BiLSTM-based Soil-Structure Interface Modelling

Pin ZHANG[1], Yi YANG[2,*] and Zhen-Yu YIN[3]

Abstract: Deep learning algorithm bidirectional long-short term memory (BiLSTM) neural network is employed to model behaviors of the soil-structure interface in this study, as a pioneer research work to investigate the feasibility of using DL to model interface behaviors. Datasets are collected from 12 constant normal stress and 14 constant normal stiffness sand-structure interface tests. A modelling framework with the integration of BiLSTM is thereafter proposed. The results indicate the BiLSTM-based model can accurately capture the responses of interface behaviors including volumetric dilatancy and strain hardening on the dense samples, and volumetric and strain softening on the loose samples, respectively. The effects of surface roughness, soil relative density and normal stiffness on the interface behaviors are also investigated using the BiLSTM-based model. The predicted normal stress, shear stress and normal displacement show good agreement with measured results.

Keywords: Deep learning; BiLSTM; interface; soil-structure interaction; sand; constitutive relation

[1] PhD candidate, Department of Civil and Environmental Engineering, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong, China; Email: pin-cee.zhang@connect.polyu.hk

[2] Assistant Professor, Department of Civil Engineering, Chu Hai College of Higher Education, Tuen Mun, N.T. Hong Kong, China; Email: yyang@chuhai.edu.hk

[3] Associate Professor, Department of Civil and Environmental Engineering, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong, China; Email: zhenyu.yin@polyu.edu.hk

[*] Correspondence to Dr. Yi YANG, Tel: +852 2972 7311; Fax: +852 2972 7324; Email: yyang@chuhai.edu.hk

Introduction

A fundamental understanding of the strength and deformation behaviors of the soil-structure interface is crucial for the geotechnical design such as piles, retaining walls and soil nails(Jin et al. 2019; Potts and Day 1999; Ye et al. 2017; Zhou et al. 2020). The implementation of a shearing test on the soil-steel interface is an extensively applied method to investigate the responses of the interface. The involved tests include direct shear test, ring shear test, ring torsion shear test and simple shear test, and the characteristics of such interface shearing tests have been summarized by Kishida and Uesugi (1987). In particular, simple shear testing apparatus has been extensively employed on the interface test due to its simplicity (Evgin and Fakharian 1996).

Numerous experimental tests have been conducted to investigate the effects of various factors on the responses of the soil-structure interface, such as surface roughness, surface hardness, particle morphology and gradation (Eid et al. 2015; Han et al. 2018; Hu and Pu 2004). With the development of computer hardware, the discrete element method (DEM) has been applied to investigate the microscopic behaviors of particles, such as rotation, breakage and fabric anisotropy, and provides a method for deeply explaining the macroscopic phenomena of the interface (Huang et al. 2019; Jing et al. 2018). Two failure modes were summarized based on the experimental and numerical observations, which is the elastic perfect-plastic failure mode along with the smooth interface and the strain localization including strain softening and dilatancy along with the rough interface (Hu and Pu 2004). Accordingly, the linear elastic constitutive model and non-linear constitutive model with a hyperbolic stress-strain relationship in the normal and tangential directions were developed to simulate the responses of the interface (D’Aguiar et al. 2011). A fraction of such constitutive models has been successfully implemented into finite element code for the simulation of geotechnical engineering (D’Aguiar et al. 2011; Stutz et al. 2017). The analysis methods associated with conventional constitutive models are limited to a fixed framework, that is, the assumed behaviors of the interface are ideal and cannot escape from the given mode. Meanwhile, excessive knowledge of domain experts is required to calibrate the parameters of constitutive models.

Machine learning (ML) has recently gained much attention to model mechanical behaviors of soils owning to its excellent capability of solving complex nonlinear problems with the interaction of high-dimensional parameters (Chen et al. 2019; Zhang et al. 2019; Zhang et al. 2020). In comparison to the physics-driven constitutive model, the ML-driven model directly learns mechanical behaviors of soils from the raw data without making any assumptions. Moreover, the application scopes and accuracy of the ML-driven model can be improved with the increasing number of datasets, and parameters are not required to be calibrated for the ML-driven model (Feng et al. 2002; Gao et al. 2016; Zhang et al. 2020). Various ML algorithms such as support vector machine (Kohestani and Hassanlourad 2016; Zhao et al. 2014), genetic algorithms (Cabalar and Cevik 2011) and evolutionary polynomial regression (Cuisinier et al. 2013; Javadi et al. 2012) have been applied to simulate the mechanical behaviors of soils. Besides, deep learning (DL) algorithms with more complex topology and stronger representation learning capability have recently been successfully employed to reproduce more complicated behaviors of soils, such as cyclic responses (Zhang et al. 2020) and multiscale hydro-mechanical coupling responses (Wang and Sun 2018). Nevertheless, behaviors of the soil-structure interface have not been modelled using ML or DL algorithms, thereby the feasibility and efficiency of such algorithms for this issue deserve to be investigated.

This study aims to use the DL algorithm to model the behaviors of the soil-structure interface. A comprehensive evaluation with qualitative and quantitative analysis of the performance of the DL-based model is implemented. To examine the generalization ability of the DL-based model, the effects of surface roughness, relative density of soils and normal stiffness on the responses of the interface are investigated using DL.

Modelling framework using deep learning

Applied methodology

Recurrent neural network (RNN) is characterized by a cyclic connection topology, which means the data conveys along with two directions: vertical flow from input to output layers and horizontal flow between hidden layers. Therefore, the output at the time step t is affected by both current input and previous information stored in the hidden neurons. Such characteristic is more suitable for modelling mechanical behaviors of soils than other ML algorithms (Gorji et al. 2020; Wang and Sun 2018; Zhang et al. 2020), because the responses of soils at a given phase are related to the current status and the previous stress or strain history. Nevertheless, conventional RNN merely learns the information stored in the nearby units while the information stored in the far units is discarded. In other words, RNN only learns the short-term history information. Moreover, the back-propagated gradients of the conventional RNN suffer from either increase or decrease at each time step, resulting in exploding or vanishing gradients (LeCun et al. 2015). To overcome such issues, a memory cell with an entity termed as "gate" is proposed. Thereafter, a series of algorithms motivated by the "gate" mechanism have been proposed and successfully applied in many domains (Zhang et al. 2020), such as long short-term memory (LSTM) (Hochreiter and Schmidhuber 1997), bidirectional LSTM (BiLSTM) (Graves and Schmidhuber 2005) and gated recurrent unit (GRU) (Cho et al. 2014).

Herein, GRU refines the memory cell of LSTM and forms a simpler architecture, thereby the implementation of GRU is easier with a less computational cost. There is no convincing evidence to evaluate the superiority of various algorithms. The memory cell of BiLSTM is the same as LSTM, but BiLSTM utilizes both positive and reverses sequential information by concatenating the outputs of hidden layers. Therefore, BiLSTM enhances the interaction of sequential datasets, but the number of weights and biases increases two times in comparison with LSTM. Considering the datasets and features used in soil constitutive modelling are limited, the complexity of the framework of DL-based constitutive model is acceptable, thereby the effect of an increasing number of weights and biases on the computational efficiency can be neglected. To this end, BiLSTM is ultimately employed to simulate the behaviors of the soil-structure interface in this study.

