Prediction of Smart Concrete Properties Using AI Techniques

Mr.Anuj Ramteke[1], Mrs. Aboli Ravikar[2]
[1, 2] Dept of Civil Engineering ME Construction & Management
2Associate Professor,Dept of Civil Engineering
[1, 2] Dr. D. Y. Patil Unitech Society's
Dr.D.Y.PatilInstituteofTechnology,

Abstract- *Smart concrete is transforming construction by combining self-sensing and self-healing characteristics, allowing for real-time structural health monitoring. However, because of different material compositions and environmental conditions, it is difficult to estimate its attributes, such as strength, durability, and crack resistance. The use of sophisticated computational techniques like machine learning (ML) and deep learning (DL), which are subsets of artificial intelligence (AI), is growing. These techniques offer potent answers by evaluating enormous datasets to precisely predict tangible behavior. These models increase prediction accuracy and lessen the need for expensive laboratory testing, increasing the viability of smart concrete for next-generation infrastructure. Convolutional Neural Networks (CNNs) improve fracture identification through image analysis, while Artificial Neural Networks (ANNs) and Support Vector Machines (SVMs) are AI models that are excellent at predicting mechanical features. By evaluating data from embedded sensors and historical records, AI optimizes mix designs and predicts probable faults. Case studies show that AI is beneficial, with ANN-based self-healing concrete achieving 93% accuracy in repair prediction and CNN-powered bridge monitoring reducing inspection expenses by 25%. These achievements demonstrate AI's significance in making building more intelligent and sustainable.*

Despite its potential, problems exist, such as data scarcity, model interpretability, and computing demands. Future research focuses on Explainable AI (XAI) for increased transparency, hybrid models that combine physics and machine learning, and edge AI for real-time decision-making. As AI evolves, its combination with smart concrete will result in safer, more durable infrastructure, opening the door for intelligent cities and resilient buildings. AI and smart concrete have the potential to reinvent current construction by combining innovation and pragmatism for a more sustainable future.

Keywords- Artificial Intelligence, Machine Learning, Smart Concrete, Artificial Neural Networks, Self-healing Concrete.

I. INTRODUCTION

Smart concrete represents a watershed moment in construction materials by combining self-sensing, self-healing, and real-time monitoring capabilities. Smart concrete, unlike ordinary concrete, actively detects damage and initiates self-repair using technologies such as conductive fillers (e.g., carbon nanotubes), microencapsulated healing chemicals, and embedded sensor networks.Smart concrete is gaining popularity in response to rising demand for sustainable, long-lasting, and low-maintenance infrastructure. Its inbuilt sensors provide constant structural health data, allowing for predictive maintenance.

Artificial intelligence (AI), specifically machine learning models like as Artificial Neural Networks (ANNs), is crucial for analyzing sensor data, anticipating concrete behavior, and optimizing mix designs. AI models have been extremely effective in calculating compressive strength, exceeding standard empirical methods, particularly for combinations including unusual chemicals.

Advanced techniques, such as hybrid multilayer perceptrons and Evolutionary ANNs, enhance prediction accuracy and efficiency, allowing for real-time decision-making and minimizing dependency on destructive testing.

Smart concrete and artificial intelligence are transforming current construction by improving safety, performance, and sustainability in infrastructure building.

II. REVIEW OF LITERATURE

Chakravarthy et al.(2023)– ML Models For SSC with Biomedical Waste Ash[1]. In this paper, the author discusses the use of incinerated biomedical waste ash (IBMWA) in self-compacting concrete (SCC) as a sustainable alternative material. Along with IBMWA, lightweight expanded clay aggregate (LECA) and ground granulated blast furnace slag (GGBS) were employed to partially replace traditional components. To estimate the compressive strength of these mixes, the researchers used three machine learning models: Artificial Neural Networks (ANN), Gradient Tree Boosting (GTB), and CatBoost Regressor. GTB and CBR outperformed ANN in prediction accuracy. The study emphasizes the potential of mixing waste resources and machine learning to create environmentally friendly, high-performance concrete.

Azizifar & Babajanzadeh (2018) – Predicting SSC Strength with Silica Fume[2].Azizifar and Babajanzadeh used machine learning to forecast the compressive strength of self-compacting concrete (SCC) that contained silica fume. They used Multivariate Adaptive Regression Splines (MARS) and Gene Expression Programming (GEP) to model the nonlinear interactions between major mix factors. Both methods accurately measured the impacts of silica fume and other components, demonstrating the importance of data-driven approaches in SCC design.

