



Evaluating structural safety of trusses using Machine Learning

Tran-Hieu Nguyen, Anh-Tuan Vu

Hanoi University of Civil Engineering, Vietnam

hieunt2@nuce.edu.vn, <https://orcid.org/0000-0002-1446-5859>

antuanva@nuce.edu.vn



ABSTRACT. In this paper, a machine learning-based framework is developed to quickly evaluate the structural safety of trusses. Three numerical examples of a 10-bar truss, a 25-bar truss, and a 47-bar truss are used to illustrate the proposed framework. Firstly, several truss cases with different cross-sectional areas are generated by employing the Latin Hypercube Sampling method. Stresses inside truss members as well as displacements of nodes are determined through finite element analyses and obtained values are compared with design constraints. According to the constraint verification, the safety state is assigned as safe or unsafe. Members' sectional areas and the safety state are stored as the inputs and outputs of the training dataset, respectively. Three popular machine learning classifiers including Support Vector Machine, Deep Neural Network, and Adaptive Boosting are used for evaluating the safety of structures. The comparison is conducted based on two metrics: the accuracy and the area under the ROC curve. For the two first examples, three classifiers get more than 90% of accuracy. For the 47-bar truss, the accuracies of the Support Vector Machine model and the Deep Neural Network model are lower than 70% but the Adaptive Boosting model still retains the high accuracy of approximately 98%. In terms of the area under the ROC curve, the comparative results are similar. Overall, the Adaptive Boosting model outperforms the remaining models. In addition, an investigation is carried out to show the influence of the parameters on the performance of the Adaptive Boosting model.

KEYWORDS. Machine Learning; Classification; Adaptive Boosting; Structural Safety; Truss Structure.

Citation: Nguyen, T.-H., Vu, A.-T., Evaluating structural safety of trusses using Machine Learning, *Frattura ed Integrità Strutturale*, 58 (2021) 308-318.

Received: 19.08.2021

Accepted: 29.08.2021

Published: 01.10.2021

Copyright: © 2021 This is an open access article under the terms of the CC-BY 4.0, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

INTRODUCTION

Assessing the safety of an existing structure is a common problem in practice. For example, many constructions after a long period of use have been seriously degraded. Safety assessment is a key factor in deciding whether to rehabilitate or demolish these constructions. A second example might be: a steel structure exposed to a corrosive environment needs to be regularly checked the structural safety due to the development of rust. For one more example, a structure subjected to an extreme impact, such as seismic loads, fire loads, or blast loads should be evaluated the safety state before continuing use.



The traditional safety assessment of structural elements is often investigated based on experimental tests and numerical simulations. For instance, the research work conducted by Capuzucca and Bonci [1] indicated that the state of damage of Carbon Fiber Reinforced Polymer (CFRP) laminate elements can be evaluated by measuring the reduction of vibration frequencies. Subsequently, a hybrid approach that combines the proper orthogonal decomposition with radial basis function (POD-RBF) and the cuckoo search optimization algorithm to determine the position and dimensions of notches in CFRP beams was proposed in Ref. [2]. The proposed approach was validated with experimental results in Ref. [1]. In 2019, a technique coupling POD-RBF, the extended isogeometric analysis method, and the Jaya algorithm was developed [3]. This technique allows to accurately identify the locations of cracks in a steel plate based on strain readings. In [4], the damages in the FRP-strengthened reinforced concrete (RC) beams were investigated using Finite Element Modeling. Recently, a two-stage approach for detecting damages was introduced, in which the modal strain energy change ratio is used to forecast the damages' locations and the slime mould algorithm is then employed to quantify the damages [5]. This approach is verified with the experimental results of a 3D four stories frame. Such methods can accurately evaluate the safety of structures but very time-consuming.

In some cases, there is a need for a model that can rapidly predict the safety state of structures in order to promptly evacuate people from the whole structure if it is unsafe. After that, the structure still needs to be exactly re-checked by engineers. The rapid safety assessment model can be integrated with the structural health monitoring system, in which information obtained from sensors is fed into such model as inputs. The output of the model is a prediction of whether the structure is safe or unsafe. This is a kind of binary classification problem and it can be solved by machine learning (ML) algorithms. In the literature, there are many previous studies that have applied ML for structural health monitoring.

In several works, Artificial Neural Networks (ANNs) were used to forecast the damages in the plates instead of solving the inverse problem by optimization algorithms [6,7]. Besides, it can be observed that the topic of Deep Learning has been received great attention from researchers. Since AlexNet, the winner of the ImageNet challenge, was first introduced in 2012, the convolutional neural network (CNN) architecture has become extremely popular in the field of computer vision and it has been used in many domains. In the field of civil engineering, the CNN architecture and its variations such as U-Net have been applied to detect cracks on concrete [8-10] as well as welded joints [11], corrosions on steel surfaces [12], damages on buildings [13], etc. For handling time series data, the RNN architecture and its variation LSTM are state-of-the-art [14-16].