Modelling strategy

The selection of features determines what factors can be considered by the BiLSTM -based model, thereby it is related to the application scopes of the BiLSTM -based model. To improve the generalization ability of the DL-based model, the features are required to involve inherent properties of the studied object, state parameters and historical information of stress or strain. The inherent properties are used to describe the characteristics of soils and structural surfaces, such as normal stiffness k_n, relative density I_{D0} and normalized roughness R_n. Such features as input parameters ensure that the BiLSTM-based model can be used to investigate behaviors of the interface of various soils and structures. State parameters are employed to represent experimental steps, such as initial normal stress σ_{n0}, shear displacement w and shear displacement increment Δw. The outputs of the DL-based model at the previous time steps are added to the input parameters for considering loading and deformation history. Overall, the input and output parameters are summarized in Table 1 and the BiLSTM-based model for modelling behaviors of the soil-structure interface can be mathematically expressed by:

$$\left(u^{t+1}, \tau^{t+1}, \sigma_n^{t+1}\right) = f\left(u^t, \tau^t, \sigma_n^t, w^t, \Delta w^t, \sigma_{n0}, k_n, I_{D0}, R_n\right) \tag{1}$$

where u^{t+1}, σ_n^{t+1}, τ^{t+1} and Δw^t are the predicted normal displacement, shear stress, normal stress and shear displacement increment at the tth step, respectively; u^t, σ_n^t, τ^t and w^t are the normal displacement, shear stress, normal stress and shear displacement at the tth step. Herein, $R_n = R_{max}\,(L = D_{50})/D_{50}$, in which $R_{max}\,(L = D_{50})$ represents the relative height between the highest peak and the lowest valley along with a surface profile over the gauge length D_{50} (diameter through which 50% of sands pass). The value of Δw is assigned beforehand, and it increases with the increasing shear displacement. Therefore, the value of Δw is prescribed as 0.01 ($w \le 0.5$ mm), 0.02 ($0.5 < w \le 1$ mm), 0.05 ($1 < w \le 3$ mm), 0.1 ($3 < w \le 4$ mm), and 0.2 ($w > 4$ mm).

It should be noted that the architecture of the memory cell (see Fig. 1) used for positive and reverse directions in the BiLSTM is the same, but the values of weights and biases are different. ${}^f\mathbf{W}$, ${}^f\mathbf{U}$ and ${}^f\boldsymbol{b}$ are the weights matrices and biases vector used for the positive data flow, and ${}^b\mathbf{W}$, ${}^b\mathbf{U}$ and ${}^b\boldsymbol{b}$ are used for the reverse data flow, as presented in Fig. 1. Given a set of input parameters $\boldsymbol{x} = [u^t, \sigma_n^t, \tau^t, w^t, \Delta w^t, \sigma_{n0}, k_n, I_{D0},$

R_n] at the tth step, the calculation principle of the BiLSTM-based model with two hidden layers is introduced. Herein, the calculation of the positive data flow is introduced for revealing the mechanism of BiLSTM, whereas the calculation of the reverse data flow is dismissed for brevity, because the calculation of data flow for both directions is similar except the used weights matrices and bias vector. results of the forget, input and output gates are obtained by:

(1) First, the input data $\boldsymbol{x}^t$ pass through the first hidden layer, and the outputs at the forget, input and output gates can be obtained using:

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

$$ {}^{f}\boldsymbol{i}^{t} = \sigma\left({}^{f}\mathbf{W}_{i}\boldsymbol{x}^{t} + {}^{f}\mathbf{U}_{i}\, {}^{f}\boldsymbol{h}^{t-1} + {}^{f}\boldsymbol{b}_{i} \right) \tag{3} $$

$$ {}^{f}\boldsymbol{o}^{t} = \sigma\left({}^{f}\mathbf{W}_{o}\boldsymbol{x}^{t} + {}^{f}\mathbf{U}_{o}\, {}^{f}\boldsymbol{h}^{t-1} + {}^{f}\boldsymbol{b}_{o} \right) \tag{4} $$

where subscript f, i and o denote the matric weight and bias vectors used in the forget, input and output gates, respectively. The result of memory is thereafter obtained by:

$$ {}^{f}\tilde{\boldsymbol{c}}^{t} = tanh\left({}^{f}\mathbf{W}_{c}\boldsymbol{x}^{t} + {}^{f}\mathbf{U}_{c}\, {}^{f}\boldsymbol{h}^{t-1} + {}^{f}\boldsymbol{b}_{c} \right) \tag{5} $$

$$ {}^{f}\boldsymbol{c}^{t} = {}^{f}\boldsymbol{f}^{t} \odot {}^{f}\boldsymbol{c}^{t+1} + {}^{f}\boldsymbol{i}^{t} \odot {}^{f}\tilde{\boldsymbol{c}}^{t} \tag{6} $$

where $\odot$ denotes the element-wise product. Next, the output of the hidden layer is obtained by:

$$ {}^{f}\boldsymbol{h}^{t} = {}^{f}\boldsymbol{o}^{t} \odot tanh\left({}^{f}\boldsymbol{c}^{t} \right) \tag{7} $$

Similar to the calculation of the positive data flow, the output of the hidden layer for the reverse data flow ${}^{b}\boldsymbol{h}^{t}$ can also be obtained using Eqs. (2)–(7) as long as replacing the ${}^{f}\mathbf{W}$, ${}^{f}\mathbf{U}$ and ${}^{f}\boldsymbol{b}$ with the ${}^{b}\mathbf{W}$, ${}^{b}\mathbf{U}$ and ${}^{b}\boldsymbol{b}$. The ultimate output of the first hidden layer is obtained by

$$ \boldsymbol{h}_{1}^{t} = {}^{f}\boldsymbol{h}^{t} \oplus {}^{b}\boldsymbol{h}^{t} \tag{8} $$

where $\oplus$ denotes concatenation operation.

(2) Second, from the first hidden layer to the second hidden layer. The calculation at the second layer still uses the same equations with the first layer, i.e., from Eqs. (2)–(8) as long as replacing the $\boldsymbol{x}_t$ with the $\boldsymbol{h}_1^t$ as the input data at the second hidden layer and using the output of hidden layer from the (t–1)th step, weights matrics and biases vectors at this layer. Finally, the output $\boldsymbol{h}_2^t$ can be obtained.

(3) Finally, from the second hidden layer to the output layer. The final output of the BiLSTM-based model at the tth step is obtained by:

$$\boldsymbol{y} = \mathbf{W} \times \boldsymbol{h}_2^t + \boldsymbol{b} \tag{9}$$

where $\mathbf{W}$ and $\boldsymbol{b}$ are the weights and biases used in the output layer. The linear activation function is used. Following Eqs. (2)–(9), the mathematic expression of the BiLSTM-based model is established.

Database

Data source

The datasets used in this study refer to Praai (2013). The results of 12 constant normal stress (CNL) tests and 20 constant normal stiffness (CNS) tests on the standard Fontainebleau sand were collected, and its properties are presented in Table 2. Such tests involve the responses of the soil-steel interface under different relative densities of soil, normal stresses, normal stiffness and surface roughness, as shown in Table 3, which are sufficient enough to develop the BiLSTM-based model and evaluate its generalization ability. However, the data points of each test are relatively limited and are interfered with by experimental and measurement errors. Numerous datasets with useful information are the basis for developing the DL-based model. The raw experimental datasets are thus preprocessed.

Data preprocessing

As presented in the Eq. [8], after the value of Δw is prescribed, the relationships of w–u, w–σ and w–τ can be interpolated for increasing the number of datasets, meanwhile, the remaining input parameters maintain constant values. Herein, the piecewise cubic Hermitian interpolation polynomials (PCHIP) method is first used, considering such an interpolation method does not change the shape of the raw relationship curve

(Moler 2004). Next, the Savitzky-Golay filter is selected to denoise the experimental data, because it can smoothen the data without distorting the tendency of original data (Savitzky and Golay 1964). Fig. 2 presents the results of data pre-processing on a representative experiment. It can be seen from the curves that the fidelity of raw data is maintained and meanwhile the noise is dismissed. Such factors improve the quality and quantity of datasets, thereby ensuring that the useful information can be learned by the BiLSTM. Therefore, 24 experimental tests with 3063 datasets and 8 experimental tests with 1064 datasets are used as training and testing sets, respectively.