Hoang(2022) – Multi-Algorithm ML Study on SSC Strength[3].In his 2022 study, Hoang compared several advanced machine learning algorithms to predict the compressive strength of self-compacting concrete (SCC). Addressing the complexity of SCC mixtures with multiple supplementary materials, he evaluated Levenberg–Marquardt Artificial Neural Networks (ANN), Genetic Expression Programming (GEP), Deep Neural Network Regression (DNNR), and Support Vector Regression (SVR). Using a detailed dataset of mix components, the study assessed each model’s ability to accurately capture nonlinear relationships and forecast compressive strength, highlighting the strengths of modern AI techniques over traditional methods.

Asteris et al.(2021) – Hybrid Ensemble Models for Concrete Strength[4]. In their 2021 study, Asteris et al. developed hybrid ensemble models to improve the prediction of concrete compressive strength by combining multiple machine learning algorithms. Using surrogate modeling techniques with Support Vector Machines (SVM) and Decision Trees (DT), the hybrid approach enhanced accuracy, stability, and generalization compared to individual models. Tested on a dataset of various concrete mix components, the ensemble effectively captured complex nonlinear behaviors, benefiting from the complementary strengths of SVM’s handling of high-dimensional data and DT’s ability to model threshold effects.

Behnood & Golafshani (2018) – ANN with Grey Wolf Optimizer for Silica Fume Concrete[5].Behnood and Golafshani (2018) used a hybrid model that used an Artificial Neural Network (ANN) and a Multi-Objective Grey Wolf Optimizer (GWO) to forecast the compressive strength of silica fume concrete. Because the complex, nonlinear effects of silica fume render typical models ineffective, the GWO was utilized to optimise the ANN's weights and biases, enhancing accuracy and avoiding local minima. This bio-inspired hybrid technique showed great promise for addressing complex concrete mix behaviors, providing a dependable and efficient tool for forecasting concrete qualities in construction.

Golafshani et al.(2020)– ANFIS and ANN Hybridized with Grey Wolf Optimizer[6].Golafshani et al. (2020) created hybrid models to forecast the compressive strength of regular and high-performance concrete by merging the Adaptive Neuro-Fuzzy Inference System (ANFIS) and Artificial Neural Networks (ANN) with the Grey Wolf Optimizer (GWO). ANFIS used fuzzy logic and neural networks to deal with uncertainty and nonlinear patterns, whereas GWO optimized model parameters for faster learning and convergence. When tested on several real datasets, these hybrid models outperformed standard approaches, attaining excellent accuracy and robust predictions.

Awoyera et al.(2020) –ANN and GEP for Geopolymer SSC[7].Awoyera et al. (2020) used Artificial Neural Networks (ANN) and Gene Expression Programming (GEP) to estimate the compressive strength of geopolymer self-compacting concrete (SCC), which is made with aluminosilicate-rich ingredients such as fly ash instead of Portland cement. Using a dataset of mix parameters and curing conditions, both models demonstrated high prediction accuracy, with ANN marginally outperforming in error reduction and generalization. While ANN excelled at capturing complicated nonlinearities, GEP supplied interpretable equations to help understand material behavior. The study proved that both methods are effective for quick, non-destructive strength prediction, which will help progress sustainable concrete technology.

Farooq et al.(2021) – Comparative Study Using SVM,ANN, and GEP[8].Farooq et al. (2021) investigated Artificial Neural Networks (ANN), Support Vector Machines (SVM), and Gene Expression Programming (GEP) for estimating the compressive strength of self-compacting concrete (SCC) containing fly ash. Using a dataset of mix and curing parameters, ANN demonstrated the highest accuracy, followed by SVM, while GEP generated interpretable equations but had lower accuracy. The study emphasized ANN's great capacity to describe complex nonlinear behavior and SVM's durability with high-dimensional data, making them useful for predicting SCC intensity.