Obviously, deep learning algorithms like CNN, RNN outperform conventional ML algorithms on unstructured data due to their automatic feature extraction capability. However, they don't work well on tabular data. For such kinds of data, the conventional ML algorithms, for example, Decision Tree (DT), Support Vector Machine (SVM), are good choices. Recently, ensemble methods have been particularly preferred, shown by their applications in Kaggle data science competitions. The concept of ensemble methods is to create a strong classifier from several weak classifiers. Some commonly used ensemble methods are Random Forest (RF), Extreme Gradient Boosting (XGBoost), Gradient Tree Boosting (GTB). In the literature, Zhang et al. [17] utilized DT and RF for evaluating the structural safety state of RC buildings after an earthquake. In [18], the GTB algorithm was used to evaluate the safety of trusses. In addition, a comparative study of ML algorithms for predicting the load-carrying capacity of steel frames was conducted [19]. The results showed that two ensemble methods, including RF and GTB, achieved better performance than the remaining ones.

Among ensemble methods, AdaBoost, short for Adaptive Boosting, was the first successful boosting algorithm developed for binary classification [20]. This algorithm has been applied to predict the failure modes and the load-bearing capacity of RC columns [21], and the compressive strength for concrete [22]. However, according to the literature review, there has not been a study related to the application of AdaBoost for classifying the safety of structures.

This paper aims to investigate the capability of the ensemble method AdaBoost in evaluating the safety of truss structures. For this purpose, the performance of AdaBoost is compared with two popular classification algorithms, i.e., ANN and SVM, in terms of the accuracy and the area under the ROC curve. The comparison is conducted on three well-known truss structures of 10 bars, 25 bars, and 47 bars. Additionally, an investigation on the influence of the parameters on the performance of the AdaBoost model is also carried out.

CLASSIFICATION OF THE SAFETY STATE OF TRUSSES USING MACHINE LEARNING

Machine Learning-based framework for safety classification of trusses

The framework for structural safety classification of truss structures using ML is presented in Fig. 1. The main characteristics of truss structures are truss members' sectional areas and these values are used as inputs of the ML model. The output is the safety state of structures. The inputs of training data are generated using a sampling method,

which is the Latin Hypercube Sampling (LHS) method in this paper. The cross-sectional areas that have just been created are assigned to truss members and the structure is then analyzed using the finite element method (FEM). Based on the results obtained from the structural analysis such as stresses inside members and the deflection, design constraints are checked. If all constraints are satisfied, this structure is safe. Otherwise, if any constraint is violated, this structure is considered unsafe. Inputs and outputs are also saved into the database. Once sufficient data has been collected, the training process begins. If the accuracy of the trained model when verifying on the testing dataset is acceptable, this ML model is ready to use.

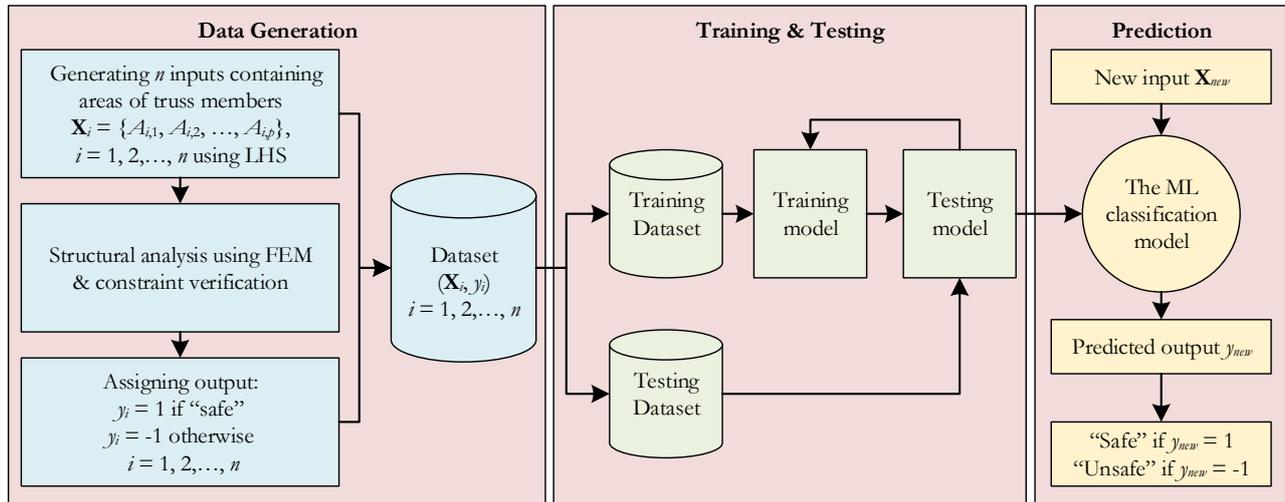


Figure 1: Overview of the ML-based framework for safety classification of trusses.

The above procedure can be applied to all ML algorithms. In this study, three powerful classification algorithms are taken for comparison including SVM, ANN, and AdaBoost. The brief introductions of these algorithms are presented in the following sections.

Support Vector Machine

SVM was initially developed to solve binary classification problems. For separating data, the best hyperplane is found by maximizing the margin between it and the support vectors as shown in Fig. 2. Other kernel functions such as sigmoid kernel, polynomial kernel, or RBF kernel can be used to separate nonlinear data.

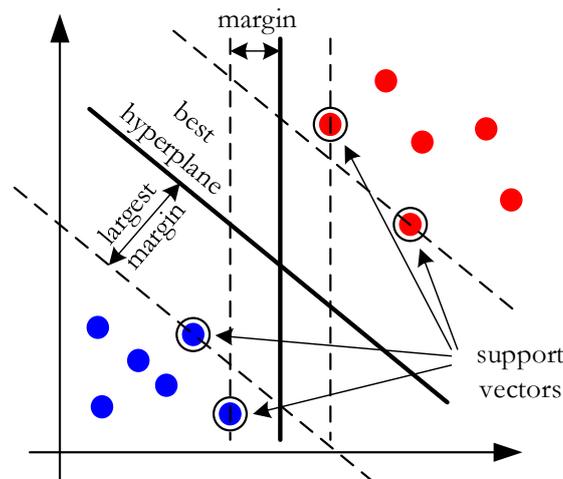


Figure 2: Support Vector Machine.

