ce/papers Proceedings in civil engineering

Ernst & Sohn A Wiley Brand

ORIGINAL ARTICLE

eurosteel AMSTERDAM 2023

Ernst & Sohn A Wiley Brand

Application of machine learning approaches for modelling crack growth rates

Mohamed El Amine Ben Seghier [1] | Vagelis Plevris[2]

Correspondence

Dr. Mohamed El Amine Ben Seghier
Department of Civil Engineering and Energy Technology, OsloMet—Oslo Metropolitan University
0167 Oslo, Norway
Email: moseg7662@oslomet.no
[1] Department of Civil Engineering and Energy Technology, OsloMet—Oslo Metropolitan University, 0167 Oslo, Norway
[2] Department of Civil and Architectural Engineering, Qatar University, Doha, Qatar

Abstract

The scientific community has widely accepted the use of machine learning techniques to tackle complex engineering problems. Among the most intriguing problems is finding the correlation between alloy steel properties and cyclic fatigue and crack growth rate. Employing machine-learning models can provide more robust and accurate predictive models to address such challenges. This paper presents the application of four machine learning models, namely decision trees (DT), random forest (RF), adoptive boosting (AdaBoost), and gradient boosting regression tree (GBRT) to predict the crack growth rate of steel/alloys. The study utilizes a large database gathered from literature to construct the predictive models and compares the results using various statistical metrics and graphical representation. The study's findings demonstrate the effectiveness and suitability of machine learning techniques to handle complex databases related to fatigue problems.

Keywords

Crack growth rate, Machine learning, Gradient boosting regression tree, comparative metrics, root mean square error.

1 Introduction

Operating structures in hostile environments, such as those found in offshore or in the sea, are subjected to a variety of continuous loads that generate fatigue and are exposed to a variety of conditions that promote corrosion progression [1–3]. Because these structures are built of different steel and alloys, they are more prone to fatigue difficulties under such loadings. Typically, cracks will develop and propagate over time, causing a significant influence on the safety of such structures as well as negative environmental implications [4]. As a result, crack growth prediction has become a hot topic in both the scientific and industrial worlds. Predicting crack growth rates is critical for building safe and reliable structures, as well as determining the remaining useful life of existing structures[5]. Many factors, including material qualities, stress circumstances, and environmental influences, can influence the rate at which a crack grows.

Researchers have developed a variety of models and approaches for estimating crack growth rates in the last decades, including empirical models, numerical simulations, and analytical methods. Empirical models are typically used to predict crack growth rates in specific materials and loading conditions using experimental data [6,7]. The Paris-Erdogan equation [8], for example, is a frequently used empirical model that connects the rate of crack formation in steel to the stress intensity factor, material parameters, and loading circumstances, which is widely validated experimentally. Numerical simulations employing finite element analysis (FEA), on the other hand, are useful tools for forecasting crack propagation rates in materials such as steel and alloys[9,10]. FEA is breaking down a large structure into small pieces and modeling the behavior of each element using mathematical equations. This strategy has been widely used to address such a problem. For predicting crack growth rates, analytical approaches based on fracture mechanics are commonly employed [11]. The premise of fracture mechanics is that fractures in materials can be viewed as microscopic defects that can expand under particular loading conditions [12]. Fracture mechanics determine the crack growth rate and the structure's remaining usable life by examining the geometry of the crack and the loading circumstances. However, earlier approaches were discovered to be much more costly and time demanding when it came to running large experimental programs, being limited to a narrow range of utilization, or being dependent on basic fitting models. On the other side, advances in processing power and data storage allow for faster and less expensive data handling and analysis. As a result, advanced data interpretation methods, such as machine learning algorithms, can be used to address such difficult problems.

The literature indicates machine learning modeling's efficiency in tackling complicated nonlinear issues in a range

© 2023 The Authors. Published by Ernst & Sohn GmbH.

