Journal of Civil, Construction and Environmental Engineering
2025, Vol. 10, No. 3, pp. 104-114
https://doi.org/10.11648/j.jccee.20251003.11

Research Article

A Machine Learning Model for Pile Settlement Prediction Using Majority Voting-Based Feature Selection

Hafeez Husain Bello[1, 2, *], You Wang[1], Shamsudeen Lawal[3]

[1]School of Civil Engineering, Central South University, Changsha, China
[2]Department of Civil Engineering, Ahmadu Bello University, Zaria, Nigeria
[3]School of Traffic and Transportation Engineering, Central South University, Changsha, China

Abstract

Pile foundations are deep foundations commonly employed in bridge construction, high-rise buildings, trains, and situations requiring high bearing capacity and minimal settlement. Accurate prediction of pile settlement is essential for ensuring the safety and stability of deep foundations, yet traditional methods like in-situ load tests are often costly and impractical. The cone penetration test (CPT) is one of the most frequent in-situ tests for pile analysis because, like a model pile, the measured cone resistance and sleeve fiction can be used to estimate pile unit toe and shaft resistances, respectively. In this paper, a machine learning (ML) framework for pile settlement prediction with a genetic algorithm (GA) majority voting (MV) feature selection (FS) strategy to enhance model performance is presented. Three tree-based algorithms, each with a unique approach for tree development and feature handling—categorical boosting (CB), light gradient boosting (LGB), and random forest (RF) are selected for this purpose. The dataset was compiled from fifty-six pile case histories in different countries have been compiled including static loading tests which include maintained load tests and constant rate of penetration tests, shaft, and toe resistances which comprise CPT and CPTu (undrained CPT) sounding, the pile geometric and mechanical properties, the loads applied from the load tests as the model inputs, and recorded settlement values for the piles from the tests as the model output to be predicted. The CB model, coupled with the GA-MV approach, achieved the best predictive accuracy, yielding an R^2 of 0.926 and RMSE of 5.92 mm upon testing, while feature importance analysis identifies applied load (P) and pile length (L) as key predictors of settlement. Also, an overall decrease of the RMSE by 11.19% was observed between the CB-GAMV model (5.92 mm) and the CB-All features model (6.68 mm), and 9.41% between the CB-GAMV model and the CB-GA model (6.54 mm) on the validation set.

Keywords

Pile Foundation Settlement, Cone Penetration Test, Machine Learning, Majority Voting, Genetic Algorithm

1. Introduction

Piles, which serve as deep foundation elements, are vital for ensuring structural stability by transferring axial loads through unsuitable soil conditions or water to stronger, load-bearing strata. A fundamental requirement in pile design is the evaluation of load-settlement behavior, which is influenced by multiple factors. These include the non-linear re-

*Corresponding author: hhbello@csu.edu.cn (Hafeez Husain Bello)

Received: 25 March 2025; **Accepted:** 6 May 2025; **Published:** 11 June 2025

sponse of soil due to its inherent variability, the structural properties of the pile, the installation technique, and the nature of the load test [1]. Traditionally, in-situ load tests are used to assess this behavior. However, these methods are costly, time-consuming, and often impractical for smaller projects [2]. According to Poulos and Davis [3], the immediate (elastic) settlement constitutes the primary component of total pile settlement, with consolidation settlement contributing further in clayey soils. In contrast, Murthy [4] notes that for piles in sandy soils, the immediate settlement accounts for nearly the entirety of the final settlement.

Various approaches have been explored to model the load-settlement behavior of piles and enhance prediction reliability. These include empirical and semi-empirical methods that utilize data from in-situ tests such as the standard penetration test (SPT), cone penetration test (CPT), and pile geometry parameters [5-9]. Among these, CPT serves as a critical method for subsurface soil characterization and pile load-settlement prediction [3]. The test involves pushing a cone-tipped rod into the ground at a steady rate while recording cone-tip resistance (q_c) and sleeve friction (f_s). These measurements provide essential information on soil density, shear strength, and compressibility, which are fundamental for accurate pile design and performance assessment.

In recent years there has been a surge in interest in applying ML and artificial intelligence (AI) techniques to address challenges in civil engineering. Various ML-based approaches have been employed to solve diverse engineering problems with varying degrees of success. Among these, artificial neural networks (ANN) have been extensively utilized in predicting pile foundation behavior, including bearing capacity and settlement [10-13]. Additionally, other ML algorithms, such as support vector machines (SVM), k-nearest neighbors (KNN), decision trees (DT), random forests (RF), gradient boosting machines (GBM), and extreme gradient boosting (XGBoost), have also been explored in related studies [14-18].

Ardalan [19] employed polynomial neural networks (NN) combined with genetic algorithms (GA) to predict pile shaft capacity using cone penetration test (CPT) and undrained shear strength data. Meanwhile, Nejad and Jaska developed two distinct ANN models to simulate the full load-settlement response of piles, incorporating SPT data [20] and CPT data [21] from published literature. These models considered test type, pile characteristics, installation method, and soil parameters along the pile length to account for soil variability. The ANN models demonstrated superior predictive accuracy compared to conventional prediction methods.

Kardani [22] explored the potential of enhancing six machine learning algorithms through particle swarm optimization (PSO) for predicting pile bearing capacity in cohesionless soils. The study identified effective stress as the most significant predictive parameter and demonstrated that the optimized XGB model achieved superior performance, exceeding the predictive capability of the conventional β-method. Ismail [23] employed a hybrid PSO/higher-order neural network approach to model the load-settlement behavior of concrete piles using SPT data, incorporating considerations of load test type and installation effects on pile performance. Zhang [24] implemented a neural network-based time series model to forecast settlement in a chimney foundation, utilizing nine months of field monitoring data for model development and validation.

2. Dataset Description

The database used for the analysis in this study was adopted from Nejad and Jaska [21]. This database includes 499 pile settlement historical cases from 24 piling projects with 56 tests in various countries including Australia, Belgium, Canada, Ireland, Italy, Japan, the Netherlands, the UAE, the UK, and the USA.

