Journal of Science and Technology in Civil Engineering, HUCE, 2024, 18 (4): 98–108

ENSEMBLE LEARNING METHODS FOR THE MECHANICAL BEHAVIOR PREDICTION OF TRI-DIRECTIONAL FUNCTIONALLY GRADED PLATES

Dieu T. T. Do[a,*]

[a]*Faculty of Information Technology, Ho Chi Minh City University of Foreign Languages - Information Technology, 828 Su Van Hanh road, District 10, Ho Chi Minh City, Vietnam*

Article history:
Received 06/9/2024, Revised 21/10/2024, Accepted 05/12/2024

Abstract

This paper aims to enhance computational performance for behavior prediction of tri-directional functionally graded plates using ensemble learning methods such as random forest, extreme gradient boosting, and light gradient boosting machine. Furthermore, the effectiveness of these methods is verified by comparing their results with those of artificial neural networks. The present investigation focuses on the buckling problem of tri-directional functionally graded plates. In this study, data pairs consisting of input and output data are generated using a combination of isogeometric analysis and generalized shear deformation theory to ensure the accuracy of the dataset. The input data in this case are eighteen control points used to characterize material distribution; the output data are total ceramic volume fraction and non-dimensional buckling load. Based on this dataset, the effect of hyperparameters in machine learning models on accuracy and computational cost is investigated to determine models with optimal hyperparameters, referred to as optimal models. The performance of the optimal models in predicting plate behavior is compared to each other. Furthermore, in terms of computational time and accuracy, the light gradient boosting machine model gives the best results compared to the others.

Keywords: tri-directional functionally graded plates; buckling; artificial neural network; ensemble learning; random forest; extreme gradient boosting; light gradient boosting machine.

https://doi.org/10.31814/stce.huce2024-18(4)-08

1. Introduction

Functionally graded materials (FGMs) are novel composite materials with locally customized properties that show gradual compositional and structural variations throughout their volume. Numerous FGMs are frequently observed in nature; for example, FGMs are found in plants and seashells, as well as in bone structures. Since the initial proposal by Niino et al. [1] to produce a thermally graded metal-to-ceramic phase, FGMs have already been the subject of extensive research. FGMs' structures and compositions can be precisely designed for specialized multifunctional characteristics. For this reason, FGMs are highly desirable for a wide range of applications, such as biomedical implants, sensors, aerospace engineering, civil engineering, and so on [2–5].

Many research investigations have been conducted on unidirectional FGMs because this type of FGMs has been utilized the most widely [6–9]. Despite their widespread use, unidirectional FGMs may not always be the most effective method for designing structures that withstand harsh environments. Thus, it stands to reason that bi- or tri-directional FGMs might be more successful in harsh environments, and numerous studies about bi- or tri-directional FGMs have been suggested [10–14]. For example, Tang Ye et al. [14] used the generalized differential quadrature method to predict dynamic behaviors of tri-directional functionally graded beams by solving the governing equation. The

*Corresponding author. *E-mail address:* dieudtt@huflit.edu.vn (Do, D. T. T.)

results showed that, in contrast to uni- or bi-directional FGMs structures, the tri-directional FGMs indexes can offer a much larger range for tuning/tailing mechanical behaviors. Tao et al. [10] utilized isogeometric analysis (IGA) to investigate thermal postbuckling and thermally induced postbuckled flutter of the tri-directional FGMs. In comparison to uni-directional FGMs, tri-directional FGMs showed a greater variety of postbukling paths and more complex deformations. This paper thus goes into further detail on the significance of tri-directional FGMs.

As mentioned in the above studies, numerical methods such as the finite element method and IGA have been used to analyze the behavior of tri-directional FGM structures. Nevertheless, applying such analytical techniques will take a lot of computational cost, particularly when dealing with structural optimization problems. As a result, machine learning algorithms have been suggested as an alternative to analytical methods for quickly predicting the behavior of functionally graded structures as in the studies [8, 13, 15–19]. For instance, Vaishali et al. [18] used support vector machine model to investigate stochastic dynamic characterization of functionally graded shells. Such approach reduces significant computational costs while ensuring the accuracy of results. Do et al. [19] utilized artificial neural networks (ANN) and extreme gradient boosting (XGBoost) for predicting dynamic behaviors of functionally graded plates. These methods not only ensure the accuracy of obtained results but also reduce computational time. Moreover, Do et al. [13] employed ANN to predict free vibration and buckling problems of tri-directional functionally graded plates. By using this method, the material optimization process to determine an optimal material distribution in the plate can be carried out much faster. From the above discussions, it can be seen that the number of studies related to applying machine learning methods to the tri-directional FG plate problems is still very limited. In particular, to the author's knowledge, ensemble learning methods have not been used for tri-directional FG plate problems.