This is an open access article under the terms of the Creative Commons Attribution-NonCommercial License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited and is not used for commercial purposes.

of domains, particularly engineering [13–15]. Given the increasing availability of data, researchers are using machine learning to speed up the prediction of material properties by training computer algorithms on large amounts of data and then using them to build complex functions that describe the relationship between the input-output variables [16]. Raja et al.[17] used a backpropagation neural network (BPNN) and an extreme learning machine (ELM) to estimate the fatigue crack growth of Al 2014 alloy, concluding that the ELM performed better on the dataset. Dieu et al. [18] used deep learning models such as multi-layer neural networks and the long-short term memory approach to forecast fatigue crack propagation, and found the latter to be more effective. Wang et al. [19] examined the performance of three machine learning techniques in predicting the FCGR of Al2024-T351 alloy (ELM, radial basis function network (RBFN), and genetic algorithms optimized backpropagation network (GABP)).

Motivated by the foregoing, this study attempts to construct and compare machine learning models for more correctly forecasting crack growth rates. Decision trees (DT), random forest (RF), adoptive boosting (AdaBoost), and gradient boosting regression tree (GBRT) are the four machine learning algorithms used. Furthermore, a database of 163 datasets is used to train and test the models, and various comparison criteria are employed to evaluate the machine learning models' performance.

2 Machine learning approaches

2.1 Decision Trees (DT)

Decision trees are a prominent machine learning technique that is widely utilized in many domains, including material science and engineering, where this basic algorithm is a sort of supervised learning algorithm that is used for both classification and regression issues [20,21]. Decision trees operate by recursively partitioning the data into smaller subsets based on the values of various input variables, with the goal of creating a model that can reliably predict the output variable based on the input variables. Creating a decision tree entails picking the most significant variables and breaking the data into subsets depending on their values, which continues until a stopping requirement, such as achieving a maximum tree depth or a minimum amount of data points in a subset, is satisfied.

Decision trees have various advantages, including the simplicity of learning and analyzing, and handling both numerical and categorical data. Decision trees can also handle missing values and outliers. However, decision trees can be overfitted, which occurs when the model gets too complicated and performs well on training data but badly on test data. Cross validation, ensemble approaches, and regularization can all be used to overcome this issue. Overall, decision trees are a powerful and versatile machine learning technique that is a strong option for resolving the current problem in this study.

2.2 Random Forest (RF)

Random forest is a strong machine learning algorithm that belongs to the ensemble learning class[22]. It includes numerous decision trees, each of which is trained on a random subset of the input features and data samples. Randomness is applied at two levels using random forest: during the building of the decision trees and during the selection of the input features[23].

To put it another way, during the creation of each decision tree, a random subset of the data samples is chosen, and the tree is built by selecting the optimal split point among a random subset of the input features. This process is repeated for each tree in the forest, yielding a collection of decision trees that are completely independent of one another. To generate a prediction for a new data point, each tree in the forest guesses the outcome independently, and the final outcome prediction is made by collecting the majority vote of all the trees. Thus, by minimizing the correlation between the individual trees, this strategy efficiently minimizes the variance of the model and aids in the prevention of overfitting.

Random forest outperforms other machine learning algorithms in several ways, including its capacity to handle high-dimensional data, noisy data, and missing data. It is also computationally efficient and can be trained with great accuracy on huge datasets. Because of these characteristics, it has become a popular choice for a wide variety of applications, including complicated classification, regression, and feature selection problems in engineering. To summarize, random forest is a powerful and versatile machine learning method capable of resolving the current problem in our study.

2.3 Adaptive boosting (AdaBoost)

Adaptive boosting (AdaBoost) is a prominent ensemble learning method for increasing the performance of weak classifiers by combining numerous weak classifiers, such as decision trees, to build a strong classifier[24]. Thus, when weak classifiers are merged in an ensemble, their performance improves dramatically. The key idea behind AdaBoost is to assign weights to each training instance and focus on those instances that are difficult to correctly classify. AdaBoost then iteratively trains a sequence of weak classifiers on the training data, assigning higher weights to misclassified instances and lower weights to correctly classified instances at each iteration. The misclassified cases are then given a higher probability of being chosen for training the next classifier in the sequence, with the final classifier being formed by integrating the weighted predictions of all the weak classifiers in the series[25].