Ghorpade & Koneru (2018) – Pattern Recognition Neural Network For SCC Grades[9].Ghorpade and Koneru (2018) used a pattern recognition neural network (PRNN) to classify self-compacting concrete (SCC) mixes into strength grades rather than forecasting actual compressive strengths. Using a dataset of mix design characteristics, the model accurately classified SCC into classes such as M30, M40, and M50. This approach streamlines quality control and enables faster, automated decision-making in concrete manufacturing.

III. OBJECTIVE

The prediction of concrete compressive strength is evolving as smart materials and AI approaches become more prevalent. Traditional empirical approaches frequently fail when used to modern concretes incorporating pozzolans, admixtures, or industrial byproducts. This paper investigates the limitations of traditional methodologies and assesses the ability of AI models, such as Artificial Neural Networks (ANNs), Evolutionary ANNs (EANNs), and hybrid systems, to reliably predict compressive strength. By collecting relevant information and evaluating model performance, the study hopes to demonstrate AI's effectiveness as a speedier, more adaptive alternative to destructive testing. Furthermore, it investigates smart concrete technologies that incorporate self-sensing and self-healing characteristics in order to enable sustainable, intelligent infrastructure. Smart materials and artificial intelligence work together to provide a game-changing solution for efficient, data-driven decision-making in modern building.

IV. METHODOLOGY

This study investigates how well machine learning predicts self-compacting concrete's compressive strength. Using reliable databases like Elsevier, Springer, SCOPUS, Web of Science, IEEE, the Turkish Journal of Engineering, and ScienceDirect, a thorough literature review was carried out. These resources were chosen because they are pertinent to real-world AI applications in research. In order to reflect a contemporary, data-driven approach, the methodology concentrated on finding, assessing, and summarizing studies that combine machine learning with concrete property prediction. This study investigates how well machine learning predicts self-compacting concrete's compressive strength. Using reliable databases like Elsevier, Springer, SCOPUS, Web of Science, IEEE, the Turkish Journal of Engineering, and ScienceDirect, a thorough literature review was carried out. These resources were chosen because they are pertinent to real-world AI applications in research..In order to reflect a contemporary, data-driven approach, the methodology concentrated on finding, assessing, and summarizing studies that combine machine learning with concrete property prediction.

Fig.1 Diagram illustrating the methodology adopted for selecting pertinent articles for this study.

V. CASE STUDY

PREDICTION OF CONCRETE COMPRESSIVE AND FLEXURAL STRENGTH-Numerous approaches have been utilized globally to estimate the compressive and flexural strength of concrete. These methods are typically categorized into empirical strategies and computational modeling techniques. Approaches Empirical models, which frequently relate strength to variables like the water-to-cement ratio, reproduce experimental results under particular circumstances. Although they are helpful in demonstrating how mix components affect strength, they need to be thoroughly validated to guarantee correctness. To improve predictions, some models additionally include correction factors.

Computational Modeling Similar to finite element analysis, computational modeling uses microstructure data and thermodynamic equations to simulate tangible behavior. In order to simulate hydration, pixel-based techniques arrange cement particles in a specific area; adding experimental data enhances realism and accuracy.

Mechanical ModelingConcrete behavior is represented by mechanical models, such as the spring-and-dashpot analogy, which associate age effects with a dashpot and the cement matrix with a spring. Although these models are helpful for forecasting compressive strength, they frequently lose accuracy because they are unable to capture time-dependent behavior, particularly in early-age strength. Statistical Methods Multiple linear regression is one statistical model that uses empirical data to determine the associations between variables. Despite being simple to use, they frequently lack the flexibility of more sophisticated approaches and require big datasets, depending on how well the selected model fits.

When predicting concrete strength, regression approaches aid in quantifying the correlations between inputs and results. Despite their occasional complexity, they provide high accuracy, particularly when combined with supporting data such as slump tests and correlation analysis. By optimizing input factors, a reviewed study that employed a multivariable power equation achieved a 99.99% correlation with test results. The reliability of the model was further enhanced by standardization. In primarily linear systems, regression is still effective and frequently outperforms sophisticated models like neural networks.

Artificial Intelligence Applications In order to handle difficult tasks like pattern recognition, learning, and optimization—all of which are crucial in civil engineering—artificial intelligence (AI) mimics human reasoning. To address uncertainty and represent nonlinear phenomena, methods like ANFIS, neural networks, and genetic algorithms are employed. Even while AI models are quite accurate and versatile, their real-world performance depends on big, high-quality information.

