ signifies the element-wise product of vectors.

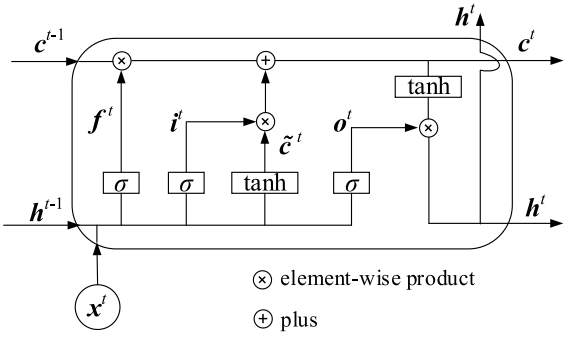

Fig. 2. Structural diagram of an LSTM memory cell (Guan and Yang, 2023).

2.3. BO

Although applying LSTM or MLP to predict CPT cone resistance profiles is promising, the selection of hyperparameters for a neural network often requires manual optimization. To address this issue, this paper employs a global optimization algorithm, BO, to identify the optimal combination of model hyperparameters that minimize the model error. BO utilizes a probabilistic model to fit real objective functions, where the next most probable point is selected for evaluation based on the fitting results. The framework comprises two key components: the probabilistic model and the acquisition function, which will be discussed in more detail in the following sections.

2.3.1. Probabilistic model

The probabilistic model is developed using a widely used Gaussian process (Tao et al., 2022), with the assumption that the responses follow a multidimensional normal distribution. Correspondingly, the error ε is assumed to follow an independent Gaussian distribution:

$$p(\boldsymbol{\varepsilon}) = N\left(0, \begin{bmatrix} k(x_1,x_1) & \cdots & k(x_1,x_n) \\ \vdots & \ddots & \vdots \\ k(x_n,x_1) & \cdots & k(x_n,x_n) \end{bmatrix}\right) \tag{3}$$

where $\boldsymbol{\varepsilon} = \boldsymbol{y} - f(\boldsymbol{x})$, in which $\boldsymbol{y}$ is the observation, and $f(\boldsymbol{x})$ is the model response; k denotes the covariance function; and x_i ($i = 1, 2, \ldots, n$) represents the input value (i.e. hyperparameters of MLP and LSTM). Accordingly, the likelihood distribution is given by

$$p(\boldsymbol{y}|\boldsymbol{f}) = N\left(\boldsymbol{f}, \begin{bmatrix} k(x_1,x_1) & \cdots & k(x_1,x_n) \\ \vdots & \ddots & \vdots \\ k(x_n,x_1) & \cdots & k(x_n,x_n) \end{bmatrix}\right) \tag{4}$$

where $\boldsymbol{f}$ is the assembly of $f(\boldsymbol{x})$. The joint distribution can be expressed as follows:

$$\begin{bmatrix} \boldsymbol{y} \\ f_* \end{bmatrix} \sim N\left(0, \begin{bmatrix} \boldsymbol{K} & \boldsymbol{K}_*^{\mathrm{T}} \\ \boldsymbol{K}_* & \boldsymbol{K}_{**} \end{bmatrix}\right) \tag{5}$$

where f* is the predicted model response, $\boldsymbol{K}$ is the covariance matrix of the assumed prior distribution, $\boldsymbol{K}_*$ is the covariance matrix of the observed set, and $\boldsymbol{K}_{**}$ is the covariance matrix of new-added samples.

2.3.2. Acquisition function

Acquisition functions are employed to select the next probable point that enables the model's optimal performance. The acquisition function can obtain the posterior distribution through the observed dataset $D_{1:n}$, thereby guiding the next evaluation point x_{n+1}. Expected improvement (EI), which is a widely used acquisition function, is employed here:

$$\alpha_n(x, D_{1:n}) = (\upsilon^* - \mu_n(x))\phi\left(\frac{\upsilon^* - \mu_n(x)}{\varphi_n(x)}\right) + \varphi_n(x)\phi\left(\frac{\upsilon^* - \mu_n(x)}{\varphi_n(x)}\right) \tag{6}$$

where α_n is the expectation; υ* is the current optimum function value; ϕ is the standard normal distribution probability density function; and φ_n and μ_n are the variance and mean value, respectively.

2.4. Bayesian-optimized MLP/LSTM

The selection of the hyperparameters of MLP and LSTM can be considered using BO in the following formula:

$$\boldsymbol{x}^* = \arg\ \min f(\boldsymbol{x}) \quad (\boldsymbol{x} \in \boldsymbol{X}) \tag{7}$$

where $f(\boldsymbol{x})$ is the objective function (also known as the model response); $\boldsymbol{x}$ represents a set of hyperparameters; $\boldsymbol{X}$ denotes the space of hyperparameter combinations; and $\boldsymbol{x}^*$ signifies the $\boldsymbol{x}$ that makes $f(\boldsymbol{x})$ obtain an optimized solution. In this contest, the mean squared error (MSE) of the neural network response is used as the objective function $f(\boldsymbol{x})$:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(\tilde{y}_i - y_i)^2 \tag{8}$$

where n represents the number of output elements; and $\tilde{y}_i$ and y_i denote the predicted and true values of a neural network, respectively.

In summary, this paper uses the Gaussian process to search for the optimal hyperparameter combination $\boldsymbol{x}$ for the neural networks. The posterior distribution is obtained from the observed dataset. Furthermore, the next evaluation point is selected by the EI acquisition function. Thus, the BO can improve step by step until the optimal hyperparameter combination is obtained.

3. Implementation of BO-MLP and BO-LSTM

3.1. Data source and processing

The selected cone resistance profiles from previously reported calibration chamber tests are summarized in Table 1, where cases 1–45 represent the experimental results, and cases 46–64 are derived from numerical simulations. Each case represents a complete q_c profile with depth. It is established that the CPT results are influenced by various factors. In this paper, five well-recognized factors that greatly affect q_c profiles are considered: the relative density D_r, initial effective vertical stress σ_v, lateral earth pressure coefficient K_0, saturation condition (SC, dry or saturated), and boundary condition (BC) (Huang and Hsu, 2005; Yang et al., 2010). The five most commonly used BCs (BC1–BC5) are summarized in Table 2. It should be noted that these boundary conditions are represented by the numbers 1–5 in the dataset, respectively. In calibration chambers, the soil stress field is generated artificially with the pre-set σ_v and K_0 instead of gravity consolidation. Accordingly, the soil stress can be considered constant throughout the calibration chamber. It is also assumed that the above five factors are constant.

For each q_c curve, the initial whole profile is processed into 50 points using interpolation. As can be observed in Table 1, a total of 64 groups of data were used for training and testing the network. About 80% of the data (52 groups) were employed to train the neural networks, while the remaining 20% (12 groups) marked with the symbol "*" in Table 1 were selected as the testing dataset to evaluate whether the model overfits or underfits. This split ratio of training and testing datasets has been well-validated for both MLP (Wu et al., 2023) and LSTM neural networks (Zhang et al., 2020b).

The quality of the input data may significantly affect the predictive performance of DL models. Normalization is an effective method for rescaling variables with different scales, thereby reducing their influence on the model performance and lowering the computational costs. The following equation is used to normalize the input data to the common range of 0–1:

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{9}$$

where x is the raw input variable before normalization; x_{norm} is the input variable after normalization; and x_{min} and x_{max} are the minimum and maximum values of the input variable, respectively.

Furthermore, in many cases, especially in DEM simulations (Arroyo et al., 2011), the q_c profile usually exhibited large oscillations, which would diminish the quality of the training data. Therefore, a sliding window approach was introduced to smooth the raw data (Zhang et al., 2020b). The smoothed value x_s can be calculated by

$$x_s = \frac{1}{w}\sum_{i=n-w+1}^{n} x_i \tag{10}$$

where w is the window size. The mean value of the datasets within a given window is then assigned as the new value of the studied parameter under study. A larger window size may result in a smoother sequential curve, but it also increases the possibility of deviating from the original curve. Accordingly, the window size is set to 4–30 according to the oscillation degree, to ensure the reliability and smoothness of the datasets. Fig. 3 shows the data before and after smoothing for cases 61 and 64 from the DEM (see Table 1). The magnitude and trend of the sequential curves are largely consistent with the original results, provided that the enhanced smoothness is taken into account.

