Journal of Science and Transport Technology
University of Transport Technology

Optimizing the architecture of the artificial neural network by genetic algorithm to improve the predictability of pile bearing capacity based on CPT results

Tuan Anh Pham*, Thi Huong Lan Vu
University of Transport Technology, Hanoi 100000, Vietnam

Article info
Type of article:
Original research paper

***Corresponding author:**
E-mail address:
anhpt@utt.edu.vn

Received:
November 13, 2021
Accepted:
December 7, 2021
Published:
December 18, 2021

Abstract: This paper presents the results of applying the Artificial Neural Network (ANN) model in determining pile bearing capacity. The traditional methods used to calculate the bearing capacity of piles still have many disadvantages that need to be overcome such as high cost, complicated calculation, time-consuming. Currently, Artificial Intelligence (AI) is a useful tool that is applied in many fields to save time and costs. The study develops an ANN model and optimizes the architecture, using the Genetic Algorithm (GA) to determine the pile bearing capacity. A dataset of 108 pile static compression results is used to train and test the model. The results of the study are compared with the experimental formula according to Vietnamese nation standard TCVN 10304:2014, showing that the ANN model with well optimized, allowing prediction of pile bearing capacity close to experimental results and better than the formula in nation standard. Specifically, the ANN model gives 12% and 32.4% better performance, respectively, than the empirical formula on R2 and RMSE criteria, respectively. The results of the study are a premise for the application of AI in solving pile problems in the field of construction.

Keywords: pile bearing capacity; CPT result; genetic algorithm; artificial neural network

1. Introduction

In the field of construction, pile foundations are increasingly proving to be an effective foundation solution when applied to projects requiring large load capacity. When calculating and designing piles, designers are required to determine the pile bearing capacity in advance by different methods. The most accurate method is the field pile static compression tests, but this method is expensive, time-consuming, and is usually only used for a few test piles on site. In addition, the dynamic load test (PDA) method is also relatively popular, however, this method gives a large error, because the wave propagation is disturbed by many factors. Therefore, a series of studies have proposed empirical formulas, which allow approximating the bearing capacity of piles, on the basis of soil properties and geometrical parametersof piles [1], [2], [3], [4]. The aboveempirical formulas all have a domain of applications well as finite precision, depending on the data that researchers use to build the formula. In addition, the use of the finite element method to simulate and determine the pile load capacity has been widely applied, however, these mathematical models are very sensitive to input parameters and types of soil model, leading to large skewed calculation results if these parameters are selected incorrectly.

Recently, the application of Artificial Intelligence (AI) and Machine Learning (ML) models in the construction field, especially in the field of building foundations, has been showing remarkable achievements. Some published literatures used the ANN model to predict the ultimate bearing capacity of piles such as Pham et al [5]-[8], Momeni [9], [10]. Some studies used other ML models in determining pile bearing capacity such as Yong et al (2020) [11] used the AN-GP model, Ghorbani et al (2018) [12] presented the ANFIS model, Pham and Vu (2021) [13] developed ensemble learning model. The above studies show that ML models have an impressive ability to predict pile bearing capacity when achieving high accuracy. However, optimizing the models as well as proposing improvements to increase the accuracy of the model is necessary and always welcomed by researchers. In addition, in the above studies, most of the optimal architecture of the model is selected manually and gradually tested the cases. That causes a huge waste of time and resources, and at the same time, it is unlikely to find the best model among all possible models.

In this study, an ANN model was developed to predict pile load based on Static Penetration Test (CPT) results. The model's architecture is optimized through a Genetic Algorithm (GA), which is a powerful algorithm in the family of evolutionary algorithms. This helps researchers avoid having to search for models by the manual solution to select good models. The calculation results of the model are compared with the experimental results of static compression of piles and the formula according to Vietnamese pile foundation design standards (TCVN 10304: 2014) to certify the superior ability of the model in determining the bearing capacity of piles. Finally, the permutation feature importance technique is applied to detect the most important input variables to estimating pile bearing capacity.

