Architecture
Image Studies

Vol.6, Issue 4, pp.702-710, 2025
DOI: https://doi.org/10.62754/ais.v6i4.668

https://journals.ap2.pt/index.php/ais/index

Artificial Intelligence Framework for Concrete Compressive Strength Prediction

Hung K. Nguyen[1], Tu T. Nguyen[2]

Abstract

This study develops and validates a robust Artificial Intelligence (AI) framework for predicting concrete compressive strength. A hybrid dataset of 1,274 samples was established by combining 244 locally tested specimens with 1,030 data points from a previously published source. Eight input parameters, including cement, blast furnace slag, fly ash, water, superplasticizer, coarse aggregate, fine aggregate, and curing age, were used. Python libraries were employed with two key preprocessing steps: a logarithmic transformation of concrete age to address the non-linear strength gain behavior and MinMax scaling to normalize input variables. The performance of the ANN was compared with Random Forest (RF) and Extreme Gradient Boosting (XGBoost) models. All machine learning models demonstrated strong predictive capability, with the XGBoost achieving the best performance, yielding a Mean Absolute Error (MAE) of 0.106 MPa and a Coefficient of Determination (R^2) of 0.999 on the independent test dataset. The proposed framework offers a highly accurate and interpretable tool for practical applications in quality assurance, concrete mix optimization, and data-driven decision-making within the construction industry.

Keywords: *Artificial Intelligence; Machine Learning; Soft Computing; Concrete Compressive Strength; Non-linear Prediction.*

Introduction

Concrete remains the most widely used construction material worldwide due to its excellent mechanical performance, durability, and adaptability across diverse structural applications [1, 2]. Among its various properties, compressive strength is the most critical indicator of quality and performance, directly reflecting a structure's load-bearing capacity, durability, and overall safety. Moreover, other key mechanical parameters, such as Young's modulus and tensile strength, are often derived from or correlated with compressive strength. The strength of concrete is influenced by numerous factors, including the water-to-cement ratio, cement type, admixtures, aggregate characteristics, curing conditions, mix proportions, and testing methods [3]. Given these complex interactions, the ability to accurately predict compressive strength based on mix composition and curing parameters offers significant advantages for both design optimization and quality control in concrete production.

Conventional evaluation of concrete compressive strength requires casting and curing cylindrical or cubic specimens, followed by destructive testing at standard ages, typically 3, 7, and 28 days. While this approach is reliable and widely accepted, it is destructive, time-consuming, and resource-intensive, often delaying construction progress as teams wait for results [4, 5]. Because the relationships among concrete's constituent materials and its final strength are highly non-linear and interdependent, traditional statistical techniques, such as multiple linear regression, struggle to model these interactions effectively. As a result, their predictive accuracy is often limited when estimating strength from mix composition and curing conditions [6, 7].

Recent advances in soft computing techniques, including ANN, Random Forest, and XGBoost, have shown strong potential for predicting concrete properties and addressing complex engineering problems [8-15]. These methods effectively capture the non-linear, multi-dimensional relationships among material parameters that traditional statistical models often fail to represent. For instance, Pham

[1] Civil Engineering Department, Lac Hong University, Dong Nai, Vietnam
[2] Civil Engineering Department, Lac Hong University, Dong Nai, Vietnam, Email: tunt@lhu.edu.vn, (Corresponding Author)

et al. [9] developed an ANN model using 190 geopolymer concrete samples and achieved highly accurate strength predictions with minimal error. Furthermore, another study comparing ANN and Adaptive Neuro-Fuzzy Inference System (*ANFIS*), for Fiber-Reinforced High-Strength Self-Compacting Concrete, found that ANN provided better predictive performance than ANFIS [11].

Beyond the field of concrete materials, AI-based models have been widely utilized to address a diverse range of engineering challenges [16–24]. Nguyen and Dinh [16] successfully employed ANN to predict bridge deck condition ratings using 2,572 records from the National Bridge Inventory, achieving an impressive accuracy of 98.5%. Likewise, Guijo-Rubio et al. [17] used an ANN to estimate solar radiation from satellite data, achieving high accuracy and outperforming models like Support Vector Regression. AI-based models have also been used to predict the fire resistance rating of timber structures [18], detect structural damage [19], and identify polymeric materials [20]. By learning from large datasets and uncovering hidden variable relationships, an AI-based model provides accurate, reliable predictive tools that reduce reliance on costly and time-consuming experimental testing.

