

Article

AI-Enabled Sustainable Soil Stabilization for Resilient Urban Infrastructure: Advancing SDG 9 and SDG 12 through Hybrid Deep Learning and Environmental Assessment

Ishwor Thapa¹ and Sufyan Ghani^{1,2*}¹ Department of Civil Engineering, Sharda University, Greater Noida, India² Tailings Division, WSP India Private Limited, India* Correspondence: dr.sufyan.ghani@gmail.com**How To Cite:** Thapa, I.; Ghani, S. AI-Enabled Sustainable Soil Stabilization for Resilient Urban Infrastructure: Advancing SDG 9 and SDG 12 through Hybrid Deep Learning and Environmental Assessment. *Bulletin of Computational Intelligence* 2025, 1(1), 3–30.

Received: 28 May 2025

Revised: 16 June 2025

Accepted: 19 June 2025

Published: 30 June 2025

Abstract: Urban areas face increasing challenges in constructing resilient infrastructure on weak or unstable soils, especially amid the impacts of climate change and rapid urbanization. This study introduces an innovative AI-driven framework that integrates advanced hybrid deep learning architectures namely Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), and Transformer models for accurately predicting the Unconfined Compressive Strength (UCS) of Nano-silica (NS) stabilized soils. A key innovation of this research lies in the development of a novel CNN-Transformer hybrid model, which outperforms traditional and standalone AI models, achieving an R^2 of 0.97 and $RMSE$ of 0.22 reducing prediction error by over 49%. Robustness was further validated using a 10,000-iteration Monte Carlo simulation. In addition to predictive modeling, this study pioneers a comparative Life Cycle Assessment (LCA) between NS and cement-based stabilization, revealing that NS reduces CO₂ emissions by 55%, lowers energy consumption by 73%, and improves material efficiency. Furthermore, a user-friendly Graphical User Interface (GUI) tool has been developed, enabling real-time optimization of NS dosage for practical implementation in urban projects. This research not only contributes a high-performance predictive tool but also supports sustainable construction practices, aligning with SDG 9 (Industry, Innovation, and Infrastructure) and SDG 12 (Responsible Consumption and Production).

Keywords: sustainable infrastructure; nano-silica stabilization; hybrid deep learning; life cycle assessment; urban resilience

1. Introduction

The rapid pace of urbanization and the increasing demand for resilient infrastructure pose significant challenges for civil engineering. Traditional methods of soil stabilization often rely on high-resource materials and can have adverse environmental impacts. In recent years, the application of NS in geotechnical engineering has emerged as a promising solution, offering enhanced soil stability while minimizing the environmental footprint. Nano-silica, due to its unique properties at the molecular level, significantly improves the strength and durability of soils, making it an ideal material for stabilizing weak or unstable soil conditions commonly found in urban construction sites. Despite its potential, the application of NS requires precise optimization to achieve the best results in terms of both soil performance and cost-effectiveness. Conventional approaches for determining the ideal stabilization parameters, such as soil composition and NS dosage, often rely on trial-and-error or basic



empirical methods, which may not fully capture the complex interactions between the materials. This limitation necessitates a more advanced, data-driven approach to predict and optimize the outcomes of NS stabilization in real-world applications.

The process of soil stabilization is a significant component of civil engineering that includes all activities aimed at increasing the bearing capacity of soil for different construction uses. In the past, soil stabilization methods mostly focused on chemical additives or mechanical processes but nanotechnology has transformed this by providing new ways to improve soil performance. Amongst these alternatives, the employment of nano-materials like NS appears to be a promising way of improving soil properties from a nano-scale perspective [1–4]. Nanotechnology offers unique opportunities to change soil characteristics and improve its overall stability, making it an area of considerable interest in geotechnical engineering research [5]. Investigation of the durability properties of soil stabilized with NS is a particular focus in the field. Understanding how NS affects the strength and durability of stabilized soils is critical towards improving stabilization techniques and ensuring long life spans of civil engineering projects. However, experimental processes carried out in the conventional soil mechanics testing laboratory are generally time-consuming and expensive. In addition, determining suitable nano-materials and their best proportions according to different soil types is complicated. Moreover, predicting performance over a long period becomes problematic when it comes to stabilized soils as well. Also, scaling the laboratory tests done in the lab to field applications and ensuring that there is the same performance across various sites become difficult. Moreover, data integration and management can be a cumbersome process since management and integration of large data arising from diverse sources such as lab tests, field tests, and simulations must always guarantee consistency in quality as well as performance.