2. Research Methods

2.1. Artificial Neural Networ

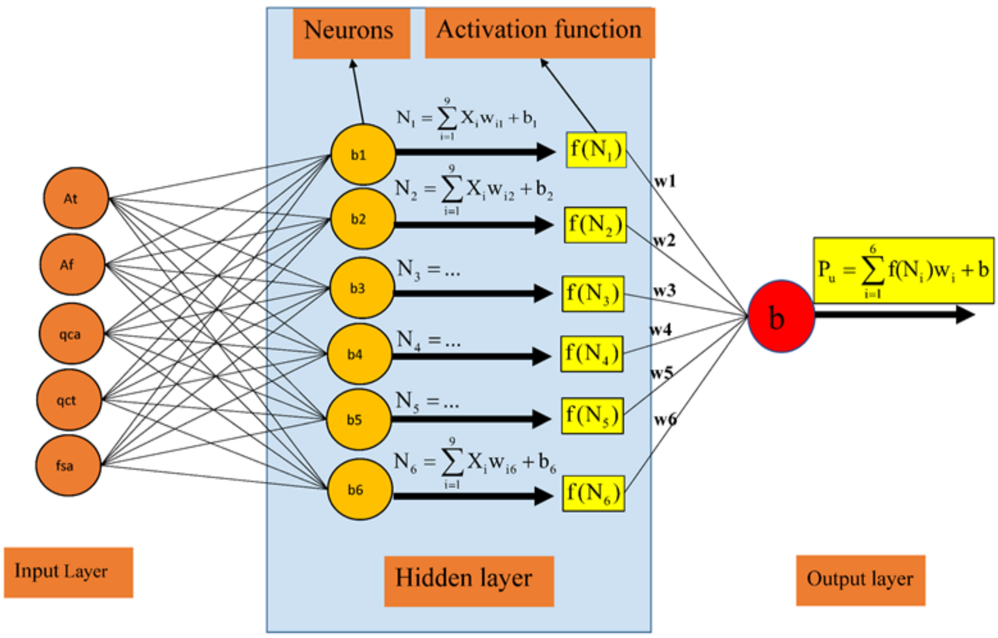

Fig 1. ANN model algorithm diagram

The ANN model is one of the most popular algorithms in the family of machine learning algorithms. This model was first introduced by McCulloch and Pitts (1943) [14]. Through a long development process, ANN has become one of the most popular ML models and is applied in all fields of science and technology. The ANN model used to predict the load capacity in this study is shown in Fig 1. In the ANN model, a network of neurons is linked together by weights. An ANN model consists of at least 3 layers: input layer, hidden layer, and output layer. The output signal of any j^{th} hidden node in the network is calculated as follows:

$$N_j = f(\sum_{i=1}^{n} X_i w_{ij} + b_j) \quad (1)$$

Where: N_j is the output signal of a node; X_i is the i^{th} input variable; w_{ij} is the connection weight between the input variable i and node j; b_j is the offset of node j; f() is the activation function of the hidden node; w_j is the weight connecting the hidden node j and the output; b is the bias of the output node; n is the number of hidden nodes of the layer.

The ANN must be trained before it can be used, the training is a process of optimizing the weights and biases so that the output of the model is closest to the measured results from the experiment. In this study, the training algorithm used is the back-propagation using the gradient descent optimization technique. In which, the error is propagated back from the output layer to the input layer. On the basis of optimizing model weights to minimize errors, the AI network will achieve high performance when applied to new or unseen data.

2.2. Genetic Algorithm

The genetic algorithm is one of the most powerful optimization algorithms, first introduced by Holland [15]. This algorithm uses Darwin's theory of evolution as a foundation. The algorithm allows optimizing multivariable functions by considering the variables of the function as the chromosomes of a population. This population continuously evolves through the generations, by selecting the best genes and passing them on to the next generation. Weak individuals will be eliminated from the population so that their genes cannot continue to be inherited. In general, the genetic algorithm goes through iterations (called generations), and in each generation, the following process is repeated:

Step 1. Select the best individuals in the population (based on that individual's performance on the training set)

Step 2. Mating individuals for the purpose of creating a new generation.

Step 3. Allow some individuals in the younger generation to mutate. A mutation is a process of replacing some random gene in a chromosome sequence, giving evolution a better chance of finding a gene.

Step 4. Eliminate weak individuals.

In this study, the parameters related to the architecture of the ANN model are considered to be the genes of the population. The individual with the best genes in the last generation will be used as the best model, used for training and testing.