Despite the success of the AI approach, challenges remain in modeling early-age non-linear strength gain and ensuring model generalizability to local materials. This study aims to address these issues by creating a hybrid dataset that merges 244 local experimental samples with Yeh's 1,030-sample international benchmark [25], improving both relevance and robustness. A logarithmic transformation of curing age is introduced to better capture concrete's maturation behavior. The study also compares a predictive ANN with other AI models to obtain the best advanced non-linear approaches for predicting concrete compressive strength.

Materials and Methods

A concise yet comprehensive overview of the methodology adopted in this study was presented, including the rationale for selecting the AI-based techniques and the overall workflow used to develop the predictive models. In addition, the procedures used for experimental data collection were briefly described in the subsequent sections.

Methodology

The research methodology for this study was systematically designed and executed in three distinct phases, as illustrated in the flowchart in Figure 1. The first phase (i) includes data collection of a 1,274-sample hybrid dataset from 244 local experimental samples and the 1,030-sample international benchmark dataset from Yeh [25]. The phase concluded with splitting the preprocessed data into an 80% training set and a 20% testing set. The second phase (ii), model development and training, involved the construction and training of the predictive ANN models. In parallel, two other powerful machine learning models, RF and XGBoost, were also developed for comparative analysis. The last phase (iii) covers performance evaluation and interpretation. The final phase was dedicated to rigorously evaluating the trained models and interpreting their results.

The performance of the ANN, RF, and XGBoost models was assessed on the independent test set using standard metrics (MAE, RMSE, and R^2). A detailed analysis of the ANN model's training process and prediction accuracy was conducted. The methodology was completed in a final validation step, where the trained ANN model was tested against a completely new set of 20 independent experimental samples to verify its real-world generalization capability.

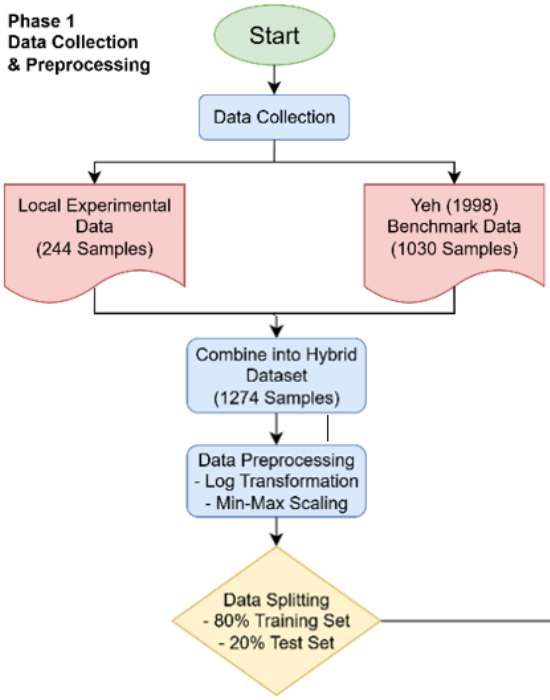

Figure 1. Research Methodology Flowchart.

Experimental Program to Collect Data

A key component of this study is the integration of local experimental data to ensure the model's relevance to regional materials and practices. A total of 244 concrete samples were prepared and tested in a laboratory under local climatic conditions. The concrete mixes were prepared using locally sourced materials. This included regular local Portland cement, natural river sand as fine aggregate, and crushed stone as coarse aggregate. A polycarboxylate-based superplasticizer was used to improve workability. The mix designs corresponded to common local concrete grades used, specifically B15 and B20, following TCVN 3118:2022 [26]. For each mix design, three cubic specimens of 150×150×150 mm were cast. Specimens were cured in a standard water tank at 20 ± 2°C until the testing date.

Compressive strength tests were conducted at multiple curing ages: 3, 7, 14, 28, and 60 days. The compression tests conformed to the requirements of TCVN 3118:2022 [26] and were conducted at the Laboratory of Structural Engineering, Lac Hong University, using the Phoenix Compression Testing Machine. The maximum compression capacity of the testing equipment is 2000 kN. The compression tests were implemented with a constant loading speed of 70kN/10s until the test specimen failed. The maximum force for each specimen was documented. The final strength for each data point was recorded as the average of three specimen tests.