This method has been used successfully in a range of engineering applications, including object identification and difficult prediction issues [26]. However, one of AdaBoost's weaknesses is its sensitivity to noisy data, which can lead to overfitting. The cross validation technique can be used to address this issue. As a result, AdaBoost may be a viable solution to the current challenge in our study.

2.4 Gradient Boosting Regression Tree (GBRT)

Gradient Boosting Regression Tree (GBRT) is a powerful and commonly used machine-learning algorithm for classification and regression problems, particularly in solving engineering problems [27]. GBRT is an ensemble method that combines multiple weak models to create a strong predictive model. It works by iteratively adding decision

25097075, 2023, 3-4, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/cepa.2639 by Cochrane Qatar, Wiley Online Library on [06/02/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

trees to the ensemble, with each new tree learning to predict the residuals (i.e., the differences between the actual and predicted values) of the previous trees. This process is repeated until the ensemble reaches a desired degree of accuracy or a halting threshold is reached[28].

The capacity of GBRT to represent complex nonlinear interactions between predictor factors and response variables is one of its fundamental strengths. The technique is capable of handling both continuous and categorical predictor variables, as well as missing data. Another advantage of GBRT is that it does feature selection automatically and can handle high-dimensional data. This makes it especially effective in applications with a large number of predictor variables. On the other hand, GBRT is computationally demanding and, if not correctly calibrated, is susceptible to overfitting. To avoid overfitting and obtain good performance, the model's hyperparameters, such as the learning rate, the maximum depth of the trees, and the number of trees in the ensemble, must be carefully chosen. To address these concerns, the cross validation technique might be used. As a result, this technique has the potential to be a powerful solution to the problem outlined in our study.

3 Database description and analysis

To predict crack growth rate in steel and alloys using machine-learning techniques, it is essential to have access to comprehensive and reliable databases. For this purpose, in this research, we selected a database consisting of 163 samples of Alloy 600, which had been previously published by Shi et al.[29], as a case study to implement our proposed framework. It should be noted that this database is related to corrosion fatigue problem. The statistical characteristics of the variables for this case study are presented in Table 1, which include the minimum and maximum values, mean, and standard deviation of each variable. Eight selected input variables from these characteristics are used to train their machine learning techniques. These variables are chosen based on their influence on the crack growth rate, which are the target output variable (CGR in cm/s). The eight input variables included temperature (T in °C), stress intensity (SI in $MPa.m^{0.5}$), pH, conductivity (HTC in µS/cm), electrochemical potential (ECP in V_{she}), yield strength (YS in MPa), B(OH)3 concentration (in ppm), and LiOH concentration (in ppm).

Figure 1 presents the Pearson correlation matrix for all variables in the database. This matrix is a useful metric to assess the strength of relationships between variables. As shown in the figure, most of the variables exhibit only moderate or weak correlations, particularly with the target variable, CGR. For example, the highest correlation coefficient between CGR and any input variable was found to be 0.62 for Stress Intensity Factor (SI) and Conductivity (HTC), which is considered moderate. This suggests that the selected variables are appropriate inputs for the machine learning techniques used in this study, but also highlights the complexity of the problem.

Table 1 Statistical description of the CGR database used in the development of machine learning models

Variables	Min	Max	mean	SD
Temperature, T (°C)	289	360	322.45	16.83
Stress intensity, SI ($MPa.m^{0.5}$)	4.6	101	38.16	19.95
pH	5.52	9.19	7.33	0.78
Conductivity, HTC (µS/cm)	0.97	1116	251.42	222.35
Electrochemical potential	-1.1	-0.56	-0.82	0.10
Yield strength, YS (MPa)	211	500	359.67	70.56
B(OH)3 ,(ppm)	0	1800	707.97	451.48
LiOH, Li (ppm)	0	10	2.20	1.95
Crack growth rate, CGR (cm/s)	4.88E-10	6.28E-07	4.27E-08	8.7E-08

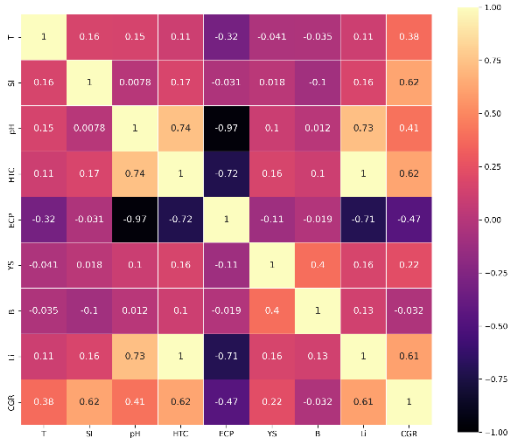


Figure 1 Correlation matrix between all the utilized variables.