2.3. Data preparation

Data used to build and test the pile bearing capacity prediction model collected from different sources have been published. Specifically, the dataset includes 108 data reports on pile load tests, summarized by Ghorbani (2018) [12]. The input parameters used to build the model are selected according to the recommendations of the published studies [1]-[4]. To be more specific, the selected input variables include cross-sectional area of the pile tip (A_t), the shaft area (A_f). Soil properties are shown through parameters obtained from static penetration test (CPT) results, including average tip penetration resistance along the pile shaft (q_{ca}), average cone tip resistance at pile tip (q_{ct}), average frictional resistance along the pile shaft (f_{sa}). The ultimate pile bearing capacity is considered as the single output variable (denoted as P_u), which is determined based on the static pile load test results. With the total amount of data is not large, the data is divided into 2 sets: the training set accounts for 75% and the test set accounts for 25% of the total data. This division ensures that the amount of training data is large enough for the model to learn the general relationships between input and output but the test data is still enough to evaluate the performance of the model one by one. objective way. In which, the training set will be used to build the model and the test set will be used to evaluate the model. The statistics of the input variables are shown in Table 1.

Table 1. Statistics of the input parameters used in the study

	A_t	A_f	q_{ca}	f_{sa}	q_{ct}	P_u
Unit	(cm^2)	(m^2)	(Mpa)	(kN)	(Mpa)	(kN)
Min	20	5.45	0.83	9.39	0.25	60
Mean	1736	26.46	5.84	101.89	8.82	1965
Median	1230	17.98	5.38	81.91	7.63	1340
Max	7854	194.65	24.7	349.64	27.11	10910
SD	1674	26.35	4.23	66.29	6.19	1702.2

2.4. Performance Indicators

In this study, performance indicators including correlation coefficient squared (R^2), root mean square error (RMSE) were used to evaluate and compare models, specifically as follows:

$$\mathrm{RMSE} = \sqrt{\frac{1}{k}\sum_{i=1}^{k}\left(y_i - \overline{y}_i\right)^2} \tag{2}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{k}\left(y_i - \overline{y}_i\right)^2}{\sum_{i=1}^{k}\left(y_i - \overline{y}\right)^2} \tag{3}$$

Where, k is the number of data samples, y_i and $\overline{y}_i$ are the experimental and model-predicted results, $\overline{y}$ is the mean of y_i.

Specifically, R^2 characterizes the correlation between the two results. The closer R^2 is to 1, the closer the two results are to each other. RMSE characterizes the average error between 2 results, the smaller the RMSE, the higher the prediction accuracy. A model is considered better when it simultaneously ensures both these criteria.

3. Results and Discussions

3.1. Optimal results of ANN model by GA

In this section, the genetic algorithm is used to optimize the architecture of the ANN model. Fig 2 shows the structure of a typical chromosome in a population. It can be seen that this chromosome has 4 length genes, each gene corresponds to some architecture of the ANN model. This study selects the four most important parameters of the network architecture: the number of hidden neurons, the activation function, the learning rate, and the number of training epochs. All parameter meanings, as well as value ranges, are mentioned in Table 2. It can be seen that, if the parameters are manually selected, the number of models to be tested can be up to thousands of cases. In this case, the genetic algorithm allows finding a good model with much less time and resources.

In the process of optimizing model parameters by genetic algorithm, the maximum number of individuals in the population selected from the beginning is 30 individuals. The number of training generations is controlled so that after 20 generations, the performance is not improved, the results are considered converged and the loop stops. Statistics of initialization parameters of the genetic algorithm are shown in Table 3. This algorithm uses a 5-fold cross-validation technique on training set to evaluate performance instead of the testing set. This operation is intended to hide the testing set during optimization, treating it as a new dataset that has never been exposed to the model to avoid overfitting. The testing set is only used to evaluate the performance of the final selected model.

The optimal results by the genetic algorithm are shown in Fig 3.

It can be seen that the optimization process converges at about generation 29, with the R^2 criterion reaching 0.72 on the 5-fold cross-validation set. This value remains constant until generation 50, satisfying the stopping condition of the algorithm. The results of the optimal parameters are shown in Table 4. This model will be used to evaluate the performance in the next section.