2.3. Dataset Curation and Description

Table 1 presents the statistical summary of the variables used in this study for predicting concrete compressive strength. The 1,274 samples dataset comprises eight input parameters, including cement, blast furnace slag, fly ash, water, superplasticizer, coarse aggregate, fine aggregate, and curing age, and one output variable, compressive strength.

Table 1. Descriptive Statistics of the Combined Dataset (N = 1,274)

Variable	Unit	Minimum	Maximum	Mean	Std. Deviation
Cement	kg/m³	198.6	540	370.4	110.6
Blast Furnace Slag	kg/m³	0.0	142.5	82.7	57.3
Fly Ash	kg/m³	0.0	0.0	0.00	0.00

Water	kg/m³	160	228	204.2	29.9
Superplasticizer	kg/m³	0.0	2.5	0.74	1.14
Coarse Aggregate	kg/m³	801	1145	974.4	77.8
Fine Aggregate	kg/m³	594	992.6	773.6	80.2
Age	days	3	365	45.7	63.2
Compressive Strength	MPa	2.33	82.6	35.8	16.7

The cement content ranges from 198.6 to 540 kg/m³, with an average of 370.4 kg/m³, reflecting diverse mix proportions. Blast furnace slag content varies between 0 and 142.5 kg/m³ (mean = 82.7 kg/m³), indicating partial cement replacement in certain mixtures. Water content ranges from 160 to 228 kg/m³ (mean = 204.2 kg/m³), and superplasticizer dosage varies from 0 to 2.5 kg/m³ (mean = 0.74 kg/m³). The coarse and fine aggregates show mean values of 974.4 and 773.6 kg/m³, respectively, within typical limits for structural concrete. The curing age spans from 3 to 365 days (mean = 45.7 days), including both early and long-term strength development stages. The compressive strength ranges from 2.33 to 82.6 MPa, with an average of 35.8 MPa and a standard deviation of 16.7 MPa, indicating a wide range of concrete quality.

Data Preprocessing

To enhance model stability and predictive accuracy, two key preprocessing steps were applied using Python's Scikit-learn library [27]. First, a logarithmic transformation was performed on the age variable to address the highly non-linear relationship between curing time and compressive strength. The transformation, defined as *Log_Age = log(1 + Age)*, effectively linearizes the early-age strength development, allowing the model to better capture the temporal strength gain behavior. Second, Min-Max scaling was applied to normalize all input features within a uniform range, ensuring that variables with larger numerical magnitudes did not dominate the learning process. This normalization step is essential for achieving stable and efficient convergence, particularly in ANN training. Following preprocessing, the dataset was partitioned into two subsets: 80% for model training and 20% reserved as an independent test set for performance evaluation.

Model Development

In the subsequent sections, the structure of the proposed predictive models will be introduced. The proposed models aimed at establishing the non-linear relationship between various inputs to the compressive concrete strength. The performance evaluation criteria applied to assess these models were also clearly outlined to ensure transparency and reproducibility. The proposed model was developed in Python using Keras, NumPy, and Pandas libraries [28, 29].

ANN Architecture and Hyperparameter Optimization

The design of an effective ANN requires careful selection of its architecture and hyperparameters using the Keras libraries. The entire workflow, from data handling with NumPy and Pandas to model building, was implemented in a Python environment to ensure reproducibility and transparency. The final ANN architecture was determined through an iterative process of experimentation to balance model complexity with performance on the validation set. The goal of this hyperparameter tuning was to find a configuration that could effectively capture the non-linear relationships in the data without overfitting [30]. The optimized architecture is detailed in Figure 2 and Table 2.