To eliminate the effect of scale difference of input parameters on the training process, all datasets are normalized using the Min-Max scaling method and mapped into the range (–1, 1).

$$x_{norm} = \frac{x - x_{\min}}{x_{\max} - x_{\min}}\left(\bar{x}_{\max} - \bar{x}_{\min}\right) + \bar{x}_{\min} \tag{10}$$

where $x_{\max}$ and $x_{\min}$ are the maximum and minimum values of the parameter x, respectively; $\bar{x}_{\max}$ and $\bar{x}_{\min}$ are the 1 and −1, respectively.

Modelling results of BiLSTM-based model

Training of the BiLSTM-based model

To detect the overfitting of the BiLSTM, the 10-fold cross-validation method is used. The original training set is randomly divided into 10 subsets, the model is thus trained 10 times with the same initial weights and biases at each training epoch. At each time, the model is trained based on 9 random subsets, and is tested on the 1 remaining subset. The ultimately propagated error at each epoch is the mean square error (MSE) on the 10 validation subsets rather than merely on one testing set. Because the excellent performance of a DL-based model on one test set may be just coincident, the model performance evaluated based on one testing set may lead to misunderstanding of the model performance. The results of 10-fold cross-validation method do not rely on the spilt results of training and testing sets, thereby it can eliminate such issue and give a fair evaluation of the model performance. In this way, the BiLSTM-based model with the integration of the 10-fold cross-validation is much more robust and the overfitting issue can also be detected.

Mean square error (MSE) is used for evaluating the difference of predicted and measured results during the training process, thereby the ultimate loss function can be expressed using

$$\text{Loss function} = \frac{1}{10n}\sum_{i=1}^{n}\left(y_i^p - y_i^m\right)^2 \tag{11}$$

where n is the total number of datasets; 10 is the number of folds; y_i^p and y_i^m are predicted and measured values, respectively.

Based on the prescribed loss function, the training process is activated for developing the BiLSTM-based model. The configurations of the BiLSTM-based model are determined using the trial-and-error method, and the results are presented in Table 4. The BiLSTM-based model with four layers (one input layer, two hidden layers with memory cell and one output dense layer) shows optimum performance. The optimum number of hidden neurons in each hidden layer is identified as 60. The activation function used in the hidden layer is *ReLU*, meanwhile, the linear activation function is applied in the output layer. The *Adam* optimizer is used to update the weights and biases of the BiLSTM, which makes use of the advantages of AdaGrad and RMSProp (Kingma 2015) and has been extensively used in many domains. The batch size determines the number of datasets to be fed to BiLSTM for training at each round. Considering the number of datasets after data pre-processing in each experiment is roughly identical to 120, the batch size is thus set as 120 so that ensures BiLSTM can learn the entire information of an entire experiment test at each round. The 200 epochs are large enough to guarantee the convergence of training. The final loss value generated during the training process is presented in Fig. 3. It can be seen that the loss value is convergent on both training and testing sets.

One of the important parameters in BiLSTM is the time step, which determines the span of the history information. The additional history information can improve the learning capability of the BiLSTM-based model, but too long history information also degrades the learning efficiency and causes overfitting. To select the appropriate time step, the performance of the BiLSTM-based model with the time step from 1 to 5 is investigated, and the corresponding training process is presented in Fig. 4. It should be noted that the performance of the BiLSTM-based model is dramatically affected by the initial weights and biases. To fairly evaluate the performance of the BiLSTM-based model, the model with each time step is trained 10 times with different weights and biases. Therefore, 10 MSE values are generated at each epoch, and they are represented using a boxplot. It can be observed in Fig. 4 that the error on the training set decreases with the increasing time step as expected, and the convergence rate increases. However, the overfitting issue appears as the time step increases to 4, causing the larger error on the validation set than that generated by the model with the time step less than 3. As the time step is 1, the MSE decreases to the minimum value with the epoch reaching 200, thereafter the increasing epoch leads to the increase in the MSE value, which is attributed to under-fitting. As the time step exceeds 300, similar status is observed in the model with the time step of 2. The model with the time step of 3 presents stable convergence on the validation set, and the MSE value on the training set is also acceptable. Therefore, the optimum time step of the BiLSTM-based model is identified as 3 in this study, which means that the current behavior of the interface is affected by the stress or strain status of the previous three steps.

Performance of the BiLSTM-based model

Fig. 5 presents the scatter plots of the predicted and measured normal stress, shear stress and normal displacement. The predicted results on both training and testing sets show excellent agreement with the measured results, and all data points are close to the line with the slope of 1. To quantitatively evaluate the performance of the BiLSTM-based model. Absolute and relative error indicators, i.e., mean absolute error (MAE) and mean absolute percentage error (MAPE), are calculated.

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}\left|y_i^p - y_i^m\right| \tag{12}$$

$$\mathrm{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i^p - y_i^m}{y_i^m}\right| \times 100\% \tag{13}$$

As presented in Fig. 5, the MAE and MAPE values for the three outputs on both training and testing are small, and the errors on the testing set are roughly three times larger than that generated on the training sets. The MAPE value for the predicted shear displacement is much larger than the remaining two outputs, which is attributed to the small value of the measured shear displacement. The MAE value is indiscernible, thereby the prediction accuracy for the shear displacement is guaranteed. The results of error analysis indicate that the BiLSTM-based model exhibits high accuracy and strong generalization ability on unknown datasets.

To further reveal the performance of the BiLSTM-based model, Fig. 6 presents the predicted relationships of w–u, w–σ and w–τ on each testing set. The predicted relationships on the training set are not presented for brevity, because they are better than the results on the testing set. Fig. 6 presents the predicted and measured behaviours of the interface under the CNL test. Regarding the rough interface with the dense sample, it can be seen from Fig. 6(a) that the shear stress increases with the increasing shear relative displacement until reaching the peak value, thereafter decreased to residual shear stress. In the case of the loose sample, the slight softening is observed. In Fig. 6(b), the dilative behavior of dense samples can be clearly observed whereas the volumetric contraction occurs on the loose sample. Regarding the smooth interface, the softening behavior on both dense or loose samples is negligible. The peak and residual shear stress decrease to 60%–70% of values generated on the rough interface, because the less dilative and contractive behaviors on the dense and loose samples are observed, respectively.

The predicted results under the CNS test for the rough interface with the normal stiffness of 2000 kPa/mm are presented in Figs. 7a–7c. When the sample is subjected to normal stiffness, behaviors of the interface are changed dramatically, but the BiLSTM-based model still accurately identifies such behaviors. As shown in Figs. 7(b) and (c), on the dense sample, the BiLSTM-based model can predict the softening and dilative behaviors. On the loose sample, at the beginning of the shearing phase, the interface contracts and consequently leads to the significant degradation of the normal stress. During the shearing phase, the interface continuously contracts and the normal stress degrades as well as shear stress (see Fig. 7(a)).

Figs. 7d–7f present the behaviors of the smooth interface under the CNS condition for the dense sand sample. The observed trends differ from the responses in the rough interface test. The dilative behavior significantly mitigates as well as the variation of stress states. The increase of normal stress can only be observed in the case of k_n = 5000 kPa/mm (see Fig. 7(d)), and the strain-softening does not occur (see Fig. 7(e)) during the shearing phase. The normal displacement roughly maintains constant (see Fig. 7(f)). Overall, under the CNS condition, the behaviors of smooth interface dramatically differ from the rough interface, and the BiLSTM-based model can still accurately capture such responses.

Overall, from the perspective of the predicted results, the relationships of w–u, w–σ and w–τ of the soil-structure interface with different surface roughness, relative density and normal stiffness can be accurately predicted using the BiLSTM-based model, and outperform the modelling results using the theoretical formulations presented by Praai (2013).