3.2. Training strategy

For the cone resistance profile varying with depth, two strategies are proposed for the depth-dependence of the q_c profile. The first strategy postulates a one-to-one mapping relationship between depth and q_c, where one depth corresponds to a specific q_c value. The second strategy postulates the depth-dependent q_c profile as a whole sequence, wherein the increase of depth is similar to the time.

The first strategy may be regarded as a regression task, in which the inputs comprise D_r, σ_v, K_0, SC, BC, and the value of depth z, while the output is the corresponding q_c value, as illustrated in Fig. 4a. In this context, the MLP is employed to fulfil this task. In the second strategy, sequence-to-sequence training in LSTM is employed, where the entire q_c curve is regarded as sequential data. The training and predictive strategy is shown in Fig. 4b. The input comprises a sequence of data, including D_r, σ_v, K_0, SC, BC, and depth z in all time steps for a single case in Table 1. The output sequence represents the entire cone resistance profile q_c in all time steps. The MSE between the measured and predicted cone resistance profiles is employed as the loss function to train the LSTM. The above five influencing factors are treated as sequential inputs, maintaining their constant values across all steps. The output sequence is required to retain the same length as the input sequence, enabling the LSTM to perform the sequence-to-sequence training.

3.3. Neural network structures

3.3.1. MLP

The MLP structure employed in this context comprises four layers: an input layer, two hidden layers, and an output layer, as shown in Fig. 1. The main hyperparameters of the MLP are enumerated in Table 3. To train the MLP, three key hyperparameters are optimized using BO: the number of neurons in the first and second hidden layers (N_{n1} and N_{n2}) as well as the initial learning rate (η_{MLP}). Furthermore, the range of N_{n1} and N_{n2} was set to 5–30

Table 1
Summary of used cases of calibration chamber test.

Case No.	D_r	σ_v (MPa)	K_0	SC	BC	Reference
1	0.878	0.1	0.45	Dry	5	Kluger et al. (2021)
2	0.748	0.2	0.45	Dry	5	
3	0.838	0.1	0.45	Saturated	5	
4*	0.78	0.2	0.45	Saturated	5	
5	0.952	0.1	0.45	Dry	5	
6	0.97	0.1	0.45	Saturated	5	
7	0.668	0.2	0.45	Saturated	5	
8*	0.606	0.2	0.45	Dry	5	
9	0.918	0.2	0.45	Saturated	5	
10	0.918	0.2	0.45	Dry	5	
11	0.65	0.056	0.393	Dry	5	Huang and Hsu (2005)
12	0.84	0.056	0.393	Dry	5	
13	0.25	0.16	0.463	Dry	5	
14	0.5	0.16	0.463	Dry	5	
15*	0.65	0.16	0.463	Dry	5	
16	0.84	0.16	0.463	Dry	5	
17	0.5	0.056	0.786	Dry	1	
18	0.5	0.056	0.786	Dry	5	
19*	0.65	0.056	0.786	Dry	1	
20	0.65	0.056	0.786	Dry	5	
21	0.84	0.056	0.786	Dry	1	
22	0.84	0.056	0.786	Dry	5	
23	0.5	0.07	0.4	Dry	1	Bałachowski (2006)
24*	0.5	0.1	0.4	Dry	1	
25	0.5	0.15	0.4	Dry	1	
26	0.5	0.2	0.4	Dry	1	
27	0.5	0.35	0.4	Dry	1	
28	0.5	0.4	0.4	Dry	1	
29	0.8	0.05	0.4	Dry	1	
30	0.8	0.07	0.4	Dry	1	
31	0.8	0.1	0.4	Dry	1	
32	0.8	0.15	0.4	Dry	1	
33	0.8	0.25	0.4	Dry	1	
34*	0.8	0.3	0.4	Dry	1	
35	0.8	0.4	0.4	Dry	1	
36	0.8	0.05	0.4	Dry	3	
37	0.8	0.07	0.4	Dry	3	
38*	0.8	0.15	0.4	Dry	3	
39	0.8	0.2	0.4	Dry	3	
40*	0.8	0.25	0.4	Dry	3	
41	0.33	0.025	1	Saturated	1	Pournaghiazar et al. (2013)
42	0.33	0.05	1	Saturated	1	
43*	0.33	0.1	1	Saturated	1	
44	0.61	0.03	1	Saturated	1	
45	0.61	0.05	1	Saturated	1	
46	0.718	0.15	0.5	Dry	3	Chen et al. (2020)
47*	0.615	0.15	0.5	Dry	3	
48	0.395	0.15	0.5	Dry	3	
49	0.231	0.15	0.5	Dry	3	
50	0.23	0.1	1	Dry	3	Schnaid (1990)
51*	0.63	0.1	1	Dry	3	
52	0.86	0.1	1	Dry	3	
53	0.65	0.075	0.5	Dry	3	
54	0.65	0.15	0.5	Dry	3	
55	0.65	0.3	0.5	Dry	3	
56	0.752	0.06	1	Dry	1	Arroyo et al. (2011)
57	0.752	0.1	1	Dry	1	
58	0.768	0.2	1	Dry	1	
59*	0.776	0.3	1	Dry	1	
60	0.784	0.4	1	Dry	1	
61	0.907	0.1	1	Dry	1	
62	0.914	0.14	1	Dry	1	
63	0.922	0.2	1	Dry	1	
64	0.929	0.3	1	Dry	1	

Note: '*' represents the testing dataset.

to reduce the complexity of the network structure and prevent overfitting. The adaptive moment estimation (Adam) is employed to optimize the weight and bias due to its superior performance (Ruder, 2016), and the Relu activation function is adopted for the hidden layers. The batch size used for each training iteration is set to 128. The maximum number of training epochs is selected as 200, given the relatively limited data (Zhang et al., 2020a; Guan and Yang, 2023).

3.3.2. LSTM

The LSTM structure is composed of three layers: a sequence input layer, an LSTM layer, and a fully connected layer. The main

Table 2
Five boundary conditions in the calibration chamber.

BC	Top and bottom boundary		Lateral boundary	
	Stress	Strain	Stress	Strain
BC1	Constant	–	Constant	–
BC2	–	0	–	0
BC3	Constant	–	–	0
BC4	–	0	Constant	–
BC5	Constant	–	Servo–controlled	

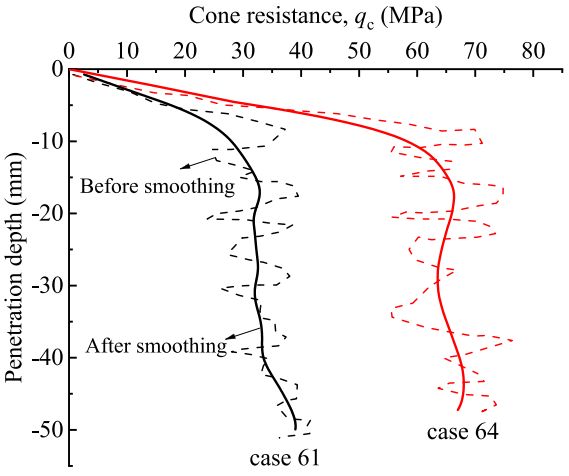

Fig. 3. Data smoothing.