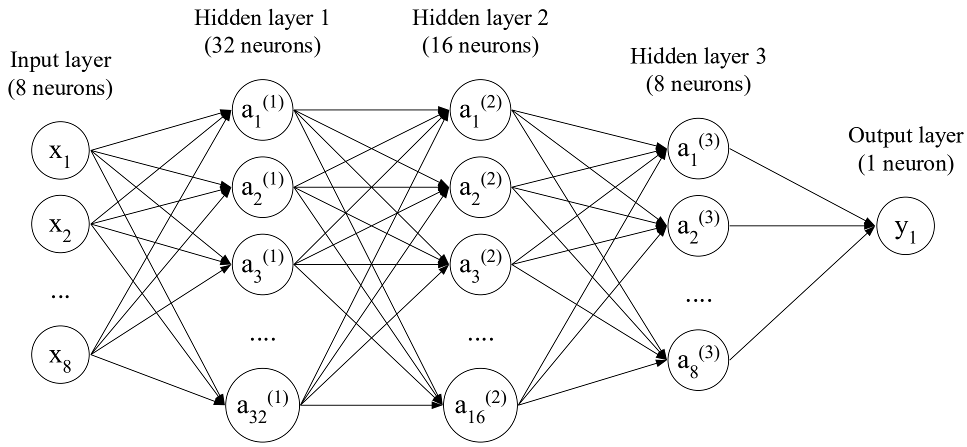

Figure 2. Architecture of the Proposed ANN Model

The network consists of an input layer with 8 neurons (one for each input feature), three sequential hidden layers, and a single output neuron. This deep structure allows the model to learn hierarchical features from the data. The hidden layers have a tapering structure (32, 16, and 8 neurons, respectively), which helps the network refine features from general patterns in the first layer to more specific ones in the subsequent layers before the final prediction.

Table 2. Optimized Hyperparameters for the ANN Model

Hyperparameter	Selected Value	Justification
Environment	Python	High flexibility and extensive libraries for deep learning [28].
No. Hidden Layers	3	Provides sufficient depth to model complex non-linearities [31].
Neurons in hidden layer	32 - 16 – 8	A tapering structure that refines features from general to specific.
Activation Function	ReLU	Prevents vanishing gradients and is computationally efficient.
Output Activation	Linear	Standard for regression tasks to predict continuous values.
Optimizer	Adam	Efficient adaptive learning rate algorithm [32].
Loss Function	MSE	Standard loss function for regression problems.
Number of Epochs	300	Determined by observing convergence on the learning curve.
Batch Size	16	Batch size for regular weight updates and stable convergence.

Ensemble Tree Models

For comparative purposes, two leading ensemble tree models were also evaluated: (i) the RF model, an ensemble method that builds multiple decision trees on different sub-samples of the data and averages their predictions to reduce variance and prevent overfitting [33]. And (ii) XGBoost model**,** a highly efficient implementation of gradient boosting that builds trees sequentially, with each new tree correcting the errors of the previous ones. It is known for its high performance on structured/tabular data [34].

Model Performance Evaluation

The performance of each model was assessed by calculating performance matrices including R^2, MSE, and RMSE. The R^2 measures the correlation between input and output parameters using Eq. 1.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \overline{y})^2} \quad \text{(Eq. 1)}$$

where y_i is the i th actual output, $\overline{y}$ is the mean of the actual outputs, $\hat{y_i}$ is the i th predicted output, and n is the total number of data samples.

MSE is the average squared difference between predicted outputs and actual outputs. MSE can be computed using Eq. 2.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \quad \text{(Eq. 2)}$$

And RMSE can be calculated by Eq. 3.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \quad \text{(Eq. 3)}$$

Results and Discussion

As noted earlier, the proposed AI-based models, configured with seven input variables and one output parameter, were developed to predict the compressive strength of concrete. Their predictive capability was thoroughly evaluated across the training, validation, and testing datasets using the coefficient of determination and the mean squared error. In addition, a supplementary experimental dataset consisting of 20 records was utilized to further assess the models' performance under real-world conditions.

Analysis of the ANN Model

Figure 3 shows the training and validation MAE of a model over 300 epochs. Both the training MAE (blue line) and validation MAE (orange line) decrease sharply during the first few epochs, indicating rapid initial learning. After the initial drop, the MAE gradually declines and stabilizes as the epochs progress, with both curves closely tracking each other, suggesting minimal overfitting. By the final epochs, the MAE reaches a very low and steady value, indicating that the model has converged and achieved high accuracy on both the training and validation datasets.