The study conducted on fly ash-treated fine-grained soils showed significant improvements in UCS and CBR, driven by chemical oxides like SiO_2 , Al_2O_3 , and CaO . Among three regression models (PQ, IA, FQ), the full quadratic model gave the most accurate UCS predictions. Simplification using geochemical indices, especially the Hydraulic Index (HI), maintained accuracy while improving interpretability. Curing time and CaO were identified as the most influential factors, highlighting the role of grouped oxides in enhancing soil strength for geotechnical applications [6]. The impact of chemical oxides on the unconfined compressive strength (UCS) of clay soils, emphasizing their role in modifying liquid limit (LL) and plasticity index (PI). Using a full quadratic model, the analysis showed that oxides such as SiO_2 and Al_2O_3 significantly reduce LL and PI, thereby enhancing UCS and soil stability. The findings support the use of targeted oxide additives to improve soil strength and ensure safer, more durable construction [7].

Machine learning algorithms like ANN, SVM, DT, and RF are increasingly used in geotechnical engineering for tasks such as soil classification, landslide prediction, and forecasting material properties [8]. The comparative study incorporates five metaheuristic algorithms along with SVR to predict the UCS of stabilized soil. Assessed with K-Fold, R2, RMSE, and MAE, SVR-HGS excelled in one case, while SVR-PSO performed better in another. SHAP and PDP examinations emphasized cement as the primary factor and pinpointed ideal A-line values to reduce stabilizing agents while keeping UCS intact [9]. The study investigates the use of cement and fly ash to stabilize clayey soil for subgrade pavement. Different combinations of cement and fly ash levels (ranging from 2–8% and 4–36% respectively) were evaluated for UCS at different curing times. Predictive models were created using Classification and Regression Tree (CART), Random Forest, and multiple regression, with CART showing the highest accuracy ($R^2 = 0.90$). DL models can be trained on existing datasets of soil properties and UCS measurements, allowing for rapid prediction of UCS for different soil-stabilizer combinations without the need for extensive physical testing [10]. Furthermore, employing DL methods in studies on soil mechanics offers a cost-efficient option [11–15]. In addition, lab tests frequently involve significant costs linked to tools, supplies, and staff. By utilizing current data and computational power, deep learning algorithms can make predictions for much less cost than running multiple physical experiments. This affordability allows researchers with limited resources to conduct thorough studies on the behaviour of NS-stabilized soil. In addition to being efficient and cost-effective, DL provides the benefit of improved accuracy. DL models are highly proficient at recognizing complex patterns and relationships in extensive datasets, making them essential for comprehending the intricate behaviour of stable soils [16]. Through the analysis of large quantities of data, these models can offer extremely precise predictions of UCS, exceeding the abilities of conventional analysis techniques. This increased level of precision boosts the dependability of research results and aids in making well-informed decisions in engineering scenarios.

A lot of investigations have a related problem whereby few samples are used in the construction and validation of models. The application limits for these models may be constrained by their small sample sizes which often fail to capture all the variability in the real soil conditions [17]. Feedforward neural networks trained on Monte Carlo data using the Modified Cam Clay model effectively estimate footing limit pressure and displacements with high accuracy and low computational cost, aiding shallow foundation failure prediction in

geotechnical engineering [18]. Moreover, studies also differ in their choices of input parameters, potentially missing crucial factors that could greatly affect the accuracy of the model. Furthermore, the effectiveness of the models in making accurate predictions can be significantly affected by the quality and range of the data employed in their training and testing stages. To solve these issues and raise the dependability and relevance of predictive models, efforts to improve data quality, standardize input parameters to better model evaluation, and guarantee transparent reporting are required. This research included 509 test samples with varying soil types (CI, CL-ML, MI), NS levels, and curing periods. This data was utilized to train and validate deep learning and hybrid deep learning models that predict the UCS of NS-stabilized soils, ensuring accurate and robust forecasts. Also, this study closes the knowledge gap regarding the increased performance and durability of NS-stabilized soil. Previous studies have focused on the immediate effects of NS on soil properties; however, this work aims to provide a comprehensive evaluation of the long-term characteristics of NS-treated soil over an extended period. The objective is to improve understanding of the interactions between NS level, length of curing, and soil variables that affect stabilized soil strength over time.