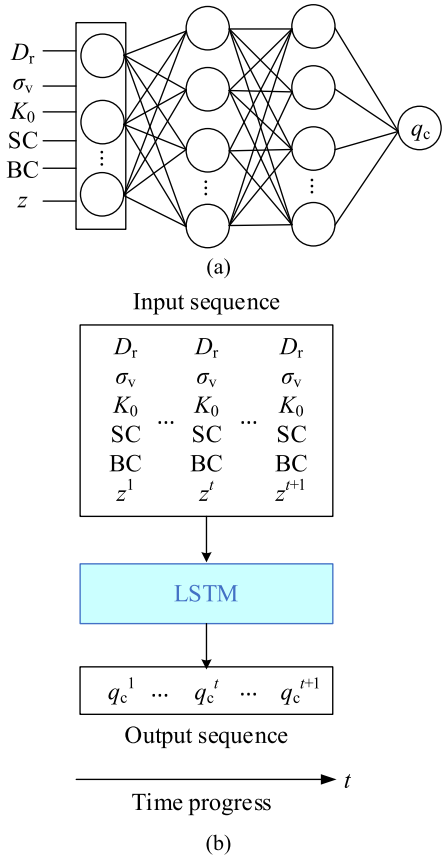

Fig. 4. Two training strategies: (a) Depth-q_c one-to-one mapping for the MLP; and (b) Sequence-to-sequence training for the LSTM.

hyperparameters of the LSTM are listed in Table 4. In this study, three hyperparameters – the number of neurons in the hidden layer (N_n), the initial learning rate (η_{LSTM}), and the regularization parameter (L_2) – are determined with the aid of BO. The optimization ranges covering the general values are also provided in Table 4. The Adam optimizer is utilized, and the batch size is set to 60. The number of epochs is set to 250 (Zhang et al., 2020b).

3.3.3. Evaluation index

The discrepancy between predicted and target values is measured using the mean squared error (MSE, see Eq. (8)) as the loss function. Additionally, two more indicators are utilized: the coefficient of determination (R^2) and the mean absolute percentage error (MAPE):

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \tilde{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{11}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{\tilde{y}_i - y_i}{y_i}\right| \times 100\% \tag{12}$$

where $\bar{y}$ is the mean true value.

3.4. BO-MLP/BO-LSTM modeling

Fig. 5 presents a flowchart illustrating the BO-MLP and BO-LSTM modeling process. This process can be described by six principal steps:

(1) Step 1. The raw datasets are obtained under different soil and penetration conditions, as presented in Section 3.1. Subsequently, the dataset is normalized and divided into two distinct sets: a training set and a testing set.
(2) Step 2. The MLP and LSTM neural networks are established, and the hyperparameters of the network models are determined. The range of hyperparameters to be optimized is also set. Further details can be found in Section 3.3.
(3) Step 3. The BO model is constructed. The MSE given in Eq. (8) is determined as the objective function to be minimized.
(4) Step 4. The neural network is trained under the current combination of hyperparameters with the training set. The value of the objective function under the current combination of hyperparameters is calculated and then returned to BO. According to the probabilistic model and acquisition function, the next group of hyperparameters is selected to update the training until reaching the maximum number of iterations.
(5) Step 5. The combination of hyperparameters with the optimal model performance after BO is then output. Thereafter, the optimized hyperparameters are employed for training the neural network.
(6) Step 6. The testing dataset is utilized to evaluate the working performance of the established neural network.

Table 3
Main hyperparameters of MLP network.

Hyperparameter	Description	Value
N_h	Number of hidden layers	2
N_{n1}	Number of neurons in the first hidden layer	[5, 30]#
N_{n2}	Number of neurons in the second hidden layer	[5, 30]#
Optimizer	Algorithm for optimizing weights and biases	Adam
MaxEpochs	Maximum number of rounds used for training	200
Batch size	The number of samples in each training process.	128
η_{MLP}	Initial learning rate	[1×10^{-4}, 1×10^{-2}]#

Note: "#" represents the hyperparameters to be optimized by BO.

Table 4
Main hyperparameters of LSTM model.

Hyperparameter	Description	Value
N_h	Number of hidden layers	1
N_n	Number of neurons in the hidden layer	[10, 80]#
Optimizer	Algorithm for optimizing weights and biases	Adam
η_{LSTM}	Initial learning rate	[1×10^{-4}, 1×10^{-2}]#
L_2	L_2 regularization parameter	[1×10^{-5}, 1×10^{-2}]#
MaxEpochs	Maximum number of rounds used for training	250
Batch size	The batch size used for each training iteration	60

Note: "#" represents the hyperparameters to be optimized by BO.

4. Performance of the DL models

4.1. Effectiveness of BO

When using BO to optimize the neural networks, the iteration number of the objective function is set to 20 to prevent overfitting (Huang et al., 2023). The process is displayed in Fig. 6. With the process of BO, the minimum objective value (i.e. the MSE of the neural network) decreases to a minimal value, signifying a good effect achieved by using BO. For the MLP neural network, the optimized combination of hyperparameters was $N_{n1} = 19$, $N_{n2} = 12$, and $\eta_{MLP} = 3.9 \times 10^{-3}$, while for the LSTM neural network, it was $N_n = 49$, $\eta_{LSTM} = 2.1 \times 10^{-5}$, and $L_2 = 9.8 \times 10^{-3}$. These hyperparameters were subsequently used to train the neural networks.

To demonstrate the effectiveness of BO for MLP and LSTM, the evaluation indices R^2 and MAPE are depicted in Fig. 7, which are also compared to the MLP and LSTM without optimization. For both the MLP and LSTM, the R^2 with BO is higher in both the training and testing sets than that without BO, and the MAPE with BO is much lower than that without BO. This indicates that BO performs well in searching for suitable hyperparameters. Furthermore, the MLP and LSTM achieve similar performance in the training and testing sets. This illustrates that both strategies proposed in this paper are viable for reproducing cone resistance profiles.

4.2. Results of developed models

4.2.1. Training results

The loss function of the two models with the optimized hyperparameters is shown in Fig. 8a and b, respectively. For both models, the MSE rapidly reduces to a relatively small value. The MSE for the testing set converges to a small value but shows no tendency to increase, indicating sufficient training without overfitting.

To evaluate the accuracy of the developed model, a comparison between the predicted and the measured q_c values for the testing set is presented in Fig. 9. As expected, the predicted q_c values from the neural networks are in close agreement with the measured values. The R^2 of the BO-LSTM and BO-MLP neural networks both achieve high values, approximately 95%. This indicates that both the MLP and LSTM neural networks perform well in predicting cone resistances among the selected raw datasets. However, some discrepancies between the predicted and measured values remain. One potential explanation for this is that the fitting degree of neural networks is not possible to reach 100% to fit all discrepancies using DL approaches (Zhang et al., 2020b).

To further evaluate the suitability of the optimized hyperparameters of the neural networks for the given cases, Fig. 10 shows the variation of R^2 with the number of cases used in the neural networks. It can be observed that R^2 for LSTM and MLP present a general tendency of increasing with the number of cases used in both training and testing sets. However, the R^2 of the testing set in the used cases 52–58 slightly decreases for both neural networks. This may be attributed to the fact that the results of these cases are not identical to those of other research, thereby reducing the overall quality of the dataset.

4.2.2. Cone resistance profiles

Fig. 11 presents a comparison between the predicted q_c profiles from BO-MLP and BO-LSTM and the corresponding true data. Each subplot in Fig. 11 illustrates three instances of successful predictions (red lines) and three instances of less optimal predictions (black lines). As can be observed in Fig. 11, the variation tendency of cone resistance with the normalized depth z/z_{total} (ratio of the current penetration depth over the total penetration depth) can be successfully reproduced by the two neural networks during penetration processes, both in the training and testing sets. Given that the coefficient of determination R^2 has reached relatively large values for both the training and testing datasets, it can be concluded that neural networks can accurately predict a significant number of cases. The predictions by the BO-MLP and BO-LSTM are similar to the true data, especially for the stable q_c values. However, some cases in the BO-MLP and BO-LSTM exhibit discrepancies between the measured and predicted values, particularly in the more problematic cases. As shown in the example of case 4 in Fig. 11d, the actual stable q_c is about 23 MPa, while the prediction reaches 21 MPa, resulting in a relative error of around 10%. However, this discrepancy is not significant in practical applications.

4.2.3. Error evaluation

To comprehensively assess the performance of the developed models, the relative error (RE) is employed to reflect the percentage of deviation between the measured values and the predictions:

$$RE = \left|\frac{\tilde{y}_i - y_i}{y_i}\right| \tag{13}$$

Fig. 12 depicts the frequency and cumulative probability of relative error distribution for the testing set in the BO-MLP and BO-LSTM models. The distribution of relative error for the BO-MLP and BO-LSTM is similar. The majority of datasets have a cumulative probability larger than 70% and 60% for BO-MLP and BO-LSTM, respectively, with relative errors below 10%. Additionally, the BO-MLP has a higher frequency of errors smaller than 5%. These findings indicate that the developed DL models are sufficiently accurate for calibration chamber tests.