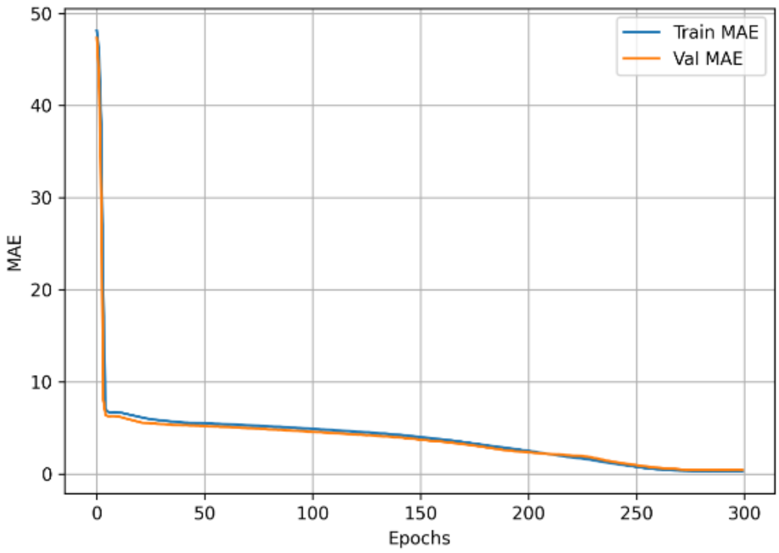

Figure 3. Learning Curve of the ANN Model.

Comparative Performance of Predictive Models

All three models were trained and evaluated on the same data splits, and the performance on the independent test set with 255 samples is summarized in Table 3. As can be seen, among the models, XGBoost achieved the highest accuracy, with the lowest MAE = 0.106 MPa and RMSE = 0.149 MPa, and an R^2 of 0.999, indicating near-perfect agreement with experimental results. RF also performed very well (MAE = 0.123 MPa, RMSE = 0.173 MPa, R^2 = 0.997), outperforming ANN (MAE = 0.376 MPa, RMSE = 0.525 MPa, R^2 = 0.994). These results highlight the superior capability of ensemble tree-based models in capturing the complex, non-linear relationships governing concrete strength.

Table 3. Performance Comparison of ANN, RF, and XGBoost Models on the Test Set

Model	MAE (MPa)	RMSE (MPa)	R^2
ANN	0.376	0.525	0.994
RF	0.123	0.173	0.997
XGBoost	0.106	0.149	0.999

Model Validation on Independent Experimental Data

A scatter plot, Figure 4, comparing the predicted compressive strength versus the actual compressive strength of concrete samples, likely obtained from a machine learning model. Each point represents a single data instance, with the x-axis showing the experimentally measured compressive strength (MPa) and the y-axis showing the predicted values (MPa). A red dashed line corresponding to *y* = *x* is included as a reference to indicate perfect prediction accuracy. Data points that lie closer to this line demonstrate high prediction accuracy, while deviations indicate discrepancies between the predicted and actual strengths. The plot suggests that most predictions closely follow the to *y* = *x* line, particularly in the mid-range strengths, indicating strong model performance.

To assess the model's performance in a real-world scenario, the XGBoost model was selected and tested on a set of 20 independent experimental samples that were not part of the training or testing datasets. This validation model was specifically designed for a more streamlined application, using only the four most critical input variables: Cement, Water, Fine Aggregate, and Coarse Aggregate. As can be seen in Figure 4, the scatter plot shows some variability at lower compressive strengths, indicating that the model slightly underestimates or overestimates extreme values.

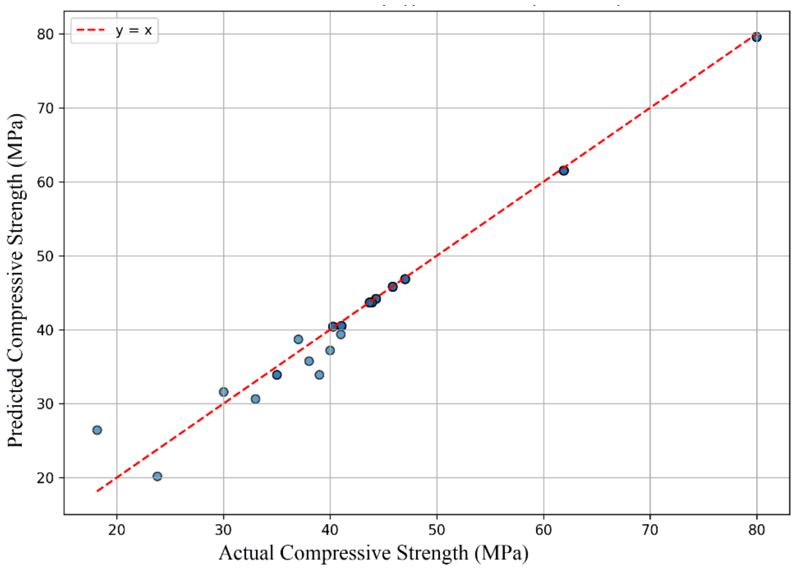

Figure 4. Predicted Vs. Actual Compressive Strength for the Xgboost Model on the Test Set.