Conclusions

A bidirectional long-short term memory (BiLSTM) neural network-based model for investigating the behaviors of the soil-structure interface was proposed in this study, as a pioneer research work to investigate the feasibility of the DL algorithm to model interface behaviors. BiLSTM is characterized by the sequence prediction capability and all simulation results indicated the BiLSTM-based model was suitable for modelling behaviors of soil-structure interface with small prediction errors. Meanwhile, the BiLSTM-based model can accurately capture responses of behaviors of the soil-structure interface, such as volumetric dilatancy and strain hardening on the dense samples, and volumetric and strain softening on loose samples, respectively. The results generated by the BiLSTM-based model are more accurate than that generated by the theoretical formulations. Meanwhile, the effects of surface roughness, the relative density of soil and normal stiffness on the interface behaviors can also be accurately captured by the same BiLSTM-based model.

Data Availability Statement

All data used during the study are available from the corresponding author by request.

Acknowledgements

This research was financially supported by the Research Grants Council (RGC) of Hong Kong Special Administrative Region Government (HKSARG) of China (Grant No.: UGC/FDS13/E02/20, 15217220, N_PolyU534/20).

References

Cabalar, A. F., and Cevik, A. (2011). "Triaxial behavior of sand–mica mixtures using genetic programming." *Expert Syst. Appl.*, 38 (8), 10358-10367.

Chen, R. P., Zhang, P., Kang, X., Zhong, Z. Q., Liu, Y., and Wu, H. N. (2019). "Prediction of maximum surface settlement caused by EPB shield tunneling with ANN methods." *Soils Found.*, 59 (2), 284–295.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, v., Bougares, F., Schwenk, H., and Bengio, Y. (2014). "Learning phrase representations using RNN Encoder–Decoder for statistical machine translation." *arXiv:1406.1078v3*.

Cuisinier, O., Javadi, A. A., Ahangar-Asr, A., and Masrouri, F. (2013). "Identification of coupling parameters between shear strength behaviour of compacted soils and chemical's effects with an evolutionary-based data mining technique." *Comput. Geotech.*, 48, 107-116.

D'Aguiar, S. C., Modaressi-Farahmand-Razavi, A., dos Santos, J. A., and Lopez-Caballero, F. (2011). "Elastoplastic constitutive modelling of soil–structure interfaces under monotonic and cyclic loading." *Comput. Geotech.*, 38 (4), 430-447.

Eid, H. T., Amarasinghe, R. S., Rabie, K. H., and Wijewickreme, D. (2015). "Residual shear strength of fine-grained soils and soil–solid interfaces at low effective normal stresses." *Can. Geotech. J.*, 52, 198–210.

Evgin, E., and Fakharian, K. (1996). "Effect of stress paths on the behavior of sand–steel interfaces." *Can. Geotech. J.*, 33 (6), 853-865.

Feng, X. T., Li, S. J., Liao, H. J., and Yang, C. X. (2002). "Identification of non-linear stress-strain-time relationship of soils using genetic algorithm." *Int. J. Numer. Anal. Methods Geomech.*, 26 (8), 815-830.

Gao, W., Ge, M., Chen, D., and Wang, X. (2016). "Back analysis for rock model surrounding underground roadways in coal mine based on black hole algorithm." *Eng. Comput-Germany*, 32 (4), 675-689.

Gorji, M. B., Mozaffar, M., Heidenreich, J. N., Cao, J., and Mohr, D. (2020). "On the potential of recurrent neural networks for modeling path dependent plasticity." *J. Mech. Phys. Solids*, 143, 103972.

Graves, A., and Schmidhuber, J. (2005). "Framewise phoneme classification with bidirectional LSTM and other neural network architectures." *Neural Netw.*, 18 (5-6), 602-610.

Han, F., Ganju, E., Salgado, R., and Prezzi, M. (2018). "Effects of Interface Roughness, Particle Geometry, and Gradation on the Sand–Steel Interface Friction Angle." *J. Geotech. Geoenviron. Eng.*, 144 (12).

Hochreiter, S., and Schmidhuber, J. (1997). "Long short-term memory." *Neural Comput.*, 9, 1735–1780.

Hu, L., and Pu, J. (2004). "Testing and modeling of soil-structure interface." *J. Geotech. Geoenviron. Eng.*, 130 (8), 851-860.

Huang, M., Chen, Y., and Gu, X. (2019). "Discrete element modeling of soil-structure interface behavior under cyclic loading." *Comput. Geotech.*, 107, 14-24.

Javadi, A. A., Faramarzi, A., and Ahangar-Asr, A. (2012). "Analysis of behaviour of soils under cyclic loading using EPR-based finite element method." *Finite Elem. Anal. Des.*, 58, 53-65.

Jin, Y.-F., Yin, Z.-Y., Zhou, W.-H., and Huang, H.-W. (2019). "Multi-objective optimization-based updating of predictions during excavation." *Eng. Appl. Artif. Intel.*, 78, 102-123.

Jing, X. Y., Zhou, W. H., Zhu, H. X., Yin, Z. Y., and Li, Y. M. (2018). "Analysis of soil-structural interface behavior using three-dimensional DEM simulations." *Int. J. Numer. Anal. Methods Geomech.*, 42 (2), 339-357.

Kingma, D. P. (2015). "Adam: a method for stochastic optimization." *International Conference on Learning Representations*, San Diego, CA.

Kishida, H., and Uesugi, M. (1987). "Tests of the interface between sand and steel in the simple shear apparatus." *Géotechnique*, 37 (1), 45-52.

Kohestani, V. R., and Hassanlourad, M. (2016). "Modeling the mechanical behavior of carbonate sands using artificial neural networks and support vector machines." *Int. J. Geomech.*, 16 (1), 04015038.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). "Deep learning." *Nature*, 521, 436-444.

Moler, C. B. (2004). *Numerical computing with MATLAB*, SIAM.

Potts, D. M., and Day, R. A. (1999). "The effect of interface properties on retaining wall behaviour." *Int. J. Numer. Anal. Methods Geomech.*, 22 (12), 1021-1033.

Praai, S. (2013). "Behaviour of soil-structure interfaces subjected to a large number of cycles. Application to piles." Ph.D, Grenoble Alps University, Grenoble, France.

Savitzky, A., and Golay, M. J. E. (1964). "Smoothing and differentiation of data by simplified least squares procedures." *Anal. Chem.*, 36 (8), 1627–1639.

Stutz, H., Mašín, D., Sattari, A. S., and Wuttke, F. (2017). "A general approach to model interfaces using existing soil constitutive models application to hypoplasticity." *Comput. Geotech.*, 87, 115-127.

Wang, K., and Sun, W. (2018). "A multiscale multi-permeability poroplasticity model linked by recursive homogenizations and deep learning." *Comput. Meth. Appl. Mech. Eng.*, 334, 337-380.

Ye, X., Wang, S., Wang, Q., Sloan, S. W., and Sheng, D. (2017). "Numerical and experimental studies of the mechanical behaviour for compaction grouted soil nails in sandy soil." *Comput. Geotech.*, 90, 202-214.

Zhang, P., Chen, R. P., and Wu, H. N. (2019). "Real-time analysis and regulation of EPB shield steering using Random Forest." *Automat. Constr.*, 106, 102860.

Zhang, P., Yin, Z.-Y., Zheng, Y., and Gao, F.-P. (2020). "A LSTM surrogate modelling approach for caisson foundations." *Ocean Eng.*, 204, 107263.

Zhang, P., Yin, Z. Y., Jin, Y. F., and Chan, T. H. T. (2020). "A novel hybrid surrogate intelligent model for creep index prediction based on particle swarm optimization and random forest." *Eng. Geol.*, 265, 105328.

Zhang, P., Yin, Z. Y., Jin, Y. F., and Ye, G. L. (2020). "An AI-based model for describing cyclic characteristics of granular materials." *Int. J. Numer. Anal. Methods Geomech.*, 44 (9), 1315-1335.

Zhao, H., Huang, Z., and Zou, Z. (2014). "Simulating the stress-strain relationship of geomaterials by support vector machine." *Math. Probl. Eng.*, 2014, 1-7.