In machine learning, ensemble learning is an effective technique that provides a reliable means of enhancing model performance and predictive accuracy. Ensemble learning methods are useful in the machine learning toolbox because they combine the strengths of several individual models to produce results that frequently outperform any single model. Therefore, ensemble learning methods such as random forest, extreme gradient boosting (XGBoost), and light gradient boosting machine (LightGBM) have been applied to various fields [20–27]. For example, Qui [20] used XGBoost for damage diagnosis for trusses based on incomplete free vibration properties. Kulkarni et al. [26] used White Shark-assisted Extreme Gradient Boost (WS_ExGB) model to diagnose microvascular diseases. Wherein XGBoost was utilized as a classification model, and White Shark was used to optimize classification parameters. From the above discussions, it can be seen that ensemble learning methods have not been used to predict behaviors of the tri-directional FG plates under compression.

In this paper, behavior prediction of tri-directional FG plates under uni-axial compression will be investigated using ANN and ensemble learning methods including random forest, XGBoost, and LightGBM. Data pairs will be created by using IGA to ensure the accuracy of the data. The accuracy and effectiveness of the analysis method were verified in the study [13]. Eighteen control points used to control material distribution in the plate are considered inputs while total ceramic volume fraction and non-dimensional buckling load are outputs. The effect of hyperparameters of each model on the accuracy and computational time are also investigated to select optimal machine learning models. Finally, results obtained by these optimal models are compared to each other in terms of accuracy and computational costs.

2. Ensemble learning methods

In machine learning, ensemble learning is the process of training multiple models to solve a common problem and combining their predictions to improve overall performance. In other words, an ensemble model makes predictions that are more accurate than those of a single model by combining several different models. Ensemble learning methods can be used to handle various tasks such as classification, clustering, and regression. Ensemble learning methods can be divided into three major categories: bagging, boosting, and stacking.

2.1. Random forest

Random forest proposed by Breiman [28] is an extension of the bagging method. This method combines feature randomness and bagging to generate an uncorrelated forest of decision trees. This is one of the main distinctions between random forest and decision trees. Random forest only chooses part of the features, whereas decision trees take into account all potential feature splits. The random forest algorithm works as follows:

- Step 1: From the given dataset, the algorithm chooses random samples.

- Step 2: For every sample chosen, the algorithm will produce a decision tree. After that, it will receive a prediction from every decision tree that was built.

- Step 3: Following that, each predicted result will be put to a vote. It will use mean for a regression problem and mode for a classification problem.

- Step 4: The algorithm will then decide which prediction result received the most votes to become the final prediction.

Random forest has the following advantages and disadvantages:

- Advantages: high accuracy, robustness to noise, non-parametric nature, estimating feature importance, handling missing data and outliers, and handling numerical and categorical data.

- Disadvantages: computational complexity, memory usage, longer prediction time, lack of interpretability, and overfitting.

2.2. Extreme gradient boosting (XGBoost)

XGBoost, which is proposed by Chen et al. [29], is one of ensemble learning algorithms known as a boosting algorithm that is used for regression and classification problems. XGBoost is renowned for its accuracy, speed, and efficiency. A stronger model in XGBoost is created by combining several weak models. It is employed based on decision trees, where a random selection of features and a subset of the data are used to train each tree. XGBoost begins with a single decision tree, which is then utilized to generate predictions using the training set. After that, the second model uses the first model's residuals as its target to train the model based on these residuals. Every subsequent model is trained using the residuals from the preceding model during a predetermined number of iterations of this process.

XGBoost has the following advantages and disadvantages:

- Advantages: high accuracy, speed and scalability, performance, flexibility, interpretability, and regularization.

- Disadvantages: complexity, computational resources, overfitting, and hyperparameter tuning.

2.3. Light gradient boosting machine (LightGBM)

LightGBM, proposed by Ke et al. [30], is an gradient boosting algorithm that builds a stronger learner by gradually adding weak learners in a gradient descent approach. LightGBM optimizes training time and memory usage with techniques such as gradient-based one-side sampling. LightGBM

builds decision trees that expand leaf-wise, meaning that, depending on the gain, only one leaf is split for each condition. Sometimes, particularly with smaller datasets, leaf-wise trees can overfit. Overfitting can be prevented in part by restricting the depth of tree. LightGBM employs a histogram-based approach in which data is grouped into bins using a distribution histogram. The data is split, the gain is calculated, and iterations are performed using the bins rather than individual data points. Additionally, this method can be optimized for a sparse dataset. Besides, exclusive feature bundling is a feature of LightGBM that reduces dimensionality and increases speed and efficiency by combining exclusive features.

