Computers and Geotechnics 169 (2024) 106248

Contents lists available at ScienceDirect

Computers and Geotechnics

journal homepage: www.elsevier.com/locate/compgeo

A data-driven approach to optimize the design configuration of multi-sleeve cone penetrometer probe attachments

Danrong Zhang [a,*], Nimisha Roy [b], J. David Frost [c]

[a] School of Computational Science and Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA
[b] School of Computing Instruction, Georgia Institute of Technology, Atlanta, GA 30332, USA
[c] School of Civil and Environmental Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA

ARTICLE INFO

Keywords:
Machine learning
Cone penetration probe
Multi sleeve attachments
Device optimization
Feature selection
Random forest

ABSTRACT

This study uses a data-driven approach to address the complexities associated with research focused multi-sleeve Cone Penetration Test (CPT) devices, particularly focusing on the multi-friction attachment (MFA) and multi-piezo-friction attachment (MPFA) CPT devices. Hindered by time-consuming assembly and susceptibility to sensor stream losses due to extensive electronic components, these advanced devices demand optimization to transform from research devices to practice-suitable devices. This study aims at optimizing the design of the multi-sleeve CPT devices using machine learning, with soil type classification performance as the primary metric for device configuration effectiveness. The research scope centers not on using machine learning for soil classification but on refining the design of multi-sleeve CPT devices. A two-phase data-driven approach is adopted, testing various feature combinations across eight machine learning models. The first phase involves identifying the most suitable model for the dataset, followed by a refinement of features to balance sensor number minimization and soil classification accuracy. The result is a proposed configuration for a multi-sleeve CPT device, simplifying the original design while maintaining robustness, thereby enhancing cost-efficiency and operational effectiveness in geotechnical practice. This work sheds light on how the integration of machine learning can guide the design optimization of geotechnical instruments.

1. Introduction

The performance of geotechnical structures, such as deep foundations, synthetic liners, and earth retaining systems, hinges on the behavior of soil-geomaterial interfaces and their ability to transfer loads (DeJong & Frost, 2002). These interfaces, encompassing contact between geological and man-made materials or varying geological strata, play a pivotal role in the behavior of geotechnical systems. Therefore, precise prediction and assessment of geotechnical interfaces are essential, directly impacting the structural integrity and cost-effectiveness of geotechnical design (Hebeler & Frost, 2006). Instrumented cone penetration test (*CPT*) devices are used in geotechnical engineering to generate profiles of response data that are subsequently analyzed to yield reliable indicators of both soil index properties as well as engineering properties for use in designing, constructing and monitoring the performance of geo-structures. Its ease of implementation, familiarity within the construction industry, and high classification accuracy make *CPT* widely used for research and practice (Hebeler et al., 2018). *CPT* measures the penetration resistance of a conical tip (*qt*) inserted into the ground, the frictional force (*fs*) that the soil exerts on a smooth sleeve located just above the cone tip, and the pore pressure (*u2*) recorded close to the penetrating tip as the probe is inserted into the subsurface (Begemann, 1965; Douglas, 1981).

A limitation of this tool is that it conventionally measures the frictional response of the soil when sheared against a surface with fixed very low roughness (smooth sleeve). However, researchers in the past have shown that a more robust characterization of interface and soil strength can be achieved when the soil is sheared against surfaces with a range of different roughness values (DeJong, 2001; DeJong & Frost, 2002; Frost et al., 2013). These findings have led to the development of more specialized *CPT* tools that include multi-sleeve attachments to be used behind a conventional *CPT* probe or as stand-alone devices behind an un-instrumented tip, thereby reflecting a multiple-sensor approach. The multi-friction attachment (*MFA*) for the cone penetrometer can record four individual sleeve friction measurements (*fs1*, *fs2*, *fs3*, and *fs4*) at each elevation within a sounding, in addition to the conventional *CPT qt*,

* Corresponding author.
E-mail addresses: dzhang373@gatech.edu (D. Zhang), nroy9@gatech.edu (N. Roy), david.frost@ce.gatech.edu (J.D. Frost).

https://doi.org/10.1016/j.compgeo.2024.106248

Available online 26 March 2024

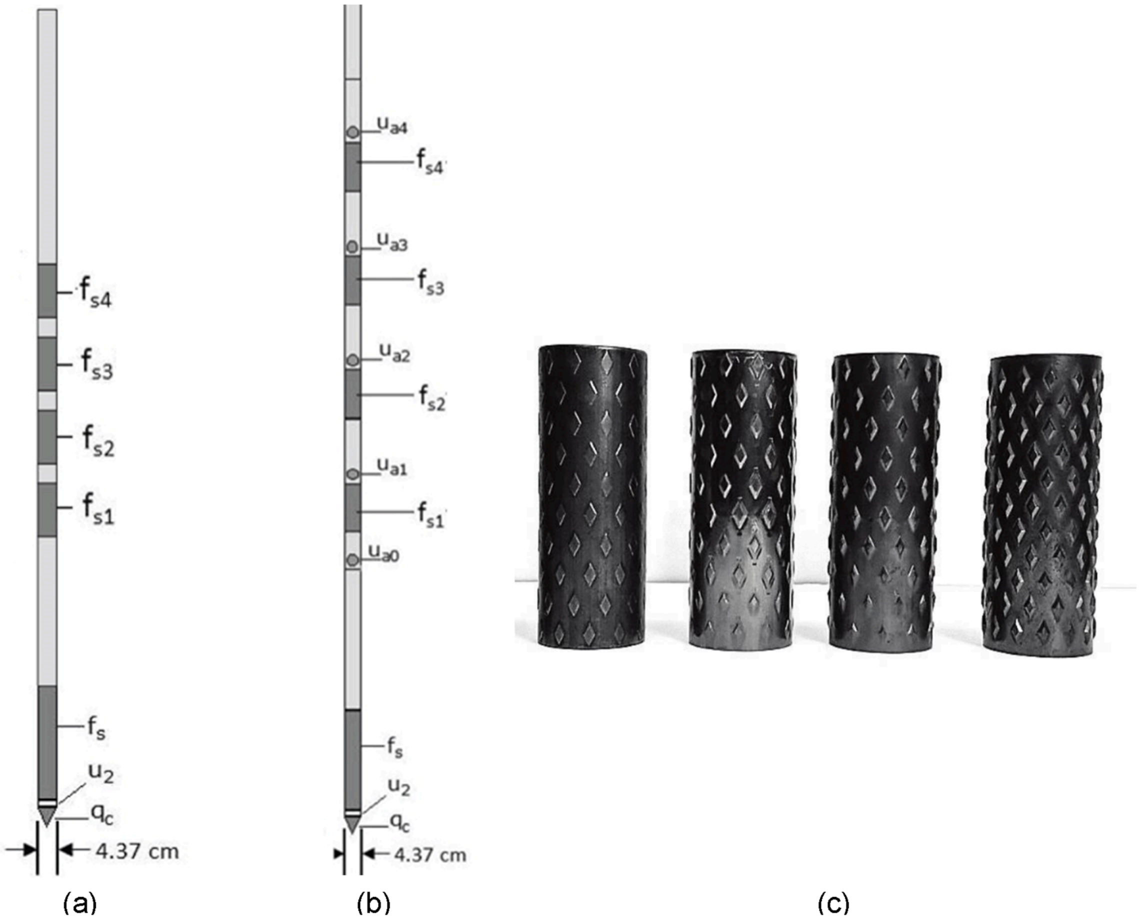

Fig. 1. MFA and MPFA configured with conventional CPT module: (a) MFA design detail; (b) MPFA design detail; (c) Sleeves with increasing surface roughness of fs1, fs2, fs3 and fs4 (Hebeler et al., 2018).

Table 1
Description of *CPT* sounding tested with multi-sleeve probes.

CPT Sounding Dataset #	Sensor Streams	Main site soil type	Device type
1	*qt, u2, fs, fs1, fs2, fs3, fs4*	Sand and some silt layers from Vermont, USA	*CPT-MFA*
2	*qt, u2, fs, fs1, fs2, fs3, fs4*	Sand and some silt layers from Vermont, USA	*CPT-MFA*
3	*qt, u2, fs, ua0, fs1, ua1, fs2, ua2, fs3, ua3, fs4, ua4*	Silty sand from South Carolina, USA	*CPT-MPFA*
4	*qt, u2, fs, ua0, fs2, ua2, fs3, fs4*	Silty sand from South Carolina, USA	*CPT-MPFA*
5	*qt, u2, fs, ua0, fs1, ua1, fs2, fs3, ua3, fs4, ua4*	Clay from Western Australia	*CPT-MPFA*
6	*qt, u2, fs, fs1, fs2, fs3, fs4*	Clay from Western Australia	*CPT-MFA*

fs and *u2* measurements, as shown in Fig. 1 (a) (DeJong & Frost, 2002). The multi-piezo-friction attachment (*MPFA*) for the cone penetrometer can obtain four independent measures of interface response (*fs1*, *fs2*, *fs3*, and *fs4*) and five independent measures of dynamic pore water pressure along the shaft (*ua0*, *ua1*, *ua2*, *ua3*, and *ua4*) of a penetrated probe, in addition to conventional *CPT* measurements (*qt, fs,* and *u2*), shown in Fig. 1 (b) (Hebeler & Frost, 2006). These enhancements of the basic *CPT* device include the ability to configure the device with multiple sleeves with different surface textures and multiple pore pressure sensors to yield 7 to 12 streams of data, respectively, compared to the three streams from conventional CPT devices. While the enhancements to existing *CPT* devices have been shown to provide a much richer data set for stratigraphic evaluation and soil classification, their current embodiments are clearly more oriented to research rather than practice because they take significant time to assemble before each deployment and are subject to more frequent sensor stream losses due to a large number of electronic cables and connectors involved and the harsh environment that they are used in.

Over the past decade, there have been significant advancements in data-driven approaches within the engineering field (Giustolisi et al., 2007; Laucelli et al., 2023). These developments have paved the way for reducing the number of sensors in multi-sensor CPT devices, aided by AI technologies. This reduction can decrease both fabrication costs and the risk of sensor failure during use. In addition, recent studies have demonstrated that incorporating machine learning models to classify soils from conventional *CPT* data yields promising results (Moon et al., 2022; Rauter & Tschuchnigg, 2021; Reale et al., 2018; Tsiaousi et al., 2018; Wu et al., 2021; Zhang et al., 2021). Compared to these studies, the basis of the present study is to identify the least number of essential features for *MFA-CPT* and *MPFA-CPT* devices with the help of machine learning. To be specific, the objective is to reduce the overall number of sensors while not losing the benefits of additional data streams corresponding to rougher sleeve surfaces and associated different pore pressures. The effectiveness of the reduction is evaluated through machine learning performance in soil type classification. By optimizing sensor count without sacrificing the quality, the industry stands to benefit from reduced operational costs and improved reliability in geotechnical investigations. This integration of machine learning in the design of multi-sleeve CPT devices also points out the potential of using AI to guide the

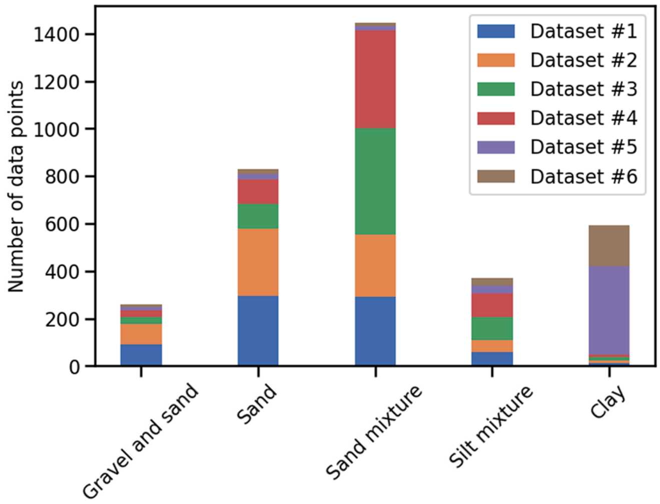

Fig. 2. Soil type distributions across the six datasets.