Zhou, J., Yu, J., Gong, X., El Naggar, M. H., and Zhang, R. (2020). "The effect of cemented soil strength on the frictional capacity of precast concrete pile–cemented soil interface." *Acta Geotech.*, 15, 3271–3282.

Table

Table 1 Input and output parameters/variables

Type	Parameter/variable	Definition
Input	σ_{n0}	Initial normal stress
	k_n	Normal stiffness
	I_{D0}	Relative density
	R_n	Normalized roughness
	Δw^{t+1}	Shear displacement increment
	w^t	Shear displacement
	σ_n^t	Normal stress
	τ^t	Shear stress
	u^t	Normal displacement
Output	σ_n^{t+1}	Normal stress
	τ^{t+1}	Shear stress
	u^{t+1}	Normal displacement

Note: t denotes values of parameters/variables at the step t

Table 2 Properties of standard Fontainebleau sand (after Praai 2013)

d_{50} (mm)	G (g/cm^3)	$\rho_{d,\ max}$ (g/cm^3)	$\rho_{d,\ min}$ (g/cm^3)	e_{max}	e_{min}	C_u
0.23	2.65	1.72	1.42	0.866	0.545	1.72

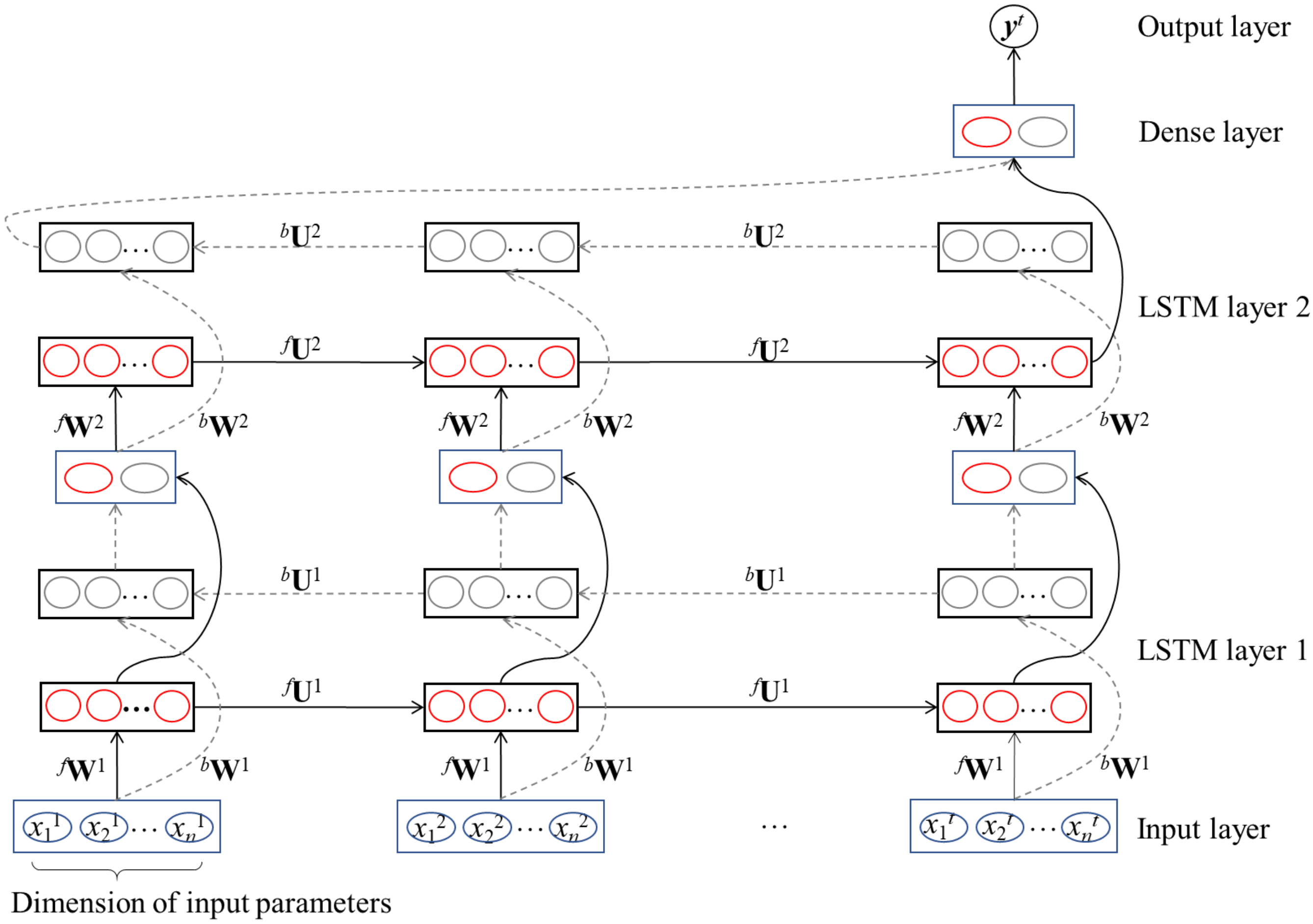
Table 3 Summary of experimental tests

Experiment type	I_{D0}	σ_{n0} (kPa)	k_n (kPa/mm)	R_n
CNL	0.9, 0.3	60, 120, 310	0	0.87 (Rough), 0.06 (Smooth)
CNS	0.9, 0.3	100	1000, 2000, 5000	0.87 (Rough)
	0.9, 0.3	60, 310	1000, 5000	0.87 (Rough)
	0.9	60, 100, 310	1000, 5000	0.06 (Smooth)