LightGBM has the following advantages and disadvantages:

- Advantages: higher efficiency, faster training speed, lower memory usage, compatibility with large datasets, and better accuracy.

- Disadvantages: overfitting, and hyperparameter tuning.

3. Numerical examples

This study investigates the buckling behaviors of tri-directional SUS304/Si_3N_4 square plate, as shown in Fig. 1, with length-to-thickness ratio a/h set to be 10, boundary condition CCCC, under uni-axial compression. The properties of the plate are estimated by Mori-Tanaka scheme. Material properties of the FG plate are given as

$$\begin{aligned} &\text{Ceramic } Si_3N_4\text{: } E_c = 348.43 \text{ GPa}, v_c = 0.24, \rho_c = 2370 \text{ kg/m}^3 \\ &\text{Metal SUS304: } E_m = 201.04 \text{ GPa}, v_m = 0.3262, \rho_m = 8166 \text{ kg/m}^3 \end{aligned} \tag{1}$$

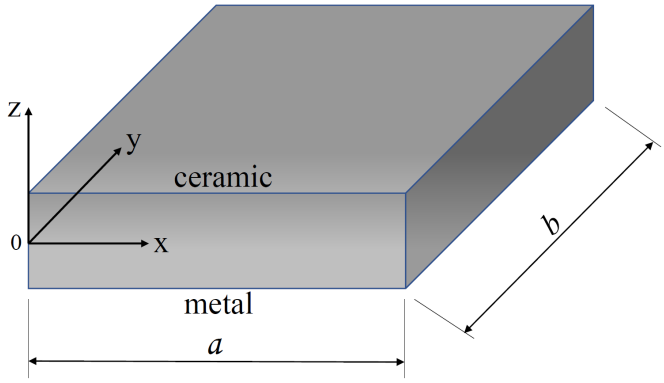

Figure 1. The functionally graded plate model

A combination of IGA and GSDT is employed to analyze the behaviors of the FG plate under uni-axial compression. The NURBS function describes material distribution in the plate in all three directions. Control points in the NURBS function control the material distribution and 18 control points are considered inputs in dataset for the training process in machine learning algorithms. The accuracy of the method was verified in the study [13]. Dataset consisting of 20000 data pairs used for the training process in machine learning methods is created by this analysis tool. In which the output data are total ceramic volume fraction and non-dimensional buckling load. In this study, the train/test ratio is set at 90/10 for all algorithms. The influence of hyperparameters in machine learning models on accuracy and computational cost is investigated to find models with optimal hyperparameters, referred to as optimal models. After that, a comparative analysis is done between the optimal models' predictions of plate behavior.

The training process is carried out using Python 3.7 on a laptop running Windows 11, 64-bit with an Intel® CoreTM i7-8550U CPU @1.80 GHz 2.00 GHz, 12.0 GB RAM.

3.1. Applying ANN for predicting the behavior of tri-directional FG plates under uni-axial compression

Firstly, an ANN architecture with one hidden layer, 100 nodes in each hidden layer, batch size 100, and 1000 epochs is considered in the first example to find an optimal combination of activation functions and optimizers. Specifically, five activation functions consisting of linear, sigmoid, tanh, softplus, and ReLU, as well as four optimizers consisting of SGD, Adagrad, RMSpop, and Adam are investigated. The accuracy and computational time obtained by such combinations are shown in Table 1. In which, MSE and MAPE symbolize mean square error and mean absolute percentage error, respectively. The table indicates that the optimizers SGD and Adam produce the best results, with MAPE less than 1%, indicating an accuracy of over 99%. A combination of Adam optimizer and ReLU function gives the best result with MSE of 1.85E-07 for both training and test sets, MAPE for training and test of 0.0738% and 0.0739%, respectively. Therefore, ANN architecture with the combination of Adam and ReLU is used in subsequent investigations.