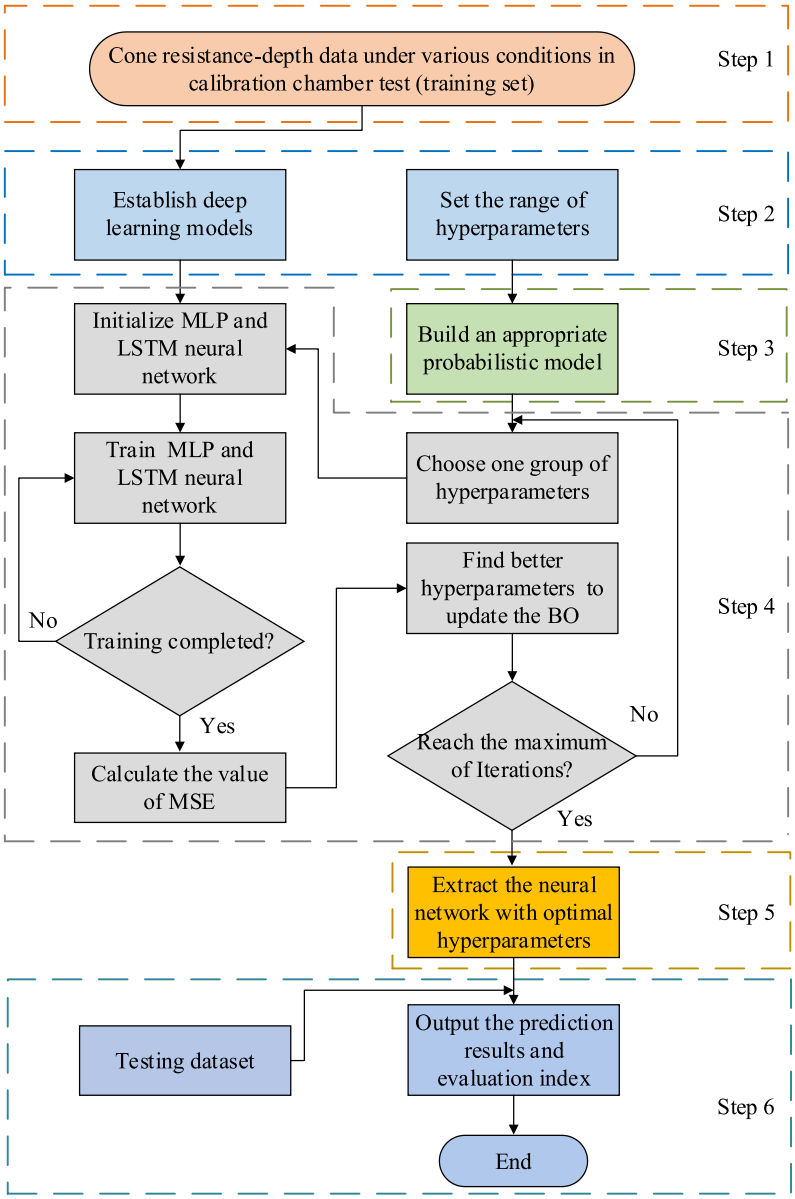

Fig. 5. Flow chart for establishing BO-MLP/BO-LSTM models.

5. Further validation and application

5.1. Validation with numerical solutions

This section aims to further compare the neural networks with numerical solutions for other cases that are not included in Table 1. These cases have been designed to consider the variation of relative density, vertical effective stress, and saturation conditions of sand, as listed in Table 5. Subsequently, numerical simulations have been performed to install the CPT in a calibration chamber for these cases, resulting in the production of cone resistance profiles. Details on the CEL model setup and soil constitutive relationships can be found in Appendices A and B, respectively.

The CEL model is initially evaluated and verified through a comparison with calibration chamber tests conducted by Kluger et al. (2021) on Ticino sand. The results of the testing conducted under two initial effective vertical stresses (σ_v) of 100 kPa and 200 kPa are presented in Fig. 13. It can be observed from Fig. 13 that there is a notable alignment between the numerical solutions and the testing results, which lends credibility to the assertion that the CEL modeling can facilitate the generation of q_c profiles in calibration chamber tests.

Fig. 14 shows the comparisons of cone resistance profiles obtained from the two neural networks and the numerical simulations for the artificial cases presented in Table 5. The q_c profiles obtained from DL models are fair close to the numerical simulations, although some cases like No. 2 in Table 5 still demonstrate a degree of discrepancy between the two methods. However, the stable values of cone resistance show a better agreement between the numerical simulations and the neural networks. It is important to note that the deployment of numerical models still requires complex model setup and time-consuming running. In contrast, a well-trained neural network can swiftly (typically in a few seconds) generate reliable q_c curves under specific soil conditions.

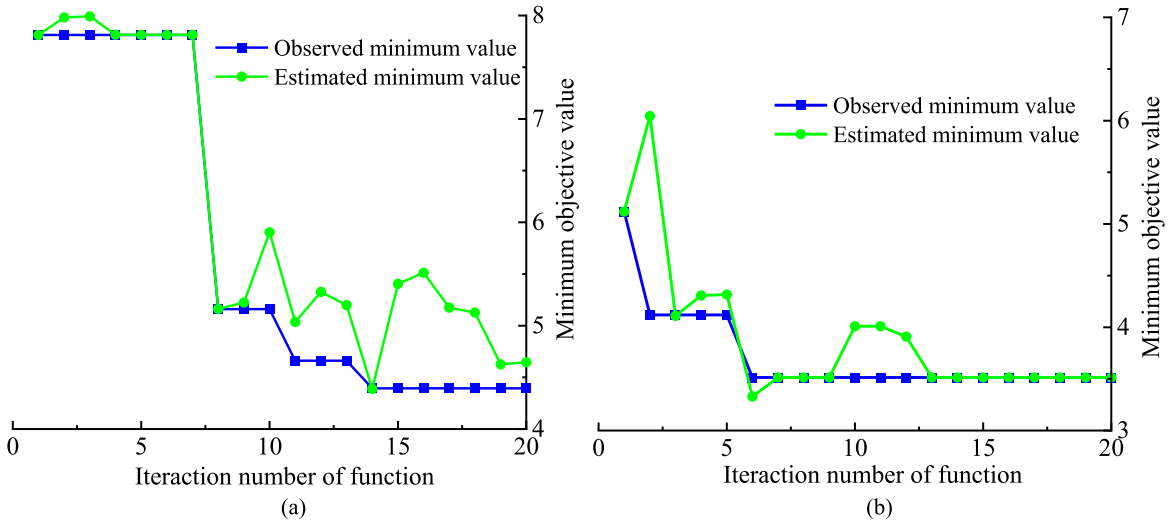

Fig. 6. Process of BO for (a) MLP and (b) LSTM models.

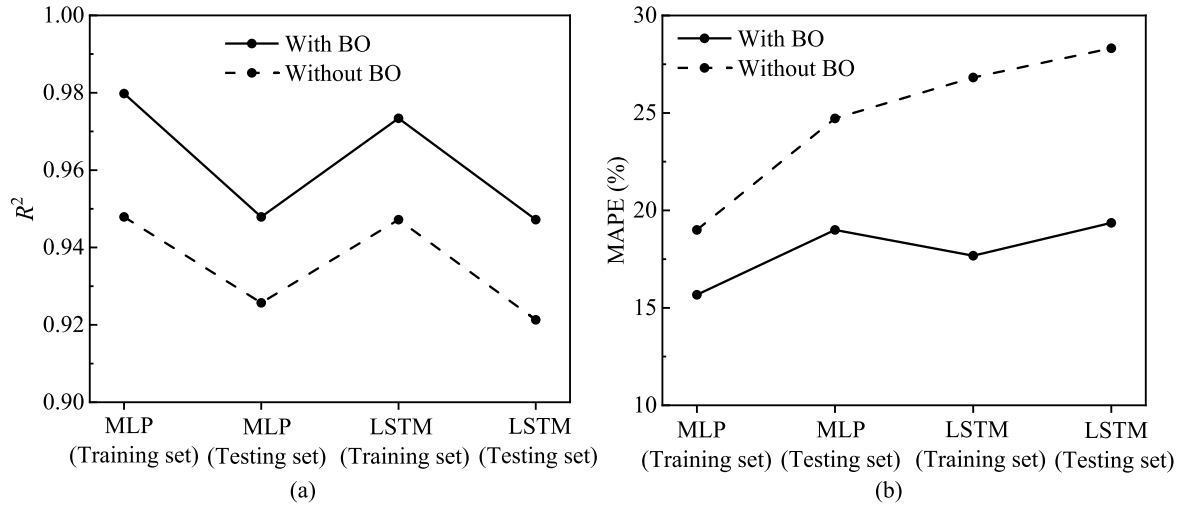

Fig. 7. Evaluation indices for neural networks with and without BO: (a) R^2; and (b) MAPE.