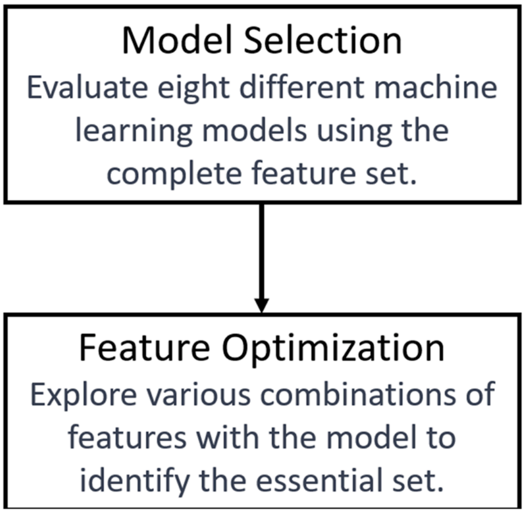
Model Selection
Evaluate eight different machine learning models using the complete feature set.
Feature Optimization
Explore various combinations of features with the model to identify the essential set.

Fig. 3. Workflow of determining the minimal essential set of sensor measurement features.

Table 2
Classification results of different machine learning models using conventional CPT features only with CPT-MFA and CPT-MPFA as input datasets.

Model	Precision	Recall	F1 score	Accuracy
Decision Tree	0.80	0.81	0.81	0.83
AdaBoost	0.84	0.78	0.80	0.84
Random Forest	**0.87**	0.84	**0.85**	**0.88**
XGBoost	0.86	0.84	**0.85**	0.87
Tabnet	0.83	0.81	0.82	0.84
KNN	0.84	**0.85**	0.84	0.85
SVM	0.75	0.67	0.70	0.77
Conventional neural network	0.71	0.62	0.64	0.71

design optimization of geotechnical instruments.

2. Dataset

The datasets of *CPT* soundings used in this study, as detailed in Table 1, were obtained from 3 distinct sites: Vermont, South Carolina, and Western Australia, collected by DeJong (2001) and Hebeler (2005). Five different soil types are included, namely gravel and sand mixtures, sands, sand mixtures, silt mixtures, and clays. A detailed typical soil profile for each site can be found in Appendix A.

Table 3
Classification results of different machine learning models using conventional CPT features and MFA features with CPT-MFA and CPT-MPFA as input datasets.

Model	Precision	Recall	F1 score	Accuracy
Decision Tree	0.88	0.89	0.88	0.89
AdaBoost	0.93	0.90	0.91	0.92
Random Forest	**0.95**	**0.93**	**0.94**	**0.94**
XGBoost	0.94	**0.93**	0.93	**0.94**
Tabnet	0.92	0.90	0.91	0.92
KNN	0.92	**0.93**	0.92	0.93
SVM	0.81	0.77	0.78	0.82
Conventional neural network	0.66	0.63	0.63	0.71

Table 4
Classification results of different machine learning models using conventional *CPT* only with *CPT-MPFA* as input datasets.

Model	Precision	Recall	F1 score	Accuracy
Decision Tree	0.85	0.84	0.84	0.90
AdaBoost	0.87	0.82	0.83	0.91
Random Forest	**0.90**	0.84	0.86	**0.92**
XGBoost	0.89	**0.89**	**0.89**	**0.92**
Tabnet	0.87	0.86	0.87	0.91
KNN	0.85	0.85	0.84	0.90
SVM	0.87	0.75	0.78	0.87
Conventional neural network	0.60	0.53	0.46	0.72

Table 5
Classification results of different machine learning models using conventional *CPT* features and *MPFA* features with *CPT-MPFA* as input datasets

Model	Precision	Recall	F1 score	Accuracy
Decision Tree	0.91	0.89	0.90	0.92
AdaBoost	**0.96**	0.93	**0.94**	**0.95**
Random Forest	0.95	0.93	**0.94**	**0.95**
XGBoost	0.95	0.92	0.93	**0.95**
Tabnet	0.92	0.93	0.92	0.94
KNN	0.92	**0.94**	0.93	**0.95**
SVM	0.91	0.88	0.89	0.91
Conventional neural network	0.83	0.70	0.73	0.83

Datasets #1, #2, and #6 include soundings with a conventional CPT and an *MFA* that provides four independent measurements of interface response, from friction sleeves at different positions behind the cone penetrometer (*fs1*, *fs2*, fs3*,* and *fs4)*, in addition to the conventional CPT measurements (*qt*, *fs*, and *u2* measurements). On the other hand, datasets #3, #4, and #5 use *MPFA* and include soundings which provide four independent measures of interface response (*fs1*, *fs2*, *fs3*, and *fs4*) and five independent measures of dynamic pore water pressure along the shaft (*ua0*, *ua1*, *ua2*, *ua3*, and *ua4*), in addition to the conventional CPT measurements (*qt*, *fs*, and *u2*). However, in dataset #4, *fs1*, *ua1*, *ua3*, and *ua4* were damaged, so data was collected only from the *qt*, *fs*, *u2*, *ua0*, *ua2*, *fs2*, *fs3*, and *fs4* sensors in that dataset. Since the sensors are independent, the damage only resulted in loss of data from the corresponding sensor. For dataset #5, the *ua2* sensor was damaged; hence data was collected from the *qt*, *fs*, *u2*, *ua0*, *ua1*, *ua3*, *ua4, fs1*, *fs2*, *fs3*, and *fs4* sensors, as noted in Table 1. The sensors that are used in all six datasets are *qt*, *u2*, *fs*, *fs2*, *fs3*, *fs4*, and *ua0*. The corresponding typical sensor record streams for each dataset can be found in Appendix B. In each dataset, the surface roughness of *fs1*, *fs2*, *fs3*, and *fs4* varies. However, for the purposes of this study, specific surface roughness variations in the different locations have been disregarded to avoid overcomplicating the analysis since in general, roughness increased with distance behind the CPT (DeJong, 2001). As a result, *fs2* in dataset #1 will be treated as equivalent to *fs2* in any other dataset, regardless of differences in surface roughness. This same approach is applied to *fs3*

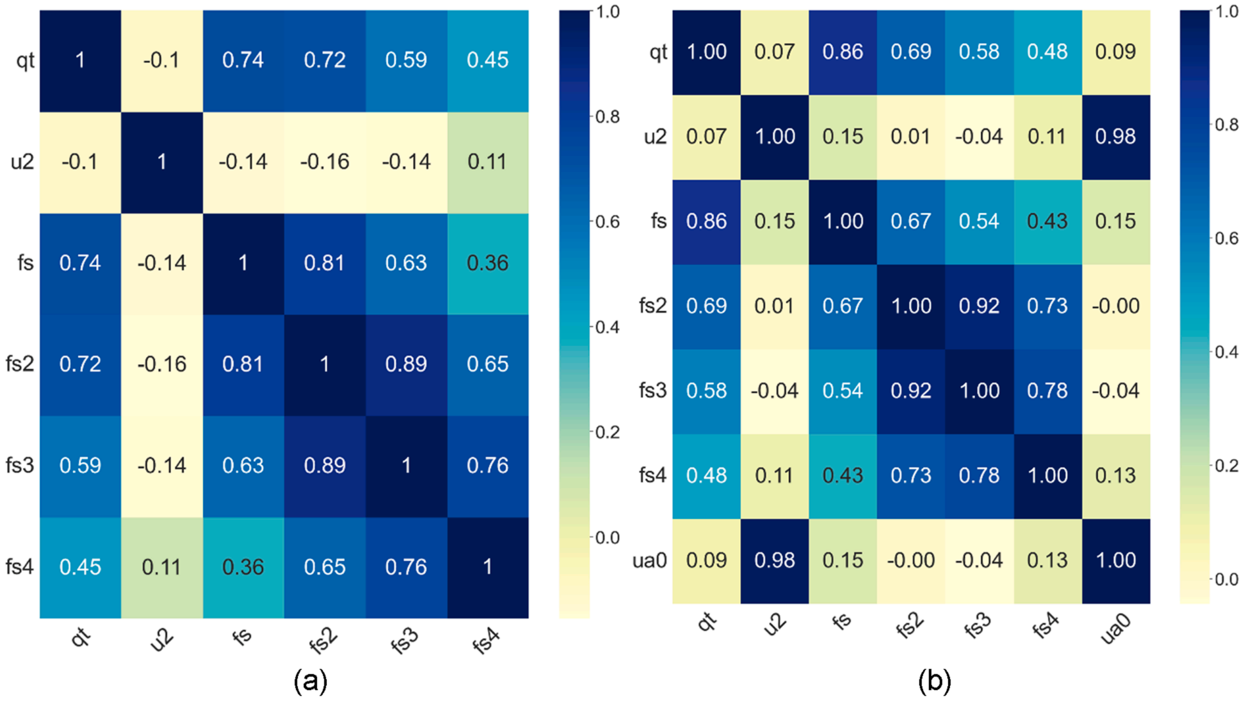

Fig. 4. Correlation Matrix: (a) features dependence for soil classification applied to all six datasets, (b) features dependence for soil classification applied to *CPT-MPFA* datasets only.

Table 6
Classification Results for different feature combinations using conventional *CPT* measurements only with *CPT-MFA* and *CPT-MPFA* datasets.

Measurement Type	Features	Precision	Recall	F1 score	Accuracy
Conventional *CPT* measurements only	*qt, fs, u2*	0.87	0.84	0.85	0.88
	qt, u2	0.75	0.71	0.73	0.77
	qt, fs	0.76	0.71	0.73	0.75
	fs, u2	0.78	0.75	0.76	0.80

Table 7
Classification Results for different feature combinations using conventional CPT measurements and MFA measurements with *CPT-MFA* and *CPT-MPFA* datasets.

Measurement Type	Features	Precision	Recall	F1 score	Accuracy
CPT-MFA measurements	*qt, fs, u2, fs2, fs3, fs4*	0.95	0.93	0.94	0.94
	qt, fs, u2, fs3, fs4	0.93	0.92	0.92	0.93
	qt, fs, u2, fs2, fs3	0.94	0.91	0.93	0.93
	qt, fs, u2, fs2, fs4	0.93	0.92	0.93	0.94
	qt, fs, u2, fs4	0.92	0.90	0.91	0.92
	qt, fs, u2, fs3	0.92	0.90	0.91	0.92
	qt, fs, u2, fs2	0.91	0.89	0.90	0.91

and *fs4*. It is important to note that while these features are treated as identical in this work, their actual surface roughness may differ somewhat across the datasets.

2.1. Soil type distribution

The soil types in the above datasets are divided into gravel and sand mixture, sand, sand mixture, silt mixture, and clay according to their respective *Ic*, where *Ic* is the soil behavior type index (Robertson, 1991). The six datasets described in Table 1 comprise a total of 3501 unique data points. The detailed distribution of soil types is shown in Fig. 2. It can be observed that the data is not uniformly distributed, mainly comprising sands, sand mixtures, and clays. For example, over 90% of clay data points are concentrated in two datasets. This issue is further elaborated on in Section 4.3.

3. Methodology

The primary aim of this study is to employ machine learning techniques to determine the minimal yet essential set of sensors that maintain high soil type classification accuracies in MFA and MPFA type devices. To achieve this, various feature combinations need to be tested across different machine learning models to ascertain the optimal configuration. Given the extensive number of potential experiments when exploring every possible feature combination for each model, a two-phase approach is adopted, as shown in Fig. 3. Initially, the study evaluates the performance of eight machine learning models using the full set of features to identify the model that is most compatible with the dataset. Once the most suitable model is selected, further exploration is conducted to refine the feature combinations, focusing on achieving the best balance of feature minimization and classification accuracy.

Upon determining the optimal model and feature combination, the study applies these findings to classify soil types in an unseen dataset, where resampling techniques are employed to enhance the model's performance further. The following section introduces the methodology used in this work, including the machine learning models, hyper-parameter tuning, model evaluation methods and resampling strategies.