Table 1. The accuracy and computational cost obtained by ANN with different combinations of activation functions and optimizers

Activation functions	Error types and computational time	Optimizers							
		SGD		Adagrad		RMSprop		Adam	
		Training	Test	Training	Test	Training	Test	Training	Test
Linear	MSE	2.03E-06	2.05E-06	7.43E-06	7.65E-06	1.00E-05	1.02E-05	2.26E-06	2.28E-06
	MAPE	0.1483	0.1473	0.4160	0.4211	0.6168	0.6209	0.2108	0.2096
	Time (second)	268.77		273.53		273.53		279.52	
Sigmoid	MSE	2.75E-06	2.77E-06	5.53E-04	5.42E-04	3.52E-05	3.35E-05	3.46E-06	3.46E-06
	MAPE	0.2042	0.2028	3.9426	3.8728	0.8752	0.8510	0.3245	0.3219
	Time (second)	274.08		278.66		280.00		289.38	
Tanh	MSE	5.95E-06	5.87E-06	3.49E-05	3.50E-05	3.23E-05	3.18E-05	5.00E-07	5.03E-07
	MAPE	0.3718	0.3673	0.9485	0.9561	1.0581	1.0445	0.1135	0.1132
	Time (second)	275.06		272.93		276.79		285.11	
Softplus	MSE	2.90E-06	3.04E-06	5.15E-05	5.05E-05	1.96E-05	1.86E-05	4.96E-06	4.99E-06
	MAPE	0.2432	0.2486	1.1726	1.1590	0.7215	0.7045	0.3885	0.3888
	Time (second)	298.73		300.62		303.07		307.97	
ReLU	MSE	3.82E-06	3.82E-06	1.17E-04	1.17E-04	1.13E-05	1.13E-05	1.85E-07	1.85E-07
	MAPE	0.2496	0.2472	1.7079	1.7252	0.6485	0.6474	0.0738	0.0739
	Time (second)	269.83		274.01		275.46		273.56	

Next, the impact of the number of hidden layers and nodes in each hidden layer in ANN architecture on accuracy and computational time is investigated. Gained results are tabulated in Table 2. According to the table, the computational time of ANN increases as the number of hidden layers or nodes increases. ANN ensures the accuracy of predicting behaviors of the FG plate when MAPE is less than 0.3% in all examined cases. ANN architecture with 2 hidden layers and 200 nodes in each hidden layer creates the lowest error in all investigated cases; therefore, this architecture is used for the next investigation to identify an optimal ANN architecture.

Finally, the effect of the number of epochs chosen in the set 100, 300, 500, 1000, 2000 on the accuracy and computational time is investigated and obtained results are presented in Table 3. As shown in the table, as the number of epochs increases, accuracy increases but so does computational time. Therefore, a value for epoch will be selected based on usage requirements in order to maintain a balance between the algorithm's computational cost and accuracy. The convergence history of the

Table 2. The accuracy and computational cost obtained by ANN with various configurations of the number of hidden layers and nodes in each hidden layer

No. nodes in each hidden layer	Error types and computational time	1 hidden layer		2 hidden layers		3 hidden layers	
		Training	Test	Training	Test	Training	Test
10	MSE	2.61E-06	2.58E-06	2.62E-06	2.66E-06	6.95E-07	7.14E-07
	MAPE	0.2071	0.2028	0.2643	0.2644	0.1440	0.1452
	Time (second)	268.59		284.31		290.91	
50	MSE	1.59E-06	1.59E-06	1.07E-06	1.10E-06	2.08E-07	2.06E-07
	MAPE	0.2294	0.2295	0.1913	0.1921	0.0792	0.0785
	Time (second)	270.65		300.65		317.42	
100	MSE	1.85E-07	1.85E-07	7.58E-07	7.63E-07	5.49E-07	5.50E-07
	MAPE	0.0738	0.0739	0.1525	0.1524	0.1245	0.1243
	Time (second)	273.56		334.81		359.71	
150	MSE	1.91E-07	1.94E-07	4.03E-07	4.06E-07	1.02E-06	1.03E-06
	MAPE	0.0647	0.0647	0.1186	0.1188	0.1947	0.1950
	Time (second)	281.70		345.44		408.88	
200	MSE	1.59E-07	1.57E-07	8.63E-08	9.01E-08	1.45E-07	1.47E-07
	MAPE	0.0654	0.0647	0.0447	0.0450	0.0704	0.0707
	Time (second)	286.04		386.67		479.41	

ANN model with epochs of 1000 is shown in Fig. 2. The figure illustrates that there is no overfitting and that test and training losses converge to zero.