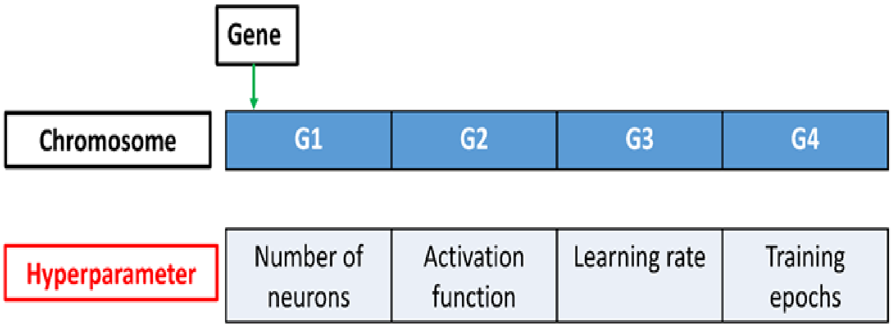

Fig 2. Structure of chromosomes and genes in GA

Table 2. Hyperparameters space of the model

Hyperparameter	Range
Number of neurons	2÷20
Activation function	'logistic', 'tanh', 'relu'
Learning rate	0.001, 0.01, 0.1, 0.2, 0.3
Number of epochs	1000, 2000, 3000, 4000

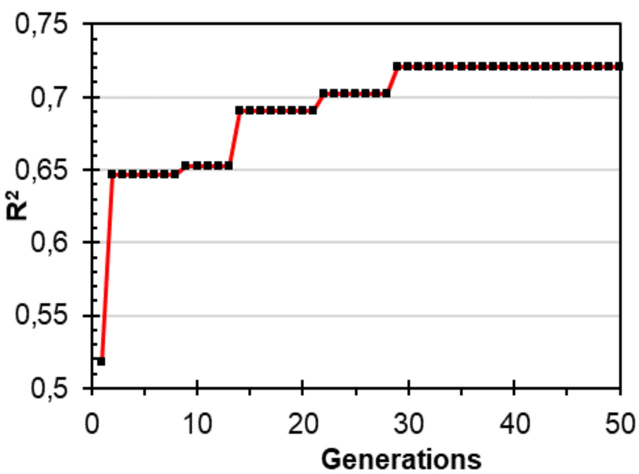

Fig 3. The optimization process of ANN model architecture by the GA based on 5-Fold technique

Table 3. Initialize parameters of the GA

Parameters	Value
Population	30
Mating rate	50%
Mutation rate	20%
Generation	Stopping after 20 generations does not improve performance.
Dataset	5-Fold CV/Training set.
Cost function	R^2

Table 4. Optimum hyperparameters of ANN model

Hyperparameter	Value
Number of neurons	15
Activation function	'relu'
Learning rate	0.1
Number of epochs	2000

3.2. Predictability of the model

The ANN model, with the optimal model architecture parameters found in the previous section, is trained on the training set, and then evaluated on the testing set. The results of the training process, using the cost function as the mean square error (MSE), are shown in Fig 4. It can be seen that after about 2000 training cycles, the results are quite convergent. Of course, increasing the number of training cycles can help reduce the error, but can cause overfitting, when the model is too suitable for the training set and does not give good results on the test set. Therefore, the value of 2000 cycles according to the result of the genetic algorithm is chosen to stop training.

The results of the regression prediction of pile load capacity of the neural network on the training set and the test set are shown in Fig 5 and simulation results are shown in Fig 6.