2.1. Dataset Properties

The dataset includes categorical variables representing pile characteristics that influence load-settlement behavior. These include the type of load test (maintained load test and constant rate of penetration test), pile type (concrete, composite, and steel), installation type (bored and driven), and pile end type (closed and open). Each category exhibits distinct settlement responses under applied loads.

The mechanical properties of the piles are defined by five key parameters: axial rigidity (*EA*), cross-sectional area at the tip (A_{tip}), perimeter in contact with soil (*O*), total length (*L*), and embedded length (*Le*). *EA*, the product of the material's modulus of elasticity and cross-sectional area, determines resistance to axial deformation. A_{tip} influences end-bearing capacity by increasing soil contact at the pile tip. *O* governs side friction resistance, which is crucial for friction piles. *L* dictates the depth of load transfer, while *Le* represents the portion embedded in the ground, affecting both frictional and end-bearing resistance.

Soil properties are represented by cone resistance (q_c) and sleeve friction (f_s) from CPT tests, which provide insights into soil strength. Higher q_c values indicate denser soils with strong end-bearing capacity, while lower values suggest compressible soils prone to settlement. The sleeve friction (f_s) assesses side friction resistance, which enhances load-bearing capacity. The soil properties are analyzed along five segments of the embedded pile length, with qc_{tip} representing the cone resistance at the pile tip using the Bustamante and Gisanelli method [25].

In total, the input features are 21: the type of test (TT), type of pile (TP), type of installation (TI), end of pile (PE), axial rigidity (*EA*), cross-sectional area of the end of the pile (A_{tip}), perimeter of the pile in contact with the soil (*O*), length of the pile (*L*), embedded length of the pile (L_{embed}), the averaged CPT results along the embedded length of the pile ($q_{c_1}, f_{s_1}, q_{c_2}, f_{s_2}, q_{c_3}, f_{s_3}, q_{c_4}, f_{s_4}, q_{c_5}, f_{s_5}$), cone tip resistance

at the end of the pile ($q_{c_{tip}}$), (21) the applied load (P), and the pile settlement (S) as the output to be predicted.

2.2. Data Preprocessing

The predictive performance of the model is directly related to the quality of input data samples. The database is divided into three sets—training, testing, and validation by Nejad et al. using cross-validation as proposed by Stone. The train set and validation set are adopted from the database with 395 and 49 data samples respectively as divided by Nejad and Jaska [21].

The categorical variables need to be transformed into numerical variables before training so they can be recognized by the ML algorithms. The TT, TP, TI, and PE are encoded into separate numerical variables [26]. This means that the TT, TI, and PE with two categories each are encoded to have two binary variables, and TP with three categories is encoded into 3 binary variables. L_e

Once the categorical variables are transformed into numerical values, the datasets are scaled. It is worthy of note that tree-based algorithms are fairly insensitive to the feature scales, but that is not the case with algorithms like SVMs and ANNs. By scaling the data, it is ensured that each feature contributes equally to the distance metrics used by these algorithms, thereby enhancing their performance and convergence speed. Normalization was adopted for this study where the features are within a common scale of [0,1].

3. Methodology

3.1. Machine Learning Algorithms

The machine learning algorithms selected for this study are popular tree-based models which are generally good for FS and interpretability. These models build multiple decision trees (DT) rather than a single DT to improve the model's overall ability to learn from the data.

3.1.1. Random Forest

Random Forest (RF) is an ensemble learning method developed by Breiman [27] based on the random subspace method for constructing decision forests by Ho [28]. It uses a technique known as bootstrap aggregation (bagging) to ensemble weak learners (DT) which are randomly built and different from each other. Building a collection of DTs with controlled variations is the main goal of the RF where low variance results in low overfitting. Therefore, the RF model adjusts the over-fitting habit of the base DT model and generally outperforms it [29]. For each node of each tree, the algorithm chooses a random subset made up of n input features, and these features should be unique to assure the diversity of the DTs and avoid the correlation of the trees. As a result, each DT in the forest makes a decision independently, and the final result can be obtained by taking the average of the prediction of every tree. If one or several features in the dataset are powerful predictors for the target, these features will be selected to split examples in many DTs, resulting in several correlated DTs in the "forest". These correlated predictors do not aid in improving the RF model performance. Mathematically, for N decision trees (N_{trees}), the prediction (y_i) for a new example X_i is obtained by:

$$y = \frac{1}{N_{trees}} \sum_{i=1}^{N_{trees}} y_i X_i \quad (1)$$

3.1.2. Light Gradient Boosting

The light gradient boosting algorithm (LGB) first introduced by Ke et al. [30] is a gradient-based learning framework built upon decision trees and boosting techniques. Unlike the XGB model, LGB utilizes histogram-based algorithms that enhance the training speed, decrease memory usage, and employ a leaf-wise growth strategy with depth constraints. This approach eliminates the need for storing pre-sorted data and allows feature values after discretization to be stored using only 8-bit integers, resulting in a memory reduction of up to one-eight of the original. Additionally, it supports categorical features directly, avoiding the need for explicit one-hot encoding, which further improves its efficiency. The traditional growth strategy for decision trees is called level-wise, which is less efficient because it processes leaves at the same level simultaneously, leading to unnecessary memory consumption. In contrast, the leaf-wise strategy is more efficient as it selects the leaves with the highest gain at each step, continuing with the branching cycle.