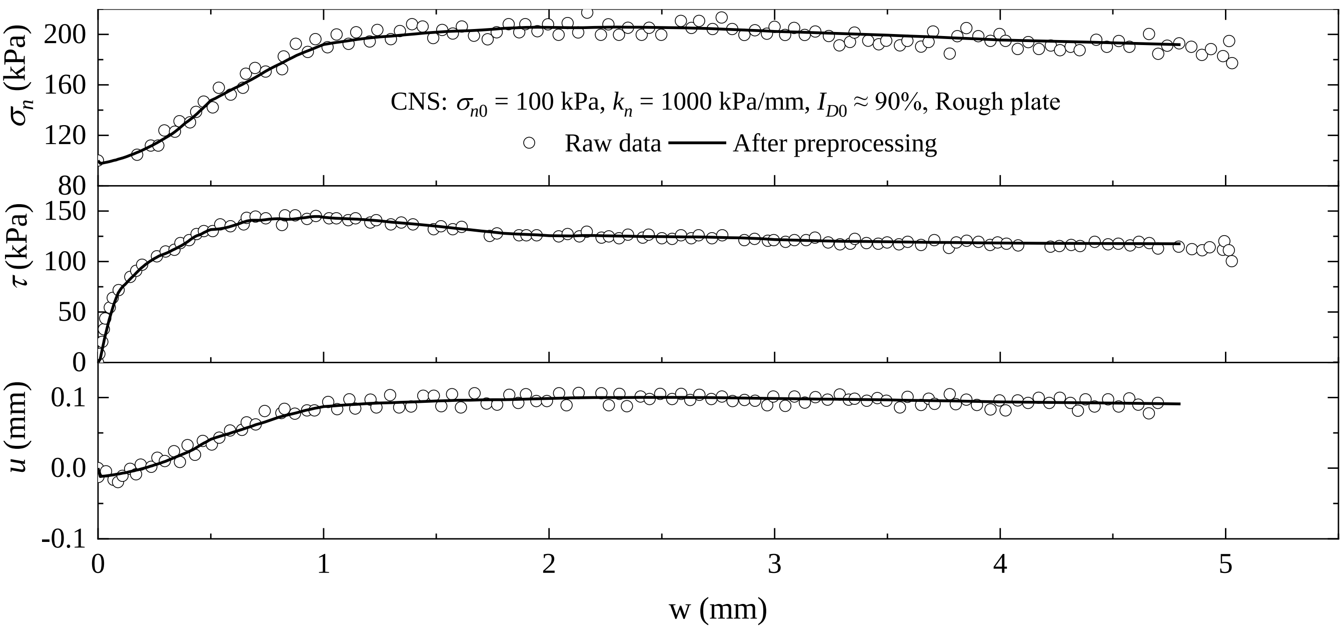
Note: 12 CNL tests and 20 CNS tests

Table 4 Configurations of the BiLSTM based model

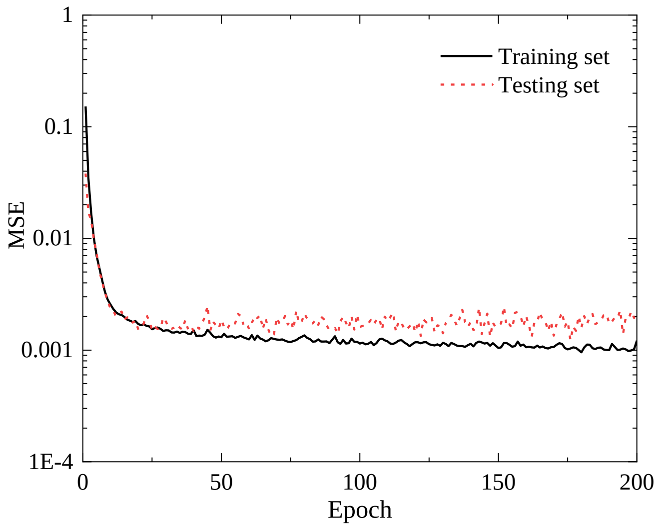
Configuration	value
Architecture	9–60(ReLU)–60(ReLU)–3(linear)
Optimizer	adam
Batch size	120
Epoch	200
Overfitting prevention	10-fold cross-validation
Time step	3

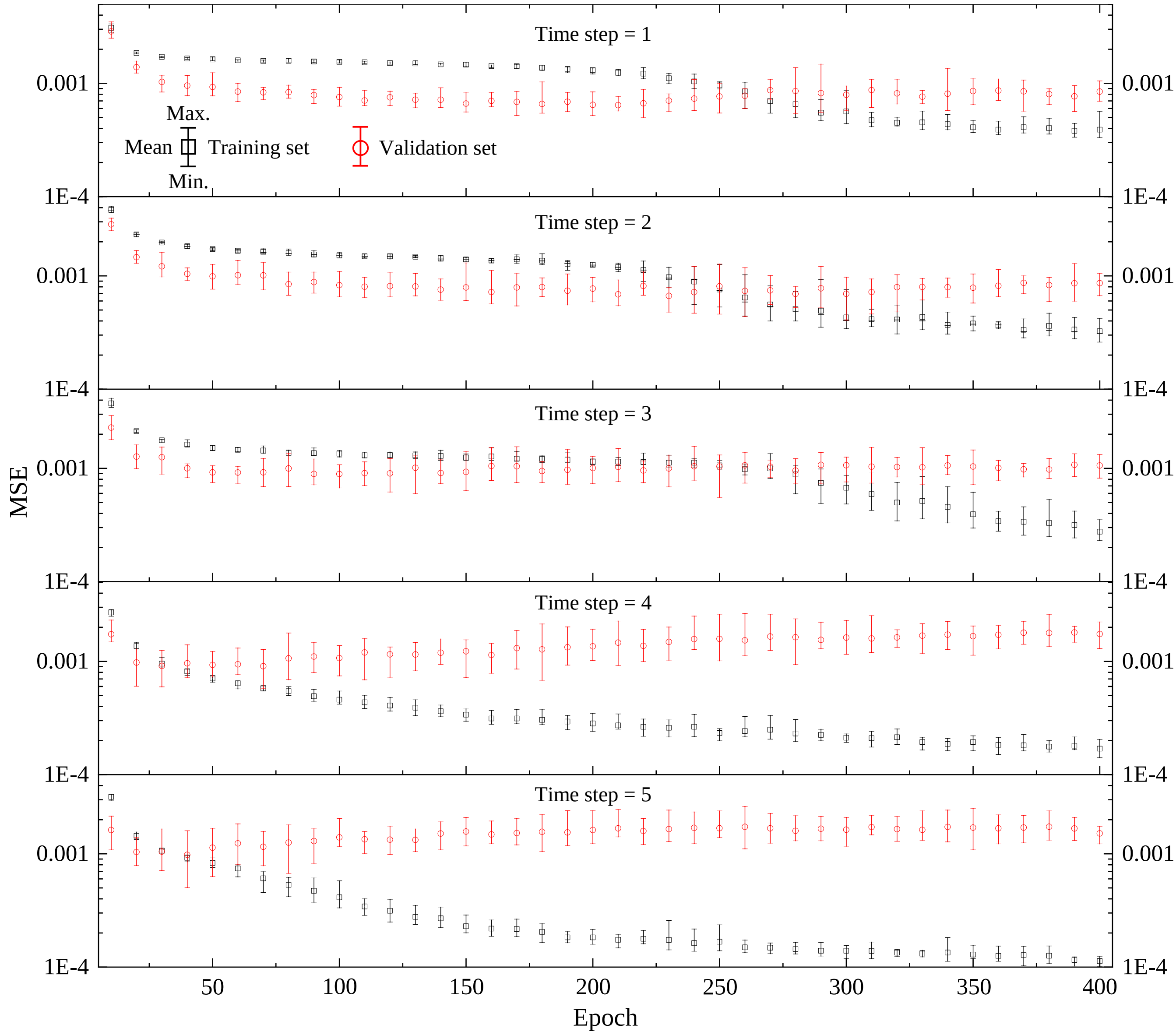


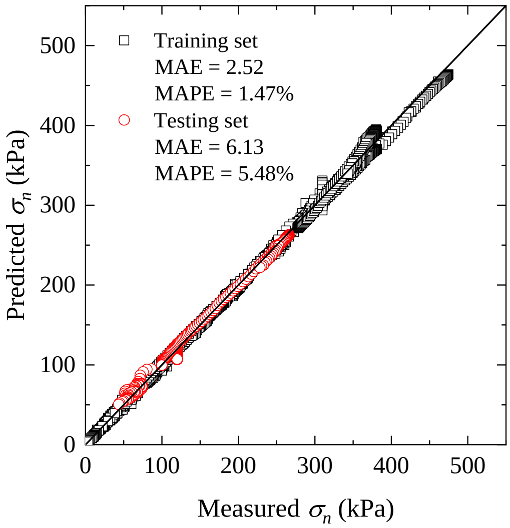
Output layer
Dense layer
LSTM layer 2
LSTM layer 1
Input layer
Dimension of input parameters



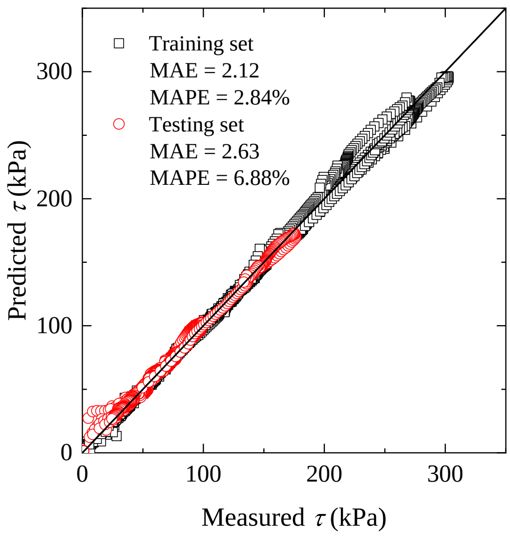
CNS: σ_{n0} = 100 kPa, k_n = 1000 kPa/mm, I_{D0} ≈ 90%, Rough plate
Raw data
After preprocessing
σ_n (kPa)
τ (kPa)
u (mm)
w (mm)