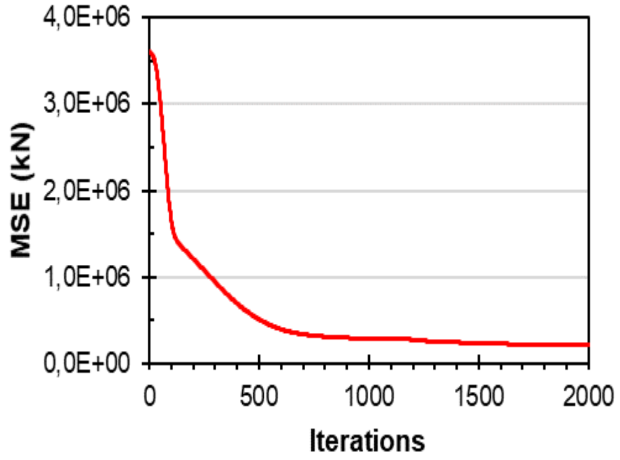

Fig 4. The cost function MSE of the model during training progress

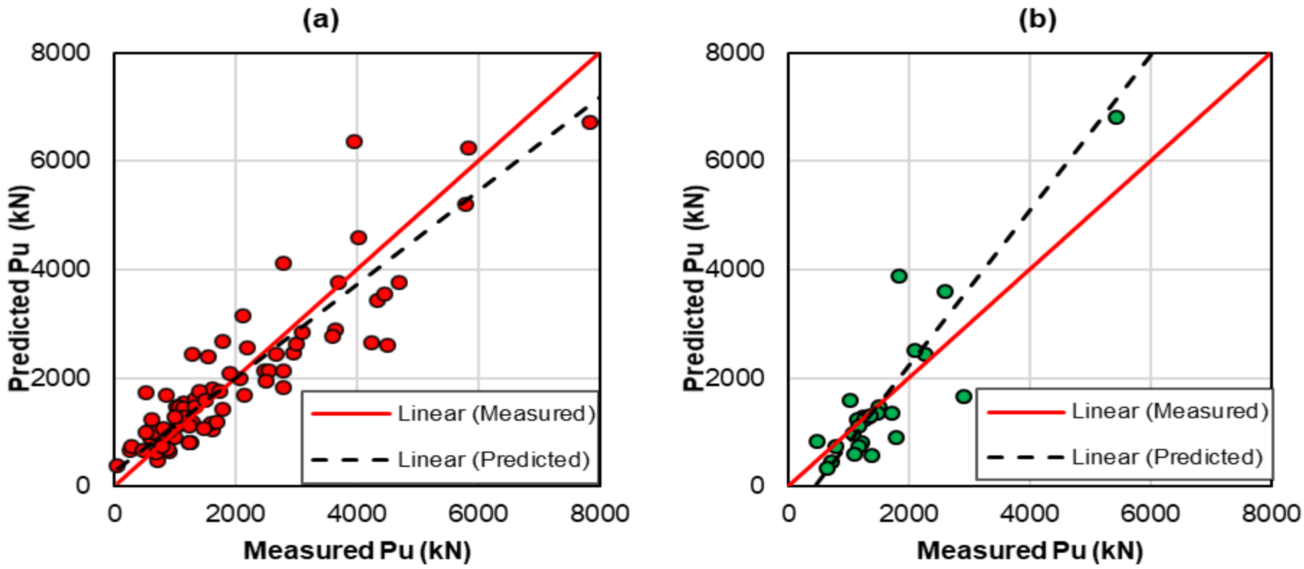

Fig 5. Regression results of the ANN model

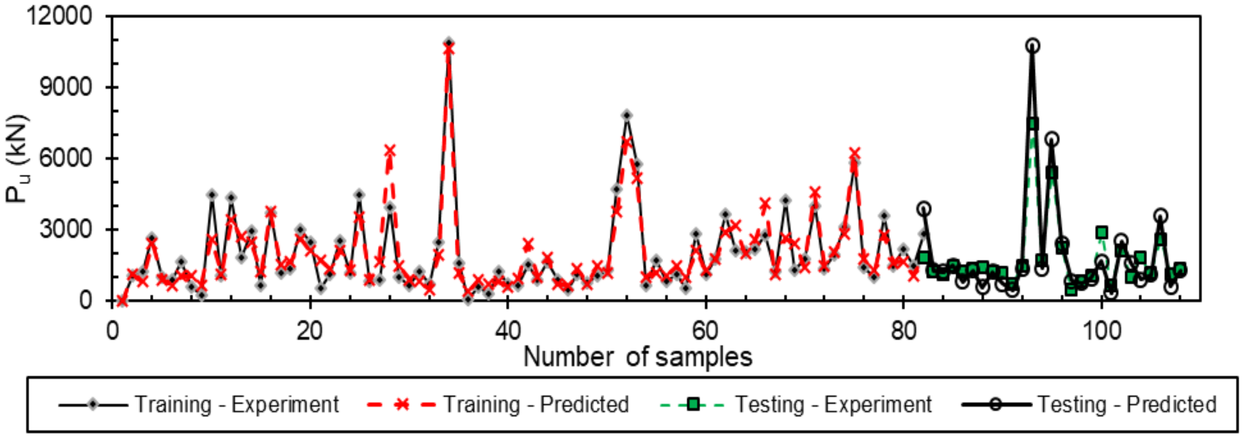

Fig 6. Visualize the prediction results of the ANN model

The analysis results show that the neural network model accurately predicts the pile load capacity. Specifically, with the training set, the criteria R^2 = 0.868 and RMSE = 640,672 kN. With the testing set, the indicators R2 = 0.911 and RMSE = 912.64 kN.

3.3. Compare the analysis results with TCVN 10304-2014 Vietnamese nation standard

The formula for calculating the pile bearing capacity according to the results of the static penetration test according to TCVN 10304:2014 is written as follows:

$$P_u = k_c.q_{ct}.A_t + \frac{q_{ca}}{\alpha_i} A_f \quad (4)$$

Where, k_c and α_i are the conversion coefficients for tip resistance and lateral resistance, respectively, see Table G2 TCVN 10304:2014. The calculation results of the formula are shown in Fig 7.

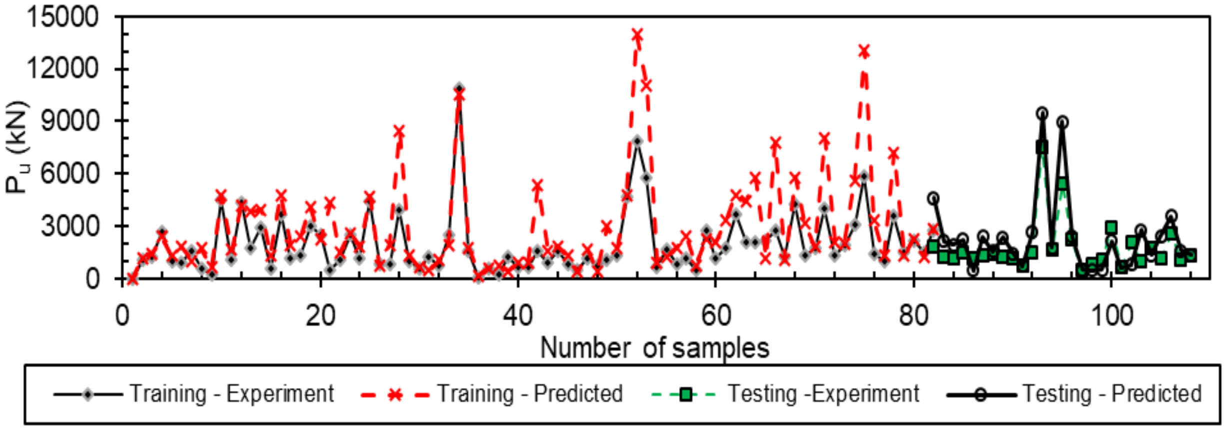

Fig 7. Visualize the prediction results of the TCVN 10304 : 2014 standard

The analytical results show that the empirical formula for determining the pile bearing capacity according to TCVN 10304-2014 predicts the bearing capacity of piles with relatively good accuracy, specifically R^2 = 0.721; RMSE = 1884,672 kN on the training set and R^2 = 0.812; RMSE = 1208.464 kN on the test set.

The results of comparing the performance of the ANN model and the formula according to TCVN 10304-2014 are shown in Table 5. It can be seen that the ANN model has 12% better efficiency when considering the R^2 criterion and 32.4% base on RMSE criteria.

4. Conclusion

The study presented the application of the ANN model to determine the bearing capacity of piles. The ANN model's architecture is optimized by the genetic algorithm, this allows the algorithm to automatically find good models compared to manual search solutions to save time and resources. The optimal results show that the neural network with 15 hidden nodes and using the 'Relu' activation function will give the best performance with the pile bearing capacity calculation data according to the CPT results. The ANN model gives superior results compared to the results calculated according to the formula according to TCVN 10304-2014 Vietnamese nation standard. Base on the analysis results, it is recommended to study and put the optimized ANN model into the foundation standards, and at the same time continue to correct the formulas in the standard to achieve higher accuracy in the actual pile design.