Artificial Neural Network

The first ANN model that was introduced in 1958 by Rosenblatt contains only three layers: an input layer, a hidden layer, and an output layer. Each layer has some neurons where the activation function is attached to mimic the nonlinear data. The state-of-the-art ANNs have more than one hidden layer that allows learning extremely complex data. Fig. 3 presents a typical architecture of an ANN model with multiple hidden layers.

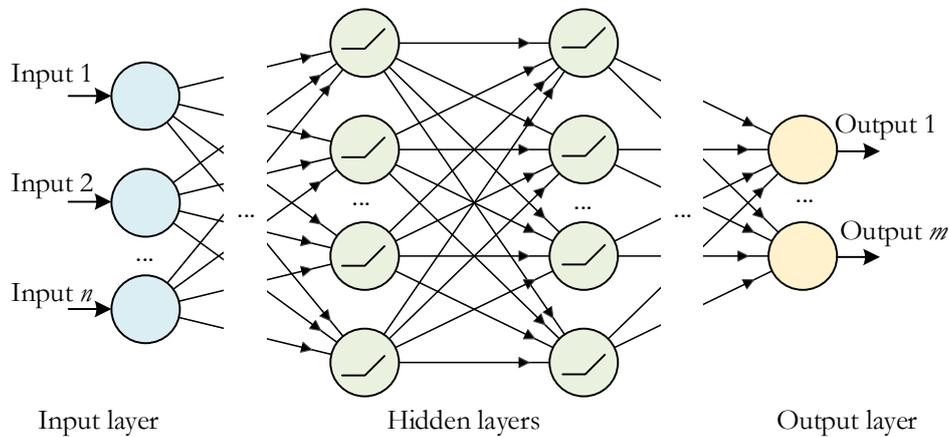


Figure 3: Artificial Neural Network.

Adaptive Boosting

The base classifier used in the AdaBoost model is often a decision tree. However, a deep decision tree can be overfitted on a specific dataset. To overcome this problem, the AdaBoost model uses several shallow decision trees which are sequentially trained by the weighted data. Each decision tree is assigned a weight based on its accuracy. The final prediction is determined by using the weighted majority voting method. The AdaBoost algorithm is illustrated in Fig. 4.

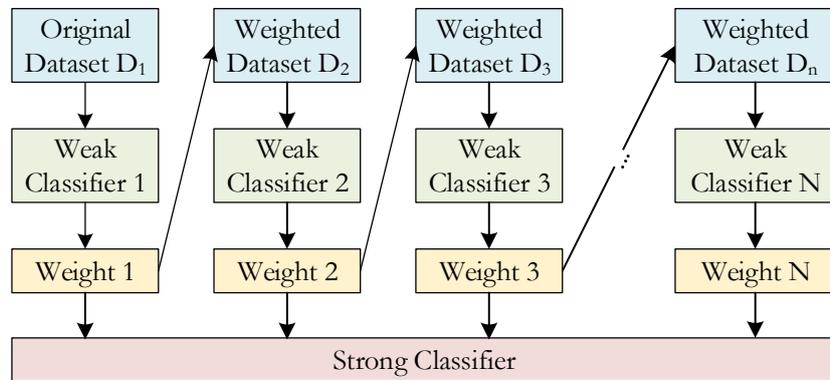


Figure 4: Adaptive Boosting.

DATASETS

Three well-known truss structures are used in this study. All these problems were initially expressed using the imperial system. In this study, the imperial units are kept without affecting the final results.

The first structure is a planar 10-bar truss which is displayed in Fig. 5. All members are made of the same material having the modulus of elasticity $E=10,000$ ksi and the density $\rho=0.1$ lb/in³. The cross-sectional areas of truss members range from 6 to 35 in². The allowable stress inside all members is ± 25 ksi, and the allowable displacement for all nodes is ± 2 in.

The second structure is a spatial 25-bar truss as presented in Fig. 6. This structure is fabricated from the same material as the 10-bar truss. There are eight groups of members with corresponding stress limits shown in Tab. 1. Members in the same

group have the same cross-sectional area varying from 0.6 to 3.5 in². The limitation of displacements along horizontal directions is ± 0.35 in. This structure subjects to two load cases independently. The first load case contains two point loads $P_1 = (0, 20, -5)$ and $P_2 = (0, -20, -5)$ acting on the node (1) and the node (2), respectively. The second load case consists of four point loads $P_1 = (1, 10, -5)$, $P_2 = (0, 10, -5)$, $P_3 = (0.5, 0, 0)$, and $P_6 = (0.6, 0, 0)$ acting on the node (1), (2), (3), and (6), respectively.