Table 3. The accuracy and computational cost obtained by ANN with different epochs

Error types and computational time	Epochs									
	100		300		500		1000		2000	
	Training	Test	Training	Test	Training	Test	Training	Test	Training	Test
MSE	4.69E-06	4.73E-06	1.76E-06	1.77E-06	9.39E-07	9.50E-07	8.63E-08	9.01E-08	5.44E-08	5.58E-08
MAPE	0.3855	0.3867	0.2358	0.2357	0.1782	0.1781	0.0447	0.0450	0.0364	0.0367
Time (second)	43.05		121.01		197.85		386.67		777.38	

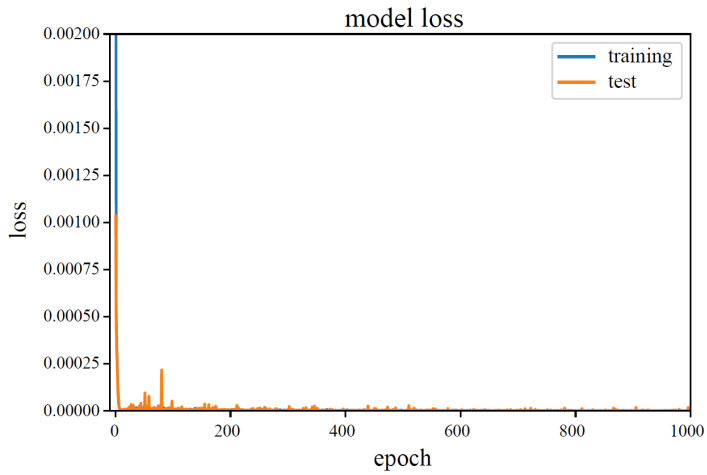

Figure 2. The convergence history of ANN model

3.2. Applying random forest for predicting the behavior of tri-directional FG plates under uni-axial compression

In this section, an optimal random forest model is selected by examining the influence of hyper-parameters such as the number of trees (n_estimators), and maximum tree depth (max_depth) on the efficiency of the method. In particular, values of n_estimators are selected in the set {10, 100, 1000, 2000, 5000, 10000}, and values of max_depth are selected in the set {1, 10, 20, 25}. Accuracy and computational time obtained by the method are presented in Table 4. From the table, it can be seen that max_depth of 20 and 25 give the best results with MAPE less than 1%, and these cases with n_estimators of 10 take around 4 seconds to train data. It takes less time to compute results from random forest with max_depth between 20 and 25 and n_estimators between 10 and 100 than from the optimal ANN architectures shown in Table 3. However, test errors obtained by random forest are always larger than corresponding training errors. While training and test errors obtained by ANN are nearly the same, optimal ANN architectures with epochs of 1000 and 10000 provide superior accuracy when compared to random forest.

Table 4. The accuracy and computational cost obtained by random forest with different combinations of n_estimators and max_depth

n_estimators	Error types and computational time	max_depth							
		1		10		20		25	
		Training	Test	Training	Test	Training	Test	Training	Test
10	MSE	2.86E-03	2.85E-03	1.23E-04	1.82E-04	9.19E-06	4.34E-05	8.46E-06	4.30E-05
	MAPE	8.9197	8.8506	1.7586	2.1139	0.2131	0.5657	0.2101	0.5622
	Time (second)	0.44		2.81		3.66		3.52	
100	MSE	2.85E-03	2.84E-03	1.01E-04	1.59E-04	3.55E-06	2.65E-05	3.48E-06	2.77E-05
	MAPE	8.9089	8.8391	1.6158	1.9733	0.1725	0.4522	0.1696	0.4485
	Time (second)	3.20		23.41		28.67		28.53	
1000	MSE	2.86E-03	2.85E-03	9.92E-05	1.57E-04	2.98E-06	2.59E-05	2.94E-06	2.61E-05
	MAPE	8.9152	8.8422	1.6048	1.9591	0.1500	0.4213	0.1503	0.4256
	Time (second)	32.11		183.29		224.55		225.86	
2000	MSE	2.86E-03	2.85E-03	9.89E-05	1.56E-04	2.95E-06	2.60E-05	2.90E-06	2.61E-05
	MAPE	8.9147	8.8403	1.6006	1.9580	0.1492	0.4225	0.1480	0.4210
	Time (second)	48.27		364.94		489.51		456.79	
5000	MSE	2.86E-03	2.85E-03	9.90E-05	1.57E-04	2.91E-06	2.57E-05	2.89E-06	2.55E-05
	MAPE	8.9161	8.8411	1.6021	1.9591	0.1479	0.4197	0.1471	0.4178
	Time (second)	137.19		910.64		1161.17		1178.84	
10000	MSE	2.86E-03	2.85E-03	9.88E-05	1.56E-04	2.88E-06	2.55E-05	2.90E-06	2.57E-05
	MAPE	8.9151	8.8399	1.6016	1.9550	0.1472	0.4199	0.1471	0.4196
	Time (second)	253.39		1810.14		2321.45		2350.46	

3.3. Applying XGBoost for predicting the behavior of tri-directional FG plates under uni-axial compression

In the XGBoost model, the influence of two hyperparameters n_estimators and max_depth on the model's effectiveness is also investigated. MSE, MAPE, and computational time obtained by these combinations are tabulated in Table 5. As shown in the table, XGBoost has improved both

accuracy and computational time compared to random forest method. In comparison to the optimal ANN architectures displayed in Table 3, combinations of max_depth from 10 and n_estimators from 100 in XGBoost provide better training errors and computational time. However, when max_depth increases, the error value of the test set also increases. This is one of the disadvantages of ensemble learning methods. Therefore, to restrict overfitting, choosing the appropriate max_depth parameter is necessary.