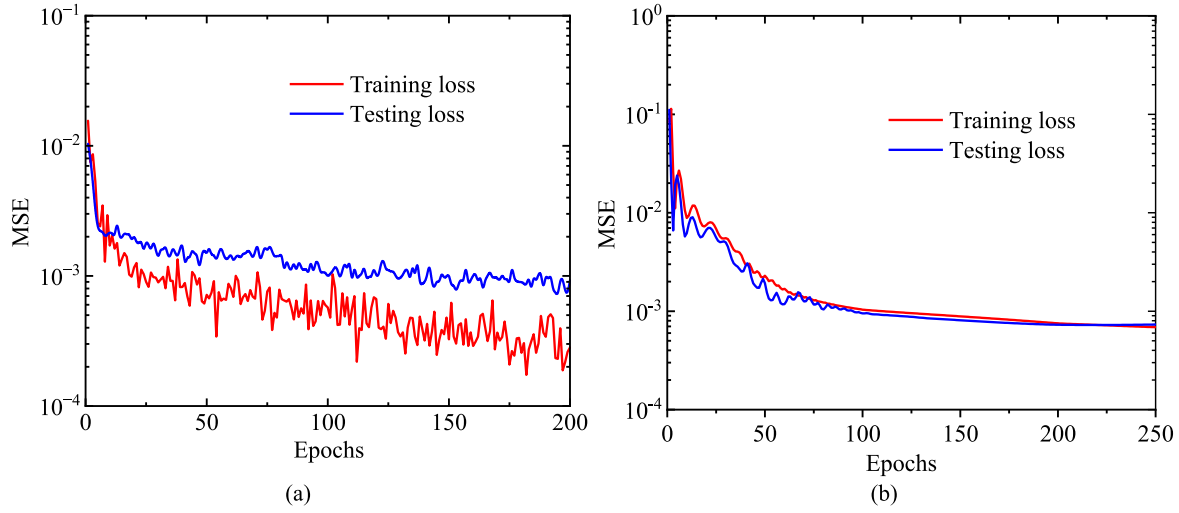

Fig. 8. Evolution of the MSE against epochs for (a) BO-MLP and (b) BO-LSTM.

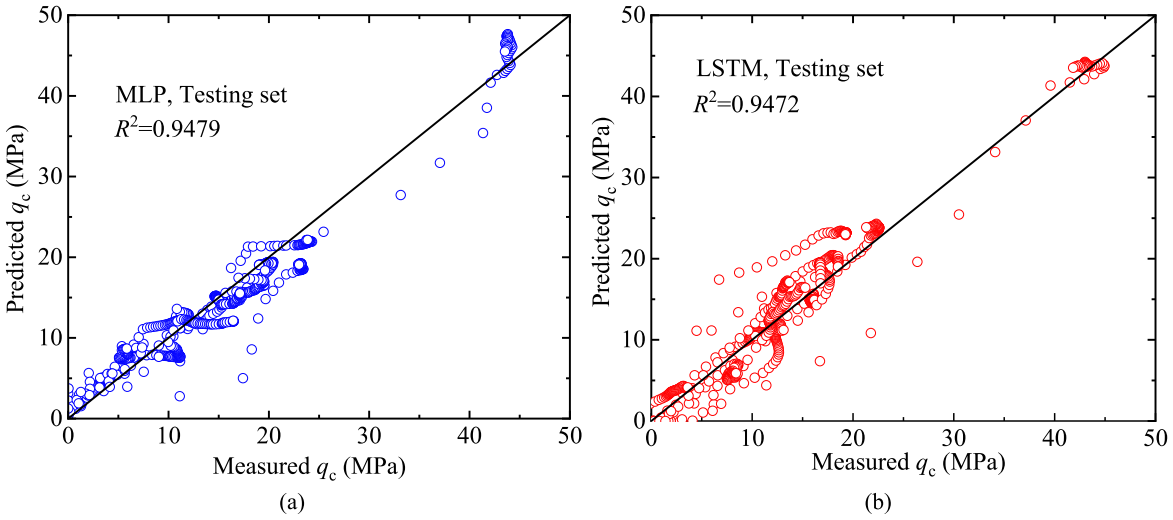

Fig. 9. Measured cone resistances against predicted values for the testing set in (a) BO-MLP and (b) BO-LSTM.

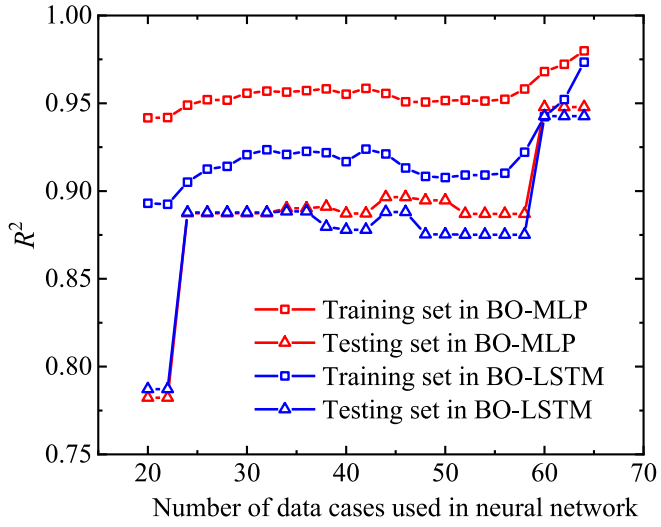

Fig. 10. The R^2 of training and testing sets with the number of data cases used in neural networks.

5.2. Application: relating q_c with soil property

As mentioned before, most of the existing correlations between q_c and soil properties (e.g. relative density D_r) were presented through a limited number of datasets for specific soil types. The trained neural networks have been proven as a reliable approach to rapidly extend the database in calibration chambers. Consequently, these correlations can be enhanced based on the extended database from the DL models. This section illustrates an example of how to potentially improve the D_r-q_c correlation based on the developed DL models.

Table 6 summarizes four reported correlations between D_r and q_c. Here, the normalized cone resistance of $Q = (q_c - \sigma_v)/\sigma_v$ was introduced to consider the influence of vertical stress whenever it was available. Fig. 15 presents the deduced correlations of D_r-Q from Eqs. (14)–(17). The 1787 groups of D_r-Q dataset generated from the developed neural networks are also presented in Fig. 15. It should be noted that only the stable q_c values from cone resistance profiles are used to relate soil properties, and the DL results take the BO-MLP as an example. The previous experimental results from Bolton and Gui (1993) and Jamiolkowski (1985) are also displayed in Fig. 15.

A comparison of the dataset from the developed neural networks shows that the linear regression of Eq. (16) is not an optimal fit for all the data, especially for D_r lower than 50%. Eq. (17) underestimates the relative density derived from the normalized cone resistance, with the resulting value approaching the lower bound of the dataset. The logarithmic form of Eqs. (14) and (15) is more consistent with the D_r-Q relationship. However, Eqs. (14) and (15) slightly overestimate the dataset. For pure experimental data in Fig. 15, the R^2 of Eqs. (14)–(17) are 68.7%, 71.3%, 75.6%, and 65.2%, respectively. This demonstrates that traditional empirical equations regarding a specific soil type are somewhat constrained in their applicability when extended to other soil states. However, the use of DL models enables the generation of hundreds of D_r-Q points, facilitating practical applications. Consequently, an enhanced correlation of D_r-Q by curve fitting to all the data in Fig. 15 is given below:

$$D_r = 28.15 \ln Q - 75.57 \tag{18}$$

Eq. (18) has a high fitting degree of 82% to all D_r-Q points. For the pure experimental data in Fig. 15, Eq. (18) still has a relatively high R^2 value of 89.6%. It is believed that the improved Eq. (18) is more reliable since the database used for the fitting included not only the published experimental data but also a large amount of data from the developed DL models.