3.1. Machine learning model

The data employed in this study consists of a typical tabular format, where each data point is represented as a vector comprising various features. In the analysis of tabular data, deep learning has seen rapid advancements over the past decade, and certain deep learning models have shown exceptional performance on specific tabular datasets. However, despite these advancements, tree-based machine learning

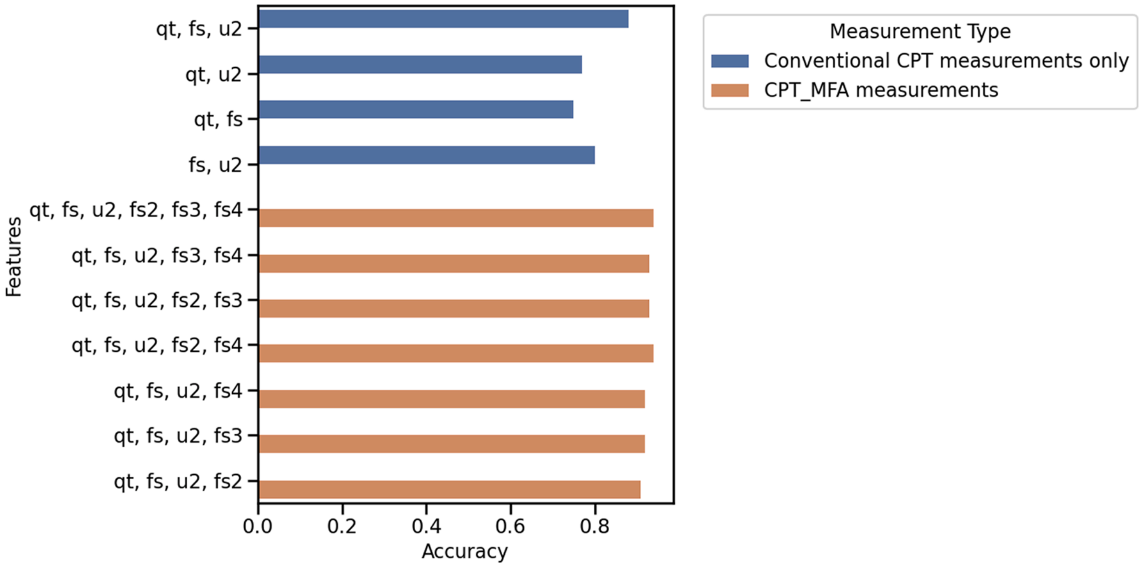

Fig. 5. Comparative accuracy of the model across various feature combinations with *CPT-MFA* measurements.

Table 8
Classification Results for different feature combinations using conventional *CPT* measurements only with *CPT-MPFA* datasets.

Measurement Type	Features	Precision	Recall	F1 score	Accuracy
Conventional *CPT* measurements only	*qt, fs, u2*	0.90	0.84	0.86	0.92
	qt, u2	0.79	0.76	0.77	0.86
	qt, fs	0.81	0.69	0.73	0.82
	u2, fs	0.84	0.78	0.80	0.86

Table 9
Classification Results for different feature combinations using conventional *CPT* measurements and *MPFA* measurements with *CPT-MPFA* datasets.

Measurement Type	Features	Precision	Recall	F1 score	Accuracy
CPT-MPFA measurements	*qt, fs, u2, fs2, fs3, fs4, ua0*	0.95	0.93	0.94	0.95
	qt, fs, u2, fs3, fs4, ua0	0.96	0.94	0.95	0.96
	qt, fs, u2, fs2, fs3, ua0	0.95	0.93	0.94	0.95
	qt, fs, u2, fs2, fs4, ua0	0.94	0.92	0.93	0.94
	qt, fs, u2, fs4, ua0	0.94	0.92	0.93	0.94
	qt, fs, u2, fs3, ua0	0.94	0.92	0.93	0.94
	qt, fs, u2, fs2, ua0	0.94	0.92	0.93	0.94
	qt, fs, u2, ua0	0.92	0.89	0.90	0.93
	qt, u2, ua0	0.90	0.89	0.89	0.91
	fs, u2, ua0	0.92	0.88	0.90	0.90
	qt, fs, ua0	0.90	0.86	0.87	0.92
	qt, fs, u2, fs4	0.92	0.89	0.91	0.93
	qt, fs, u2, fs3	0.91	0.85	0.87	0.93
	qt, fs, u2, fs2	0.91	0.86	0.88	0.92

models continue to be the leading performers (Gorishniy et al., 2021; Shwartz-Ziv & Armon, 2022). In light of this, the study extends its exploration to eight machine learning models, including four tree-based models: decision tree, adaptive boosting (AdaBoost), random forest, and extreme gradient boosting (XGBoost); one deep learning model specifically designed for tabular data - TabNet (Arik & Pfister, 2021), alongside K nearest neighbors (KNN), support vector machine (SVM), and a conventional neural network. Hyperparameter tuning is facilitated by the robust capabilities of the scikit-learn library (Pedregosa et al., 2011). The definition of the hyperparameters for each model and the corresponding search space are detailed in Appendix C.

3.1.1. *Decision tree*

Decision tree features a tree-like structure with nodes and edges, where each node represents a feature, and each edge signifies the outcome of a test. The accuracy of decision trees can be enhanced through node splitting, which divides nodes into sub-nodes. At each tree level, one feature is selected and split to ensure a maximum drop in uncertainty. Decision trees are interpretable and quick to train and test, but challenges such as overfitting and suboptimal prediction accuracy for complex tasks may arise.

3.1.2. *AdaBoost*

AdaBoost applies a sequential learning approach where multiple base learners are combined to enhance the model performance. In this work, AdaBoost is utilized with decision trees as the base learners. The algorithm iteratively adjusts to focus more on the data points that were previously misclassified, by modifying their weights. This makes them more influential in the training of subsequent learners. Each decision tree in AdaBoost contributes to the final model, and the accuracy of each tree determines its weight in the final decision. AdaBoost is effective in reducing overfitting, especially in datasets with high variance, by creating a robust classifier. However, its performance can be affected by noisy data and outliers.

3.1.3. *Random Forest*

The Random Forest algorithm is an ensemble technique that combines multiple decision trees to improve prediction performance. Each tree is trained on a randomly sampled subset of the data with replacement, using a different combination of features. This approach, known as bootstrap sampling and feature bagging, enhances the diversity and robustness of the model. The final prediction is determined by a majority vote from all trees. Random Forests typically outperform single decision tree by reducing overfitting. However, they require more computational resources, which can increase training time, and are less interpretable due to their complexity.

3.1.4. *XGBoost*

Similar to Random Forest, XGBoost also constructs multiple decision

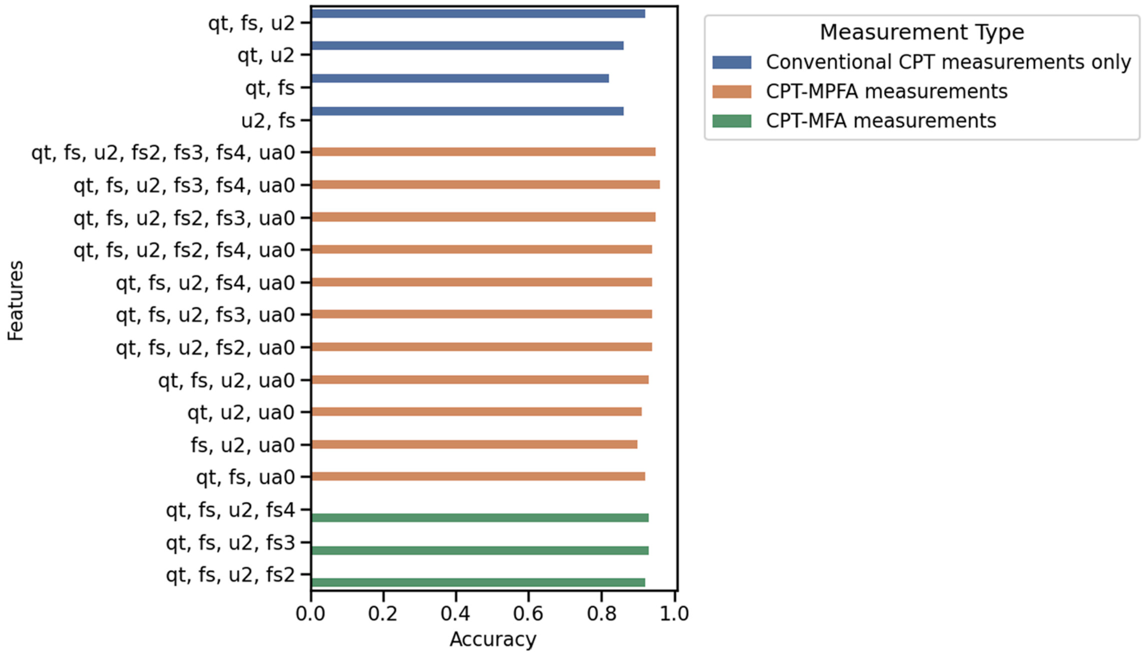

Fig. 6. Comparative accuracy of the model across various feature combinations with *CPT-MPFA* measurements.

Table 10
Model accuracy with and without depth as features.

With/without depth	Dataset	Features	Accuracy
Without depth	*CPT-MFA* and *CPT-MPFA* combined	*qt*, *fs*, *u2*, *fs2*, *fs3* and *fs4*	0.94
	CPT-MPFA only	*qt*, *fs*, *u2*, *fs2*, *fs3*, *fs4* and *ua0*	0.95
With depth	*CPT-MFA* and CPT-*MPFA* combined	Depth, *qt*, *fs*, *u2*, *fs2*, *fs3* and *fs4*	0.97
	CPT-MPFA only	Depth, *qt*, *fs*, *u2*, *fs2*, *fs3* *fs4* and *ua0*	0.97

trees. However, its approach is sequential, where each subsequent tree is designed to rectify the errors made by the previous ones. This process is driven by gradient boosting, a method that focuses on minimizing the loss function. In contrast to AdaBoost, which also builds decision trees sequentially but typically uses simpler trees, XGBoost employs more complex, deeper trees for detailed data modeling.

A notable advantage of XGBoost lies in its integration of regularization methods, which are instrumental in mitigating overfitting. Despite its strengths, XGBoost presents a more complex tuning process compared to simpler models like decision trees or random forest. Its sequential tree-building nature can also result in longer training times, particularly for extensive datasets.

3.1.5. TabNet

TabNet is an innovative deep learning model specifically designed for tabular data (Arik & Pfister, 2021). Developed by Google Cloud AI researchers, it stands out from traditional deep learning models by using sequential attention to choose which features to reason from at each decision step, making it highly interpretable. This attention mechanism allows TabNet to learn both local and global representations of features, contributing to its robust predictive performance. TabNet also employs a unique feature masking strategy, enabling it to perform feature selection dynamically during the training process. This leads to efficient learning and improved generalization on structured data. TabNet achieves state-of-the-art performance on several real-world datasets.

3.1.6. KNN

The KNN algorithm is a non-parametric method, which operates by identifying the 'k' nearest data points in the feature space to a query point and making predictions based on the labels of these neighbors. KNN's effectiveness lies in its simplicity and the intuitive nature of its mechanism, where the outcome is determined by the majority vote or average from the 'k' nearest neighbors. Unlike more complex models, KNN does not build an explicit model but makes decisions based on the localized pattern of the data. This characteristic makes KNN particularly useful in scenarios where the data distribution is not well understood. However, it can be computationally demanding with large datasets, as it involves calculating the distance between the query point and all points in the dataset for each prediction.

3.1.7. SVM

SVM seeks to identify an optimal hyperplane within an N-dimensional space to distinctly classify data points. Its primary objective is to determine a hyperplane that maximizes the margin, defined as the greatest distance between the hyperplane and the nearest data points on either side. This focus on only the most critical data points near the decision boundary makes SVM notably memory efficient. For post-training, it requires only the storage of these pivotal support vectors. Despite this efficiency, SVM tends to underperform with noisy datasets, where the clear margin needed for optimal classification is obscured by overlapping data points.

3.1.8. Conventional neural network

Neural network typically structures with an input layer, several hidden layers, and an output layer, and operates through weights and biases that facilitate data transformation across these layers. These parameters are refined via forward and backward propagation during the training process. The effectiveness of neural network generally increases with the availability of larger training datasets, as they have numerous weights and biases to learn and optimize. Note that TabNet in Section 3.1.5 is a type of neural network, however, it stands out from traditional neural network by incorporating attention mechanisms and feature selection that improve interpretability and performance in structured data analysis.

Table 11
Classification Results using an entire sounding dataset as the testing set.