FUZZY LOGIC -Fuzzy logic, introduced by Lotfi Zadeh, has played a key role in improving how computers handle complex decisions and uncertainties. His work led to the development of fuzzy inference systems like Mamdani, Sugeno, and Takagi models, which are now widely used to better simulate real-world processes. Unlike traditional logic that sees things as simply true or false, fuzzy logic allows for partial truths—meaning something can belong to multiple categories to different degrees. This makes it much better at dealing with situations where boundaries aren't clear-cut (Duan et al., 2013; Erzin, 2007; Sergio & Mauro, 1997; Silva et al., 2021; Syed et al., 2023). For example, instead of labeling people strictly as 'short' or 'tall' based on height, fuzzy logic recognizes the in-between cases. While classic set theory is often seen as too rigid, fuzzy sets provide a smoother way to represent membership. In fields like concrete technology, Fuzzy Logic Controllers (FLCs) are especially useful because they handle the complex, nonlinear behavior of materials better than traditional models, helping to predict and control compressive strength more effectively.

NEURAL NETWORKS-Neural networks have come a long way since the 1950s, evolving from simple mechanical tasks to mimicking human brain functions for complex problem-solving. These Artificial Neural Networks (ANNs) are made up of connected "neurons" that work together to process information, making them great at handling tricky, nonlinear problems. In civil engineering, ANNs are used for things like detecting structural damage, modeling materials, and designing concrete mixes (Saridemir, 2010). For example, Gandomi and Roke (2015) combined ANNs with fuzzy logic to predict the strength of self-compacting concrete, achieving a strong accuracy with an R^2 of 0.9767 using six neurons and 500 training cycles. Later, Khan et al. (2021) improved the model by tweaking the layers and neurons, reducing errors significantly. Faradonbeh et al. (2018) also showed great results, confirming that factors like the water-to-binder ratio affect strength predictions. Overall, neural networks consistently outperform traditional methods, especially for low to medium strength concrete mixes.

GENETIC PROGRAMMING-Genetic Programming (GP) is a smart approach inspired by natural evolution, where computer programs "evolve" by mimicking processes like selection, mutation, and crossover to solve complex problems. A special type called Gene Expression Programming (GEP) fine-tunes these programs by testing how well they perform and making improvements over time. For example, Faradonbeh and colleagues (2018) looked at how different population sizes affect models when fly ash partially replaces cement, focusing on predicting concrete's compressive strength after 28 days. Their model used four key inputs: water, cement, coarse aggregates, and fine aggregates. They analyzed data from 1,442 tests covering 68 different concrete mixes, with strengths ranging from 18 to 27 MPa, water-cement ratios between 0.39 and 0.62, and aggregate sizes from 25 to 100 mm. Researchers like Ferreira (2002, 2006) tested these models against real-world data and found them to be highly accurate. This shows how GP-based methods can effectively capture the complex relationships in concrete materials and their strength.

PARAMETERS IN MACHINE LEARNING MODELS-In machine learning, hyperparameters are settings chosen before training that guide how algorithms learn, such as learning rate, cluster numbers, or treedepth. Hyperparameters govern the learning process (e.g., batch size) or model structure (e.g., number of hidden layers), in contrast to trainable parameters (e.g., weights). They impact model accuracy, training duration, and computing cost and need to be predetermined. The majority of models require careful hyperparameter selection to balance underfitting and overfitting and guarantee acceptable performance on new data, even though simple models like linear regression might not require tweaking. For instance, the number of neurons per layer in a neural network is dependent on the number of layers.

UNTRAINABLE PARAMETERS-Model performance depends on hyperparameters, but if they are not adjusted correctly, overfitting—in which the model learns noise rather than real patterns—can result in subpar performance on fresh data. In regression, for instance, raising the polynomial degree may reduce training error but frequently degrades test accuracy. In order to prevent this and preserve model robustness, structural parameters such as polynomial degree are fixed during training.

HYPERPARAMETERTUNING IN MACHINE LEARNING-

Only a small number of the numerous hyperparameters have a significant effect on the model's performance; the most crucial ones are learning rate and network structure (layers, neurons). There is less of an impact from others, such as batch size and momentum (Asteris et al., 2021; Mehmannavaz et al., 2014; Neira et al., 2020). Large batches can occasionally be beneficial, but studies show that mini-batch sizes of two to thirty-two typically perform best. Because over-optimizing small parameters can increase complexity without enhancing results, careful tweaking is essential.