This method typically results in better accuracy, as it reduces errors with fewer splits. However, the leaf-wise approach can lead to deeper trees, potentially causing overfitting. To address this, LGB imposes a maximum depth limit on the tree to balance high efficiency with overfitting prevention. For a given dataset $X = \{(x_i, y_i)\}_{i=1}^{m}$, LGB is designed to find an approximation $\hat{f}(x)$ of the function $f^*(x)$ that minimizes the expected values of a given loss function $L(y, f(x))$:

$$\hat{f}(x) \quad \arg\min_{f} E_{y,x} L(y, f(x)) \quad (2)$$

LGB combines multiple T regression trees to approximate the final model, which is defined as:

$$f_T(X) = \sum_{t=1}^{T} f_t(X) \quad (3)$$

3.1.3. Categorical Boosting

Categorical boosting [31] is a relatively new and powerful variant of gradient boosting decision trees (GBDT) developed by Yandex. The GBDT process involves minimizing the residuals or errors in the model predictions to continuously refine the model. In CB, modifications are made to the GBDT, where, *ordered boosting*, a permutation-based modification to optimize *prediction shift*, and an innovative algorithm, *ordered*

target statistics (OTS) for handling categorical features. Prediction shift is a particular type of target leakage that exists in all current variants of GBDT, and the CB algorithm addresses this issue.

To handle categorical features some statistics may be computed using label values of the examples as in equation (1). However, the problem with this is that it leads to the model overfitting. To reduce the overfitting tendency of the model, CB employs a more efficient strategy, OTS which allows the use of the entire dataset for training. Each example's target statistics (TS) values are dependent on the observed history. Therefore, a random permutation σ of the training examples is introduced as an artificial "time" to translate this concept to a normal offline situation. Then, each example's TS is computed using all of the available "history", i.e., take $D_k = \{x_i: \sigma_j < \sigma_k\}$ in (2), and $D_k = D$ a test example. A more detailed explanation of the implementation of the algorithm and tree building is provided in these studies [31, 32].

$$\hat{x}_j^i = \frac{\sum_{j=1}^{n} 1_{\{x_j^i = x_k^i\}} \cdot y_j}{\sum_{j=1}^{n} 1_{\{x_j^i = x_k^i\}}} \tag{4}$$

$$\hat{x}_j^i = \frac{\sum_{x_j \in D_k} 1_{\{x_j^i = x_k^i\}} \cdot y_j + \alpha p}{\sum_{x_j \in D_k} 1_{\{x_j^i = x_k^i\}} + \alpha} \tag{5}$$

3.2. Feature Selection with Genetic Algorithm

High dimensionality of the feature space increases the computational complexity of ML models, i.e., too many features increase the model complexity, and where possible, it is recommended to remove redundant or irrelevant features, thereby, reducing model complexity and computational time [33]. To reduce the dimensionality of the model, these features may need to be dropped before the prediction step. As such, this study adopted the use of GA to extract the most relevant features.

GA is amongst the most powerful global optimization techniques used to solve various optimization problems. GA is inspired by the process of natural selection and genetics, operating on a population of potential solutions encoded as chromosomes, and typically represented in binary form. This population is iteratively evolved by the algorithm, simulating genetic processes including crossover, mutation, and selection [34].

The research methodology flowchart is shown in Figure 1.