To address this gap, the current research introduces a novel hybrid AI-driven framework that integrates Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM) networks, Recurrent Neural Networks (RNN), and Transformer models. By utilizing these state-of-the-art machine learning techniques, the study aims to predict the stability of NS stabilized soils with high accuracy, ensuring that the results are both reliable and scalable. Furthermore, the research incorporates a computational economic analysis, providing a holistic evaluation of the costs and long-term benefits associated with the use of NS in urban infrastructure development. The main objective of this study is to develop an integrated system that combines AI-driven predictions with economic considerations, thereby enabling engineers and urban planners to make informed decisions about soil stabilization materials. By optimizing the use of NS and predicting its performance under various conditions, the research seeks to enhance the sustainability, cost-effectiveness, and resilience of urban infrastructure projects. Ultimately, this study contributes to the growing body of knowledge at the intersection of geotechnical engineering and artificial intelligence, offering a path forward for more sustainable, economically viable, and environmentally responsible construction practices.

2. Research Significance

The significance of this research lies in its innovative integration of advanced artificial intelligence (AI) models and NS soil stabilization techniques to address the growing challenges in urban infrastructure development. In an era where sustainable and cost-effective solutions are paramount, this study leverages cutting-edge hybrid AI approaches incorporating Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM) networks, and Transformer models to predict and optimize the stability of NS stabilized soils. This hybrid AI model enhances the precision of slope stability predictions, a critical factor for ensuring the safety and resilience of urban infrastructure projects. Furthermore, by utilizing AI's ability to model complex, non-linear relationships within geotechnical data, this research significantly improves the understanding of soil behavior and stabilizing factors. Moreover, the research provides practical applications for urban planners and engineers by offering a robust predictive tool that helps optimize the use of materials and minimize construction costs while ensuring structural integrity. This AI-driven prediction model, combined with economic analysis, offers a strategic approach to project planning and execution, ensuring that infrastructure is not only cost-effective but also sustainable over its lifecycle. This is particularly crucial as urban areas continue to expand and face new challenges posed by climate change, rapid population growth, and increasing demand for resilient infrastructure. Ultimately, this study bridges the gap between geotechnical engineering and AI, marking a significant advancement in both fields. It provides a scalable and globally applicable solution for improving soil stabilization methods, making it an essential resource for policymakers, engineers, and sustainability experts working to create resilient and sustainable cities. The results of this research promise to shape the future of urban infrastructure and pave the way for smarter, more efficient construction practices worldwide.

3. Methodology

The process outlined in Figure 1 begins from the collection of the experimental dataset. These specimens are subjected to various laboratory examinations to identify their physical and chemical characteristics, specifically concentrating on moisture level, density, and classification of soil. The next step involves treating the soil samples with varying amounts of NS before testing. After that, the mixture is allowed to rest for different amounts of time to study changes in its performance properties. Following the completion of curing, the samples are subjected to testing for UCS through a triaxial setup to establish the soil's maximum capacity for compressive stress. The data

collected from these trials, including soil composition, NS levels, and length of curing, is compiled in a database. This database is used for developing deep-learning models to predict the UCS of NS-stabilized soils. The DL models are trained and validated using experimental data to ensure they are accurate and reliable. The final step involves assessing the results to determine the effectiveness of NS in enhancing soil properties, providing important insights into soil stabilization techniques for engineering applications.