Experiment ID	Train	Test	precision	recall	F1	accuracy
1	Dataset #2 Dataset #3 Dataset #4 Dataset #5 Dataset #6	Dataset #1 Sand and some silt layers Vermont, USA	0.39	0.33	0.30	0.55
2	Dataset #1 Dataset #3 Dataset #4 Dataset #5 Dataset #6	Dataset #2 Sand and some silt layers Vermont, USA	0.33	0.29	0.26	0.56
3	Dataset #1 Dataset #2 Dataset #4 Dataset #5 Dataset #6	Dataset #3 Silty sand South Carolina, USA	0.53	0.43	0.45	0.72
4	Dataset #1 Dataset #2 Dataset #3 Dataset #5 Dataset #6	Dataset #4 Silty sand South Carolina, USA	0.53	0.53	0.50	0.75
5	Dataset #1 Dataset #2 Dataset #3 Dataset #4 Dataset #6	Dataset #5 Clay Western Australia	0.45	0.44	0.40	0.63
6	Dataset #1 Dataset #2 Dataset #3 Dataset #4 Dataset #5	Dataset #6 Clay Western Australia	0.30	0.44	0.30	0.53

3.2. Pearson correlation coefficient

A correlation matrix maps the Pearson correlation coefficient between each pair of input features (Cohen et al., 2009). It can show the statistical relationship between two features. The range of the correlation coefficient is (−1, 1). If two features are closely correlated, the absolute value of the coefficient will be close to 1; otherwise, it will be close to 0. Including two highly correlated features when training a machine learning model is not desirable since correlated features can lead to overfitting. In this work, the correlation matrix is generated to guide the feature selection process.

3.3. Hyperparameter tuning and model evaluation

To optimize model performance, hyperparameter tuning is essential. In this study, the models undergo tuning through randomized search, coupled with 10-fold cross-validation on the training set. However, for TabNet, considering the computational cost, randomized search with 3-fold cross-validation on the training set is used. The effectiveness of the different models is assessed using key metrics: recall, precision, F1 score, and accuracy. This approach ensures a balanced evaluation of model performance across different aspects of classification effectiveness.

3.3.1. K fold cross validation

Cross-validation is a method that evaluates machine learning models and works well with limited data. It provides a better estimate of a model’s generalization capability by evaluating it on multiple subsets of the data. Usually, the dataset is evenly divided into K groups. Each time K-1 groups are used to train a model, and one group is used to evaluate the performance of the trained model. This pattern of training and evaluating is repeated K times, with choosing a different hold-out dataset for evaluation each time. The final performance metrics are computed as the average of the K iterations.

3.3.2. Randomized search

Randomized Search is a method used in hyperparameter optimization for machine learning models. It involves randomly selecting a fixed number of parameter combinations from specified distributions for the hyperparameters. This technique allows for a more efficient exploration of the parameter space, as it does not require a systematic examination of all possible combinations. While it may not guarantee finding the absolute best parameters, it provides a balance between exploration and computational efficiency, making it a practical choice in many machine learning applications.

Due to the higher computational requirements of TabNet, the search is restricted to 15 iterations. For the rest of the models, 100 iterations of randomized search are conducted. The best estimator from these randomized searches is determined based on the highest accuracy achieved.

3.3.3. Performance metrics

In multiclass classification problems, evaluating the performance of a model involves several key metrics: recall, precision, F1 score, and accuracy. Recall, also known as sensitivity, measures the proportion of actual positives that are correctly identified. Precision quantifies the proportion of predicted positives that are true positives. The F1 score provides a harmonic mean of precision and recall, offering a balance between the two by considering both false positives and false negatives. It is particularly useful when there’s an uneven class distribution. Lastly, accuracy represents the overall correctness of the model, defined as the ratio of correctly predicted observations to the total observations. It’s a straightforward metric, however, it can be misleading in imbalanced class distributions. Therefore, having the combined consideration of these four metrics is critical for a comprehensive understanding of a multiclass classifier’s performance.

3.4. Resampling

Resampling, generally used to resolve the imbalanced dataset issue, includes two methods - up-sampling and down-sampling. Up-sampling increases the number of data points in the minority classes so that it can match with the majority class. In contrast, down-sampling decreases the data points in the majority classes to match with the minority class.

In this work, for up-sampling experiment, the minority classes of the training set are up-sampled to have the same number as the majority class (the class with the highest number of data points) by an up-sample method called the synthetic minority oversampling technique (*SMOTE)*

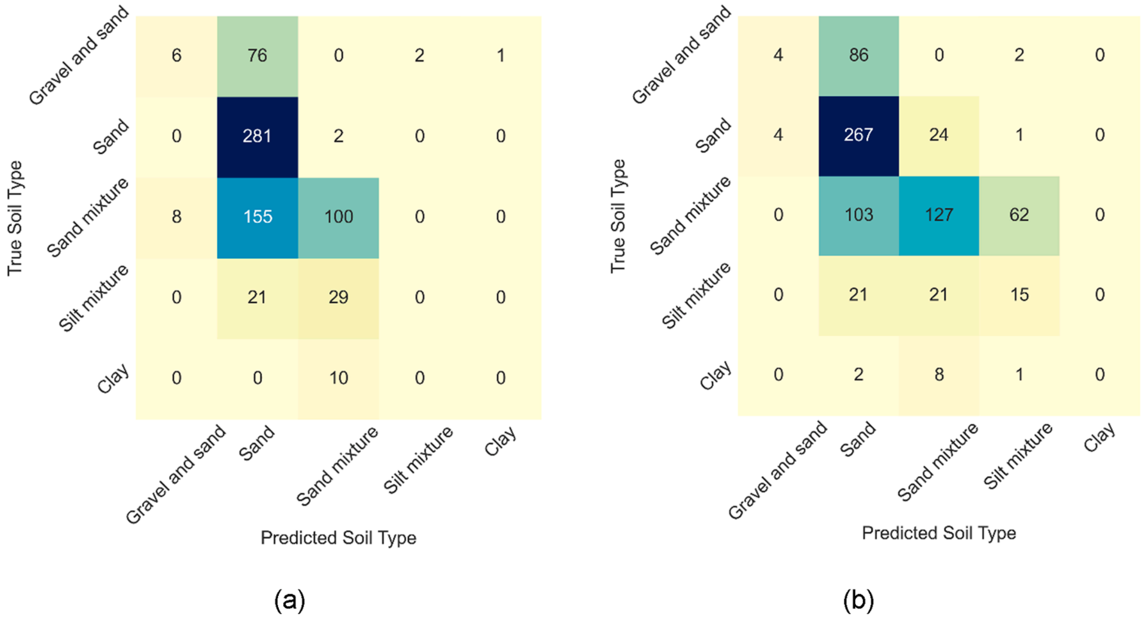

Fig. 7. Confusion matrix for experiment 1 and 2: (a) confusion matrix for experiment 1, (b) confusion matrix for experiment 2.

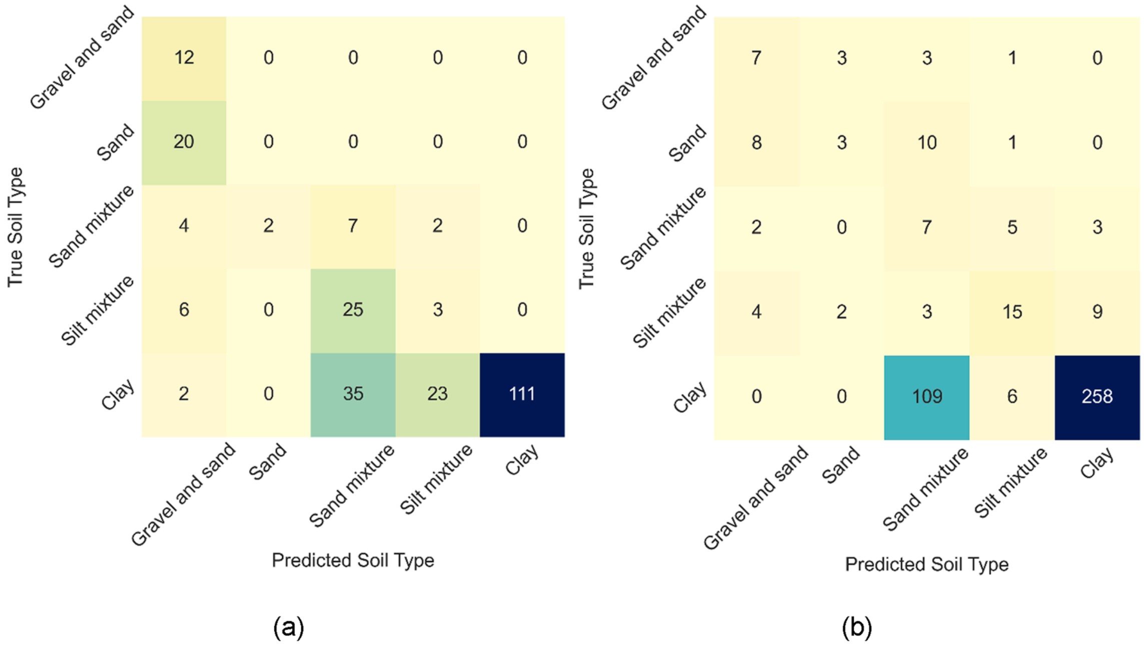

Fig. 8. Confusion matrix for experiment 5 and 6: (a) confusion matrix for experiment 5, (b) confusion matrix for experiment 6.

(Chawla et al., 2002). *SMOTE* first finds K samples that are closest in the distance to the minority class samples and then gets the difference between the minority sample (x_i) and the nearest neighbor (x_j). The synthetic new samples are generated by $x_{new} = x_i + (x_j - x_i)\cdot\delta$, whereas δ is a random number between 0 and 1.

For down-sampling experiments, the majority classes of the training set are down-sampled to have the same number as the minority class (the class with the least number of data points) by random selection.

A combined resampling experiment is also explored, where the minority classes are up-sampled and the majority classes are down-sampled. The objective is to equalize the number of instances across all classes to match the count of the mid-sized class.

4. Results

4.1. Model selection

This section explores the optimal machine learning model for soil classification, utilizing shared features across datasets. Four experiments are conducted, each having 80% of the dataset for training and 20% for testing. The comparative analysis involves a range of models including decision tree, AdaBoost, random forest, XGBoost, TabNet, KNN, SVM and conventional neural network to determine the most effective approach in accurately classifying soil types.

The randomized search method is implemented for each model to arrive at the best parameters. The training set is used to fit the randomized search and implement cross-validation to find the best

Table 12
Classification Results of experiments without resampling, with up-sampling, with down-sampling, and with combined sampling.

Experiment ID	Train	Test	Accuracy without resampling	Accuracy after up-sampling	Accuracy after down-sampling	Accuracy after combined sampling
1	Dataset #2 Dataset #3 Dataset #4 Dataset #5 Dataset #6	Dataset #1 Sand and some silt layers Vermont, USA	0.55	0.47	0.62	0.45
2	Dataset #1 Dataset #3 Dataset #4 Dataset #5 Dataset #6	Dataset #2 Sand and some silt layers Vermont, USA	0.56	0.63	0.42	0.61
3	Dataset #1 Dataset #2 Dataset #4 Dataset #5 Dataset #6	Dataset #3 Silty sand South Carolina, USA	0.72	0.75	0.74	0.75
4	Dataset #1 Dataset #2 Dataset #3 Dataset #5 Dataset #6	Dataset #4 Silty sand South Carolina, USA	0.75	0.70	0.72	0.75
5	Dataset #1 Dataset #2 Dataset #3 Dataset #4 Dataset #6	Dataset #5 Clay Western Australia	0.63	0.14	0.59	0.30
6	Dataset #1 Dataset #2 Dataset #3 Dataset #4 Dataset #5	Dataset #6 Clay Western Australia	0.53	0.42	0.63	0.54

estimator. With the best estimator of each model, the test set accuracy, precision, recall, and F1 score are compared and used to identify the optimal model.

4.1.1. CPT-MFA datasets and CPT-MPFA datasets combined

Two experiments are conducted using CPT-MFA and CPT-MPFA datasets combined. The first experiment, which functions as the baseline model, uses conventional CPT features only, namely *qt, fs* and *u2*. The second experiment uses both conventional CPT features and MFA features, including *qt*, *fs*, *u2*, *fs2*, *fs3*, and *fs4*. The results of eight different models are shown in Tables 2 and 3, respectively.

With MFA features, the overall model performance is better, with the highest precision increased by 8%, the highest F1 score and recall increased by 9% and the highest accuracy increased by 6%, compared to using conventional CPT features only. This indicates that MFA features can help machine learning models classify soil better. Random Forest is the optimal model based on both experiments since it has the highest precision, F1 score and accuracy.