6. Conclusions

This study has successfully developed Bayesian-optimized neural networks for predicting cone resistance profiles in calibration chamber tests performed on the sand. The process commenced with the training of the neural networks using 52 cases sourced from the literature, followed by the validation using a testing dataset consisting of 12 groups. BO was used to identify the optimal hyperparameters for both the MLP and LSTM networks. The results demonstrate that the neural network after BO exhibited superior

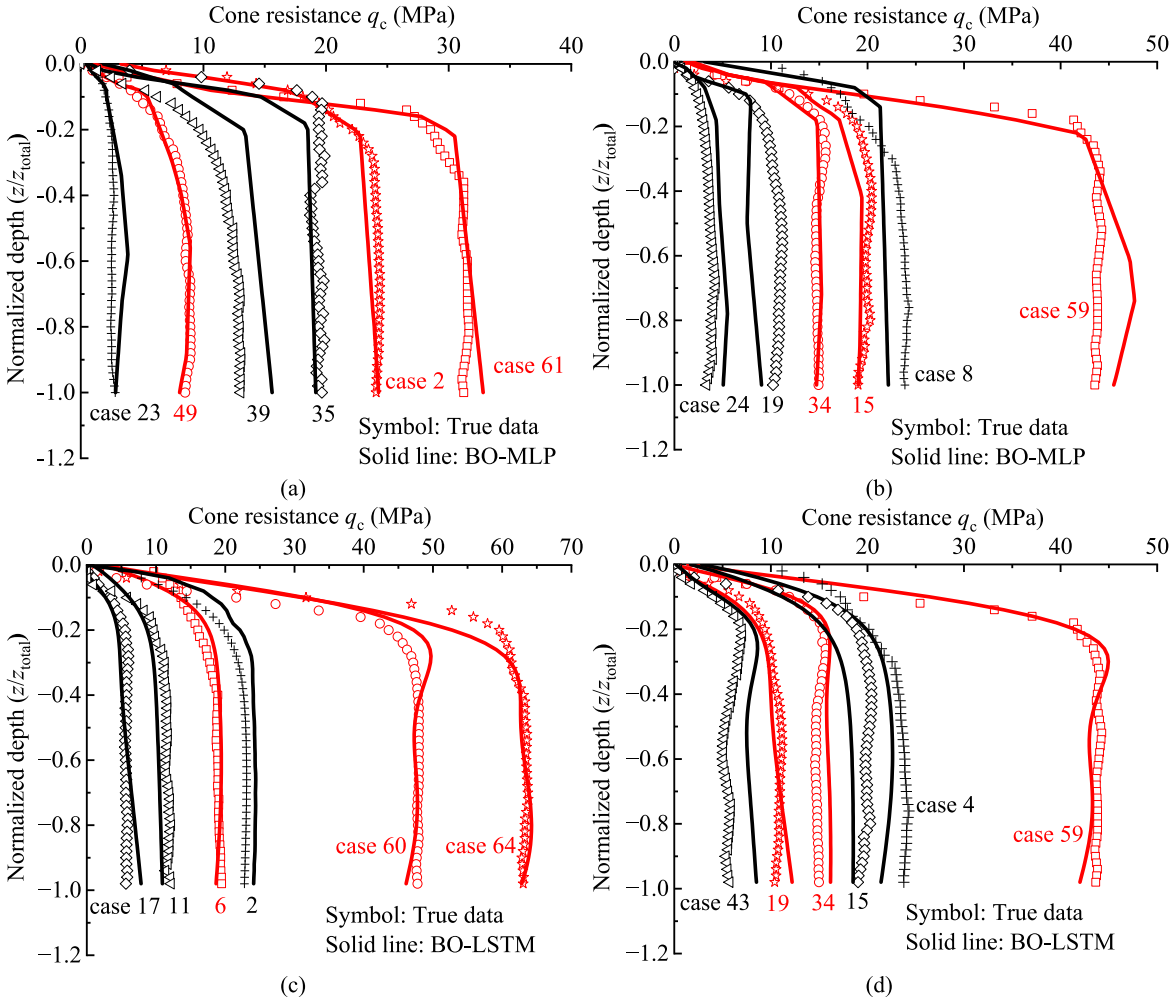

Fig. 11. Comparison of measured and predicted q_c profiles from BO-MLP in (a) training and (b) testing sets, and from BO-LSTM in (c) training and (d) testing sets. Red lines represent better predictions, while black lines represent worse cases.

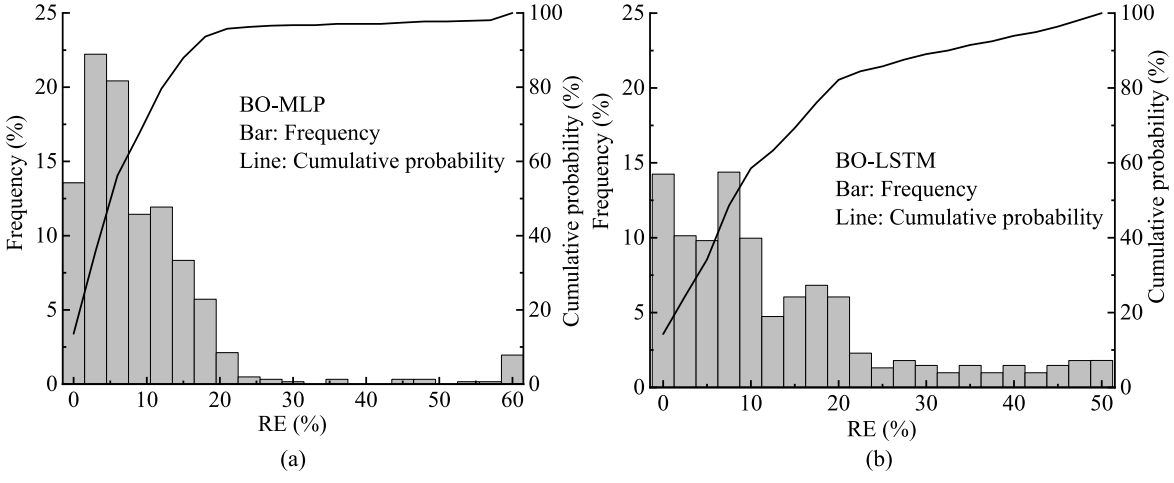

Fig. 12. Frequency and cumulative probability of relative error in the testing set for (a) BO-MLP and (b) BO-LSTM.

Table 5
Additional cases to be validated by numerical modeling.

No.	D_r	σ_v (MPa)	K_0	Saturation condition
1	0.563	0.13	0.42	Dry
2	0.82	0.15	0.45	Saturated
3	0.734	0.09	0.5	Dry
4	0.8	0.16	0.38	Dry

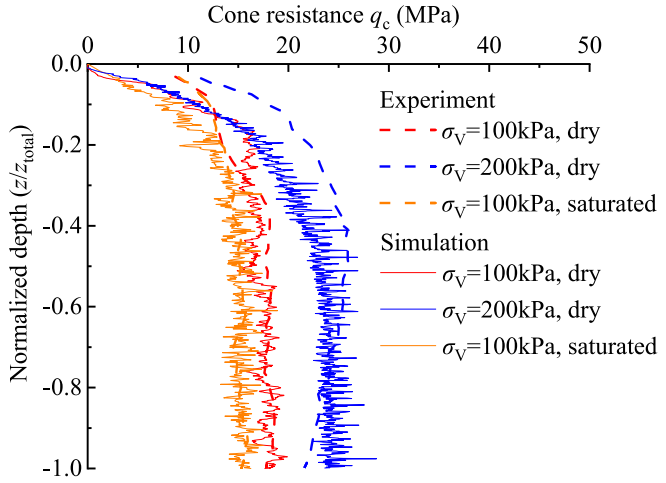

Fig. 13. Comparison of the simulated cone resistance with the experimental results (Kluger et al., 2021).