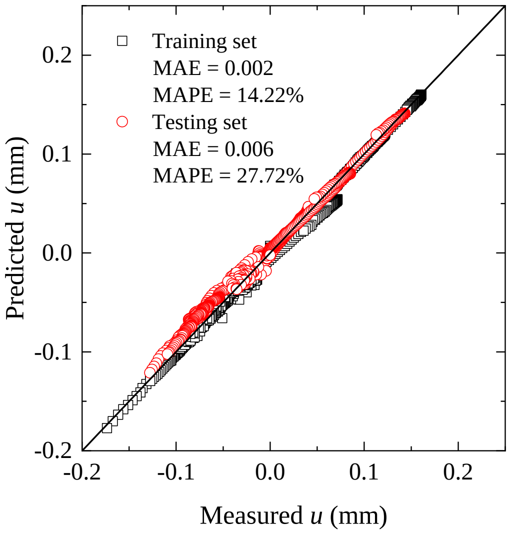

Training set
Testing set
MSE
1
0.1
0.01
0.001
1E-4
0
50
100
150
200
Epoch


Time step = 1
Time step = 2
Time step = 3
Time step = 4
Time step = 5
Max.
Mean
Min.
Training set
Validation set
MSE
Epoch
0.001
1E-4
50
100
150
200
250
300
350
400

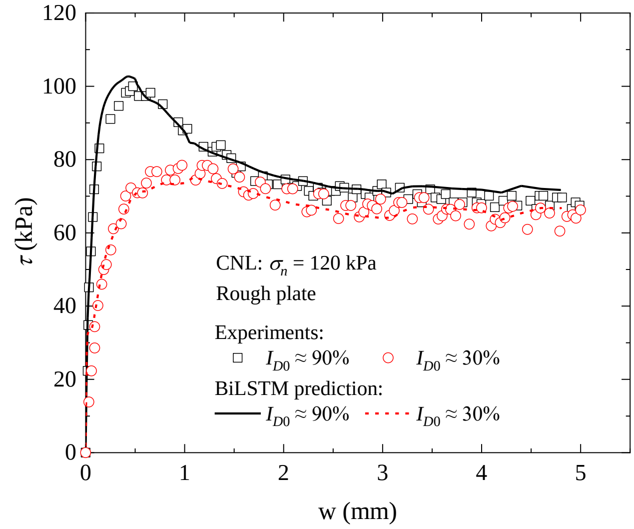


Training set
MAE = 2.52
MAPE = 1.47%
Testing set
MAE = 6.13
MAPE = 5.48%
Predicted σ_n (kPa)
Measured σ_n (kPa)
0
100
200
300
400
500

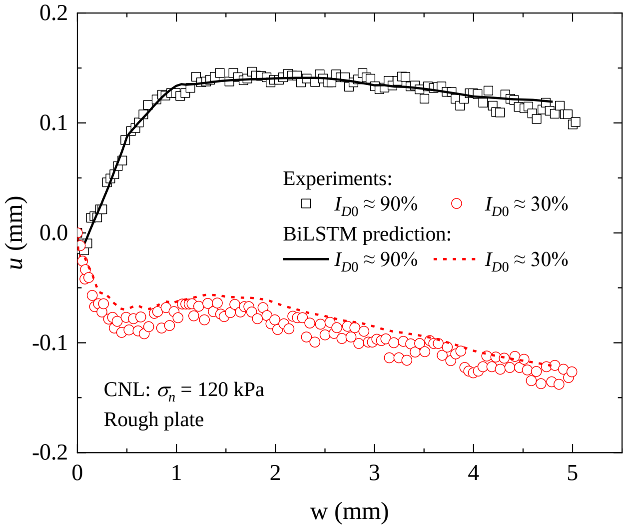

Training set
MAE = 2.12
MAPE = 2.84%
Testing set
MAE = 2.63
MAPE = 6.88%
Predicted τ (kPa)
Measured τ (kPa)
0
100
200
300
0
100
200
300



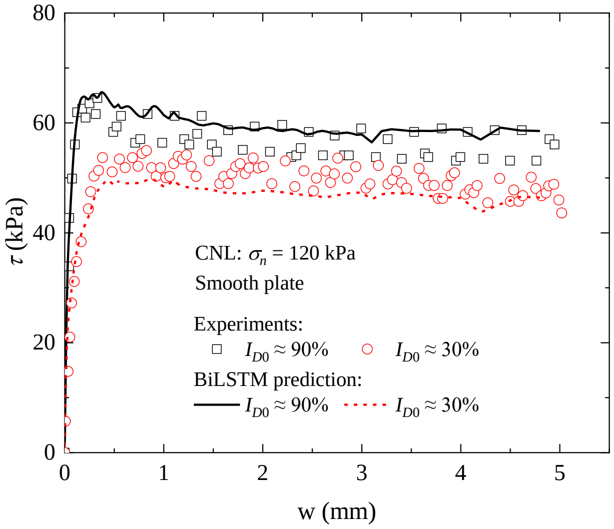
Training set
MAE = 0.002
MAPE = 14.22%
Testing set
MAE = 0.006
MAPE = 27.72%
Predicted u (mm)
Measured u (mm)
0.2
0.1
0.0
-0.1
-0.2
-0.2
-0.1
0.0
0.1
0.2



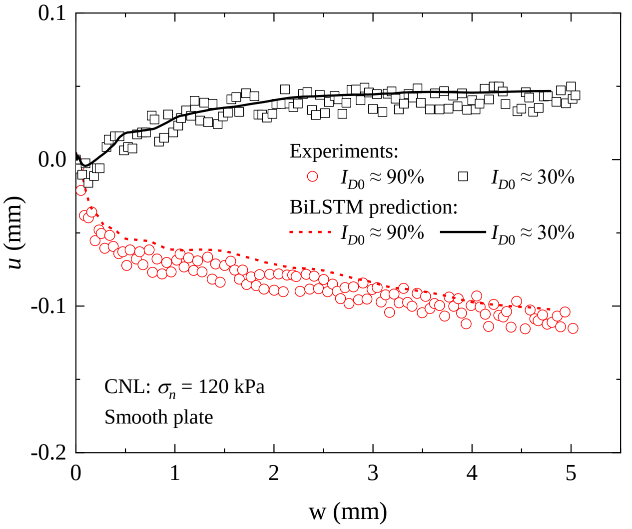
120
100
80
60
40
20
0
τ (kPa)
0
1
2
3
4
5
w (mm)
CNL: σn = 120 kPa
Rough plate
Experiments:
ID0 ≈ 90%
ID0 ≈ 30%
BiLSTM prediction:
ID0 ≈ 90%
ID0 ≈ 30%



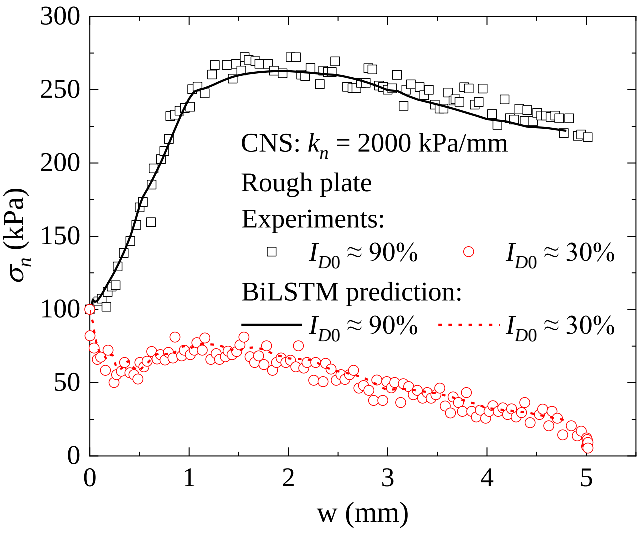
0.2
0.1
0.0
-0.1
-0.2
u (mm)
0
1
2
3
4
5
w (mm)
Experiments:
$I_{D0} \approx 90\%$
$I_{D0} \approx 30\%$
BiLSTM prediction:
$I_{D0} \approx 90\%$
$I_{D0} \approx 30\%$
CNL: σ_n = 120 kPa
Rough plate