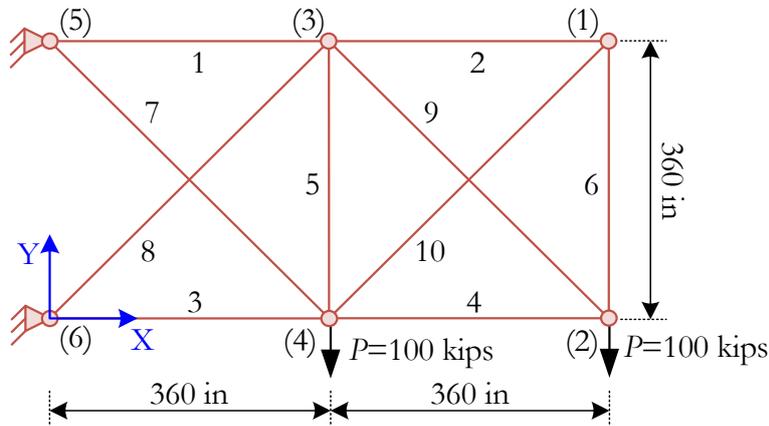


Figure 5: 10-bar truss structure.

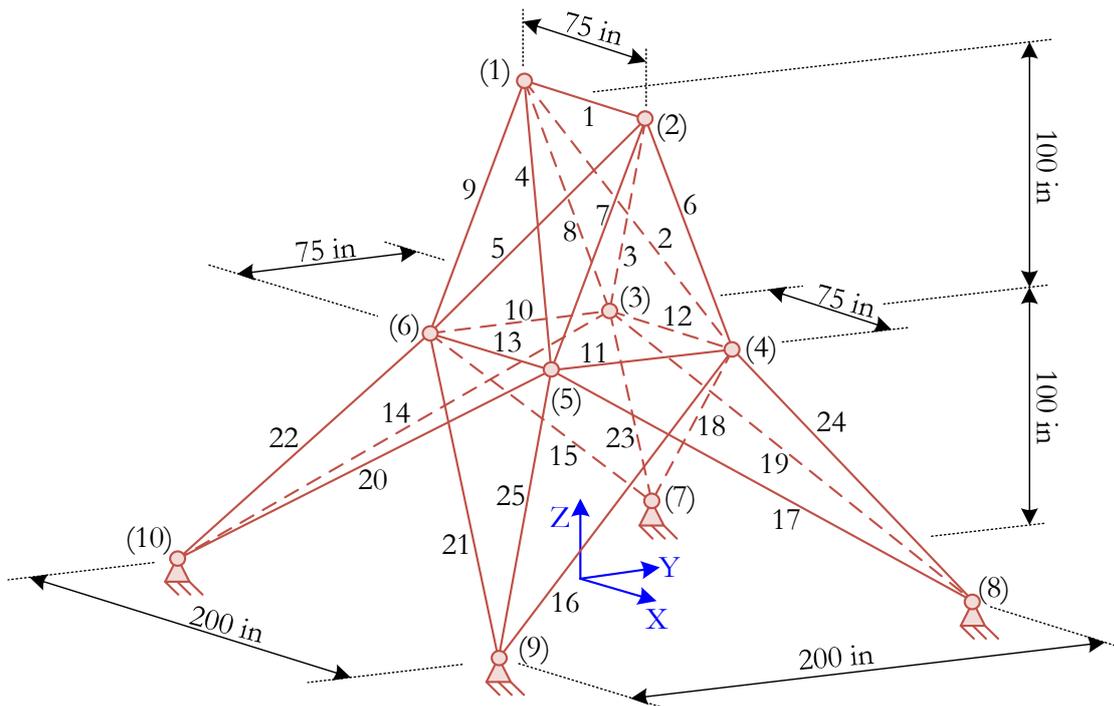


Figure 6: 25-bar truss structure.

The final structure is a planar truss as schematized in Fig. 7. This structure contains 47 members which are divided into 27 groups as shown in Tab. 2. All members in a group are assigned a cross-sectional area range from 0.1 to 35 in². The modulus of elasticity of the material used in this structure $E=30,000$ ksi and the material density $\rho=0.3$ lb/in³. This tower is subjected to one of three independent load cases. The first load case (LC1) consists of two loads of 14 kips and 6 kips applying to the node (17). The second load case (LC2) has two loads of 14 kips and 6 kips acting on the node (22). The third load case (LC3) is the sum of the two above load cases (LC1) and (LC2).



Group	Member	Maximum compressive stress	Maximum tensile stress
1	1	35.092	40.0
2	2, 3, 4, 5	11.590	40.0
3	6, 7, 8, 9	17.305	40.0
4	10, 11	35.092	40.0
5	12, 13	35.092	40.0
6	14, 15, 16, 17	6.759	40.0
7	18, 19, 20, 21	6.959	40.0
8	22, 23, 24, 25	11.082	40.0

Table 1: Limitation of stresses for eight groups of the 25-bar truss structure - unit: ksi.