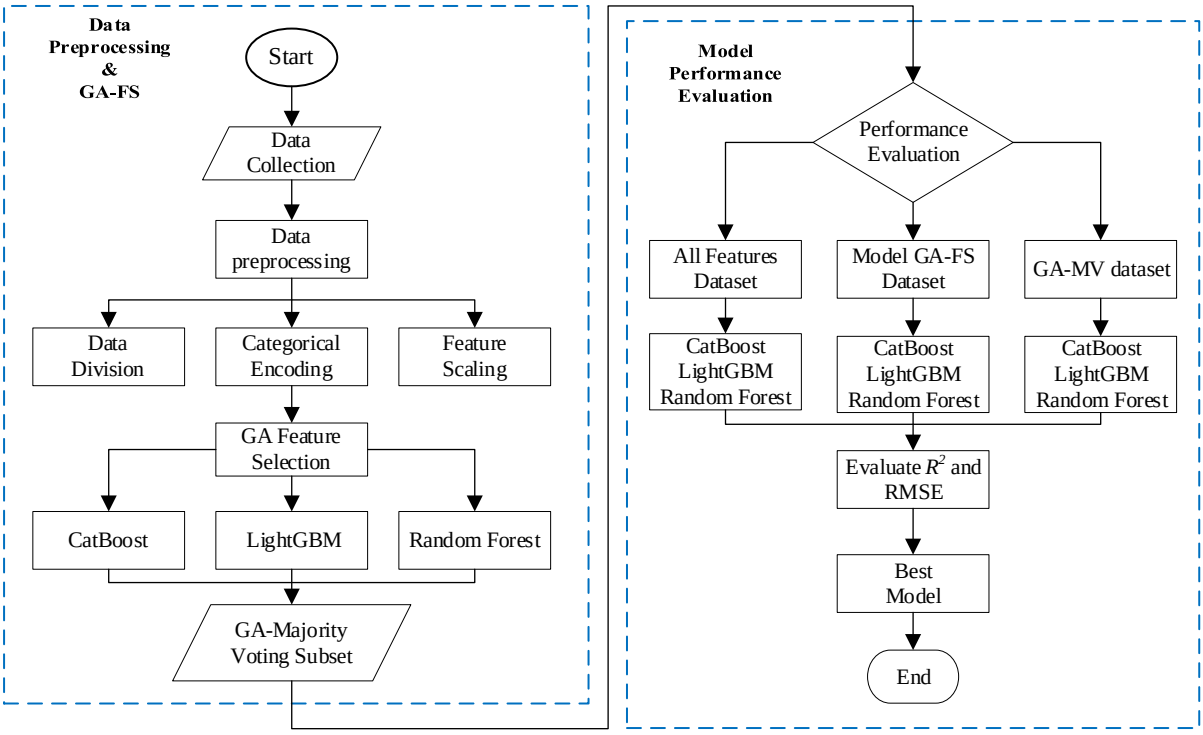

Figure 1. *Research methodology.*

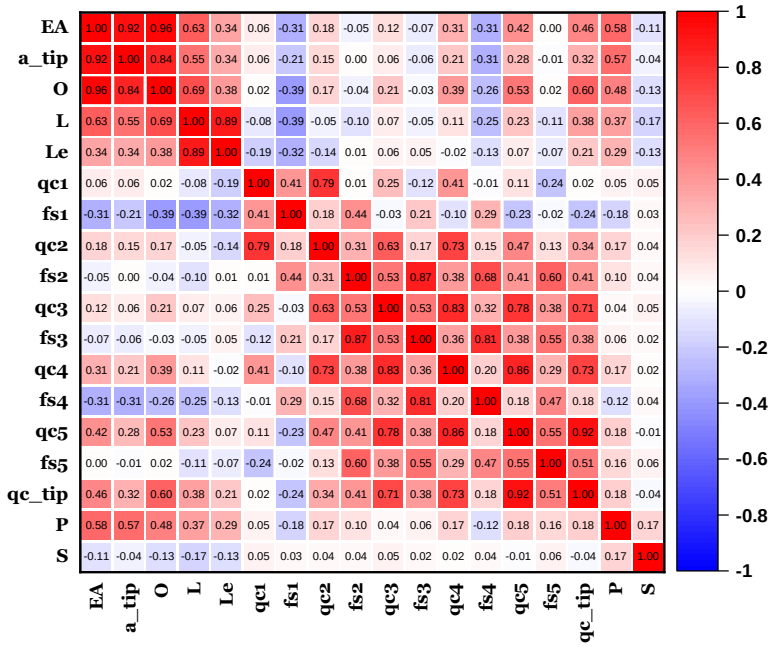

Figure 2. *Correlation plot for numerical features.*

GAs have been extensively applied to FS tasks across various domains, demonstrating their effectiveness in optimizing feature subsets for improved model performance. Rostami et al. [35] introduced a community detection-based GA for FS which clusters features based on similarity and employs a novel repair operation to enhance classification accuracy. Similarly, Nematzadeh et al. [36] proposed a distance-based mutual congestion FS method combined with a GA, specifically designed for high-dimensional medical datasets, resulting in superior performance.

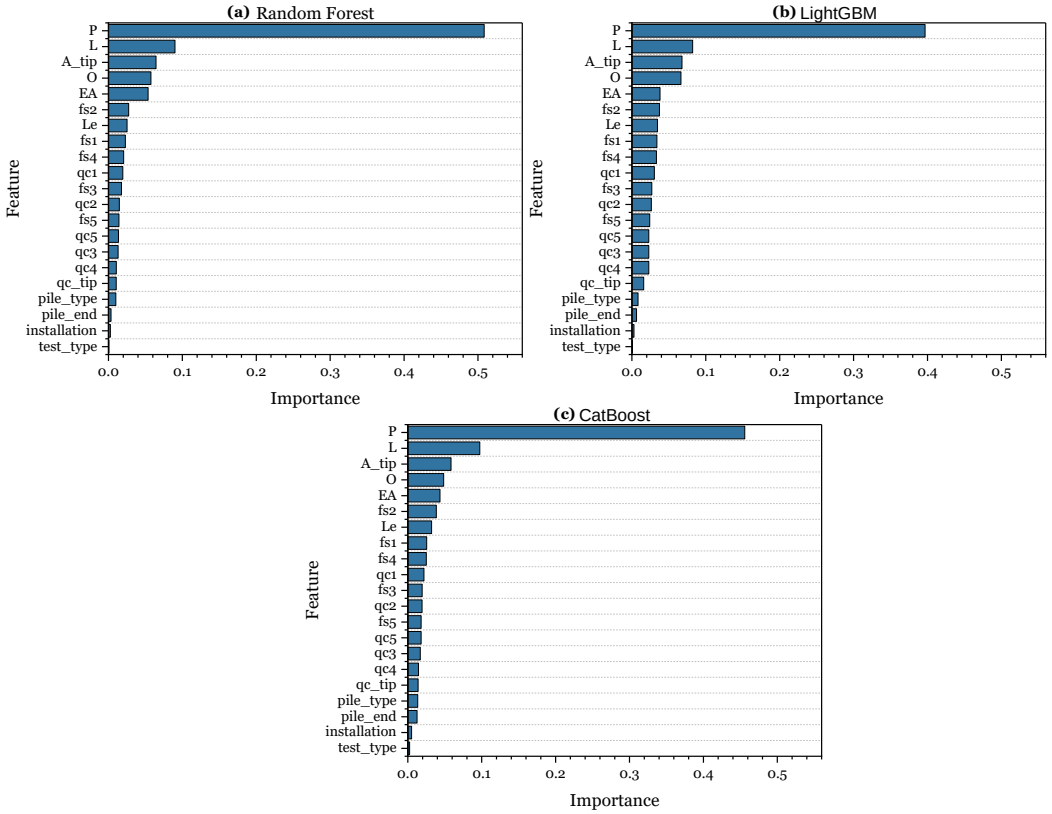

Figure 3. *Feature importance using all features.*

The adaptability of GAs to various data complexities is evident in their application to noisy label scenarios. Imani et al. [37] developed the Noise-Aware Multi-Objective Feature Selection Genetic Algorithm (NMFS-GA) to select optimal feature subsets in binary classification tasks with noisy labels, enhancing both accuracy and interpretability.

Each model has its unique method of selecting and ranking features, and this is commonly known. An example is presented in Figure 3. It can be seen that all the ML models have different rankings for each feature, and only the most relevant can be consistently ranked highly by all models.

Table 1. *GA selected feature subset.*

Features	ML Model			
	CB	RF	LGB	Voting
Test type	1	0	1	✓
Pile Type	1	1	1	✓
Installation	0	0	1	×
Pile end	1	0	0	×
EA	0	1	1	✓
A_{tip}	1	1	0	✓
O	0	1	1	✓
L	1	1	1	✓
L_e	1	1	0	✓
q_{c1}	0	1	1	✓
f_{s1}	1	0	1	✓
q_{c2}	0	1	0	×
f_{s2}	1	0	1	✓
q_{c3}	1	0	1	×
f_{s3}	0	1	0	×
q_{c4}	0	1	1	×
f_{s4}	1	1	0	✓
q_{c5}	0	0	0	×
f_{s5}	0	0	1	×
qc_{tip}	0	1	1	✓
P	1	1	1	✓
	Parameter settings			
Generation	50	50	50	
Population	100	100	100	
Crossover	0.8	0.8	0.8	
Mutation	0.03	0.03	0.03	

As such, this study proposes an FS strategy based on the concept of ensembling, where multiple models work together to improve overall performance. Instead of relying on a single model's FS process, the three ML models (RF, CB, and LGB) are each paired with GA to select the most important features.