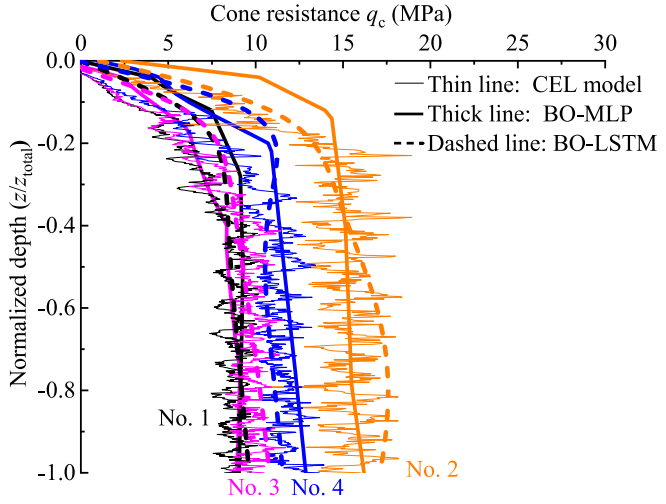

Fig. 14. Comparison of neural networks with CEL model in predicting unseen cases.

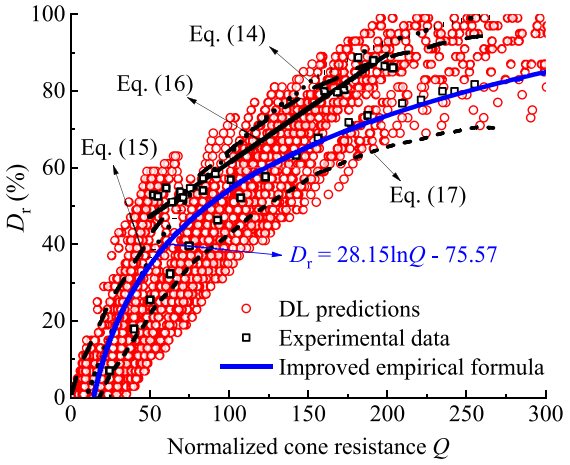

Fig. 15. Correlation between the relative density and normalized cone resistance.

performance compared to that without optimization.

Two strategies were proposed to produce the cone resistance profile: (1) depth-resistance mapping using MLP and (2) sequence-to-sequence training using LSTM. The results illustrated that both strategies yielded comparative performance and were capable of reproducing the entire cone resistance profile. Further comparisons were carried out against the numerical results from the CEL model, and a favorable agreement between the numerical and predicted results was observed, providing further evidence of the reliability of the developed DL models.

Finally, the validated DL models were used to generate a large number of datasets, correlating cone resistance with the relative density of sand to prove the validity of the derived results. The obtained R^2 value between the predicted and previous experimental data was 89.6%, indicating a notable degree of predictive ability with superior generalization.

This study has demonstrated that neural networks can be applied to provide general models that serve as a possible alternative for CPT virtual calibration chambers. This approach outperforms other previously published calibration chambers based on numerical models, as it provides pseudo-real-time predictions and is, therefore, orders of magnitude more computationally efficient. Furthermore, this work paves the way for extending CPT-based material parameter interpretations, such as soil strength and stiffness parameters under drained and undrained conditions, using other DL-based approaches.

Table 6
Summary of representative correlations of D_r-q_c-σ_v.

Empirical formula		Empirical coefficient	Reference
$D_r = \frac{1}{C_2}\ln\left[\frac{q_c}{C_0(\sigma_v)^{C_1}}\right]$	(14)	$C_0 = 60$, $C_1 = 0.7$, $C_2 = 2.91$	Lunne and Christoffersen (1983)
$D_r = C_2 \log_{10}\left[\frac{q_c/p_a}{(\sigma_v/p_a)^{C_1}} - 1\right]$	(15)	$C_1 = 0.5$, $C_2 = 68$, p_a is the atmospheric pressure	Kulhawy et al. (1990)
$D_r = AQ + B, Q = \frac{q_c - \sigma_v}{\sigma_v}$	(16)	$A = 0.2831$, $B = 32.964$	Bolton and Gui (1993)
$D_r = A_0 + B_0 \ln(q_c/\sigma_v^{0.5})$	(17)	$A_0 = -1.292$, $B_0 = 0.268$	Jamiolkowski et al. (2003)

CRediT authorship contribution statement

Mingpeng Liu: Writing – original draft, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Enci Sun:** Methodology, Data curation. **Ningning Zhang:** Writing – review & editing, Validation, Supervision. **Fengwen Lai:** Writing – review & editing, Visualization, Methodology. **Raul Fuentes:** Writing – review & editing, Validation, Supervision.

Declaration of interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The authors acknowledge the funding support from the National Natural Science Foundation of China (Grant No. 52408356) and the China Scholarship Council (CSC).

Appendices A and B. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.jrmge.2024.10.004.