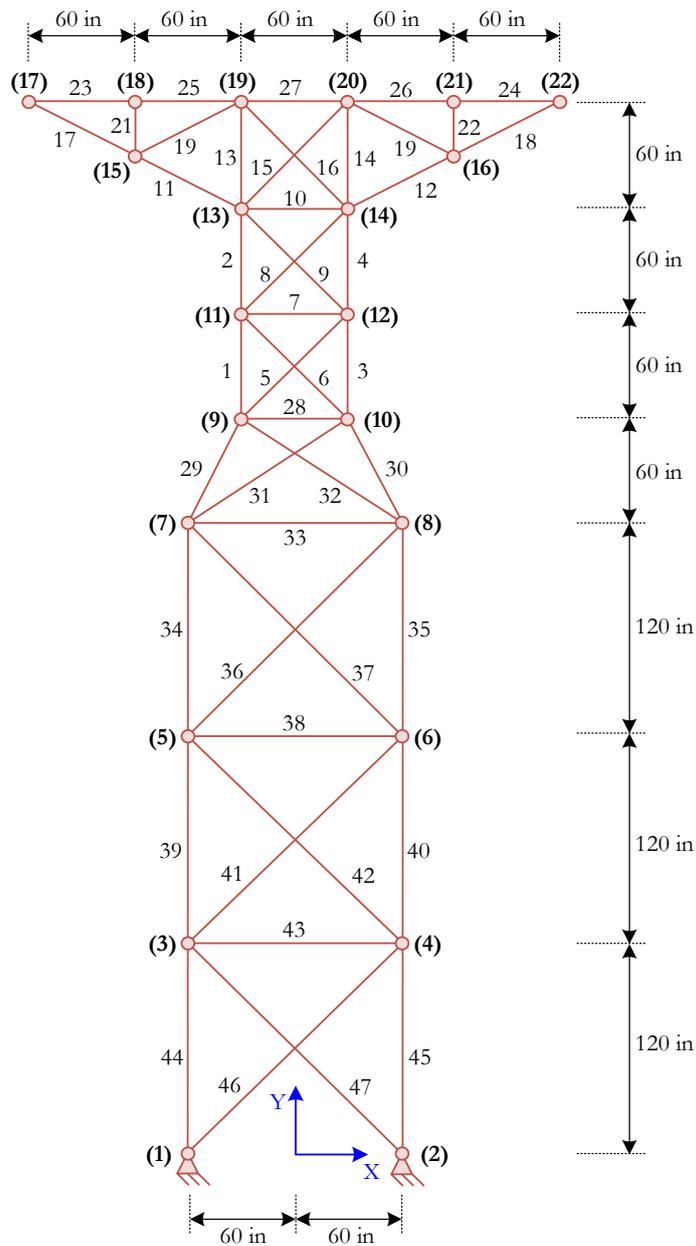


Figure 7: 47-bar truss structure.



Group	Members	Group	Members	Group	Members
1	1, 3	10	17, 18	19	33
2	2, 4	11	19, 20	20	34, 35
3	5, 6	12	21, 22	21	36, 37
4	7	13	23, 24	22	38
5	8, 9	14	25, 26	23	39, 40
6	10	15	27	24	41, 42
7	11, 12	16	28	25	43
8	13, 14	17	29, 30	26	44, 45
9	15, 16	18	31, 32	27	46, 47

Table 2: Group member of the 47-bar tower.

Stresses inside members must be smaller than limit values. The limited tensile and compressive stresses are 20 ksi and -15 ksi, respectively. The maximum buckling stress of the i th member can be calculated by the following equation:

$$\sigma_{\sigma}^i = -\frac{KEA_i}{L_i} \tag{1}$$

where: K is the buckling constant that is fixed to 3.96 in this study; A_i and L_i are the cross-sectional area and the length of i th member, respectively.

For each structure, two separated datasets are independently generated, one for training and one for testing. Each dataset contains 1000 data samples. The number of safe samples and unsafe samples in each dataset are shown in Fig. 8.

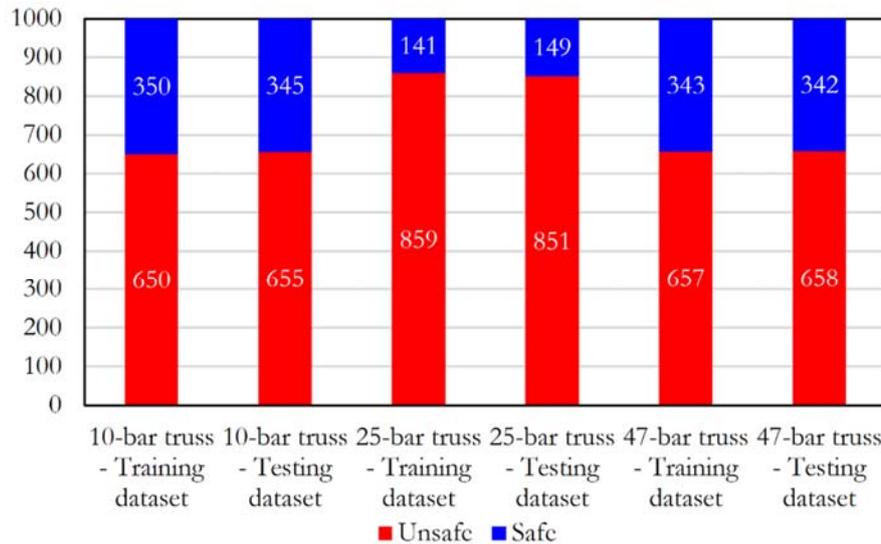


Figure 8: Distribution of safe and unsafe samples in the training and the testing datasets.

COMPARISON

Setups

ML models for evaluating the safety of truss structures are developed based on the open-source library scikit-learn [23]. There are three numerical examples and three classification algorithms used in this work, so a total of 9 models are developed. The hyperparameters tuning of ML models is conducted by combining the grid search



method and k -fold cross-validation. The training datasets are divided into five parts ($k=5$), in which four parts are used to train and the remaining one is used to validate. The final parameters of each algorithm are summarized in Tab. 3. After finding the hyperparameters, the full training datasets are fed into the ML models. When the training process is completed, these models are ready to predict the structural safety state of the testing datasets.