4 Implementation

The implementation approach must be properly stated in order to construct machine-learning models for predicting the crack growth rate (CGR). The major processes are as follows: obtaining the entire CGR database, separating it into two sets (i.e. training and testing datasets), performing the modeling process using the machine learning techniques using the training datasets and validate it using testing datasets, and finally evaluating performance. First, the collected data must be pre-processed to detect outliers and missing information, in which in our study there were no missing information or detected outliers. Then, the database is randomly divided into training and testing datasets in an 80:20 ratio. Since the input factors have different scales, the data must be normalized in the 0-1 range to eliminate the scale effect. The normalized training datasets (i.e. 80% of the total database) are used for developing the four proposed machine learning models, while the testing datasets are used for validation (i.e. 20% of the total database). To avoid overfitting difficulties, the

25097075, 2023, 3-4, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/cepa.2639 by Cochrane Qatar, Wiley Online Library on [06/02/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

k-cross validation technique is utilized during the training phase, and the number of estimators used is set to 200 for all machine learning models. After the development of the predictive models, the test datasets are used to estimate CGR output, which is then evaluated using performance metrics such as root mean square error (RMSE), mean absolute error (MAE), Nash-Sutcliffe Efficiency (NSE), coefficient of determination (R^2), and uncertainty at 95% (U95). The best performing machine learning model is then selected based on the evaluation results from both training and testing phases. The formula for the evaluation metrics is represented by the expressions below [30,31].

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(CGR_i^{exp} - CGR_i^{pre}\right)^2} \quad (1)$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|CGR_i^{exp} - CGR_i^{pre}\right| \quad (2)$$

$$NSE = 1 - \frac{\sum_{i=1}^{n}\left(CGR_i^{exp} - CGR_i^{pre}\right)^2}{\sum_{i=1}^{n}\left(CGR_i^{exp} - CGR^{avg}\right)^2} \qquad -\infty \le NSE \le 1 \quad (3)$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}\left(CGR_i^{exp} - CGR_i^{pre}\right)^2}{\sum_{i=1}^{n}\left(C_{rate}^{avg} - CGR_i^{pre}\right)^2} \quad (4)$$

$$U95 = 1.96\sqrt{SD^2 + RMSE^2} \quad (5)$$

where CGR_i^{exp}, CGR_i^{pre} and CGR^{avg} denote the *i-th* real experimental value, the predicted result and the average value, respectively. n represent the data size, while SD is the standard deviation.

5 Results and discussions

Table 2 presents the performance evaluation results of the machine learning models during the training and testing phases using the statistical metrics. The best performing machine learning model is defined as having the lowest RMSE and MAE values and the highest NSE value. In general, all proposed models, except DT, provided acceptable predictive results for CGR. According to the statistical measures, the GBRT model had the highest accuracy throughout both the training (RMSE = 6.64E-09 and MAE = 5.36E-09) and testing (RMSE = 2.11E-08 and MAE = 1.39E-08) phases. On the other hand, the DT model had the lowest performance with an overall RMSE value of 4.0483E-08 and MAE value of 2.5523E-08. Although AdaBoost showed relatively higher performance during the training phase compared to RF, the latter yielded better results than AdaBoost during the testing phase. Overall, the GBRT model yielded the highest NSE result of 0.97, followed by AdaBoost with 0.933, RF with 0.89, and finally DT with 0.76.