Overall, the distribution of points demonstrates that the model effectively captures both the trend and variance of the actual measurements. On completely unseen data, the model achieved an MAE of 0.737 MPa and an R^2 of 0.921, well within acceptable limits for practical engineering applications. Most points closely align with the reference line, reflecting a high R^2 and low MAE and RMSE, which indicate reliable predictive performance. This visualization provides a valuable diagnostic tool for evaluating the accuracy and robustness of concrete strength prediction models in engineering contexts.

Conclusion

This study developed and validated a predictive model for estimating concrete compressive strength by integrating local experimental data with an international benchmark dataset. The XGBoost model achieved an MAE of 0.106 MPa and an R^2 of 0.999 on the independent test set. Validation on 20 new samples (R^2 = 0.921) confirmed its ability to generalize to unseen, real-world mix designs. While Random Forest showed slightly lower errors, XGBoost proved highly competitive and particularly suitable for complex, multi-output, and real-time applications, highlighting the benefit of combining local experimental insights with established global data.

The proposed model offers a reliable tool for quality control and mix design optimization in the construction industry. Beyond predictive accuracy, it provides a foundation for AI-driven systems in intelligent infrastructure management, enabling data-driven, efficient, and adaptable engineering practices. By demonstrating high accuracy, robust generalization, and practical applicability, this work represents a significant step toward next-generation AI-assisted structural engineering, supporting more informed and reliable decision-making in concrete construction.

Acknowledgment

This research is funded by the Lac Hong University (LHU), Dong Nai, Vietnam. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the LHU.

References

[1] Gagg, C. R. (2014). Cement and concrete as engineering material: A historic appraisal and case study analysis. Engineering Failure Analysis, 40, 114–140.

[2] Neville, A. M. (1963). Properties of concrete. Pearson Education India.

[3] Whight, J. K. (2012). Reinforced Concrete Mechanics and Design, 6th Edition. Pearson.

[4] Lamond, J., & Pielert, J. (2006). Significance of Tests and Properties of Concrete and Concrete-Making Materials. ASTM International 100 Barr Harbor Drive, PO Box C700, West Conshohocken, PA 19428-2959. https://doi.org/10.1520/STP169D-EB

[5] Marchewka, A., Ziolkowski, P., & García Galán, S. (2025). Real-time prediction of early concrete compressive strength using AI and hydration monitoring. Scientific Reports, 15(1), 15463.

[6] Lee, S.-C. (2003). Prediction of concrete strength using artificial neural networks. Engineering Structures, 25(7), 849–857.

[7] Chou, J.-S., Chiu, C.-K., Farfoura, M., & Al-Taharwa, I. (2011). Optimizing the prediction accuracy of concrete compressive strength based on a comparison of data-mining techniques. Journal of Computing in Civil Engineering, 25(3), 242–253.

[8] Nguyen, T. T., and Dinh, K. (2020). An artificial intelligence approach for concrete hardened property estimation. Journal of Science and Technology in Civil Engineering (STCE) - NUCE, 14(2), 40-52.

[9] Pham, T. T., Nguyen, T. T., Nguyen, L. N., Nguyen, P. V. (2020). A neural network approach for predicting hardened property of Geopolymer concrete." International Journal of GEOMATE, Oct. 2020, Vol 19, Issue 74, pp.193–201.

[10] Wu, Y., Pieralisi, R., Sandoval, F. G. B., López-Carreño, R. D., & Pujadas, P. (2024). Optimizing pervious concrete with machine learning: Predicting permeability and compressive strength using artificial neural networks. Construction and Building Materials, 443, 137619.

[11] Nguyen, T. T., Pham Duy, H., Pham Thanh, T., & Vu, H. H. (2020). Compressive strength evaluation of fiber-reinforced high-strength self-compacting concrete with artificial intelligence. Advances in Civil Engineering, 2020(1), 3012139.

[12] Sun, Z., Li, Y., Yang, Y., Su, L., & Xie, S. (2024). Splitting tensile strength of basalt fiber reinforced coral aggregate concrete: optimized XGBoost models and experimental validation. Construction and Building Materials, 416, 135133.