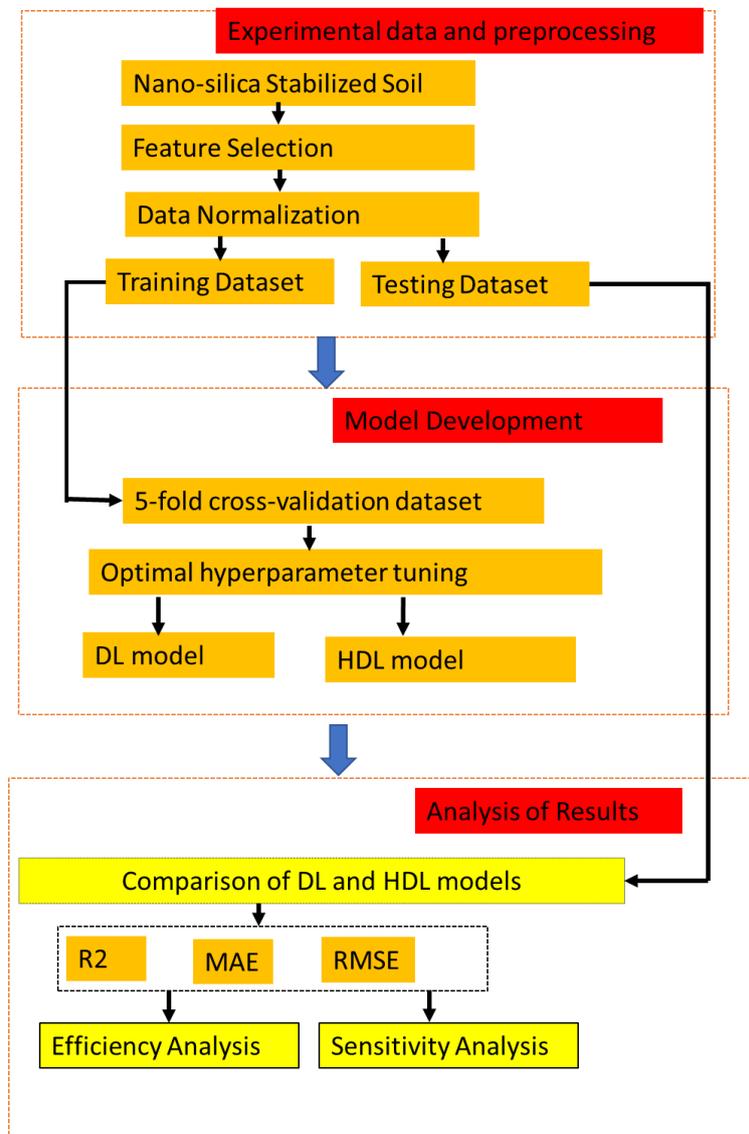


Figure 1. Methodology Flowchart.

3.1. Predictive Models

3.1.1. Convolution Neural Network (CNN)

CNN is a type of ANN known for its prominent feature of weight sharing [19]. Using weight sharing can significantly decrease the amount of weights, further lowering the complexity of the entire CNN. The CNN can take images as input and automatically extract data features using filters. Characteristics do not necessitate arduous and complex manual creation, and the entire process of feature representation is more automated. The CNN is composed of multiple network layers and each network layer has many independent neurons [20]. Figure 2 shows the operational structure of the CNN model. After the input image goes through the convolution operation, the feature extraction filter produces a feature map within the C1 layer. Afterwards, neural network operation is performed on feature map S1 by determining the connection strength between neuron layers and incorporating the appropriate bias. Ultimately, the sigmoid function produces a new feature mapping result called S2, which represents a deeper abstraction of the C1 layer. This cycle continues until ultimately, the outcome is fed into a classifier to generate a result.

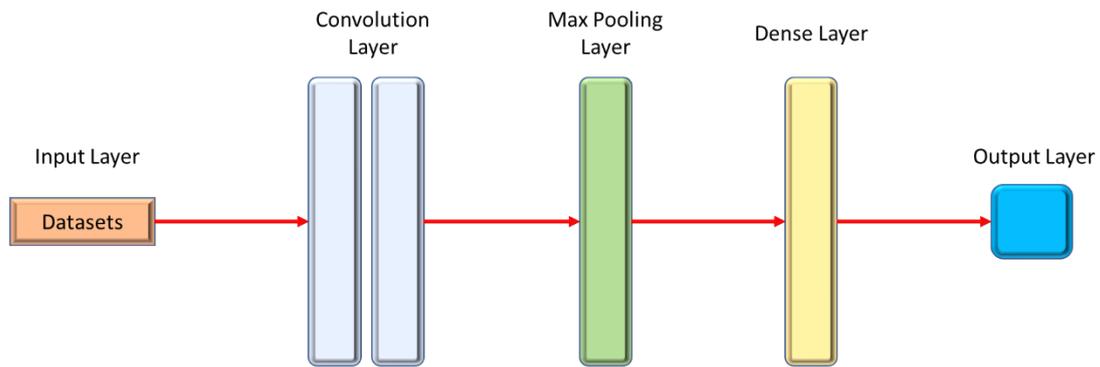


Figure 2. Architecture of CNN.