References

Arroyo, M., Butlanska, J., Gens, A., Calvetti, F., Jamiolkowski, M., 2011. Cone penetration tests in a virtual calibration chamber. Geotechnique 61, 525–531.
Bałachowski, L., 2006. Penetration resistance of Lubiatowo sand in calibration chamber tests. Arch. Hydro-Eng. Environ. Mech. 53, 311–329.
Bolton, M., Gui, M., 1993. The Study of Relative Density and Boundary Effects for Cone Penetration Tests in Centrifuge. University of Cambridge, Cambridge, UK. Department of Engineering.
Bui, D.T., Tsangaratos, P., Nguyen, V.T., Liem, N.V., Trinh, P.T., 2020. Comparing the prediction performance of a Deep Learning Neural Network model with conventional machine learning models in landslide susceptibility assessment. Catena 188, 104426.
Butlanska, J., Arroyo, M., Gens, A., O'Sullivan, C., 2014. Multi-scale analysis of cone penetration test (CPT) in a virtual calibration chamber. Can. Geotech. J. 51, 51–66.
Chen, Z., Huang, M., Shi, Z., 2020. Application of a state-dependent sand model in simulating the cone penetration tests. Comput. Geotech. 127, 103780.
Cudmani, R., Osinov, V., 2001. The cavity expansion problem for the interpretation of cone penetration and pressuremeter tests. Can. Geotech. J. 38, 622–638.
Dai, H., MacBeth, C., 1997. Effects of learning parameters on learning procedure and performance of a BPNN. Neural Network. 10, 1505–1521.
Fan, S., Bienen, B., Randolph, M., 2018. Stability and efficiency studies in the numerical simulation of cone penetration in sand. Géotech. Letture 8, 13–18.
Gao, X., Shi, M., Song, X., Zhang, C., Zhang, H., 2019. Recurrent neural networks for real-time prediction of TBM operating parameters. Autom. ConStruct. 98, 225–235.
Guan, Q.Z., Yang, Z.X., 2023. Hybrid deep learning model for prediction of monotonic and cyclic responses of sand. Acta Geotech 18, 1447–1461.
Guan, Q.Z., Yang, Z.X., Guo, N., Hu, Z., 2023. Finite element geotechnical analysis incorporating deep learning-based soil model. Comput. Geotech. 154, 105120.
Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural Comput. 9, 1735–1780.
Huang, A.B., Hsu, H.H., 2005. Cone penetration tests under simulated field conditions. Geotechnique 55, 345–354.
Huang, H.W., Li, Q.T., Zhang, D.M., 2018. Deep learning based image recognition for crack and leakage defects of metro shield tunnel. Tunn. Undergr. Space Technol. 77, 166–176.
Huang, Y., Wang, Y., Xu, Z., Wang, P., 2023. Prediction and variable importance analysis for small-strain stiffness of soil based on ensemble learning with Bayesian optimization. Comput. Geotech. 162, 105688.
Jamiolkowski, M., 1985. New developments in field and laboratory testing or soils. In: Proceedings of the 11th International Conference on Soil Mechanics and. Foundation Engineering., San Francisco, USA, pp. 57–153.
Jamiolkowski, M., Lo Presti, D., Manassero, M., 2003. Evaluation of relative density and shear strength of sands from CPT and DMT. In: Germaine, J.T., Sheahan, T.C., Whitman, R.V. (Eds.), Soil Behavior and Soft Ground Construction. American Society of Civil Engineers, Reston, USA, pp. 201–238.
Jas, K., Dodagoudar, G.R., 2023. Liquefaction potential assessment of soils using machine learning techniques: a state-of-the-art review from 1994–2021. Int. J. GeoMech. 23, 03123002.
Kluger, M.O., Kreiter, S., Stähler, F.T., Goodarzi, M., Stanski, T., Mörz, T., 2021. Cone penetration tests in dry and saturated Ticino sand. Bull. Eng. Geol. Environ. 80, 4079–4088.
Kohestani, V.R., Hassanlourad, M., 2016. Modeling the mechanical behavior of carbonate sands using artificial neural networks and support vector machines. Int. J. GeoMech. 16, 04015038.
Kulhawy, F.H., Mayne, P.W., Institute, E.P.R., Group, C.U.G.E., 1990. Manual on Estimating Soil Properties for Foundation Design. Electric Power Research Institute, Palo Alto, USA.
Lai, F., Shiau, J., Keawsawasvong, S., Chen, F., Banyong, R., Seehavong, S., 2023. Physics-based and data-driven modeling for stability evaluation of buried structures in natural clays. J. Rock Mech. Geotech. Eng. 15, 1248–1262.
Liu, M., Zhuang, P., Lai, F., 2024. A Bayesian optimization-genetic algorithm-based approach for automatic parameter calibration of soil models: application to clay and sand model. Comput. Geotech. 176, 106717.
Liyanapathirana, D.S., 2009. Arbitrary Lagrangian Eulerian based finite element analysis of cone penetration in soft clay. Comput. Geotech. 36, 851–860.
Lunne, T., Christoffersen, H.P., 1983. Interpretation of cone penetrometer data for offshore sands. In: Offshore Technology Conference. https://doi.org/10.4043/4464-MS. Houston, USA.
Lunne, T., Powell, J.J., Robertson, P.K., 2002. Cone Penetration Testing in Geotechnical Practice. CRC Press, London, UK.
Mo, P.Q., Marshall, A.M., Yu, H.S., 2017. Interpretation of cone penetration test data in layered soils using cavity expansion analysis. J. Geotech. Geoenviron. 143, 04016084.
Nhu, V.H., Hoang, N.D., Nguyen, H., Ngo, P.T.T., Thanh Bui, T., Hoa, P.V., Samui, P., Tien Bui, D., 2020. Effectiveness assessment of Keras based deep learning with different robust optimization algorithms for shallow landslide susceptibility mapping at tropical area. Catena 188, 104458.
Phoon, K.K., Zhang, W., 2023. Future of machine learning in geotechnics. Georisk 17, 7–22.
Pournaghiazar, M., Russell, A.R., Khalili, N., 2013. The cone penetration test in unsaturated sands. Geotechnique 63, 1209–1220.
Protopapadakis, E., Voulodimos, A., Doulamis, A., Doulamis, N., Stathaki, T., 2019. Automatic crack detection for tunnel inspection using deep learning and heuristic image post-processing. Appl. Intell. 49, 2793–2806.
Ruder, S., 2016. An overview of gradient descent optimization algorithms. https://doi.org/10.48550/arXiv.1609.04747.
Schnaid, F., 1990. A Study of the Cone-Pressuremeter Test in Sand. University of Oxford, Oxford, UK. PhD Thesis.
Susila, E., Hryciw, R.D., 2003. Large displacement FEM modelling of the cone penetration test (CPT) in normally consolidated sand. Int. J. Numer. Anal. Methods GeoMech. 27, 585–602.
Tao, Y., Sun, H., Cai, Y., 2022. Predictions of deep excavation responses considering model uncertainty: integrating BiLSTM neural networks with Bayesian updating. Int. J. GeoMech. 22, 04021250.
Wang, D., Bienen, B., Nazem, M., Tian, Y., Zheng, J., Pucker, T., Randolph, M.F., 2015. Large deformation finite element analyses in geotechnical engineering. Comput. Geotech. 65, 104–114.
Wang, K., Sun, W., 2018. A multiscale multi-permeability poroplasticity model linked by recursive homogenizations and deep learning. Comput. Methods Appl. Mech. Eng. 334, 337–380.
Wu, M., Xia, Z., Wang, J., 2023. Constitutive modelling of idealised granular materials using machine learning method. J. Rock Mech. Geotech. Eng. 15, 1038–1051.
Yang, Z., Jardine, R., Zhu, B., Foray, P., Tsuha, C., 2010. Sand grain crushing and interface shearing during displacement pile installation in sand. Geotechnique 60, 469–482.
Yazici, K., Taskin, A., 2023. A comparative Bayesian optimization-based machine learning and artificial neural networks approach for burned area prediction in forest fires: an application in Turkey. Nat. Hazards 119, 1883–1912.
Yousefpour, N., Correa, O., 2023. Towards an AI-based early warning system for bridge scour. Georisk 17, 713–739.
Yousefpour, N., Downie, S., Walker, S., Perkins, N., Dikanski, H., 2021b. Machine learning solutions for bridge scour forecast based on monitoring data. Transport. Res. Rec. 2675, 745–763.
Yousefpour, N., Medina-Cetina, Z., Hernandez-Martinez, F.G., Al-Tabbaa, A., 2021a. Stiffness and strength of stabilized organic soils – Part II/II: parametric analysis and modeling with machine learning. Geosciences 11, 218.
Yu, H., Zhou, H., Sheil, B., Liu, H., 2022. Finite element modelling of helical pile installation and its influence on uplift capacity in strain softening clay. Can. Geotech. J. 59, 2050–2066.
Yu, H.S., Schnaid, F., Collins, I.F., 1996. Analysis of cone pressuremeter tests in sands. J. Geotech. Eng. 122, 623–632.
Zhang, N., Arroyo, M., Ciantia, M.O., Gens, A., Butlanska, J., 2019. Standard penetration testing in a virtual calibration chamber. Comput. Geotech. 111, 277–289.
Zhang, P., 2019. A novel feature selection method based on global sensitivity analysis with application in machine learning-based prediction model. Appl. Soft Comput. 85, 105859.
Zhang, P., Yin, Z.Y., Jin, Y.F., 2021. State-of-the-art review of machine learning applications in constitutive modeling of soils. Arch. Comput. Methods Eng. 28, 3661–3686.
Zhang, P., Yin, Z.Y., Zheng, Y., Gao, F.P., 2020a. A LSTM surrogate modelling approach for caisson foundations. Ocean Eng. 204, 107263.

Zhang, P., Yin, Z.Y., Jin, Y.F., Ye, G.L., 2020b. An AI-based model for describing cyclic characteristics of granular materials. Int. J. Numer. Anal. Methods GeoMech. 44, 1315–1335.

Dr. Fengwen Lai is currently an Assistant Professor in the College of Civil Engineering at Fuzhou University, China. He obtained his PhD degree at Southeast University in July 2023. From November 2021 to February 2024, he worked as a researcher at the Geo-Section of the Delft University of Technology. From March 2022 to May 2022, he undertook a visit to the Institute of Soil Mechanics, Foundation Engineering and Computational Geotechnics of the Graz University of Technology. His background is in analytical modeling – standard finite element method – filed monitoring. His research interests lie in the field of soil–structure interaction (SSI), which encompasses a range of problems involving large displacements in various contexts, including off- and on-shore geotechnical engineering, deep excavations/tunneling, retaining structures, and trapdoor problems. His research has been published in more than 30 peer-reviewed journals specializing in geotechnical engineering, including *Géotechnique*, *Canadian Geotechnical Journal*, *Journal of Rock Mechanics and Geotechnical Engineering*, *Tunneling and Underground Space Technology*, and *Computer and Geotechnics*.