Table 5. The accuracy and computational cost obtained by XGBoost with different combinations of n_estimators and max_depth

n_estimators	Error types and computational time	max_depth							
		1		10		20		25	
		Training	Test	Training	Test	Training	Test	Training	Test
10	MSE	1.83E-03	1.81E-03	2.83E-05	6.17E-05	1.83E-05	4.55E-05	1.83E-05	4.55E-05
	MAPE	7.0768	7.0194	0.7667	1.0140	0.5782	0.7654	0.5782	0.7654
	Time (second)	0.12		0.96		1.36		1.45	
100	MSE	1.69E-04	1.78E-04	7.43E-08	1.71E-05	3.12E-08	1.77E-05	3.20E-08	1.78E-05
	MAPE	2.0780	2.1502	0.0243	0.1460	0.0133	0.1327	0.0134	0.1329
	Time (second)	0.85		6.99		7.32		7.66	
1000	MSE	1.64E-05	1.83E-05	7.43E-08	1.71E-05	3.12E-08	1.77E-05	3.20E-08	1.78E-05
	MAPE	0.6515	0.6893	0.0243	0.1460	0.0133	0.1327	0.0134	0.1329
	Time (second)	8.95		13.69		14.20		14.09	
2000	MSE	1.18E-05	1.34E-05	7.43E-08	1.71E-05	3.12E-08	1.77E-05	3.20E-08	1.78E-05
	MAPE	0.5507	0.5856	0.0243	0.1460	0.0133	0.1327	0.0134	0.1329
	Time (second)	17.08		22.49		21.68		22.26	
5000	MSE	7.18E-06	8.47E-06	7.43E-08	1.71E-05	3.12E-08	1.77E-05	3.20E-08	1.78E-05
	MAPE	0.4258	0.4590	0.0243	0.1460	0.0133	0.1327	0.0134	0.1329
	Time (second)	41.65		44.48		45.26		43.69	
10000	MSE	4.59E-06	5.64E-06	7.43E-08	1.71E-05	3.12E-08	1.77E-05	3.20E-08	1.78E-05
	MAPE	0.3362	0.3686	0.0243	0.1460	0.0133	0.1327	0.0134	0.1329
	Time (second)	81.35		82.93		82.01		82.19	

3.4. Applying LightGBM for predicting the behavior of tri-directional FG plates under uni-axial compression

The combination of n_estimators and max_depth is further explored in this section. In the LightGBM model, the number of nodes in the tree is represented as num_leaves, which controls the complexity of the tree model, set num_leaves = $2^{\text{max\_depth}}$. Table 6 presents results obtained by LightGBM. From the table, it can be seen that the LightGBM method improves the accuracy and calculation speed significantly compared with two methods: random forest and XGBoost. In comparison with ANN, LightGBM with n_estimators of 10000 and max_depth of 10 gives better results. Specifically, accuracy obtained by such LightGBM model reaches 99.9937% and 99.9633% for the training and testing process, respectively. And it takes only 33.31 seconds. While optimal ANN model with epochs of 1000 gives accuracy of 99.9553% and 99.9550% for the training and testing process. ANN requires 386.67 seconds to obtain results.

From the results obtained by all methods, it can be seen that ANN gives highly accurate predictions with more complex architecture; however, ANN requires more computational effort to get

results. In ensemble learning methods, LightGBM gives the best accuracy and computational time results. LightGBM is superior to random forest and XGBoost. Moreover, LightGBM also surpasses ANN in terms of accuracy and computational time. However, in ensemble learning methods, the hyperparameter max_depth needs to be chosen appropriately to limit overfitting.