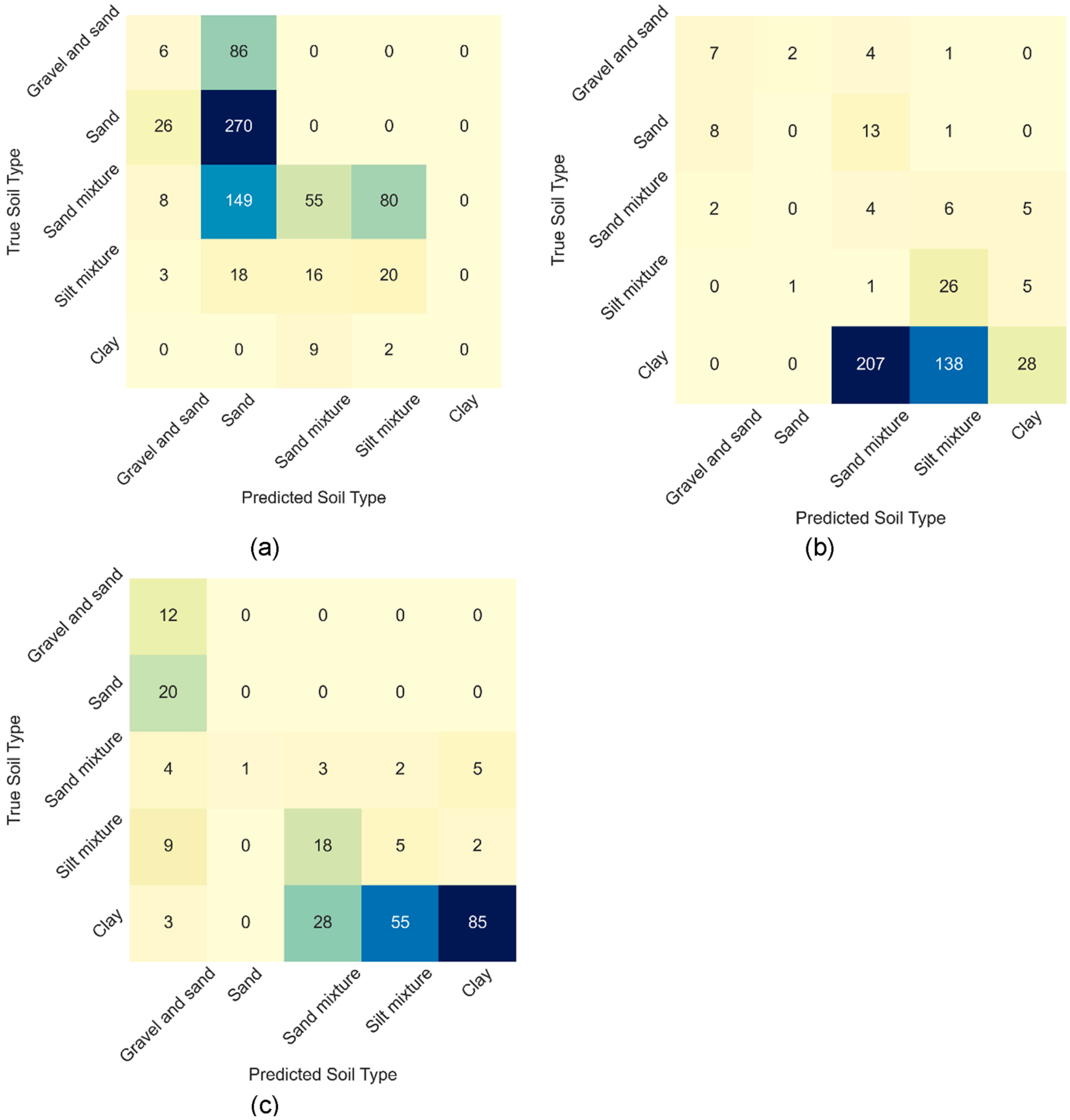

Fig. 9. Confusion matrix for experiment 1, 5 and 6 after up-sampling: (a) confusion matrix for experiment 1, (b) confusion matrix for experiment 5, (c) confusion matrix for experiment 6.

4.1.2. *CPT-MPFA datasets only*

Two additional experiments are conducted using only *CPT-MPFA* datasets. Compared with *CPT-MFA*, *CPT-MPFA* datasets share the feature of pore water along the shaft *ua0*. The first experiment functions as the baseline model, using conventional *CPT* features only, whereas seven features are considered in the second experiment, including *qt, fs, u2, fs2, fs3, fs4*, and *ua0*. The results of eight different models are shown in Table 4 and 5, respectively. Again, an increase in model performance can be observed after using *MPFA* features, which suggests the importance of MPFA features in soil type classification.

The results presented in Tables 2, 3, 4, and 5 illustrate that AdaBoost, Random Forest, XGBoost, TabNet, and KNN are effective when applied to *CPT-MFA* and *CPT-MPFA* datasets. Notably, KNN and tree-based models like Random Forest, AdaBoost, and XGBoost show higher performance in comparison to TabNet. Among all the models, Random Forest stands out slightly, affirming its efficacy and reliability in soil classification tasks. Therefore, Random Forest is chosen as the optimal machine learning model for predicting soil type and will be used in the following sections to arrive at the optimal combination of *CPT* attachment sensors to appropriately classify soils.

4.2. *Feature selection*

The goal of this section is to minimize the number of sensors required for accurate soil type classification. Therefore, the focus is on training the machine learning model with the fewest possible features, while ensuring that classification accuracy remains uncompromised. In this section, the correlation between features is first analyzed. Then, two main experiments are conducted, where the first experiment uses both *CPT-MFA* and *CPT-MPFA* dataset and the second experiment only uses the *CPT-MPFA* dataset. In both experiments, 80% of the dataset is used as the training set and 20% as the test set. Based on the previous experiment results, Random Forest is used here to compare different combinations of features. The randomized search method is implemented to arrive at the best parameters where the training set is used to fit the randomized search and 10-fold cross-validation is implemented to find the best estimator.

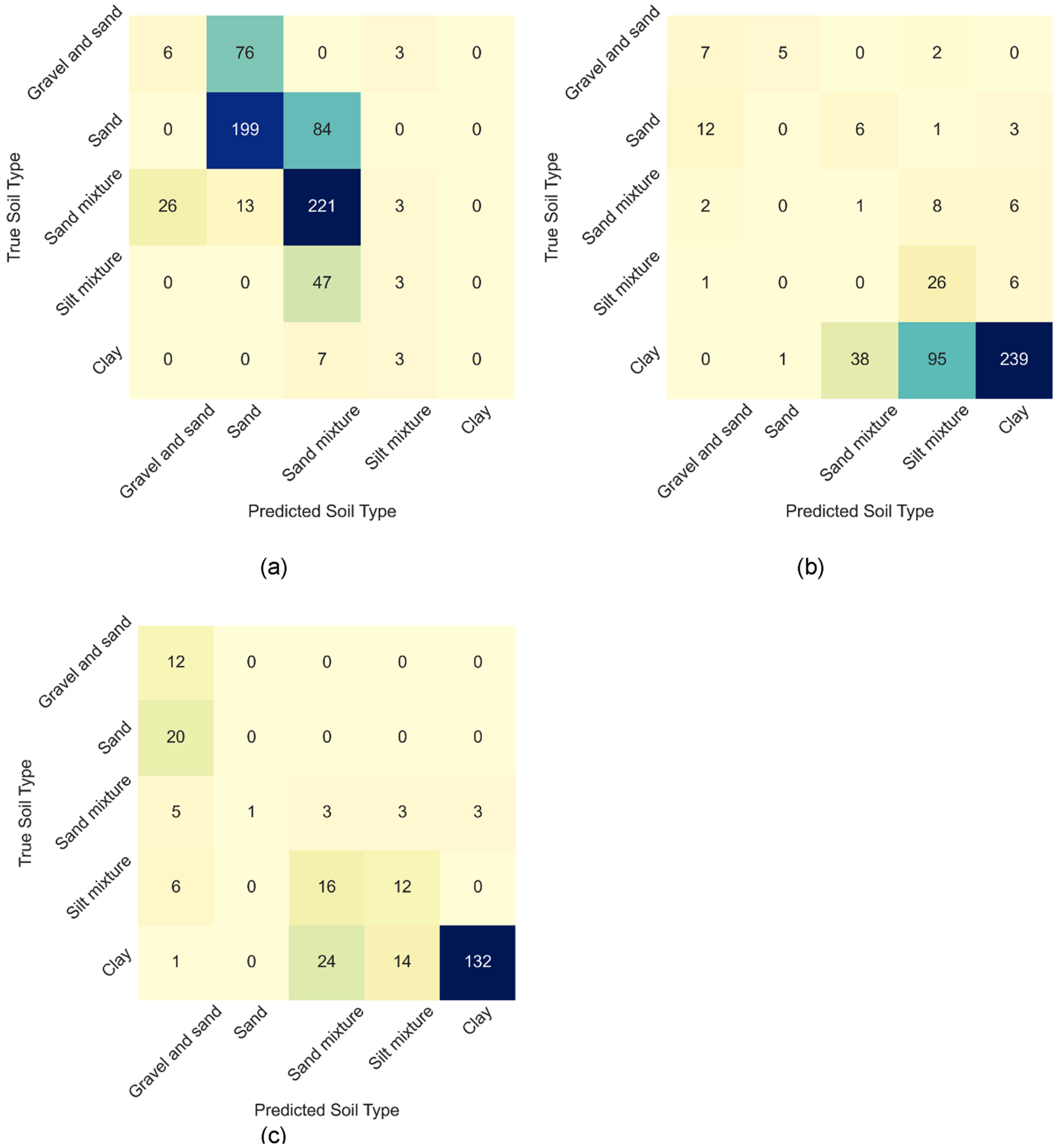

Fig. 10. Confusion matrix for experiment 1, 5 and 6 after down-sampling: (a) confusion matrix for experiment 1, (b) confusion matrix for experiment 5, (c) confusion matrix for experiment 6.

4.2.1. *Feature correlation*

There is a bilinear relationship between interface strength (*fs, fs1, fs2, fs3*, and *fs4*) and surface roughness (Uesugi & Kishida, 1986). Specifically, the interface strength increases linearly with increasing surface roughness up to a critical roughness value (Hebeler et al., 2018). Considering this factor, the relationships between *CPT* sleeve friction and multi-sleeve friction, as well as between pore pressure and dynamic pore pressure along the shaft, are investigated.

The correlation matrix for six features *qt, fs, u2, fs2, fs3* and *fs4* using the training set of all six datasets is shown in Fig. 4 (a). The analysis reveals a high correlation between *fs2* and *fs*, as well as between *fs2* and *fs3*, suggesting a redundancy in including these feature pairs in the multi-sleeve CPT device configuration. Similarly, the correlation matrix for six features, *qt, fs, u2, fs2, fs3, fs4* and *ua0* applied to the training set of *MPFA* datasets only is shown in Fig. 4 (b). Fs2 and *fs3* are highly correlated with a correlation value 0.92, and *u2* and *ua0* are highly correlated with a correlation value 0.98. This indicates that one of the features in each pair could potentially be omitted without significant loss of information.

4.2.2. *Classification results*

Tables 6 and 7 present the precision, recall, F1 score, and accuracy results for classifications using various feature combinations from both the *CPT-MFA* and *CPT-MPFA* datasets. Table 6 shows the results trained with conventional *CPT* measurements only, whereas Table 7 shows the results trained with conventional *CPT* measurements and *MFA* measurements. When compared to models trained solely on conventional *CPT* measurements, there is a marked improvement in performance when using the *CPT-MFA* measurements, as shown in Fig. 5. This notable improvement underscores the necessity and effectiveness of MFA measurements. Tables 6 and 7 also indicate a decline in performance when fewer features are employed for model training. In Table 7, by using the combination of *qt, fs, u2*, and either *fs3* or *fs4*, the accuracy peaks at 0.92. This is still relatively close to the maximum accuracy of 0.94, which

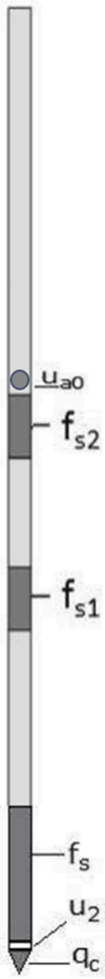

Fig. 11. Proposed design configuration of multi-sleeve cone penetrometer probe attachments.

demands an additional feature for a mere 2% accuracy boost. Given the objective of this study—to identify the most efficient feature combination with the fewest features—the combination of *qt, fs, u2*, and either *fs3* or *fs4* emerges as the optimal choice.