HYPERPARAMETER OPTIMIZATION-To reduce loss and increase model correctness on test or validation data, hyperparameter optimization determines the optimal set of hyperparameters. In order to direct the search for ideal settings, it interprets the loss as an objective function.

REPRODUCIBILITY IN MACHINE LEARNING-In machine learning, reproducibility is essential, necessitating meticulous monitoring of parameters, models, and experiments to guarantee reliable outcomes. Replication is hampered by frequent code or data changes in the absence of adequate infrastructure. These days, researchers can maintain and share datasets, configurations, and measurements with the aid of tools and web platforms. This is crucial in deep learning, where complexity makes reproducibility difficult (Khan, 2012).

DEVELOPING MACHINE LEARNING MODELS - Machine learning models can be developed using different strategies based on how much labeled data is available. As outlined by Khan (2012) and Onyelowe et al. (2021), there are four main approaches: supervised, unsupervised, semi-supervised, and reinforcement learning.

Supervised learning - Is like learning with a teacher. The model is trained on data where both the inputs and correct outputs are known. It learns from these examples and can then make accurate predictions on new, unseen data—great for tasks like spam detection or medical diagnosis.

Unsupervised learning - is more about self-discovery. The model looks at data without any labels and tries to find hidden patterns or groupings on its own. It's useful when you don't know exactly what you're looking for, like grouping similar customer behaviors or detecting unusual activity.

Semi-supervised learning - mixes both worlds. A small amount of labeled data helps guide the model, while a larger set of unlabeled data helps it learn more efficiently. This approach is helpful when labeling is expensive or time-consuming.

Reinforcement learning - is like learning by trial and error. The model interacts with an environment, makes decisions, and learns from rewards or penalties. It's commonly used in robotics, gaming, and self-driving cars.

Each method has its strengths depending on your goals and the data you have.

CATEGORIES OF MACHINE LEARNING MODELS - Machine learning tasks are generally divided into *classification* and *regression* problems (Prasad et al., 2019). *Classification* assigns data to specific categories (e.g., spam vs. not spam), while *regression* predicts continuous numerical values (e.g., house prices). Some algorithms are flexible enough to handle both.

Popular classification algorithms include - Support Vector Machines (SVM), Random Forest, Decision Tree, Logistic Regression, Naive Bayes, K-Nearest Neighbors (KNN)

Common regression algorithms include - K-Nearest Neighbors Regression, Decision Tree Regression, Random Forest Regression, Neural Network Regression, Linear Regression.

RECOGNIZING AND MITIGATING OVERFITTING IN MACHINE LEARNING - Overfitting occurs when a model picks up on the noise and anomalies in the training data in addition to the underlying patterns. High accuracy on training sets may arise from this, but on fresh, untested data, it frequently leads to subpar performance (Dias & Pooliyadda, 2001; Nguyen et al., 2019).

Overfitting can be caused by - Limited training data, Irrelevant features or noise in the dataset, Overly complex models, Excessive training duration.

Regularization (L1, L2), learning curve analysis, and cross-validation are some of the methods used to identify and stop overfitting. Additionally, techniques like feature selection, data augmentation, dropout, and early termination aid in enhancing generalization. Building dependable models in deep learning requires controlling overfitting, which sometimes results from an over emphasis on small training features.

CAUSES OF OVERFITTING - When the training data is excessively sparse or noisy, the model may overfit and learn patterns that are not very generalizable. It frequently manifests as early gains in validation metrics that are followed by a drop. This implies that the model'sadaptation to the training datais excessive. Preventing overfitting requires striking a balance between dataset quantity and model complexity.

IDENTIFYING OVERFITTING IN MACHINE LEARNING –

Validation Loss Gap - If training loss decreases while validation loss increases, the model is likely overfitting and failing to generalize.

Learning Curve Analysis - A widening gap between high training accuracy and lower validation accuracy signals overfitting.

Regularization Check - Adding a regularization term can reveal if the model is too complex and needs simplification.
Visual Inspection - Comparing model predictions to training data can highlight if the model is memorizing examples instead of learning patterns.

TECHNIQUES TO PREVENT OVERFITTING IN ML –

Regularization (L1/L2) - Adding penalties to the loss function helps keep the model from becoming too complex by limiting the size of its parameters.