Each model runs independently with GA, producing a unique subset of selected features. Rather than using just one model's selection, an MV strategy is applied—only features chosen by at least two out of the three models are retained. This approach ensures that the final feature set captures the most consistently important features across multiple perspectives, reducing bias and enhancing model robustness.

The GA-FS was initialized using the same parameter settings for all the ML models, this was to get results based on the same parameters without any tuning bias. The features with unit 1 value indicate that the feature was selected by the model and 0 means it was not selected by the model. The voting column indicates if the feature was selected by at least two models. The selected features are presented in Table 1.

3.3. Evaluation Metrics

Commonly used metrics for evaluating model performance, as highlighted in the literature, include the correlation coefficient (R), the coefficient of determination (R^2), root mean square error (RMSE), and mean absolute error (MAE). Among these, R^2 is often favored over R because it provides a more precise and unbiased measure of model effectiveness. It quantifies the proportion of variance in the dependent variable explained by the independent variables, offering a clearer assessment of how well the model represents the actual data points. [38]

Beyond the R^2, the RMSE is widely recognized as one of the most effective error metrics for regression tasks due to its sensitivity to larger errors, which can significantly impact model performance. This makes RMSE particularly valuable when penalizing substantial deviations between predicted and observed values is crucial. By combining R^2 and RMSE, a more comprehensive evaluation of the model's predictive ability is achieved, ensuring both a good fit to the data and reliability across different conditions [39].

$$R^2 = 1 - \left(\frac{\sum_{i=1}^{N}(O_i - P_i)^2}{\sum_{i=1}^{N}(O_i - \bar{O})^2}\right) \quad (6)$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}|O_i - P_i|}{N}} \quad (7)$$

4. Results and Discussion

4.1. Model Performance Comparison

The performances of the three ML models presented in the study were evaluated using the metrics discussed in Chapter

3.3. Each of the three models was assessed using the three different feature sets: (i) all available features, (ii) features selected by the GA for each model, and (iii) features selected using the MV approach.

From Table 2, it can be observed that among the models, CB consistently outperformed RF and LGB across all feature sets.

Table 2. *Performance of all models on the train and val sets.*

Model	Features	R^2 Train / Val	RMSE (mm) Train / Val
RF	All	0.929 / 0.913	4.226 / 6.417
	RF-GA	0.927 / 0.909	4.283 / 6.538
	MV	0.928 / 0.916	4.260 / 6.313
LGB	All	0.774 / 0.638	7.538 / 13.08
	LGB-GA	0.766 / 0.635	7.665 / 13.14
	MV	0.768 / 0.640	7.635 / 13.04
CB	All	0.965 / 0.906	2.954 / 6.668
	CB-GA	0.959 / 0.909	3.202 / 6.537
	MV	0.963 / 0.926	3.060 / 5.922

When using the majority voting-selected features, CB achieved the highest R^2 and the lowest RMSE on the val set, indicating better predictive accuracy and generalization. While all three models showed improvements when using their GA-selected feature subsets compared to using all features, the MV approach led to further enhancements, especially in reducing RMSE.

Compared to individual GA-selected feature subsets by each model, the GAMV approach resulted in slightly improved performance across all models, particularly in the val sets, with significant gains observed in RMSE reduction. This suggests that this method has the likelihood to improve model accuracy while reducing redundancy.

For instance, in the case of RF and LGB, the performance with GAMV features was closer to their best-case scenario, often matching or slightly exceeding the performance with their GA-selected features.

4.2. Final Model Evaluation

Based on the comparative analysis, CB was identified as the best-performing model. It consistently exhibited higher R^2 values and lower RMSE and MAE scores, regardless of the FS method used. The superior performance of the CB model can be attributed to its ability to handle categorical variables efficiently, its robustness against overfitting, and its use of ordered boosting, which minimizes prediction bias.

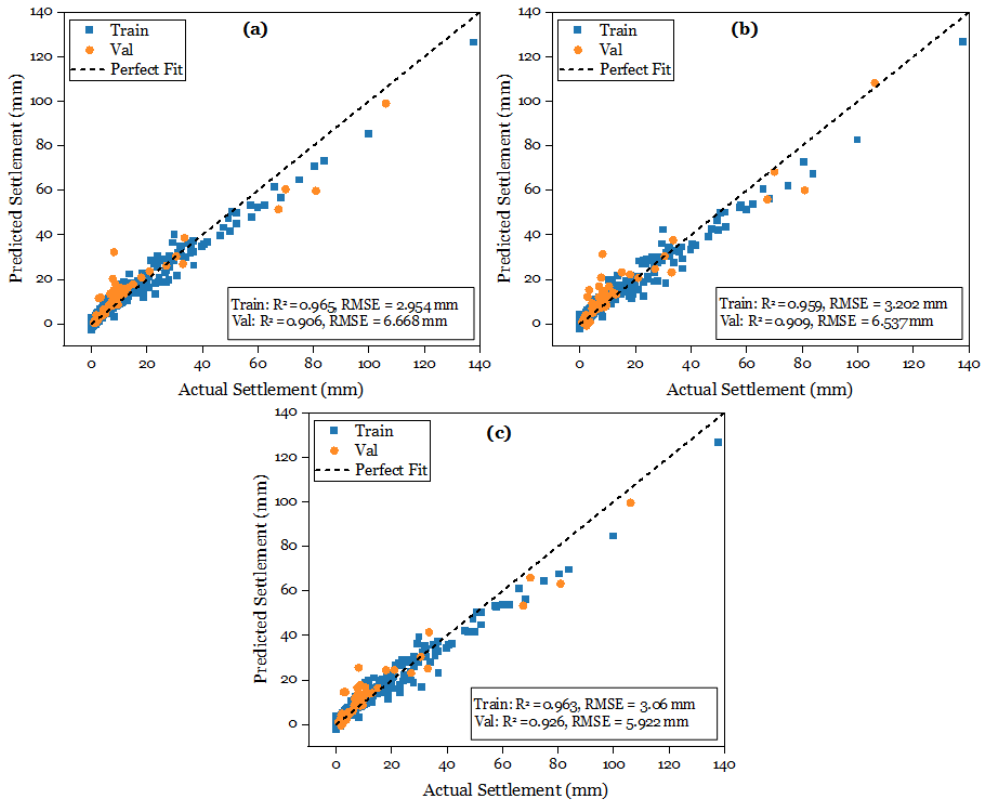

Figure 4. *Regression plots for CB: (a) All features; (b) CB-GA; and (c) CB-GAMV.*