In a CNN structure, there are three key layers: the convolutional layer, the pooling layer, and the dense layer with rectified linear activation. The convolution operation is applied to combine every convolution layer (layers 1, 3, and 5) with the respective kernel size (3, 4, and 4). After each convolution layer, the feature maps are subjected to a max-pooling process. Max-pooling helps reduce the dimensions of the feature map. In this research, the stride for max-pooling and convolution is established at 2, with kernel size parameters determined through the brute force method. Layers 1, 3, 5, 7, and 8 utilize the (LeakyRelu) activation function. The final layer (layer 9) consists of 5 output neurons, 20 output neurons, and 30 output neurons within every fully connected layer. The hyperparameters for DL models are presented in Table 1. Google Colab library was used to run all the DL models.

Table 1. Hyperparameters for Deep learning models.

Model Type	Hyperparameter	Range (Tuned via Hyperband)
CNN	Filters (Conv1D)	32, 64, 96, 128
	Kernel Size	Min (3, input length)
	Pooling Size	1
	Dense Layer Units	64
	Optimizer	Adam
	Loss Function	MSE
	LSTM	LSTM Units
RNN	Dense Layer Units	64
	Optimizer	Adam
	Loss Function	MSE
	RNN Units	32, 64, 96, 128
Transformer	Dense Layer Units	64
	Optimizer	Adam
	Loss Function	MSE
	Number of Heads	4
	Key Dimension	32
	Layer Normalization	Yes
	Dense Layer Units	64
	Optimizer	Adam
	Loss Function	MSE

3.1.2. Long Short-Term Memory (LSTM)

LSTM is a specific type of RNN that addresses the issue of long-term memory storage that other RNNs struggle with [21]. Gates and memory lines are utilized in the LSTM to remember information from the beginning phases. LSTM is composed of a chain-like structure containing four neural networks and memory blocks called cells [22]. Figure 3 depicts the structure of LSTM. The information is stored in the cells and manipulation of memory is carried out by the gates. Three gates are present: input gate, forget gate, and output gate. The purpose of the forget gate is to eliminate information in the cell state that is no longer necessary. The input gate is responsible for incorporating new information into the cell state, while the output gates retrieve and present relevant information from the current cell.