Tables 8 and 9 show the precision, recall, F1 score, and accuracy for classification using different feature combinations with *CPT-MPFA* datasets. Table 8 presents the results trained with conventional *CPT* measurements only whereas Table 9 presents the results trained with conventional *CPT* measurements and *MPFA* measurements. In Table 9, the highest accuracy of 0.96 is reached by the feature combination of *qt, fs, u2, fs3, fs4 and ua0*. Therefore, this combination of *qt, fs, u2, fs3, fs4* and is considered as the optimal choice for the dataset collected by *CPT-MPFA* device. Fig. 6 shows the increase in model accuracy after adding *MPFA* measurements into the training, which suggests the importance of MPFA measurements.

4.2.3. Including depth as a feature

When collecting *CPT* data, the depth of each data point is also a feature that can be considered a factor influencing the soil type classification. Depth not only serves as an indicator of the vertical variability of the test site, but also offers insight into the vertical stress levels present, which is intrinsically linked to *qt* and *fs* measurements. Soil samples at similar depths are often subject to comparable vertical stress conditions, thereby influencing the *qt* and *fs* readings. This relationship underscores the relevance of depth in the context of CPT data analysis. Table 10 shows the model performance with and without having depth as a training feature. An increase in model performance can be observed after including depth as a training feature.

However, it is crucial to acknowledge that depth can also introduce a spatial bias in the model. Sites that are close to each other are likely to exhibit similar soil distributions at equivalent depths due to higher spatial correlation. On the other hand, sites that are geographically distant might display a diverse range of soil types at the same depth, complicating the inclusion of depth as a training feature.

In conclusion, while depth can be a contentious feature due to potential spatial biases, its relationship with vertical stress and subsequent impact on *qt* measurements make it a valuable feature for enhancing model accuracy in soil type classifications.

4.3. Application – classification of unseen CPT dataset

The final phase of this work is to predict the soil types with a new *CPT*-sounding result based on the best-selected model and the optimal feature combination. To this end, in this section, instead of building the test set randomly, a complete *CPT*-sounding dataset will be used as the test set for the experiments. As shown in Table 11, each of the six sounding datasets is considered the test set, with the other five datasets as the training set, respectively. Random forest is used as the training model per Section 4.1, and the feature combination of *qt, fs, u2* and *fs3* is used per Section 4.2. The classification precision, recall, F1 score, and accuracy metrics achieved in each case are also depicted in Table 11.

The results in Table 11 show a classic example of class imbalance in machine learning and the spatial variation of soil property (Ching et al., 2023). The accuracy of the models drops to between 55% and 75% from the best-case scenario of 94%, as shown in Table 7. The recall, precision, and F1 score decrease to about 25–55% due to the influence of the imbalanced dataset. Experiments 1, 2, 5, and 6 in Table 11 get unexpectedly poor results. Therefore, confusion matrices are used to investigate further the frequency and reasons behind those incorrect predictions.

The confusion matrices for experiments 1 and 2, as depicted in Fig. 7, reveal a tendency of the models to incorrectly classify numerous data points as ’Sand’. Notably, out of 829 data points categorized as ’Sand’, 35.7% are from dataset #1, and 34.1% are from dataset #2. When these datasets are used as the test set, the model’s capacity to differentiate ’Sand’ from other soil types is substantially diminished due to the limited representation of ’Sand’ in the training set. This imbalance leads to a disproportionate focus on the ’Sand’ classification during model training, resulting in a higher rate of misclassification towards the ’Sand’ category in the test sets.

The confusion matrices for experiments 5 and 6, shown in Fig. 8, indicate a significant misclassification issue, with a notable number of data points, particularly those labeled as ’Clay’, being incorrectly identified as ’Sand mixture’. Out of 593 data points classified as ’Clay’, 28.8% are from dataset #6, and 62.9% are from dataset #5. When these specific datasets are used as test sets, the model’s proficiency in accurately identifying ’Clay’ is compromised. This is compounded by the fact that a substantial portion of the test set comprises ’Clay’ data points. Additionally, ’Sand mixture’ represents the most prevalent category across all six datasets (as illustrated in Fig. 2), leading to a bias in the model’s predictions, where ’Clay’ is frequently misclassified as ’Sand mixture’.

The observed misclassifications in the experiments can be attributed to the method of manually selecting training and test sets from *CPT* soundings at various locations in the United States and Australia, rather than random selection from a combined dataset. This approach results in significantly different distributions between the training and test sets, adversely affecting the accuracy. Typically, enhancing the accuracy in such scenarios would involve increasing the number of uniformly distributed data points across all classes. However, due to the limited availability of data points for multi-sleeve *CPT* in the literature, this study has instead explored resampling methods to achieve a more balanced distribution of class labels, aiming to improve the model performance.

4.3.1. Results from up-sampling and down-sampling of the datasets

In this section, three sampling methods are investigated to determine their impact on the model's performance for an unseen dataset. Table 12 shows the accuracy of each experiment before and after down-sampling, up-sampling and combined sampling.

Overall, sampling approaches can improve the model performance for unseen datasets. In most of the experiments, the highest accuracy is reached by either up-sampling or down-sampling. The performance of combined sampling is intermediate between the performance with up-sampling or down-sampling.

In experiments 1, 4, 5, and 6, the accuracy after up-sampling decreases compared with the accuracy without resampling, especially for experiment 1, 5 and 6. For experiment 5 and 6 where the test set is mainly 'Clay', the accuracy drops to 0.14 and 0.42 respectively. The models perform worse after up-sampling. From the confusion matrices shown in Fig. 9, more clay data points are predicted as 'Sand mixture' and 'Silt mixture'. For experiment 1, more data points are mis-predicted as 'Sand' after up-sampling. The result indicates that the synthetic clay data points generated by *SMOTE* in experiments 5 and 6 cannot correctly represent the 'Clay' data points in the test set. The same reasoning applies to the synthetic sand data points generated in experiment 1. Therefore, whether the synthetic data points can convincingly represent the real data points using *SMOTE* on highly imbalanced data should be taken into consideration.

Experiments 3 and 4 after down-sampling have a similar accuracy as up-sampling or without any resampling methods. The accuracy of experiments 1, 5, and 6 after down-sampling are higher compared to the results after oversampling. The confusion matrices of experiments 5 and 6 are shown in Fig. 10. Compared to the confusion matrix after upsampling, the clay data points are predicted more accurately instead of being mis-predicted as sand mixtures or silt mixtures. For experiment 1, more sand mixtures data points are correctly predicted, as shown in Fig. 10a. The result shows that the minority data points ('Clay' in experiments 5 and 6; 'Sand' in experiment 1) can be more sufficiently learned by the models after down-sampling.

5. Discussion - proposed design configuration of multi-sleeve cone penetrometer probe attachments

The analysis presented in Section 4 noted that the optimal feature combination of *CPT-MFA* device is *qt, fs, u2, fs3* and the optimal feature combination of *CPT-MPFA* device is *qt, fs, u2, fs3, fs4, ua0*. Reflecting these optimal configurations, Fig. 11 shows the proposed design configuration of a new multi-sleeve cone penetrometer probe attachment, with two friction sleeves and one independent measure of dynamic pore water pressure along the shaft. This proposed configuration can allow for superior classification of soil type and other engineering properties during site characterization without the electronic complexity and potential low robustness associated with the original 7 data stream *MFA-CPT* or 12 stream *MPFA-CPT* versions. Specifically, the proposed design results in a 6 data stream configuration consisting of a tip resistance measurement (*qt*), 3 friction sleeve measurements (*fs, fs1*, and *fs2*) using sleeves with increasing roughness and two pore pressure sensors (*u2*, and *ua0*), one measuring pore pressure in the tip region and the other measuring pore pressure generated after shearing against the most heavily textured sleeve. The additional friction sleeves and pore pressure sensor can be readily configured into a simpler attachment used behind a conventional *CPT* to improve the overall use of the device data while avoiding challenging operational issues. This can still allow for full quantification of the interface friction versus surface roughness relationship for all soils and simultaneously yield two independent measurements of pore pressure generated, one due to probe tip advancement and the other due to sleeve induced soil shearing.

6. Conclusions

The research was primarily centered on identifying the minimal yet crucial set of sensor measurements for multi-sleeve CPT devices. This was achieved by utilizing the performance of machine learning models in soil type classification as a key metric to guide the optimization process. Through a comprehensive analysis of various machine learning models and feature combinations, the study revealed that a reduced number of sensors can achieve comparable classification performance to more complex configurations. Specifically, for the CPT-MFA device, the optimal feature combination was identified as *qt*, *fs*, *u2*, and *fs3*, while for the CPT-MPFA device, it was *qt, fs, u2, fs3, fs4,* and *ua0*. Based on the findings, a new configuration for a multi-sleeve attachment for use in conjunction with a conventional CPT was identified. It consisted of a simple attachment with two additional friction sleeves and one pore pressure sensor. The proposed configuration addressed the challenges of reducing electronic complexity, time-consuming assembly, and the susceptibility of the device to sensor stream losses while maintaining robustness at the same time.

In pursuing the objective of optimizing multi-sleeve CPT devices, this research also sheds light on the efficacy of various machine learning techniques in soil type classification. While the deep learning model TabNet showed promise in predicting soil types using multi-sensor CPT data, KNN and traditional tree-based models like Random Forest, AdaBoost, and XGBoost demonstrated superior performance in this domain. Additionally, the resampling technique can somewhat improve the classification accuracy of unseen soil datasets but not fully overcome the deficiency in an initially imbalanced dataset.

In conclusion, this research marks a significant step forward in the field of geotechnical engineering, illustrating how the integration of machine learning can effectively guide the design and optimization of geotechnical instruments. Reducing sensor complexity while maintaining the performance in soil type classification leads to more cost-effective and efficient geotechnical practice, potentially benefiting the geo-infrastructure construction industry. This work not only offers practical solutions for optimizing geotechnical instrument designs but also paves the way for more sophisticated and data-driven approaches in geotechnical practice.

7. Declaration of Generative AI and AI-assisted technologies in the writing process

During the preparation of this work the authors used ChatGPT in order to improve readability and language. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

CRediT authorship contribution statement

Danrong Zhang: . **Nimisha Roy:** Data curation, Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **J. David Frost:** Conceptualization, Methodology, Formal analysis, Investigation, Resources, Writing – original draft, Writing – review & editing, Supervision, Project administration, Funding acquisition.

Declaration of competing interest

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.

Data availability

Data will be made available on request.

Acknowledgments

The work described in this manuscript was supported in part by the US National Science Foundation through Grant CMMI 1826118. This support is gratefully acknowledged. The opinions are those of the authors and not the sponsor. The contributions of former PhD students Jason DeJong (University of California, Davis), Gregory Hebeler (WSP), and Alejandro Martinez (University of California, Davis), who each conducted lab and field experiments as well as numerical simulations and analysis associated with the successive generations of the various multi-sleeve devices are acknowledged.

Appendix A

The CPT data used in this work is collected by DeJong (2001). Fig. A.1a, b and c show the soil profile of Vermont, South Carolina, and Western Australia respectively. While all three sites encompass the five soil types, there is a noticeable imbalance in the distribution of these soil types across the different locations, which is also indicated in Fig. 2.