Table 2 Performance evaluation of the machine learning models

	Metrics	DT	RF	AdaBoost	GBRT
Training phase	RMSE (cm/s)	4.19E-08	2.86E-08	1.48E-08	6.64E-09
	MAE (cm/s)	2.55E-08	1.33E-08	1.05E-08	5.36E-09
	NSE	0.7925	0.9031	0.9742	0.9948
Testing phase	RMSE (cm/s)	3.48E-08	2.30E-08	2.89E-08	2.11E-08
	MAE (cm/s)	2.54E-08	1.44E-08	1.53E-08	1.39E-08
	NSE	0.6686	0.8554	0.7715	0.8784

Scatter plots can be used to visualize the accuracy of the CGR predictions during both the training and testing phases. Each machine-learning model, namely, DT, RF, AdaBoost, and GBRT has a scatter plot, which is shown in Figures 2-5 in same respect. The determination coefficient (R^2) and the linear fitting line (dashed red-line) are included in each figure. The scatter plots reveal that the GBRT model has the highest degree of agreement between the predicted and actual values of CGR data. The GBRT model yielded an R^2 value of 0.98 during the training phase and 0.95 during the testing phase. The overall R^2 results (80%+20%) for the AdaBoost (R^2 = 0.9087) and RF (R^2 = 0.9056) models are similar, with a relative difference of only 0.34%. However, the DT-based regression model had the lowest R^2 values during both the training (i.e. 0.81) and testing (i.e. 0.69) phases.

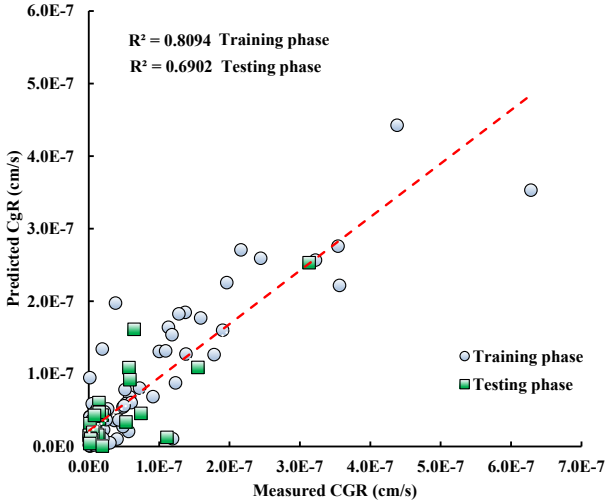


Figure 2 Scatter plot of the actual values versus the predicted results of the CGR using decision trees (DT) model

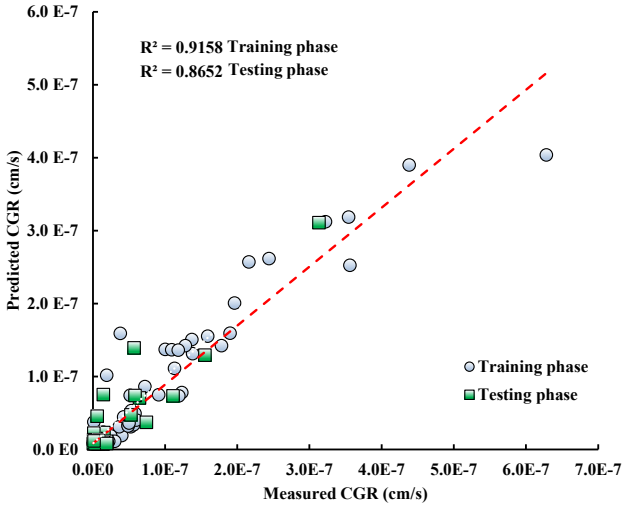


Figure 3 Scatter plot of the actual values versus the predicted results of the CGR using random forest (RF) model

25097075, 2023, 3-4, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/cepa.2639 by Cochrane Qatar, Wiley Online Library on [06/02/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