Algorithm	Hyperparameters
SVM	kernel='rbf'; C=1000; gamma=0.1
ANN	architecture (100-100-100-2); activation='relu'; solver=Adam; batch_size=5; max_iter=1000
AdaBoost	base_estimator=DecisionTreeClassifier; max_depth=2; n_estimators=100; learning_rate=1

Table 3: Hyperparameters of ML models.

Metrics

Two evaluation metrics are used in this study. The first metric is the accuracy which can be determined using the following equation:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

where: TP , TN are the numbers of positive and negative samples that are correctly classified; FP , FN are the numbers of positive and negative samples that are misclassified.

Additionally, the area under the ROC curve (AUC) is also used to compare three algorithms. The ROC curve originally developed for operators of military radar receivers is the relationship between True Positive Rate and False Positive Rate.

$$True\ Positive\ Rate = \frac{TP}{FN + TP} \tag{3}$$

$$False\ Positive\ Rate = \frac{FP}{TN + FP} \tag{4}$$

Results

Each problem is carried out 30 times. The average accuracies of three ML algorithms are summarized in Tab. 4. It can be seen that for two problems of 10-bar truss and 25-bar truss, all three algorithms achieve high accuracy (over 90%). Particularly for the 47-bar truss problem, the accuracies of the SVM and the ANN models are quite low (below 70%), while the AdaBoost models still retain the high accuracy (97.7%).

Problem	Number of features	SVM	ANN	AdaBoost
10-bar truss	10	0.959	0.937	0.936
25-bar truss	8	0.974	0.967	0.974
47-bar truss	27	0.658	0.669	0.976

Table 4: Accuracy comparison of three ML models.

Furthermore, Fig. 9 shows the typical ROC curves of these ML algorithms for three examples. There is a good agreement when comparing three algorithms in terms of the accuracy and the AUC metrics. In more detail, all SVM, ANN, and AdaBoost obtained the AUC close to one for the first two problems. However, the SVM and the ANN models obtain the AUC of 0.69 while the AdaBoost model achieves the AUC=0.99 when classifying on the 47-bar truss testing dataset.

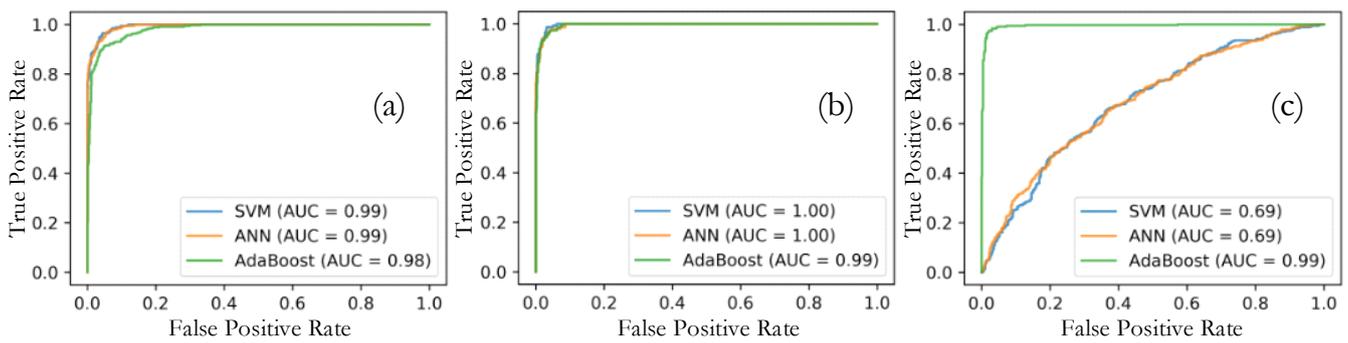


Figure 9: ROC curves of three ML algorithms for three datasets. (a) 10-bar truss – Testing dataset. (b) 25-bar truss – Testing dataset. (c) 47-bar truss – Testing dataset.

Overall, it can be noted that for problems with a small number of features, all three algorithms give accurate results. But for problems having a large number of features, the AdaBoost algorithm outperforms the two remaining algorithms.

MODEL PERFORMANCE ANALYSIS

One of the most important parts when building an ML model is the training data. In this study, data is collected through conducting a parametric finite element analysis (FEA). For large-scale structures, each FEA consumes several hours or days. Performing a large number of FEAs leads to time-consuming. Therefore, in this section, the influence of the number of samples of the training dataset is investigated. Besides, the performance of the AdaBoost model strongly depends on the number of base classifiers. Thus, the influence of the number of base classifiers is also considered in this section.

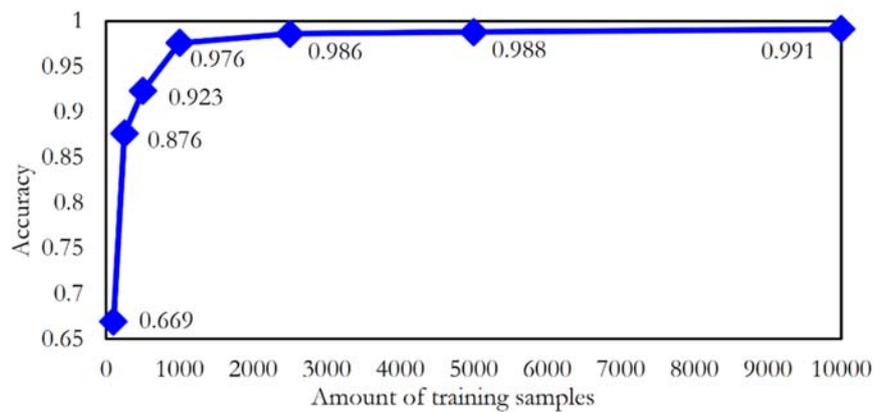


Figure 10: The influence of the training dataset amount on the performance of the AdaBoost model.