From Figure 4, all models show a strong correlation between actual and predicted settlement values; however, Figure 4 (c) stands out with the best balance between training and validation performance. It achieved a training R^2 of 0.963 and RMSE of 3.06 mm, while its validation R^2 and RMSE were 0.926 and 5.922 mm, respectively. These values indicate a high level of accuracy with minimal overfitting, making it the most reliable model in this study for settlement prediction.

In contrast, Figure 4 (a) had a slightly higher training R^2 (0.965) and lower RMSE (2.954 mm) but performed worse on the validation set (R^2 = 0.906, RMSE = 6.668 mm), suggesting a mild overfitting issue. Figure 4 (b) showed marginally better validation R^2 (0.909) than Figure 4 (a) but had higher training RMSE and comparable validation error. Visual inspection further supports Figure 4 (c)'s superiority, as its predicted values align more closely with the perfect-fit line across all settlement ranges, reflecting better consistency and accuracy. Overall, Figure 4 (c) demonstrates the generalized predictive performance.

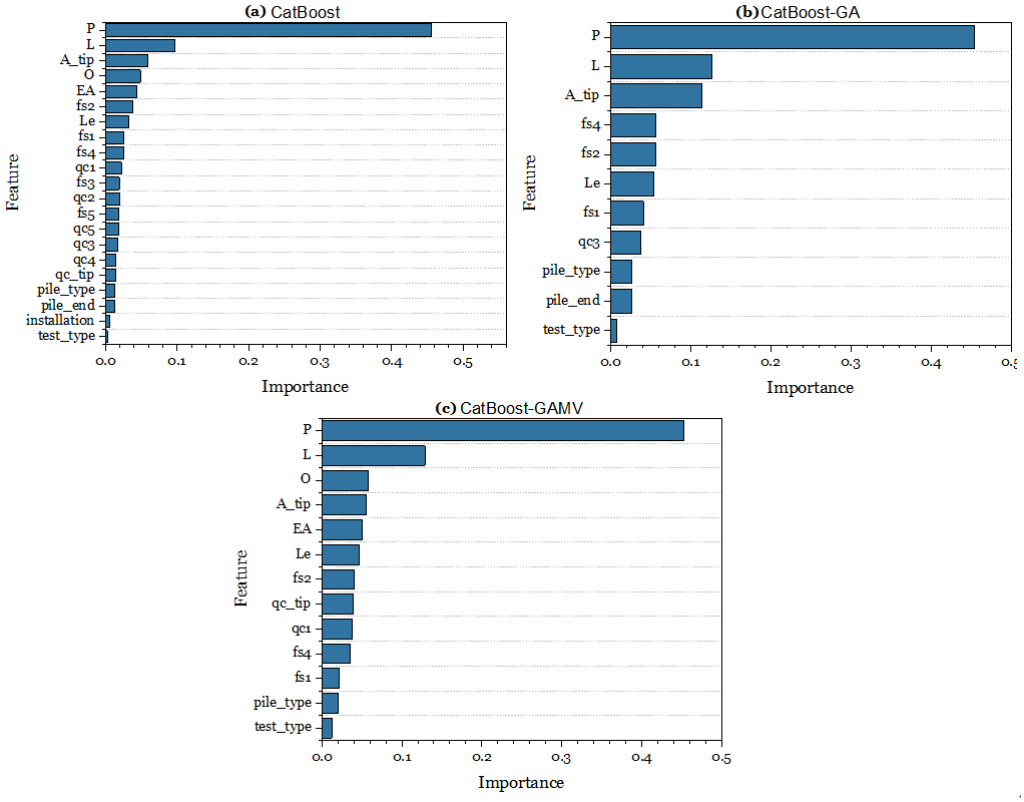

Figure 5. *Feature importance for CB: (a) All features; (b) CB-GA; and (c) MV.*

The effectiveness of CB further improved with the MV-selected features, reinforcing the hypothesis that ensembling FS enhances model reliability. This combination yielded a model that not only fit the training data well but also maintained high accuracy across varying conditions, making it the optimal choice for predicting pile settlement in this study.

4.3. Feature Importance

Feature importance analysis is crucial to interpret the model outputs and to understand the influence of various factors on pile settlement. This analysis aids in validating the model against known geotechnical principles and in identifying the most significant predictors of settlement behavior. To assess the importance of each feature, the importance values were computed for each variation of the CB model, and the resulting plots are shown in Figure 5.

The results from both models show a consensus in ranking the most influential features. *P* and *L* are ranked as the top two features in Figure 5, followed by *EA*, A_{tip}, and *O*, though alternating in Figure 5 (a) and (c), and an absence of *EA* and *O* in Figure 5 (b). These features have similar importance values and are closely ranked with each other, indicating a substantial impact on the prediction of pile settlement. However, *P* is the most important feature as all models show the highest importance values for *P*.

The soil properties have importance values that are lesser than the previously mentioned features but rank above the categorical features. These features provide additional information on soil-pile interaction and pile end-bearing capacity. Finally, the categorical features show very negligible importance values, which indicates that they only slightly

influence the model predictions in all cases.