Cross-Validation and Early Stopping - Checking the model's performance on different data splits ensures reliability, while stopping training early prevents it from over-learning the training data.

Data Augmentation - Generating new, varied samples by modifying existing data helps the model learn broader patterns.

Feature Selection - Removing irrelevant or redundant inputs simplifies the model and reduces the chance of it focusing on noise.

Dropout - Temporarily disabling some neurons during training encourages the model to develop more robust, generalized features.

Using these methods together helps build models that perform well not only on training data but also on new, unseen data with consistent accuracy.

FUZZY LOGIC APPROACH –

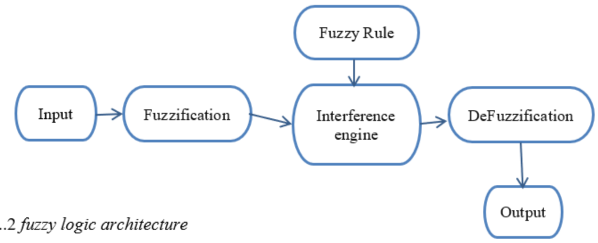

Fig..2 *fuzzy logic architecture*

A fuzzy rule base, fuzzification unit, inference engine, and defuzzification unit are the four main components of a conventional fuzzy logic system (Golafshani et al., 2020b). It begins by employing membership functions such as Gaussian or trapezoidal to transform clear inputs into fuzzy sets. Inputs may concurrently be a part of more than one fuzzy subset. The system mimics human reasoning by using language IF-THEN rules (e.g., "IF A THEN B"), which are then interpreted by an inference engine to provide fuzzy outputs.These hazy outcomes are then transformed into precise numerical values for decision-making by the defuzzification unit (Ben et al., 2022; Golafshani et al., 2020b).

PERFORMANCEEVALUATION OF THE GEP MODEL –

To ensure the reliability of the Gene Expression Programming (GEP) model, it is recommended to maintain a dataset-to-feature ratio of at least 3:1, with 5:1 considered optimal. The model's accuracy is evaluated using key metrics such as RMSE, MAE, and RSE, which measure how well the predictions align with experimental results (Ben et al., 2022; Golafshani et al., 2020b). External validation is performed through regression analysis, where slopes (k or k′) close to 1 and passing through the origin indicate accurate prediction of Self-Compacting Concrete (SCC) compressive strength (fc). The model's effectiveness heavily depends on carefully chosen fitting parameters, often refined through trial and error or experimental tuning. Critical factors impacting GEP performance include population size (number of chromosomes), head size, and the number of genes. This study explored five head sizes—8, 9, 10, 12, and 14—with three or four genes each. The head size governs the complexity of functional units, while the number of genes defines the model's structural components. These parameters are optimized to achieve the best predictive performance (Babatunde et al., 2022; Mogaraju, 2023; Shariati et al., 2021). These design decisions are fine-tuned through GEP to optimize predictive accuracy. A flowchart illustrating the GEP process is provided in Figure 3.

MODEL EVALUATION CRITERIA –

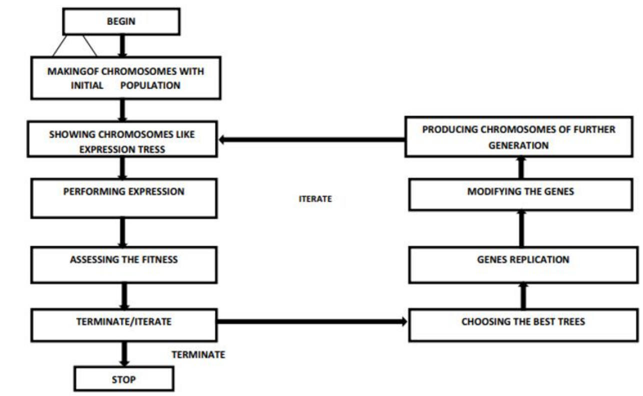

Fig.3 *Generic algorithm flow chart diagram*

The correlation coefficient (R) is widely used to assess how closely a model's predictions align with actual outcomes. However, it has notable limitations—particularly its insensitivity to scaling—which can result in misleading assessments if used in isolation. To address this, the current study incorporates a broader set of evaluation metrics, including Mean Absolute Error (MAE), Relative Squared Error (RSE), and Relative Root Mean Square Error (RRMSE). These metrics provide a more nuanced understanding of prediction accuracy by capturing both the magnitude and variability of errors. Additionally, a composite performance index proposed by Gandomi and Roke, which integrates R and RRMSE, is employed to ensure a more balanced and comprehensive evaluation of the model's performance. The mathematical foundation for the model's predictions is represented by the following generalized regression equation:

$$Fc = f(\alpha + \beta_1 Y_1 + \beta_2 Y_2 + \beta_3 Y_3 + \ldots + \beta_n Y_n) + \varepsilon \quad (1)$$

Where:
Fc = predicted compressive strength,
α = regression intercept,
β_1 to β_n = regression coefficients,
Y_1 to Y_n = input variables (predictors),
ε = error term capturing model deviations.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} (ei - mi)^2}{n}} \quad (2)$$

$$MAE = \sqrt{\frac{\sum_{i=1}^{n} |ei - mi|^2}{n}} \quad (3)$$

$$RSE = \sqrt{\frac{\sum_{i=1}^{n} (mi - ei)^2}{\sum_{i=1}^{n} (\overline{e} - ei)^2}} \quad (4)$$

$$RRMSE = \frac{1}{|\overline{e}|}\sqrt{\frac{\sum_{i=1}^{n} (ei - mi)^2}{n}} \quad (5)$$

$$R = \frac{\sum_{i=1}^{n} (ei - \overline{ei})(mi - \overline{mi})}{\sqrt{\sum_{i=1}^{n} (ei - \overline{ei})^2 \sum_{i=1}^{n} (mi - \overline{mi})^2}} \quad (6)$$

$$\rho = \frac{RRMSE}{1 + R} \quad (7)$$

EXPERIMENTAL CONFIGURATION FOR ARTIFICIAL NEURAL NETWORK - Concrete specimens were evaluated for both compressive and flexural strength at standardized curing intervals of 7, 14, 21, and 28 days, using cube specimens for compressive testing and beam specimens for flexural testing. At the end of the 28-day curing period, the cube samples were also weighed to calculate their density, offering further insight into the material's properties. The architecture of the artificial neural network (ANN) employed in the analysis is illustrated in Figure 4. Previous studies [72–75] have investigated the predictive capabilities of Gene Expression Programming (GEP) and Multivariate Adaptive Regression Splines (MARS) in estimating the 28-day compressive strength of self-compacting concrete (SCC). Further research [76–81] focused specifically on using MARS models to predict SCC strength based on a variety of input parameters, demonstrating the approach's potential for accurate, data-driven predictions. The outcomes confirmed that both GEP and MARS offered strong predictive accuracy, as demonstrated in Figures 5, 6, and 7, and summarized in Table 1.

GAPS IN KNOWLEDGE - Despite advancements in using artificial intelligence (AI) to predict the properties of self-compacting concrete (SCC), several key challenges remain that impact the accuracy and reliability of these models

Data Limitations - AI models require large, high-quality datasets to learn effectively. Variability in materials, curing processes, and testing methods makes it difficult to gather comprehensive and consistent data that accurately reflects SCC behavior.

Feature Selection - Identifying the most influential input variables, such as mix composition, curing duration, and admixtures, is essential for building robust models. This requires thorough analysis to determine which factors most affect compressive strength.

Model Transparency - While deep learning models offer strong predictive power, they often lack interpretability.Understanding how individual inputs influence predictions is critical for practical use but remains a challenge.
Generalization Challenges - AI models may perform poorly on data that differs from their training set. Techniques like transfer learning and domain adaptation are needed to improve adaptability.

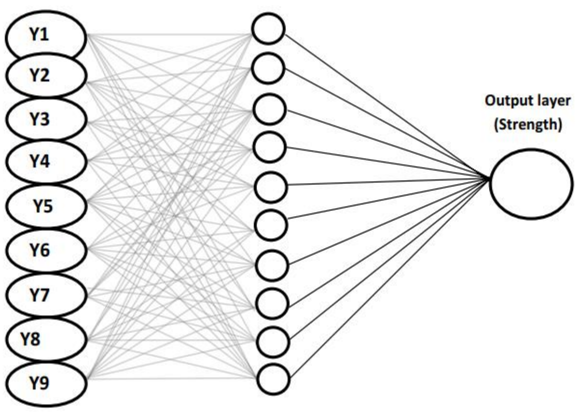

Fig.4 AAN Architecture

Validation Strategies - Accurate model evaluation requires proper metrics (e.g., MAE, RMSE) and validation techniques such as cross-validation to ensure generalizability and reliability.