Table 6. The accuracy and computational cost obtained by LightGBM with different combinations of n_estimators and max_depth

n_estimators	Error types and computational time	max_depth							
		1		10		20		25	
		Training	Test	Training	Test	Training	Test	Training	Test
10	MSE	2.81E-03	2.78E-03	1.89E-03	1.89E-03	1.47E-03	1.48E-03	1.39E-03	1.41E-03
	MAPE	8.8578	8.7226	7.2181	7.1931	6.3729	6.3777	6.1978	6.2117
	Time (second)	0.11		0.12		0.15		0.18	
100	MSE	8.23E-04	8.26E-04	1.16E-04	1.41E-04	5.93E-01	5.64E-05	2.86E-05	4.62E-05
	MAPE	4.6895	4.7073	1.6962	1.8812	0.9401	1.1664	0.8460	1.0522
	Time (second)	0.32		0.51		0.75		0.91	
1000	MSE	1.54E-05	1.76E-05	6.07E-06	9.93E-06	7.09E-07	4.44E-06	3.45E-07	3.94E-06
	MAPE	0.6325	0.6725	0.3752	0.4790	0.1019	0.2015	0.0644	0.1467
	Time (second)	1.72		3.41		5.32		6.70	
2000	MSE	9.46E-06	1.07E-05	1.81E-06	4.14E-06	7.58E+00	2.91E-06	1.96E-08	3.22E-06
	MAPE	0.4977	0.5259	0.1897	0.2762	0.0205	0.0818	0.0105	0.0649
	Time (second)	3.40		6.52		10.62		11.67	
5000	MSE	6.89E-06	7.75E-06	1.30E-07	1.32E-06	2.03E-10	2.76E-06	2.18E-11	3.17E-06
	MAPE	0.4241	0.4474	0.0401	0.0877	0.0009	0.0480	0.0003	0.0474
	Time (second)	8.52		15.22		26.19		29.30	
10000	MSE	5.52E-06	6.27E-06	5.76E-09	1.06E-06	4.67E-14	2.76E-06	9.88E-16	3.17E-06
	MAPE	0.3785	0.3994	0.0063	0.0367	1.28E-05	0.0465	2.92E-06	0.0470
	Time (second)	16.49		33.31		55.03		64.15	

4. Conclusions

In this paper, ANN and ensemble learning methods such as random forest, XGBoost, and LightGBM have been successfully applied to predict buckling behaviors of the tri-directional SUS304/Si_3N_4 square plate. These models are able to predict non-dimensional buckling load and total ceramic volume fraction with high accuracy for any given set of eighteen control points. The influence of hyperparameters in the models on the accuracy and computational time has been investigated to find the optimal models. The results showed that the LightGBM model gives better results than the remaining models. ANN can ensure accuracy in prediction but is time-consuming. Ensemble learning methods help to save computational time but setting the max_depth hyperparameter appropriately is necessary to restrict overfitting. In particular, with the proven effectiveness of the LightGBM model, it can be extended to large datasets to save computational costs while still ensuring prediction accuracy. The present methodology can be extended to more complex engineering problems such as inverse problems and shells.

Acknowledgements

We are grateful to Ho Chi Minh City University of Foreign Languages - Information Technology for their support of this research.

References

[1] Niino, M., Hirai, T., Watanabe, R. (1987). Functionally gradient materials. In pursuit of super heat resisting materials for spacecraft. *Journal of the Japan Society for Composite Materials*, 13(6):257–264.

[2] Ury, N., Bocklund, B., Perron, A., Bertsch, K. M. (2024). Automated path planning for functionally graded materials considering phase stability and solidification behavior: Application to the Mo-Nb-Ta-Ti system. *Computational Materials Science*, 244:113172.

[3] Kumar, S., Murthy Reddy, K. V. V. S., Kumar, A., Rohini Devi, G. (2013). Development and characterization of polymer-ceramic continuous fiber reinforced functionally graded composites for aerospace application. *Aerospace Science and Technology*, 26(1):185–191.

[4] Rouf, S., Malik, A., Raina, A., Irfan Ul Haq, M., Naveed, N., Zolfagharian, A., Bodaghi, M. (2022). Functionally graded additive manufacturing for orthopedic applications. *Journal of Orthopaedics*, 33: 70–80.

[5] Canpolat, Ö., Çanakçı, A., Erdemir, F. (2023). SS316L/Al2O3 functionally graded material for potential biomedical applications. *Materials Chemistry and Physics*, 293:126958.

[6] Jin, G., Su, Z., Shi, S., Ye, T., Gao, S. (2014). Three-dimensional exact solution for the free vibration of arbitrarily thick functionally graded rectangular plates with general boundary conditions. *Composite Structures*, 108:565–577.

[7] Vel, S. S., Batra, R. C. (2004). Three-dimensional exact solution for the vibration of functionally graded rectangular plates. *Journal of Sound and Vibration*, 272(3–5):703–730.

[8] Do, D. T. T., Lee, D., Lee, J. (2019). Material optimization of functionally graded plates using deep neural network and modified symbiotic organisms search for eigenvalue problems. *Composites Part B: Engineering*, 159:300–326.