[13] Nguyen, T. T., Tung, P. T., Tan, N. N., Linh, N. N., & Luc, T. T. (2022). A study of factors affecting GPR signal amplitudes in reinforced structures using deep belief networks. Infrastructures, 7(9), 123.

[14] Verma, M., & Sharma, U. (2025). Prediction and comparison of compressive strength of geopolymer concrete by using a decision tree and random forest regression model. Asian Journal of Civil Engineering, 1-16.

[15] Nguyen-Sy, T. (2024). Optimized hybrid XGBoost-CatBoost model for enhanced prediction of concrete strength and reliability analysis using Monte Carlo simulations. Applied Soft Computing, 167, 112490.

[16] Nguyen, T. T., and Dinh, K. (2020). Prediction of bridge deck condition rating based on artificial neural networks. Journal of Science and Technology in Civil Engineering (STCE) - NUCE, 13(3), 15-25.

[17] Guijo-Rubio, D., Durán-Rosal, A. M., Gutiérrez, P. A., Gómez-Orellana, A. M., Casanova-Mateo, C., Sanz-Justo, J., Salcedo-Sanz, S., & Hervás-Martínez, C. (2020). Evolutionary artificial neural networks for accurate solar radiation prediction. Energy, 210.

[18] Nguyen, T. T., Pham, H. T., & Nguyen, H. K. (2025). Data-Driven Approach to Predict Fire-Resistance Ratings of Timber Columns. Civil Engineering Journal, 11(10), 4232–4245. https://doi.org/10.28991/CEJ-2025-011-10-013

[19] Hung, D. V., Hung, H. M., Anh, P. H., & Thang, N. T. (2020). Structural damage detection using a hybrid deep learning algorithm. Journal of Science and Technology in Civil Engineering - NUCE, 14(2), 53-64.

[20] Rabiej, M., & Rabiej, S. (2021). Application of the artificial neural network for the identification of polymers based on their X-ray diffraction curves. Computational Materials Science, 186, 110042.

[21] Ibrahim, M., Haider, A., Lim, J. W., Mainali, B., Aslam, M., Kumar, M., & Shahid, M. K. (2024). Artificial neural network modeling for the prediction, estimation, and treatment of diverse wastewaters: A comprehensive review and future perspective. Chemosphere, 362, 142860.

[22] Moradi, M., Gul, F. C., & Zarouchas, D. (2024). A novel machine learning model to design history-independent health indicators for composite structures. Composites Part B: Engineering, 275, 111328.

[23] Deng, Y., Ju, H., Li, Y., Hu, Y., & Li, A. (2024). Abnormal data recovery of structural health monitoring for the ancient city wall using a deep learning neural network. International Journal of Architectural Heritage, 18(3), 389-407.

[24] Kesavan, A. (2024). Embedded intelligence in structural health monitoring using artificial neural networks (Doctoral dissertation, RMIT University).

[25] Yeh, I.-C. (1998). Modeling of the strength of high-performance concrete using artificial neural networks. Cement and Concrete Research, 28(12), 1797–1808.

[26] Ministry of Science and Technology. (2022). TCVN 3118:2022, Hardened concrete – Test method for compressive strength (No. TCVN 3118:2022).

[27] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., & Dubourg, V. (2011). Scikit-learn: Machine learning in Python. The Journal of Machine Learning Research, 12, 2825–2830.

[28] Géron, A. (2022). Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems. O'Reilly Media, Inc.

[29] Hunter, D., Yu, H., Pukish III, M. S., Kolbusz, J., & Wilamowski, B. M. (2012). Selection of proper neural network sizes and architectures—A comparative study. IEEE Transactions on Industrial Informatics, 8(2), 228–240.

[30] Goodfellow, I., Bengio, Y., Courville, A., & Bengio, Y. (2016). Deep learning (Vol. 1, Issue 2). MIT Press Cambridge.

[31] Thapa, J. (2024). Concrete compressive strength prediction by an artificial neural network approach. Journal of Engineering Issues and Solutions, 3(1), 76–90.

[32] Chollet, F. (2017). Deep Learning With Python Manning Publications Company. Manning Publications Company.

[33] Mandal, A. (2025). Predicting compressive strength of concrete using advanced machine learning techniques: A combined dataset approach. Asian Journal of Civil Engineering, 1–17.

[34] Chen, T., & Guestrin, C. (2016). Xgboost: A scalable tree boosting system. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 785–794.