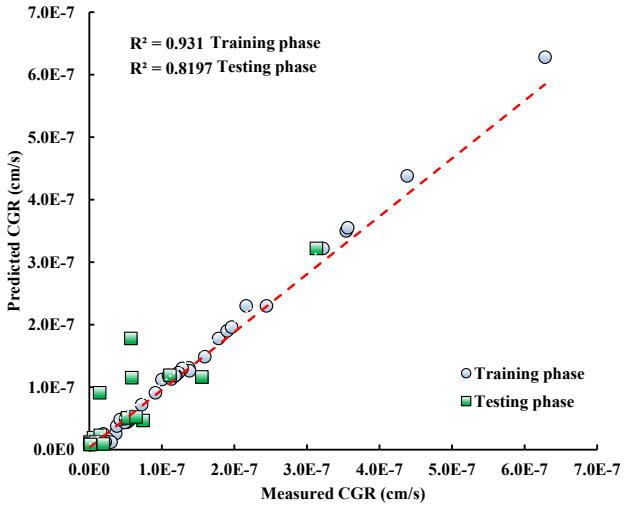


Figure 4 Scatter plots of the actual values versus the predicted results of the CGR using Adaptive boosting (AdaBoost) model

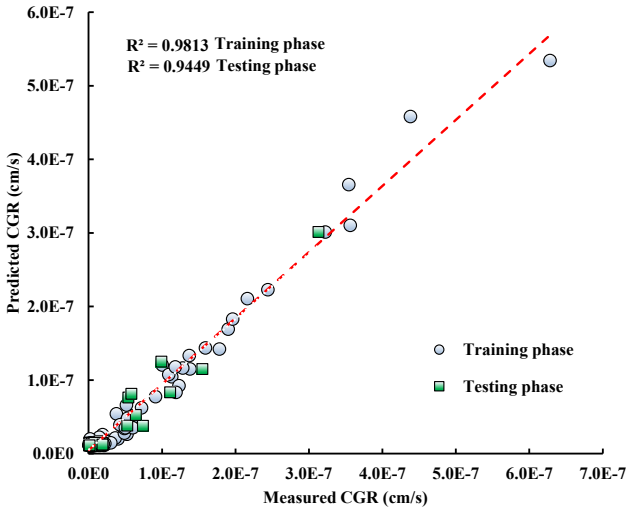
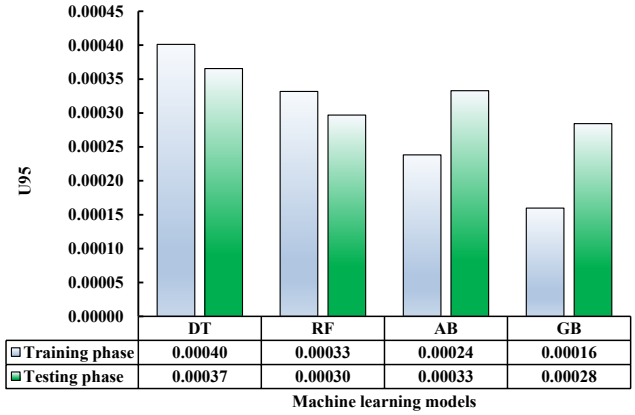


Figure 5 Scatter plots of the actual values versus the predicted results of the CGR using Gradient Boosting Regression Tree (GBRT) model

U95 is considered a global performance metric as it provides an estimate of the model's performance based on multiple metrics simultaneously, such as SD and RMSE, resulting in more comprehensive and reliable results. The CGR was predicted using the four machine learning models mentioned earlier, and the results were compared to the actual measured data based on individual performance metrics. The modeling accuracy and efficiency were demonstrated using the single ML-model; however, using global performance metrics allowed for conclusions to be drawn regarding the best and least efficient machine learning models. Figure 6 illustrates the comparative results of the global performance based on the U95 metric using the training and testing datasets. It is worth noting that the machine-learning model with the lowest U95 value is considered the best performing model. Notably, the GBRT model provides a robust and accurate performance, with U95 values of 0.00016 during the training phase and 0.00028 during the testing phase. By calculating the relative error between the U95 values of the machine learning models, it can be seen that the GBRT model improves the prediction results compared to the AdaBoost, RF, and DT models by 28.20%, 43.14%, and 53.14%, respectively.