Influence of the training dataset amount

To investigate the influence of the training dataset amount, seven training datasets of the 47-bar truss problem are generated. The numbers of samples of seven datasets are 100, 250, 500, 1000, 2500, 5000, and 10000, respectively. The amount of the testing dataset remains 1000 samples. The accuracies of seven AdaBoost models are shown in Fig. 10. It can be observed that the accuracy of the classification model significantly improves when increasing the number of samples from 100 to 250 (0.669 for 100 samples and 0.876 for 250 samples). When augmenting the training dataset to 500 samples, the accuracy achieves 0.923. The accuracies for 1000, 2500, 5000, 10000 samples are 0.976, 0.986, 0.988, 0.991, respectively. Generally, the number of 1000 samples is a good choice when balancing the accuracy and the quantity of the training data.

Influence of the number of base classifiers

An important key factor of an ensemble model is the number of base classifiers used in this model. In this study, six AdaBoost models are compared where the numbers of base classifiers are 5, 10, 50, 100, 500, and 1000, respectively. These



models are trained by the same 47-bar truss training dataset of 1000 samples. The prediction results on the 47-bar truss testing dataset are presented in Fig. 11. It can be clearly seen that the accuracy of the AdaBoost model is greatly enhanced when extending the number of base classifiers from 5 to 50. However, when increasing the number of base classifiers from 50 to 1000, the model quality does not improve.

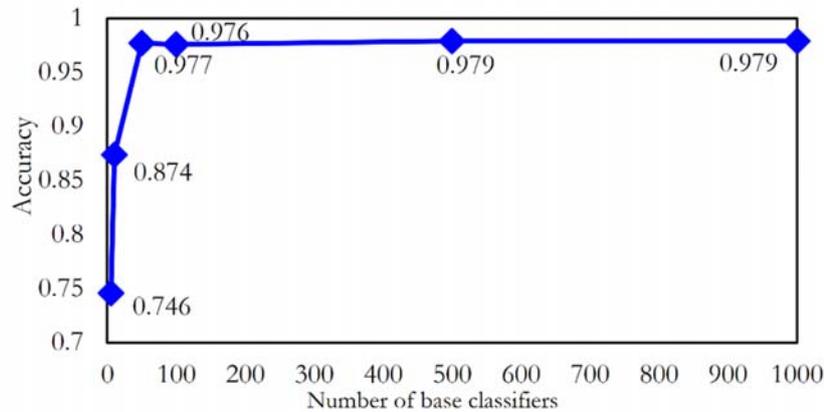


Figure 11: The influence of the number of base classifiers on the performance of the AdaBoost model.

CONCLUSIONS

The rapid evaluation of the structural safety state is an important task for escaping people when a hazard occurs. In this study, three machine learning algorithms including Support Vector Machine, Artificial Neural Network, and Adaptive Boosting are used to identify the safety state of truss structures. The results of the present work demonstrate the potential application of machine learning for structural safety evaluation. The comparative study indicates that all three algorithms achieve high accuracy (over 90%) for small-scale structures with few input features. However, for large-scale structures having many input features, the AdaBoost algorithm exhibits a strong ability in comparison with other algorithms (over 95% for the AdaBoost, about 65% for both the SVM and the ANN). Additionally, an investigation on the influence of the training dataset size and the number of base classifiers is implemented. The results of the investigation provide a good suggestion when developing machine learning models later. In the future, the study can be extended to other structures such as frames, dams, etc.

ACKNOWLEDGMENT

The first author T.-H. Nguyen was funded by Vingroup Joint Stock Company and supported by the Domestic Ph.D. Scholarship Programme of Vingroup Innovation Foundation (VINIF), Vingroup Big Data Institute (VINBIGDATA) code VINIF.2020.TS.134.

REFERENCES

- [1] Capozucca, R. and Bonci, B. (2015). Notched CFRP laminates under vibration. *Composite Structures*, 122, pp.367-375. DOI: 10.1016/j.compstruct.2014.11.062.
- [2] Samir, K., Brahim, B., Capozucca, R. and Wahab, M.A. (2018). Damage detection in CFRP composite beams based on vibration analysis using proper orthogonal decomposition method with radial basis functions and cuckoo search algorithm. *Composite Structures*, 187, pp.344-353. DOI: 10.1016/j.compstruct.2017.12.058.
- [3] Khatir, S. and Wahab, M.A. (2019). Fast simulations for solving fracture mechanics inverse problems using POD-RBF XIGA and Jaya algorithm. *Engineering Fracture Mechanics*, 205, pp.285-300. DOI: 10.1016/j.engfracmech.2018.09.032.
- [4] Magagnini, E. and Capozucca, R. (2020). Detection of Damage in RC Beams Strengthened with NSM CFRP Rectangular Rod by Finite Element Modeling. In: *Fracture, Fatigue and Wear*, Singapore, Springer, pp. 227-242.