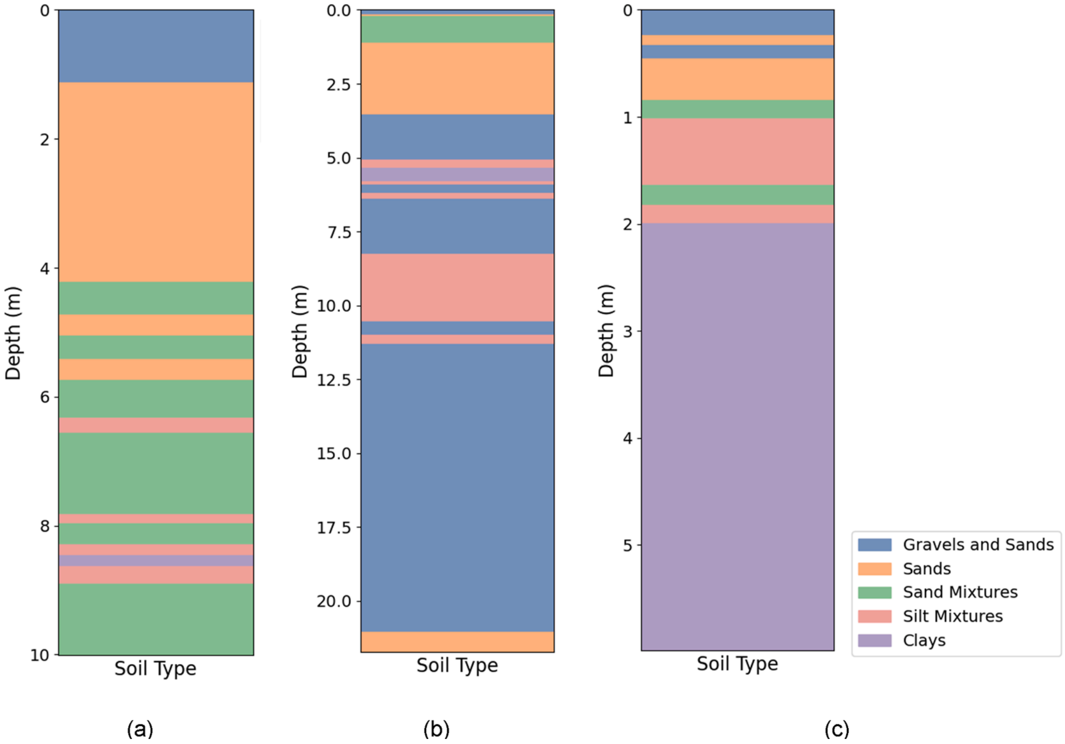

Fig. A1. Soil profile for the sites in this work: (a) Vermont; (b) South Carolina; (c) West Australia.

Appendix B

Fig. B.1 to Fig. B.6 shows the sensor record streams of dataset 1 to dataset 6, respectively. For *MFA-CPT* device, it shows the sensor record streams of *qt, u2, fs, fs2, fs3,* and *fs4.* For *MPFA-CPT* device, it shows the sensor record streams of *qt, u2, fs, fs2, fs3, fs4,* and *ua0* (DeJong, 2001). The soundings reveal notable correlations among sleeve stresses *(fs, fs2, fs3, fs4)* as well as between pore pressures *(u2, ua0)*, underscoring the critical need for optimization of *MFA* and *MPFA* devices.