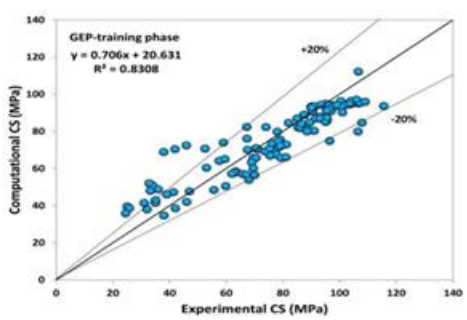

Fig.5 Scatter plot comparing the observed and predicted compressive strength during the training phase of the GEP model. Reference: Milad, B., and Valiollah, A. (2018). Civil Engineering Journal, Vol. 4, No. 7, July.

Uncertainty and Variability - Material inconsistencies and measurement errors hinder precise prediction. Techniques like ensemble modeling and Bayesian neural networks help quantify uncertainty, enhancing model trustworthiness.

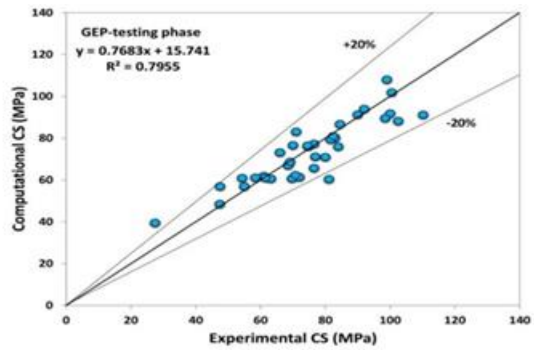

Fig. 6 Scatter plot illustrating the observed vs. predicted compressive strength during the testing phase of the GEP model. Source: Milad, B., and Valiollah, A. (2018). Civil Engineering Journal, Vol. 4, No. 7, July.

VI .CONCLUSION

This review highlights that artificial intelligence (AI) techniques are highly effective in predicting the compressive strength of self-compacting concrete (SCC), with model results closely matching experimental data. Most AI models reviewed achieved a strong correlation, with coefficients of determination (R^2) above 0.8, indicating reliable predictions of the 28-day compressive strength. Among these, Artificial Neural Networks (ANN) demonstrated particularly consistent and accurate performance, making them a preferred choice for civil engineering applications where precise property estimation is crucial.

AI models offer significant advantages by reducing the need for time-consuming and costly physical testing, thereby accelerating construction processes. Reinforcement learning methods, such as the Deep Deterministic Policy Gradient (DDPG) algorithm, show promise but require careful hyperparameter tuning to maximize accuracy by minimizing error.

A common challenge in AI modeling is overfitting, where models perform well on training data but poorly on new data. This can be addressed using techniques like cross-validation, data augmentation, dropout, and thoughtful feature selection. These strategies enhance the model's ability to generalize beyond the training set, ensuring dependable results in real-world scenarios. Overall, AI provides a powerful and efficient toolset for advancing SCC strength prediction with practical benefits for engineering projects.

REFERENCES

[1] Chakravarthy et al.(2023)– ML Models For SSC with Biomedical Waste Ash.

[2] Azizifar & Babajanzadeh (2018) – Predicting SSC Strength with Silica Fume.

[3] Hoang(2022) – Multi-Algorithm ML Study on SSC Strength

[4] Asteris et al.(2021) – Hybrid Ensemble Models for Concrete Strength.

[5] Behnood & Golafshani (2018) – ANN with Grey Wolf Optimizer for Silica Fume Concrete.

[6] Golafshani et al.(2020)– ANFIS and ANN Hybridized with Grey Wolf Optimizer.

[7] Awoyera et al.(2020) –ANN and GEP for Geopolymer SSC.

[8] Farooq et al.(2021) – Comparative Study Using SVM,ANN, and GEP.

[9] Ghorpade & Koneru (2018) – Pattern Recognition Neural Network For SCC Grad