All models consistently rank the categorical values as the least important features, with negligible feature importance values. These categorical variables contribute minimally to the models, suggesting that the pile settlement prediction is more sensitive to quantitative, site-specific soil and pile parameters rather than the categorical distinctions of pile or test types. This could be due to the inherent uniformity in pile behavior across different types or because the dataset or models do not capture significant variations that differentiate these categories.

5. Conclusions

The results indicate that applying ML models with an ensemble-based FS strategy improves predictive accuracy in pile settlement analysis. By utilizing an MV approach for FS, the dependency on a single model's bias is reduced, leading to more consistent and generalizable results. The final CB-MV model achieved good performance with an R^2 of 0.926 and an RMSE of 5.92 mm for settlements up to 140 mm on the validation set and outperformed all the other models discussed in the study.

While the study demonstrates the effectiveness of the proposed GAMV-FS strategy, some limitations remain. First, the dataset size and distribution could influence model performance, and additional validation on larger datasets with larger feature counts is recommended. Second, while CB outperformed the other models, exploring hybrid modeling approaches or deep learning techniques could provide further improvements. Lastly, this study adopted GA as the primary method for FS, subsequent studies could employ any FS method which may include metaheuristic methods such as particle swarm algorithm or differential evolution with the MV approach, or they could adopt less complicated methods like recursive feature elimination with the MV approach.

Abbreviations

ANN	Artificial Neural Network
CB	Categorical Boosting or CatBoost
CPT	Cone Penetration Test
FS	Feature Selection
GA	Genetic Algorithm
GAMV	Genetic Algorithm Majority Voting
RF	Random Forest
LGB	Light Gradient Boosting
MV	Majority Voting
SPT	Standard Penetration Test
SVM	Support Vector Machine
XGB	Extreme Gradient Boosting
Val	Validation

Acknowledgments

The authors would like to collectively thank F. Pooya Nejad and Mark B. Jaksa of the School of Civil, Environmental and Mining Engineering, University of Adelaide, Australia for making their pile load-settlement dataset publicly available.

Author Contributions

Hafeez Husain Bello: Conceptualization, Formal Analysis, Methodology, Visualization, Writing - original draft

You Wang: Supervision, Writing - review & editing

Shamsuddeen Lawal: Conceptualization, Formal Analysis, Visualization

Funding

This work is not supported by any external funding.

Data Availability Statement

The data that support the findings of this study can be found at: http://dx.doi.org/10.1016/j.compgeo.2017. 04.003.

Conflicts of Interest

The authors declare no conflicts of interest.

References

[1] Berardi R, Bovolenta R. Pile-settlement evaluation using field stiffness non-linearity. Proceedings of the Institution of Civil Engineers-Geotechnical Engineering 2005; 158: 35-44. https://doi.org/10.1680/geng.2005.158.1.35

[2] Abu-Farsakh MY, Titi HH. Assessment of direct cone penetration test methods for predicting the ultimate capacity of friction driven piles. Journal of Geotechnical and Geoenvironmental Engineering 2004; 130: 935-44. https://doi.org/10.1061/(ASCE)1090-0241(2004)130:9(935)

[3] Poulos HG, Davis EH. Pile foundation analysis and design. vol. 397. Wiley New York; 1980.

[4] Murthy VNS. Principles and practices of soil mechanics and foundation engineering. New York: Marcel Decker Inc., 2002.

[5] Meyerhof GG. Bearing capacity and settlement of pile foundations. Journal of the Geotechnical Engineering Division 1976; 102: 197-228. https://doi.org/10.1061/AJGEB6.0000243

[6] Ardalan H, Eslami A, Nariman-Zadeh N. Shaft resistance of driven piles based on CPT and CPTu results using GMDH-type neural networks and genetic algorithms. The 12th International Conference of International Association for Computer Methods and Advances in Geomechanics (IACMAG), Citeseer, 2008, p. 1850-8.

[7] Decourt L. Prediction of load-settlement relationships for foundations on the basis of the SPT, Ciclo de Conferencias Internationale. Leonardo Zeevaert, UNAM, Mexico 1985: 85-104.

[8] Karimpour-Fard M, Eslami A. Estimation of vertical bearing capacity of piles using the results CPT and SPT tests. Geotechnical and Geophysical Site Characterization: Proceedings of the 4th International Conference on Site Characterization ISC-4, vol. 1, Taylor & Francis Books Ltd; 2013, p. 1055-62.

[9] Vesic AS. Design of pile foundations. NCHRP Synthesis of Highway Practice 1977.

[10] Chan WT, Chow YK, Liu LF. Neural network: An alternative to pile driving formulas. Comput Geotech 1995; 17: 135-56. https://doi.org/10.1016/0266-352X(95)93866-H

[11] Goh ATC. Pile Driving Records Reanalyzed Using Neural Networks. Journal of Geotechnical Engineering 1996; 122: 492-5. https://doi.org/10.1061/(ASCE)0733-9410(1996)122:6(492)

[12] Lee IM., Lee JH. Prediction of pile bearing capacity using artificial neural networks. Computers and Geotechnics 1996; 18: 189-200. https://doi.org/10.1016/0266-352X(95)00027-8

[13] Teh CI, Wong KS, Goh ATC, Jaritngam S. Prediction of pile capacity using neural networks. Journal of Computing in Civil Engineering 1997; 11: 129-38. https://doi.org/10.1061/(ASCE)0887-3801(1997)11:2(129)

[14] Samui P. Prediction of pile bearing capacity using support vector machine. International Journal of Geotechnical Engineering 2011; 5: 95-102. https://doi.org/10.3328/IJGE.2011.05.01.95-102

[15] Bui XN, Jaroonpattanapong P, Nguyen H, Tran QH, Long NQ. A Novel Hybrid Model for Predicting Blast-Induced Ground Vibration Based on k-Nearest Neighbors and Particle Swarm Optimization. Scientific Reports 2019; 9: 1-14. https://doi.org/10.1038/s41598-019-50262-5