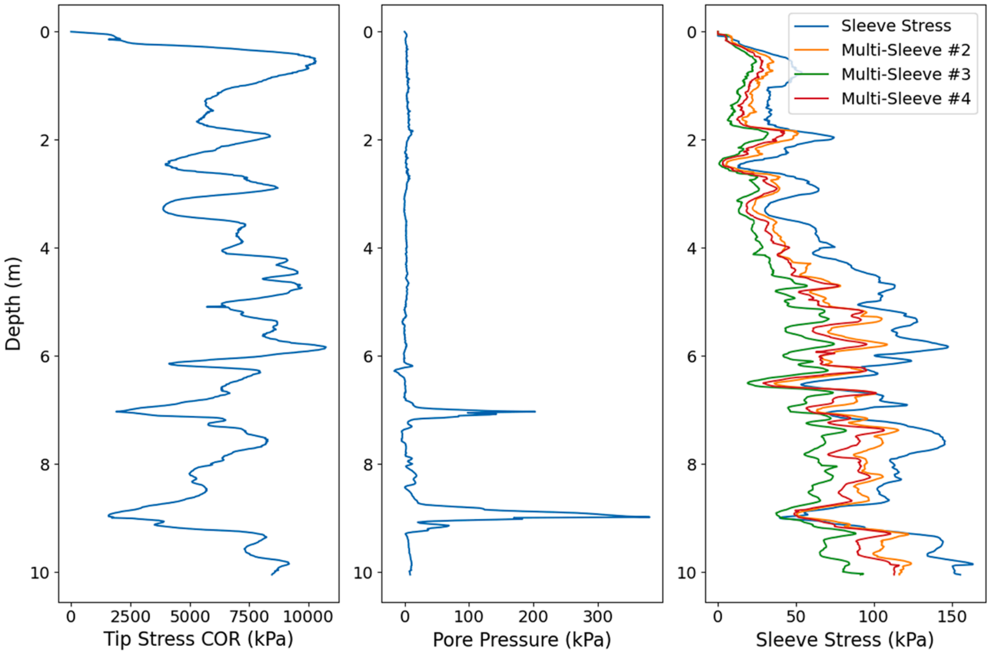

Fig. B1. Sensor record streams for dataset 1

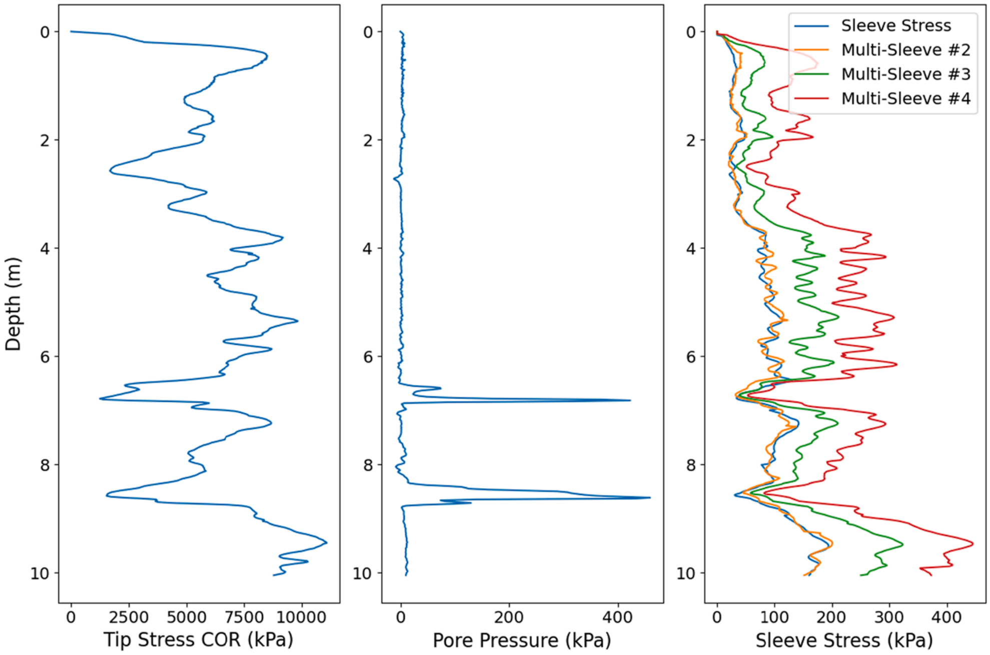

Fig. B2. Sensor record streams for dataset 2

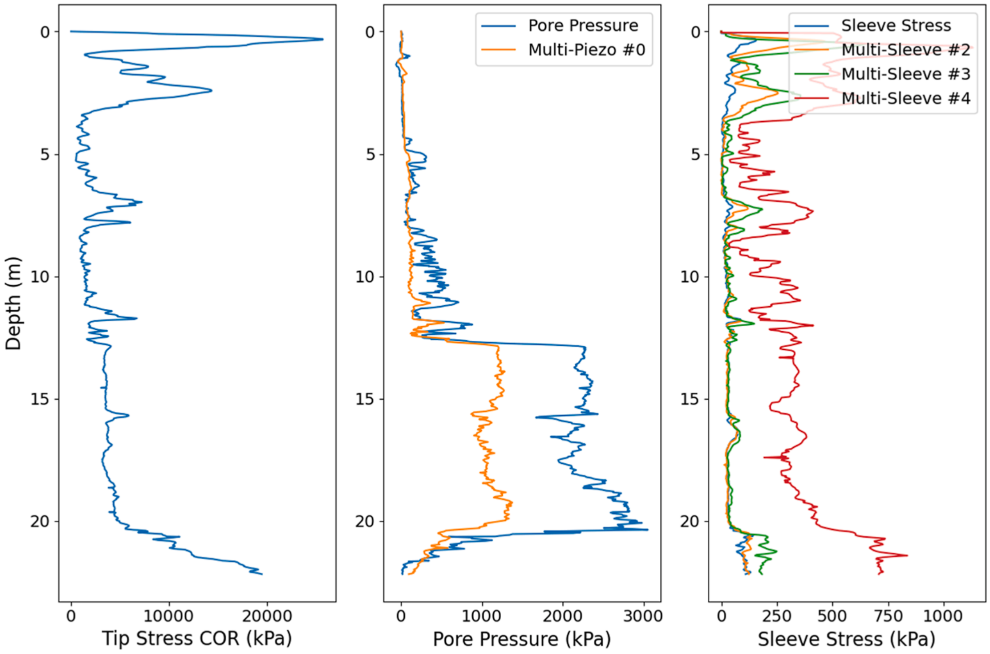

Fig. B3. Sensor record streams for dataset 3

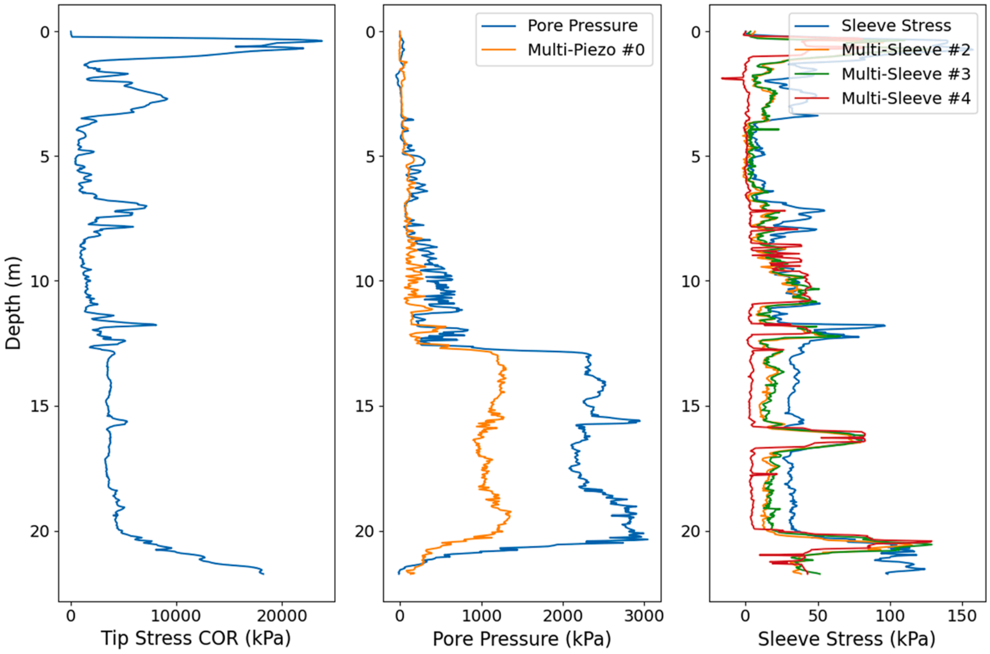

Fig. B4. Sensor record streams for dataset 4

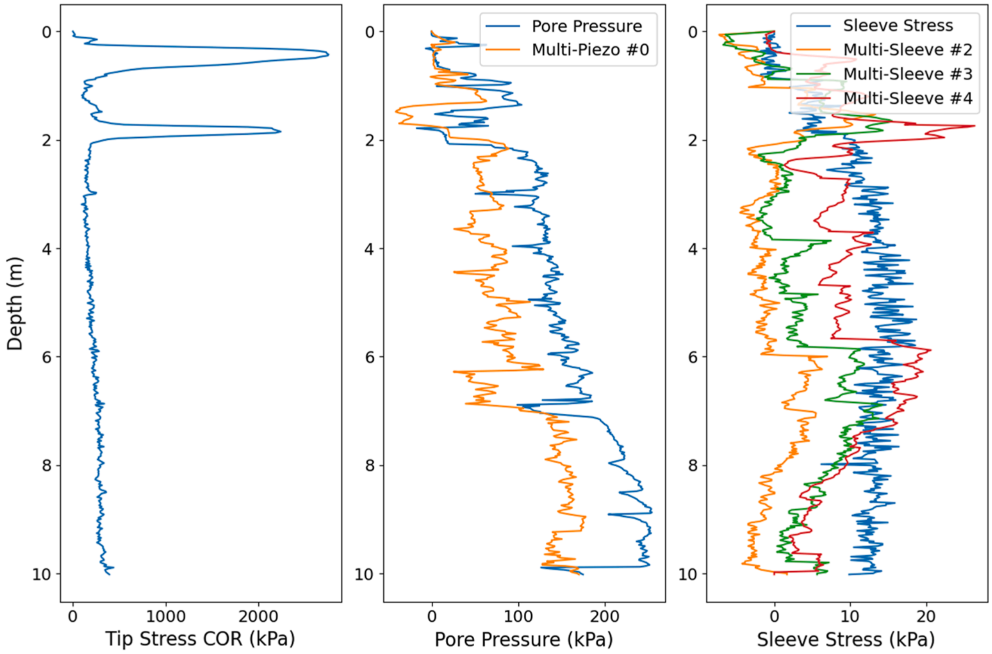

Fig. B5. Sensor record streams for dataset 5

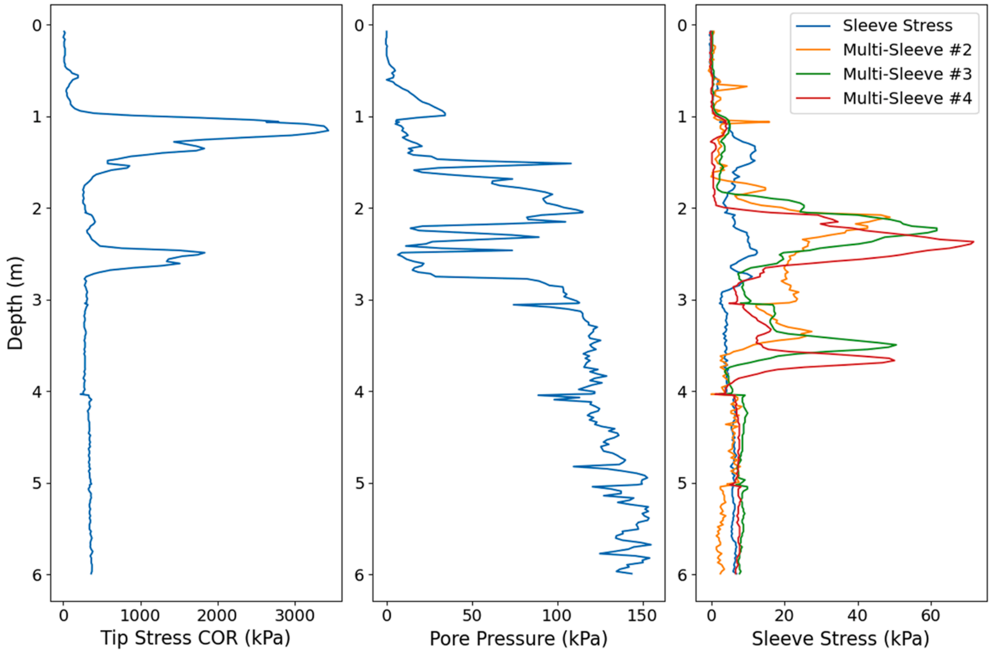

Fig. B6. Sensor record streams for dataset 6

Appendix C

Tables C.1–C.8 detail the definitions of hyperparameters for each machine learning model employed in this study, specifically: decision tree, AdaBoost, random forest, XGBoost, TabNet, KNN, SVM, and conventional neural network. Tables C.9 further delineates the hyperparameter search space for each model, enhancing the credibility and robustness of the model performance.

Table C1
Definition of decision tree hyperparameters.

Hyperparameter	Definition
Max_depth	The maximum depth of the tree, controlling overfitting by limiting the complexity.
Min_samples_split	The minimum number of samples required to split an internal node.
Min_samples_leaf	The minimum number of samples a leaf node must have.
Criterion	The function to measure the quality of a split.

Table C2
Definition of AdaBoost hyperparameters.

Hyperparameter	Definition
n_estimators	The number of base estimators (decision trees) in the ensemble.
learning_rate	The weight assigned to each classifier during each stage of the boosting process.

Table C3
Definition of random forest hyperparameters.

Hyperparameter	Definition
n_estimators	The number of trees in the forest.
max_features	The number of features when searching for the best split.
max_depth	The maximum depth of each tree.
min_samples_split	The minimum number of samples to split a node.
min_samples_leaf	The minimum number of samples required at each leaf node.

Table C4
Definition of XGBoost hyperparameters.

Hyperparameter	Definition
n_estimators	The number of boosting rounds.
learning_rate	The step size at each iteration while moving towards a minimum of a loss function.
max_depth	The maximum depth of each tree.
gamma	The minimum loss reduction needed to make partition on a leaf node.
subsample	The fraction of the training data to be randomly sampled for each tree.
colsample_bytree	The fraction of features to be randomly sampled for each tree.

Table C5
Definition of Tabnet hyperparameters.

Hyperparameter	Definition
N_d	The dimension of the decision prediction layer, influencing the model complexity.
N_a	The dimension of the attention embedding, affecting how the model focuses on input features.
gamma	The factor showing the sparsity of feature reusage in the masks.
N_steps	The number of steps in the model.

Table C6
Definition of KNN hyperparameters.

Hyperparameter	Definition
n_neighbors	The number of nearest neighbors to consider in the voting process.
weights	The weight each neighbor has in the voting process.
metric	The distance metric to calculate the proximity between data points.

Table C7
Definition of SVM hyperparameters.

Hyperparameter	Definition
C	The regularization parameter controlling the trade-off between achieving a low error on the training data and minimizing the norm of the weights.
gamma	The kernel coefficient showing how far the influence of a single training example reaches.

Table C8
Definition of Conventional neural network hyperparameters.

Hyperparameter	Definition
Hidden_layer_sizes	The number of neurons in each hidden layer.
Activation	The activation function for the neurons.
Solver	The algorithm for weight optimization.
Alpha	The L2 regularization term, which helps prevent overfitting by penalizing large weights.
Learning_rate	The step size at each iteration while moving towards a minimum of the loss function.

Table C9
Hyperparameter search space.

Model	Hyperparameters	Range
Decision tree	Max_depth	Integers from 5 to 20 (11 values)
	Min_samples_split	[2, 5, 10, 15, 20]
	Min_samples_leaf	[1, 2, 4, 6, 8]
	Criterion	[gini, entropy]
AdaBoost	Max_depth for base learner	[1, 2, 3, 4, 5]
	n_estimators	Integers from 30 to 300 (10 values)
	learning_rate	[0.01, 0.1, 0.5, 1.0]
Random Forest	n_estimators	Integers from 30 to 300 (10 values)
	max_features	[auto, sqrt]
	max_depth	Integers from 5 to 20 (11 values)

(continued on next page)

Table C9 (*continued*)

Model	Hyperparameters	Range
	min_samples_split	[2, 5, 10]
	min_samples_leaf	[1, 2, 4]
XGBoost	n_estimators	Integers from 30 to 300 (10 values)
	learning_rate	[0.01, 0.05, 0.1, 0.2]
	max_depth	Integers from 5 to 20 (11 values)
	gamma	[0, 0.1, 0.2, 0.3, 0.4]
	subsample	[0.6, 0.8, 1.0]
	colsample_bytree	[0.6, 0.8, 1.0]
TabNet	N_d	[8, 16, 32]
	N_a	[8, 16, 32]
	gamma	[1, 1.5]
	N_steps	[3, 4]
KNN	n_neighbors	[5, 7, 9, 11]
	weights	[uniform, distance]
	metric	[Euclidean, manhattan]
SVM	C	[0.1, 1, 10, 100, 1000]
	gamma	[1, 0.1, 0.01, 0.001, 0.0001]
Conventional Neural Network	Hidden_layer_sizes	[(10, 30, 10), (20,)]
	Activation	[tanh, relu]
	Solver	[sgd, adam]
	Alpha	[0.0001, 0.05]
	Learning_rate	[constant, adaptive]

References

Arik, S.Ö., Pfister, T., 2021. Tabnet: attentive interpretable tabular learning. Proceedings of the AAAI Conference on Artificial Intelligence.

Begemann, H., 1965. The friction jacket cone as an aid in determining the soil profile. Proc. 6th Int. Conf. on SMFE.

Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P., 2002. SMOTE: synthetic minority over-sampling technique. J. Artif. Intell. Res. 16, 321–357.

Ching, J., Uzielli, M., Phoon, K.-K., Xu, X., 2023. Characterization of autocovariance parameters of detrended cone tip resistance from a global CPT database. J. Geotech. Geoenviron. Eng. 149 (10), 04023090.

Cohen, I., Huang, Y., Chen, J., Benesty, J., Benesty, J., Chen, J., Huang, Y., Cohen, I., 2009. Pearson correlation coefficient. Noise Reduct. Speech Process. 1–4.

DeJong, J.T., Frost, J.D., 2002. A multisleeve friction attachment for the cone penetrometer. Geotech. Test. J. 25 (2), 111–127.

DeJong, J.T., 2001. Investigation of particulate-continuum interface mechanisms and their assessment through a multi-friction sleeve penetrometer attachment. PhD DSissertation. Georgia Institute of Technology, 360 pp.

Douglas, B. J., 1981. Soil classification using electric cone penetrometer. In: Sympsium on Cone Penetration Testing and Experience, Geotechnical Engineering Division, ASCE, St. Louis, Oct.

Frost, J., Martínez, A., Hebeler, G., 2013. Cyclic multi-piezo-friction sleeve penetrometer testing for liquefaction assessment. Geotechnical and Geophysical Site Characterization: Proceedings of the 4th International Conference on Site Characterization ISC-4.

Giustolisi, O., Doglioni, A., Savic, D.A., Webb, B., 2007. A multi-model approach to analysis of environmental phenomena. Environ. Model. Softw. 22 (5), 674–682.

Gorishniy, Y., Rubachev, I., Khrulkov, V., Babenko, A., 2021. Revisiting deep learning models for tabular data. Adv. Neural Inf. Proces. Syst. 34, 18932–18943.

Hebeler, G.L., 2005. Multi Scale Investigations of Interface Behavior. Georgia Institute of Technology, Atlanta, p. 772 pp.. PhD Dissertation.

Hebeler, G.L., Frost, J.D., 2006. A multi piezo friction attachment for penetration testing. In: GeoCongress 2006: Geotechnical Engineering in the Information Technology Age, pp. 1–6.

Hebeler, G.L., Martinez, A., Frost, J.D., 2018. Interface response-based soil classification framework. Can. Geotech. J. 55 (12), 1795–1811.

Laucelli, D. B., Enríquez, L., Saldarriaga, J., Giustolisi, O., 2023. Using symbolic machine learning to assess and model substance transport and decay in water distribution networks.

Moon, J.-S., Kim, C.-H., Kim, Y.-S., 2022. Soil classification from piezocone penetration test using fuzzy clustering and neuro-fuzzy theory. Appl. Sci. 12 (8), 4023.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., 2011. Scikit-learn: machine learning in Python. J. Mach. Learn. Res. 12, 2825–2830.

Rauter, S., Tschuchnigg, F., 2021. CPT data interpretation employing different machine learning techniques. Geosciences 11 (7), 265.

Reale, C., Gavin, K., Librić, L., Jurić-Kaćunić, D., 2018. Automatic classification of fine-grained soils using CPT measurements and Artificial Neural Networks. Adv. Eng. Inf. 36, 207–215.

Robertson, P., 1991. Soil classification using the cone penetration test: Reply. Can. Geotech. J. 28 (1), 176–178.

Shwartz-Ziv, R., Armon, A., 2022. Tabular data: Deep learning is not all you need. Information Fusion 81, 84–90.

Tsiaousi, D., Travasarou, T., Drosos, V., Ugalde, J., Chacko, J., 2018. Machine learning applications for site characterization based on CPT data. In: Geotechnical Earthquake Engineering and Soil Dynamics V. American Society of Civil Engineers Reston, VA, pp. 461–472.

Uesugi, M., Kishida, H., 1986. Influential factors of friction between steel and dry sands. Soils Found. 26 (2), 33–46.

Wu, S., Zhang, J.-M., Wang, R., 2021. Machine learning method for CPTu based 3D stratification of New Zealand geotechnical database sites. Adv. Eng. Inf. 50, 101397.

Zhang, W., Wu, C., Zhong, H., Li, Y., Wang, L., 2021. Prediction of undrained shear strength using extreme gradient boosting and random forest based on Bayesian optimization. Geosci. Front. 12 (1), 469–477.