[9] Quoc, T. H., Tu, T. M., Long, N. V. (2016). Bending and free vibration analysis of functionally graded plates using new eight-unknown shear deformation theory by finite element method. *Vietnam Journal of Science and Technology*, 54(3):402.

[10] Tao, C., Dai, T., Chen, Y. (2024). Thermal postbuckling and thermally induced postbuckled flutter of tri-directional functionally graded plates in yawed supersonic flow. *Aerospace Science and Technology*, 154:109491.

[11] Tang, H., Nguyen, N. V., Lee, J. (2024). Accelerating tri-directional material distribution optimization in functionally graded plates with an adaptive design control point variable selection. *Computer Methods in Applied Mechanics and Engineering*, 418:116474.

[12] Tang, H., Nguyen, N. V., Lee, J. (2024). Simultaneous optimal tri-directional distribution of material and porosity in functionally graded plates under free vibration. *Thin-Walled Structures*, 196:111496.

[13] Do, D. T. T., Nguyen-Xuan, H., Lee, J. (2020). Material optimization of tri-directional functionally graded plates by using deep neural network and isogeometric multimesh design approach. *Applied Mathematical Modelling*, 87:501–533.

[14] Tang, Y., Li, C.-L., Yang, T. (2023). Application of the generalized differential quadrature method to study vibration and dynamic stability of tri-directional functionally graded beam under magneto-electro-elastic fields. *Engineering Analysis with Boundary Elements*, 146:808–823.

[15] Feng, Y., Wu, D., Chen, X., Gao, W. (2024). Machine learning aided stochastic free vibration analysis of functionally graded porous plates. In *Machine Learning Aided Analysis, Design, and Additive Manufacturing of Functionally Graded Porous Composite Structures*, Elsevier, 293–311.

[16] Truong, T. T., Lee, S., Lee, J. (2020). An artificial neural network-differential evolution approach for optimization of bidirectional functionally graded beams. *Composite Structures*, 233:111517.

[17] Do, D. T. T., Thai, S., Bui, T. Q. (2022). Long short-term memory for nonlinear static analysis of functionally graded plates. *Journal of Science and Technology in Civil Engineering (STCE) - HUCE*, 16 (3):1–17.

[18] Vaishali, Mukhopadhyay, T., Karsh, P. K., Basu, B., Dey, S. (2020). Machine learning based stochastic dynamic analysis of functionally graded shells. *Composite Structures*, 237:111870.

[19] Do, D. T. T., Thai, S. (2023). Transient analysis of functionally graded plates using extreme gradient boosting. *Journal of Science and Technology in Civil Engineering (STCE) - HUCE*, 17(4):26–36.

[20] Qui, L. X. (2023). Damage identification of trusses using limited modal features and ensemble learning. *Journal of Science and Technology in Civil Engineering (STCE) - HUCE*, 17(2):9–20.
[21] Quan, T. V., Linh, N. N., Tan, N. N. (2023). Investigating compressive strength of concrete containing steel fiber by data-driven approach. *Journal of Science and Technology in Civil Engineering (STCE) - HUCE*, 17(3):65–79.
[22] Linh Khanh, P. N., Bao Ngan, N. H. (2023). Machine learning-based pedo transfer function for estimating the soil compression index. *Journal of Science and Technology in Civil Engineering (STCE) - HUCE*, 17 (1):67–78.
[23] Cong, Y., Inazumi, S. (2024). Ensemble learning for predicting subsurface bearing layer depths in Tokyo. *Results in Engineering*, 23:102654.
[24] Chen, Q., Xiao, Z., Yao, Q. (2024). Quantile control via random forest. *Journal of Econometrics*, 105789.
[25] Du, Q., Zhai, J. (2024). Application of artificial intelligence Sensors based on random forest algorithm in financial recognition models. *Measurement: Sensors*, 33:101245.
[26] Kulkarni, M. D., Deore, S. S. (2024). Detection of microvascular disease in Type-2 diabetes mellitus patients using White Shark assisted Extreme Gradient Boosted model. *Biomedical Signal Processing and Control*, 95:106326.
[27] Song, X., Chen, Z. S. (2024). Enhancing financial time series forecasting in the shipping market: A hybrid approach with Light Gradient Boosting Machine. *Engineering Applications of Artificial Intelligence*, 136: 108942.
[28] Breiman, L. (2001). Random forests. *Machine Learning*, 45(1):5–32.
[29] Chen, T., Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, ACM, 785–794.
[30] Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., Liu, T.-Y. (2017). LightGBM: A highly efficient gradient boosting decision tree. *Advances in Neural Information Processing Systems*, 30.