[16] Pham BT, Tien Bui D, Prakash I. Landslide Susceptibility Assessment Using Bagging Ensemble Based Alternating Decision Trees, Logistic Regression and J48 Decision Trees Methods: A Comparative Study. Geotechnical and Geological Engineering 2017; 35: 2597-611. https://doi.org/10.1007/S10706-017-0264-2/FIGURES/8

[17] Zhang R, Li Y, Goh ATC, Zhang W, Chen Z. Analysis of ground surface settlement in anisotropic clays using extreme gradient boosting and random forest regression models. Journal of Rock Mechanics and Geotechnical Engineering 2021; 13: 1478-84. https://doi.org/10.1016/J.JRMGE.2021.08.001

[18] Zhang W, Wu C, Zhong H, Li Y, Wang L. Prediction of undrained shear strength using extreme gradient boosting and random forest based on Bayesian optimization. Geoscience Frontiers 2021; 12: 469-77. https://doi.org/10.1016/J.GSF.2020.03.007

[19] Ardalan H, Eslami A, Nariman-Zadeh N. Piles shaft capacity from CPT and CPTu data by polynomial neural networks and genetic algorithms. Computers and Geotechnics 2009; 36: 616-325. https://doi.org/10.1016/j.compgeo.2008.09.003

[20] Nejad F, Jaksa M, Kakhi M, McCabe BA. Prediction of pile settlement using artificial neural networks based on standard penetration test data. Computers and Geotechnics 2009; 36: 1125-1133. https://doi.org/10.1016/j.compgeo.2009.04.003

[21] Nejad FP, Jaksa MB. Load-settlement behavior modeling of single piles using artificial neural networks and CPT data. Computers and Geotechnics 2017; 89: 9-21. https://doi.org/10.1016/J.COMPGEO.2017.04.003

[22] Kardani N, Zhou A, Nazem M, Shen SL. Estimation of Bearing Capacity of Piles in Cohesionless Soil Using Optimised Machine Learning Approaches. Geotechnical and Geological Engineering 2020; 38: 2271-91. https://doi.org/10.1007/s10706-019-01085-8

[23] Ismail A, Jeng D-S. Empirical Method for Settlement Prediction of Single Piles Using Higher Order Neural Network and Particle Swarm Optimization 2012: 285-94. https://doi.org/10.1061/9780784412121.030

[24] Zhang G, Xiang X, Tang H. Time Series Prediction of Chimney Foundation Settlement by Neural Networks. International Journal of Geomechanics 2011; 11: 154-8. https://doi.org/10.1061/(ASCE)GM.1943-5622.0000029

[25] Bustamante M, Gianeselli L. Pile bearing capacity by means of static penetrometer CPT: In Proceedings of the 2nd European Symposium on Penetration Testing 1982.

[26] Kosaraju N, Sankepally SR, Mallikharjuna Rao K. Categorical Data: Need, Encoding, Selection of Encoding Method and Its Emergence in Machine Learning Models—A Practical Review Study on Heart Disease Prediction Dataset Using Pearson Correlation. Proceedings of International Conference on Data Science and Applications, Singapore: Springer Nature Singapore; 2023, p. 369-382. https://doi.org/10.1007/978-981-19-6631-6_26

[27] Breiman L. Random forests. Machine Learning 2001; 45: 5-32. https://doi.org/10.1023/A:1010933404324

[28] Ho TK. Random decision forests. Proceedings of 3rd International Conference on Document Analysis and Recognition, vol. 1,1995,278-282. doi: 10.1109/ICDAR.1995.598994

[29] Bernard S, Adam S, Heutte L. Dynamic Random Forests. Pattern Recognit Lett 2012; 33: 1580-6. https://doi.org/10.1016/J.PATREC.2012.04.003

[30] Ke G, Meng Q, Finley T, Wang T, Chen W, Ma W, Ye Q, Liu T-Y. Lightgbm: A highly efficient gradient boosting decision tree. Advances in neural information processing systems 2017; 30.

[31] Prokhorenkova L, Gusev G, Vorobev A, Dorogush AV, Gulin A. CatBoost: unbiased boosting with categorical features. In: Bengio S, Wallach H, Larochelle H, Grauman K, Cesa-Bianchi N, Garnett R. Advances in neural information processing systems 2018; 31.

[32] Dorogush AV, Ershov V, Yandex AG. CatBoost: gradient boosting with categorical features support 2018. https://doi.org/10.48550/arXiv.1810.11363

[33] Chandrashekar G, Sahin F. A survey on feature selection methods. Computers & Electrical Engineering 2014; 40: 16-28. https://doi.org/10.1016/j.compeleceng.2013.11.024

[34] Holland JH. Genetic Algorithms and Adaptation. Adaptive Control of Ill-Defined Systems, Boston, MA: Springer US; 1984, p. 317-33. https://doi.org/10.1007/978-1-4684-8941-5_21

[35] Rostami M, Berahmand K, Forouzandeh S. A novel community detection based genetic algorithm for feature selection. J Big Data 2021; 8: 2. https://doi.org/10.1186/s40537-020-00398-3

[36] Nematzadeh H, Mani J, Nematzadeh Z, Akbari E, Mohamad R. Distance-based mutual congestion feature selection with genetic algorithm for high-dimensional medical datasets. Neural Comput Appl 2025: 1-16. https://doi.org/10.1007/s00521-024-10837-4

[37] Imani V, Moradi E, Sevilla-Salcedo C, Fortino V, Tohka J. Optimizing Feature Selection for Binary Classification with Noisy Labels: A Genetic Algorithm Approach. International Conference on Advances in Computing Research, Springer; 2024, 956: 392-403. https://doi.org/10.1007/978-3-031-56950-0_33

[38] Cameron AC, Windmeijer FAG. An R-squared measure of goodness of fit for some common nonlinear regression models. J Econom 1997; 77: 329-42. https://doi.org/10.1016/S0304-4076(96)01818-0

[39] Hecht-Nielsen R. Theory of the backpropagation neural network. Neural networks for perception 1992; 593-605. https://doi.org/10.1109/IJCNN.1989.118638