Acknowledgments

This research is funded by the University of Transport Technology (UTT) under grant number DTTD2021-24.

References

[1] M. A. Shahin and M. B. Jaksa, "Intelligent Computing for Predicting Axial Capacity of

Drilled Shafts," in Contemporary Topics in In Situ Testing, Analysis, and Reliability of [1] G. G. Meyerhof, "Bearing Capacity and Settlement of Pile Foundations," J. Geotech. Eng. Div., vol. 102, no. 3, pp. 197–228, 1976.

[2] I. Shooshpasha, "Prediction of the Axial Bearing Capacity of Piles by SPT-based and Numerical Design Methods," Int. J. Geomate, 2013, doi: 10.21660/2013.8.2118.

[3] A.S.Vesic, Design of Pile foundation. Transportation Research Board, National Council, 1977.

[4] J. H. Schmertmann, "GUIDELINES FOR CONE PENETRATION TEST. (PERFORMANCE AND DESIGN)," Jul. 1978.

[5] T. A. Pham, H.-B. Ly, V. Q. Tran, L. V. Giap, H.-L. T. Vu, and H.-A. T. Duong, "Prediction of Pile Axial Bearing Capacity Using Artificial Neural Network and Random Forest," Appl. Sci., vol. 10, no. 5, p. 1871, Mar. 2020, doi: 10.3390/app10051871.

[6] T. A. Pham, V. Q. Tran, H.-L. T. Vu, and H.-B. Ly, "Design deep neural network architecture using a genetic algorithm for estimation of pile bearing capacity," PLoS ONE, vol. 15, no. 12, p. e0243030, Dec. 2020, doi: 10.1371/journal.pone.0243030.

[7] T. A. Pham, V. Q. Tran, and H.-L. T. Vu, "Evolution of Deep Neural Network Architecture Using Particle Swarm Optimization to Improve the Performance in Determining the Friction Angle of Soil," Math. Probl. Eng., vol. 2021, pp. 1–17, May 2021, doi: 10.1155/2021/5570945.

[8] T. A. Pham, H.-L. T. Vu, and H.-A. T. Duong, "Improving deep neural network using hyper-parameters tuning in predicting the bearing capacity of shallow foundations," J. Appl. Sci. Eng., vol. 25, no. 2, p. 13, doi: https://doi.org/10.6180/jase.202204_25(2).0012.

[9] E. Momeni, M. B. Dowlatshahi, F. Omidinasab, H. Maizir, and D. J. Armaghani, "Gaussian Process Regression Technique to Estimate the Pile Bearing Capacity," Arab. J. Sci. Eng., vol. 45, no. 10, pp. 8255–8267, Oct. 2020, doi: 10.1007/s13369-020-04683-4.

[10] M. A. Shahin and M. B. Jaksa, "Intelligent Computing for Predicting Axial Capacity of Drilled Shafts," in Contemporary Topics in In Situ Testing, Analysis, and Reliability of Foundations, Orlando, Florida, United States, Mar. 2009, pp. 26–33. doi: 10.1061/41022(336)4.

[11] W. Yong et al., "A new hybrid simulated annealing-based genetic programming technique to predict the ultimate bearing capacity of piles," Eng. Comput., Jan. 2020, doi: 10.1007/s00366-019-00932-9.

[12] B. Ghorbani, E. Sadrossadat, J. Bolouri, and P. RahimzadehOskooei, "Numerical ANFIS-Based Formulation for Prediction of the Ultimate Axial Load Bearing Capacity of Piles Through CPT Data," Geotech. Geol. Eng., pp. 1–20, Jan. 2018, doi: 10.1007/s10706-018-0445-7.

[13] T. A. Pham and H.-L. T. Vu, "Application of Ensemble Learning Using Weight Voting Protocol in the Prediction of Pile Bearing Capacity," Math. Probl. Eng., vol. 2021, p. e5558449, Jul. 2021, doi: 10.1155/2021/5558449.

[14] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," Bull. Math. Biophys., vol. 5, no. 4, pp. 115–133, Dec. 1943, doi: 10.1007/BF02478259.

[15] J. H. Holland, P. of P. and of E. E. and C. S. J. H. Holland, and S. L. in H. R. M. Holland, Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence. MIT Press, 1992.