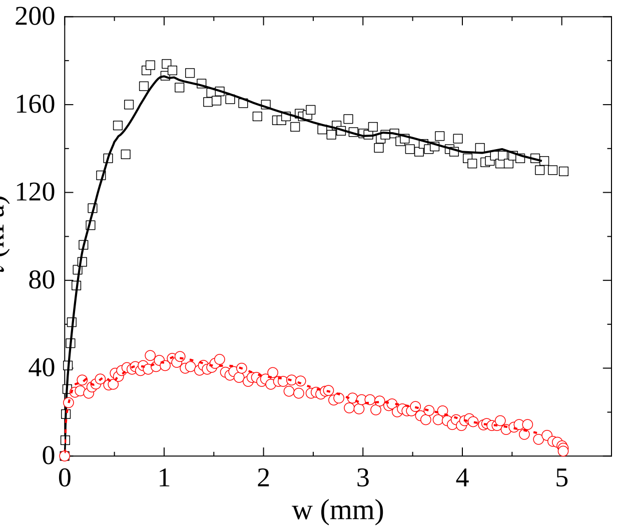
CNL: σ_n = 120 kPa
Smooth plate
Experiments:
$I_{D0} \approx 90\%$
$I_{D0} \approx 30\%$
BiLSTM prediction:
$I_{D0} \approx 90\%$
$I_{D0} \approx 30\%$
τ (kPa)
w (mm)
0
20
40
60
80
0
1
2
3
4
5



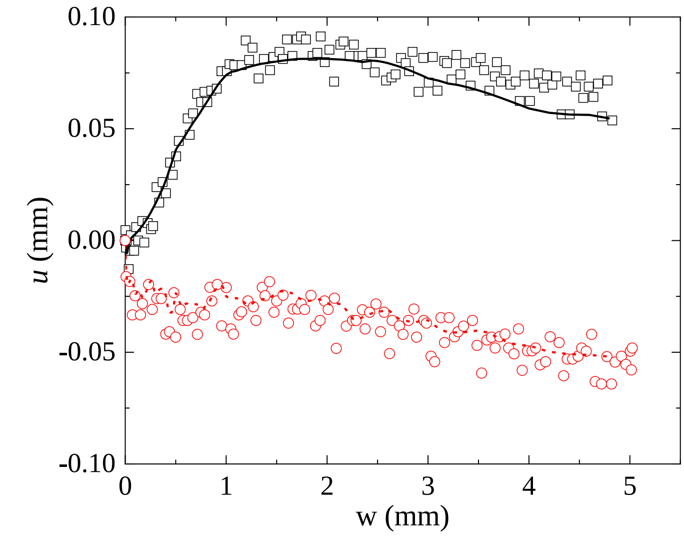
0.1
0.0
-0.1
-0.2
u (mm)
0
1
2
3
4
5
w (mm)
Experiments:
$I_{D0} \approx 90\%$
$I_{D0} \approx 30\%$
BiLSTM prediction:
$I_{D0} \approx 90\%$
$I_{D0} \approx 30\%$
CNL: σ_n = 120 kPa
Smooth plate



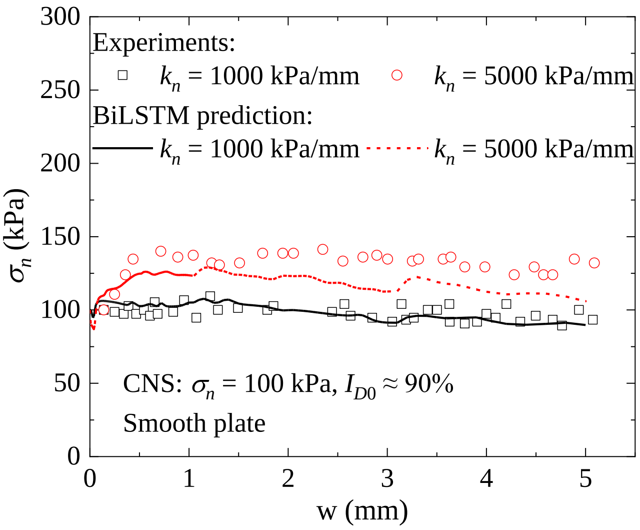
300
250
200
150
100
50
0
σ_n (kPa)
0
1
2
3
4
5
w (mm)
CNS: k_n = 2000 kPa/mm
Rough plate
Experiments:
$I_{D0} \approx 90\%$
$I_{D0} \approx 30\%$
BiLSTM prediction:
$I_{D0} \approx 90\%$
$I_{D0} \approx 30\%$

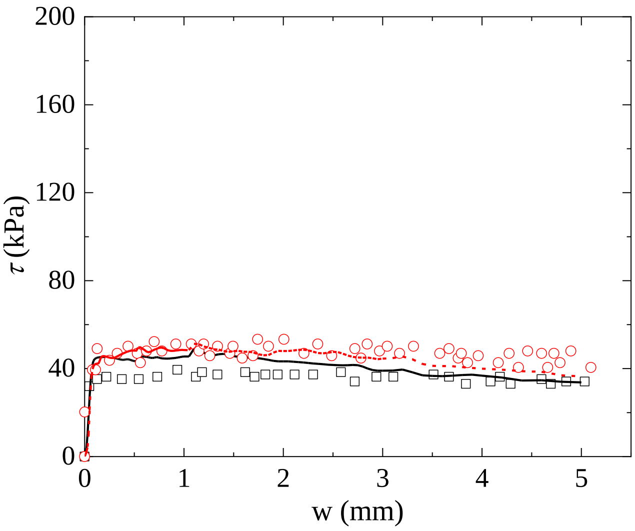


200
160
120
80
40
0
τ (kPa)
0
1
2
3
4
5
w (mm)

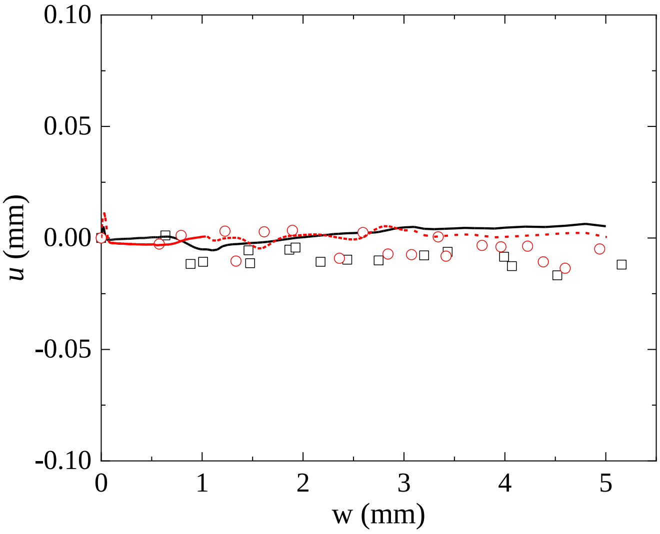


0.10
0.05
0.00
-0.05
-0.10
u (mm)
0
1
2
3
4
5
w (mm)


Experiments:
k_n = 1000 kPa/mm
k_n = 5000 kPa/mm
BiLSTM prediction:
k_n = 1000 kPa/mm
k_n = 5000 kPa/mm
CNS: σ_n = 100 kPa, $I_{D0} \approx 90\%$
Smooth plate
σ_n (kPa)
w (mm)
0
50
100
150
200
250
300
0
1
2
3
4
5



200
160
120
80
40
0
τ (kPa)
0
1
2
3
4
5
w (mm)


0.10
0.05
0.00
-0.05
-0.10
u (mm)
0
1
2
3
4
5
w (mm)