	DT	RF	AB	GB
Training phase	0.00040	0.00033	0.00024	0.00016
Testing phase	0.00037	0.00030	0.00033	0.00028

Figure 6 Performance evaluation based on global metric (i.e. U95)

6 Conclusions

This paper describes how machine-learning algorithms were used to address the critical challenge of precisely determining the crack growth rate (CGR) in steel and alloys. Based on a large database of 163 measurements, four machine learning methods were utilized to build prediction models for crack growth rate (CGR): decision trees (DT), random forest (RF), adoptive boosting (AdaBoost), and gradient boosting regression tree (GBRT). Each model's performance was assessed using both single and global performance indicators. According to the results, all proposed machine learning models performed satisfactory, with GBRT providing the highest accuracy and efficiency based on both single (e.g., R^2 = 0.98, RMSE = 6.64E-09) and global (e.g., U95 = 0.00018) measures. The GBRT model also showed significant improvement in mathematical prediction accuracy compared to the AdaBoost, RF, and DT models, by 28.20%, 43.14%, and 53.14%, respectively.

Overall, machine-learning techniques provide practical and useful solutions for dealing with fatigue issues like crack growth rate (CGR). Further development of more detailed models, as well as database expansion to incorporate datasets with different features such as the design specimen geometries and crack dimension (i.e. specimen thickness, type of crack, crack size) and the stress ratios, will aid in the comprehension and use of the suggested machine-learning model for evaluating other essential criteria

References

[1] H.M.H. Farh, M.E.A. Ben Seghier, T. Zayed, A comprehensive review of corrosion protection and control techniques for metallic pipelines, Eng. Fail. Anal. (2022) 106885.

[2] X. Fang, H. Wang, W. Li, G. Liu, B. Cai, Fatigue crack growth prediction method for offshore platform based on digital twin, Ocean Eng. 244 (2022) 110320.

25097075, 2023, 3-4, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/cepa.2639 by Cochrane Qatar, Wiley Online Library on [06/02/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

[3] N. Yazdani, P. Albrecht, Crack growth rates of structural steel in air and aqueous environments, Eng. Fract. Mech. 32 (1989) 997–1007.

[4] M.E.A. Ben Seghier, Z. Mustaffa, T. Zayed, Reliability assessment of subsea pipelines under the effect of spanning load and corrosion degradation, J. Nat. Gas Sci. Eng. (2022) 104569.

[5] R. Taiwo, M. El Amine Ben Seghier, T. Zayed, Towards sustainable water infrastructure: The state-of-the-art for modeling the failure probability of water pipes, Water Resour. Res. (2023) e2022WR033256.

[6] J.D.M. Costa, J.A.M. Ferreira, Effect of stress ratio and specimen thickness on fatigue crack growth of CK45 steel, Theor. Appl. Fract. Mech. 30 (1998) 65–73.

[7] R.L. Amaro, N. Rustagi, K.O. Findley, E.S. Drexler, A.J. Slifka, Modeling the fatigue crack growth of X100 pipeline steel in gaseous hydrogen, Int. J. Fatigue. 59 (2014) 262–271.

[8] D. Kujawski, F. Ellyin, A fatigue crack propagation model, Eng. Fract. Mech. 20 (1984) 695–704.

[9] J.C. Newman, Finite-element analysis of crack growth under monotonic and cyclic loading, ASTM International, 1977.

[10] M. Stolarska, D.L. Chopp, N. Moës, T. Belytschko, Modelling crack growth by level sets in the extended finite element method, Int. J. Numer. Methods Eng. 51 (2001) 943–960.

[11] N.E. Frost, L.P. Pook, K. Denton, A fracture mechanics analysis of fatigue crack growth data for various materials, Eng. Fract. Mech. 3 (1971) 109–126.

[12] J.D. Landes, J.A. Begley, A fracture mechanics approach to creep crack growth, ASTM STP. 590 (1976) 128–148.

[13] M.E.A. Ben Seghier, H. Carvalho, B. Keshtegar, J.A.F.O. Correia, F. Berto, Novel hybridized adaptive neuro-fuzzy inference system models based particle swarm optimization and genetic algorithms for accurate prediction of stress intensity factor, Fatigue Fract. Eng. Mater. Struct. 43 (2020) 2653–2667.