- DOI: 10.1007/978-981-15-9893-7_16.
- [5] Tiachacht, S., Khatir, S., Le Thanh, C., Rao, R.V., Mirjalili, S. and Wahab, M.A. (2021). Inverse problem for dynamic structural health monitoring based on slime mould algorithm. *Engineering with Computers*, pp.1-24. DOI: 10.1007/s00366-021-01378-8.
- [6] Zenzen, R., Khatir, S., Belaidi, I., Le Thanh, C. and Wahab, M.A. (2020). A modified transmissibility indicator and Artificial Neural Network for damage identification and quantification in laminated composite structures. *Composite Structures*, 248, p.112497. DOI: 10.1016/j.compstruct.2020.112497.
- [7] Khatir, S., Boutchicha, D., Le Thanh, C., Tran-Ngoc, H., Nguyen, T.N. and Abdel-Wahab, M. (2020). Improved ANN technique combined with Jaya algorithm for crack identification in plates using XIGA and experimental analysis. *Theoretical and Applied Fracture Mechanics*, 107, p.102554. DOI: 10.1016/j.tafmec.2020.102554.
- [8] Cha, Y.J., Choi, W. and Büyüköztürk, O. (2017). Deep learning-based crack damage detection using convolutional neural networks. *Computer-Aided Civil and Infrastructure Engineering*, 32(5), pp.361-378. DOI: 10.1111/mice.12263.
- [9] Dung, C.V. (2019). Autonomous concrete crack detection using deep fully convolutional neural network. *Automation in Construction*, 99, pp.52-58. DOI: 10.1016/j.autcon.2018.11.028.
- [10] Liu, Z., Cao, Y., Wang, Y. and Wang, W. (2019). Computer vision-based concrete crack detection using U-net fully convolutional networks. *Automation in Construction*, 104, pp.129-139. DOI: 10.1016/j.autcon.2019.04.005.
- [11] Dung, C.V., Sekiya, H., Hirano, S., Okatani, T. and Miki, C. (2019). A vision-based method for crack detection in gusset plate welded joints of steel bridges using deep convolutional neural networks. *Automation in Construction*, 102, pp.217-229. DOI: 10.1016/j.autcon.2019.02.013.
- [12] Atha, D.J. and Jahanshahi, M.R. (2018). Evaluation of deep learning approaches based on convolutional neural networks for corrosion detection. *Structural Health Monitoring*, 17(5), pp.1110-1128. DOI: 10.1177/1475921717737051.
- [13] Gao, Y. and Mosalam, K.M. (2018). Deep transfer learning for image-based structural damage recognition. *Computer-Aided Civil and Infrastructure Engineering*, 33(9), pp.748-768. DOI: 10.1111/mice.12363.
- [14] Qu, X., Yang, J. and Chang, M. (2019). A deep learning model for concrete dam deformation prediction based on RS-LSTM. *Journal of Sensors*, 2019. DOI: 10.1155/2019/4581672.
- [15] Guo, A., Jiang, A., Lin, J. and Li, X. (2020). Data mining algorithms for bridge health monitoring: Kohonen clustering and LSTM prediction approaches. *The Journal of Supercomputing*, 76(2), pp.932-947. DOI: 10.1007/s11227-019-03045-8.
- [16] Zhang, J., Cao, X., Xie, J. and Kou, P. (2019). An improved long short-term memory model for dam displacement prediction. *Mathematical Problems in Engineering*, 2019. DOI: 10.1155/2019/6792189.
- [17] Zhang, Y., Burton, H.V., Sun, H. and Shokrabadi, M. (2018). A machine learning framework for assessing post-earthquake structural safety. *Structural safety*, 72, pp.1-16. DOI: 10.1016/j.strusafe.2017.12.001.
- [18] Truong, V.H., Vu, Q.V., Thai, H.T. and Ha, M.H. (2020). A robust method for safety evaluation of steel trusses using Gradient Tree Boosting algorithm. *Advances in Engineering Software*, 147, p.102825. DOI: 10.1016/j.advengsoft.2020.102825.
- [19] Kim, S.E., Vu, Q.V., Papazafeiropoulos, G., Kong, Z. and Truong, V.H. (2020). Comparison of machine learning algorithms for regression and classification of ultimate load-carrying capacity of steel frames. *Steel and Composite Structures*, 37(2), pp.193-209. DOI: 10.12989/scs.2020.37.2.193.
- [20] Freund, Y. and Schapire, R.E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55(1), pp.119-139. DOI: 10.1006/jcss.1997.1504.
- [21] Feng, D.C., Liu, Z.T., Wang, X.D., Jiang, Z.M. and Liang, S.X. (2020). Failure mode classification and bearing capacity prediction for reinforced concrete columns based on ensemble machine learning algorithm. *Advanced Engineering Informatics*, 45, p.101126. DOI: 10.1016/j.aei.2020.101126.
- [22] Feng, D.C., Liu, Z.T., Wang, X.D., Chen, Y., Chang, J.Q., Wei, D.F. and Jiang, Z.M. (2020). Machine learning-based compressive strength prediction for concrete: An adaptive boosting approach. *Construction and Building Materials*, 230, p.117000. DOI: 10.1016/j.conbuildmat.2019.117000.
- [23] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V. and Vanderplas, J. (2011). Scikit-learn: Machine learning in Python. *The Journal of Machine Learning Research*, 12, pp.2825-2830.