[14] M.E.A. Ben Seghier, V. Plevris, G. Solorzano, Random forest-based algorithms for accurate evaluation of ultimate bending capacity of steel tubes, in: Structures, Elsevier, 2022: pp. 261–273.

[15] M.E.A. Ben Seghier, B. Keshtegar, M. Taleb-Berrouane, R. Abbassi, N.-T. Trung, Advanced intelligence frameworks for predicting maximum pitting corrosion depth in oil and gas pipelines, Process Saf. Environ. Prot. (n.d.).

[16] M.E.A. Ben Seghier, J.A.F.O. Corriea, J. Jafari-Asl, A. Malekjafarian, V. Plevris, N.-T. Trung, On the modeling of the annual corrosion rate in main cables of suspension bridges using combined soft computing model and a novel nature-inspired algorithm, Neural Comput. Appl. (2021) 1–17.

[17] A. Raja, S.T. Chukka, R. Jayaganthan, Prediction of fatigue crack growth behaviour in ultrafine grained al 2014 alloy using machine learning, Metals (Basel). 10 (2020) 1349.

[18] D.T.T. Do, J. Lee, H. Nguyen-Xuan, Fast evaluation of crack growth path using time series forecasting, Eng. Fract. Mech. 218 (2019) 106567.

[19] B. Wang, W. Zhao, Y. Du, G. Zhang, Y. Yang, Prediction of fatigue stress concentration factor using extreme learning machine, Comput. Mater. Sci. 125 (2016) 136–145.

[20] L. Rokach, O. Maimon, Decision trees, Data Min. Knowl. Discov. Handb. (2005) 165–192.

[21] S. Rovnyak, S. Kretsinger, J. Thorp, D. Brown, Decision trees for real-time transient stability prediction, IEEE Trans. Power Syst. 9 (1994) 1417–1426.

[22] M.R. Segal, Machine learning benchmarks and random forest regression, (2004).

[23] Y. Liu, Y. Wang, J. Zhang, New machine learning algorithm: Random forest, in: Inf. Comput. Appl. Third Int. Conf. ICICA 2012, Chengde, China, Sept. 14-16, 2012. Proc. 3, Springer, 2012: pp. 246–252.

[24] R. Zemel, T. Pitassi, A gradient-based boosting algorithm for regression problems, Adv. Neural Inf. Process. Syst. 13 (2000).

[25] D.D. Margineantu, T.G. Dietterich, Pruning adaptive boosting, in: ICML, Citeseer, 1997: pp. 211–218.

[26] M.E.A. Ben Seghier, D. Höche, M. Zheludkevich, Prediction of the internal corrosion rate for oil and gas pipeline: Implementation of ensemble learning techniques, J. Nat. Gas Sci. Eng. (2022) 104425.

[27] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, T.-Y. Liu, Lightgbm: A highly efficient gradient boosting decision tree, Adv. Neural Inf. Process. Syst. 30 (2017).

[28] A. Anghel, N. Papandreou, T. Parnell, A. De Palma, H. Pozidis, Benchmarking and optimization of gradient boosting decision tree algorithms, arXiv Prepr. arXiv1809.04559. (2018).

[29] J. Shi, J. Wang, D.D. Macdonald, Prediction of primary water stress corrosion crack growth rates in Alloy 600 using artificial neural networks, Corros. Sci. 92 (2015) 217–227.

[30] M. El, A. Ben, B. Keshtegar, K. Fah, T. Zayed, R. Abbassi, N. Thoi, Prediction of maximum pitting corrosion depth in oil and gas pipelines, Eng. Fail. Anal. 112 (2020) 104505. doi:10.1016/j.engfailanal.2020.104505.

[31] M.E.A. Ben Seghier, X.-Z. Gao, J. Jafari-Asl, D.-K. Thai, S. Ohadi, N.-T. Trung, Modeling the nonlinear behavior of ACC for SCFST columns using experimental-data and a novel evolutionary-algorithm, in: Structures, Elsevier, 2021: pp. 692–709.

25097075, 2023, 3-4, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/cepa.2639 by Cochrane Qatar, Wiley Online Library on [06/02/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License