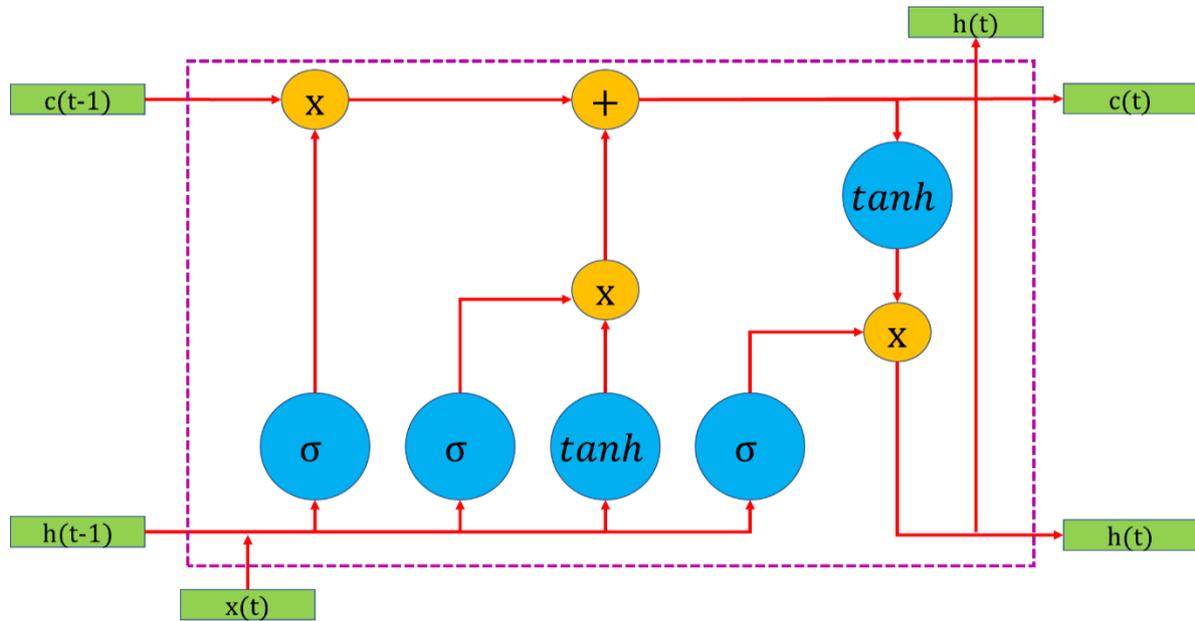


Figure 3. Architecture of LSTM.

The LSTM model includes two LSTM layers and a dense layer. There are 128 units in each LSTM layer, and the dense layer has 64 units. LSTM layers employ tanh and sigmoid activation functions, while ReLU is used by the dense layer and softmax is used by the output layer. The LSTM model is being trained with the RMSprop optimizer utilizing a learning rate of 0.05, a batch size of 50, and throughout 1000 epochs. This model is likewise developed using TensorFlow and Keras libraries in Google Colaboratory.

3.1.3. Recurrent Neural Network (RNN)

The RNN is considered one of the most effective algorithms for addressing sequential data tasks such as time series prediction, language translation, and speech recognition. It employs an inner condition [23]. The RNN architecture displayed in Figure 4 moves information from previous neurons to more recent ones. For example, in Equation (2). The most recent hidden state, h_t , is determined by both the most recent inputs, x_t , and the preceding hidden state, h_{t-1} . The most recent result, y_t , applies a linear transformation in Equation. (2) to extract time-related characteristics from h_t .

$$h_t = \tanh(Ux_t + Vh_{t-1} + b_h) \quad (1)$$

$$o_t = Wh_t + b_o \quad (2)$$

b_h and b_o represent the same biases for the entire sequence, while U , V , and W are the identical weights used. Activation functions like the sigmoid, ReLU, and tanh are employed to evaluate the importance of the input to the network. The RNN's $\tanh(x)$ function outputs the hyperbolic tangent of the input. If the input sequences are too long, RNNs may face issues with gradient vanishing or explosion [24].

The RNN model is comprised of two RNN layers and one dense layer. Each RNN layer contains 64 units, while the dense layer has 32 units. The RNN layers apply tanh for activation, the dense layer uses ReLU, and the output layer uses softmax. The RNN model is trained using the SGD optimizer with a learning rate of 0.05, a batch size of 80, and 1500 epochs.

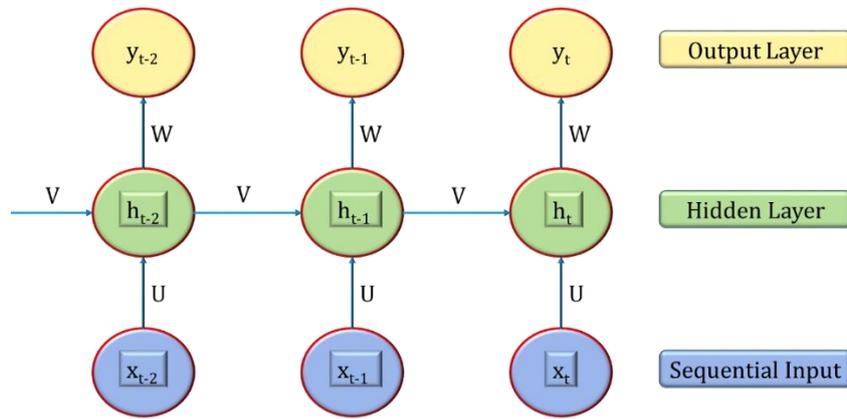


Figure 4. Schematic diagram of RNN architecture.

3.1.4. Transformer Model

The Transformer transformed NLP by addressing sequence-to-sequence tasks like machine translation and text generation without relying on recurrent structures. It uses a self-attention mechanism that improves training efficiency and performance [25]. The model has two main components: the encoder and decoder, each of multiple layers. The encoder processes input sequences into context-aware representations, while the decoder generates the output sequence. Each layer uses self-attention and feed-forward networks with residual connections and layer normalization, which enhance stability and training speed. The self-attention mechanism allows the model to focus on relevant parts of the input by computing relationships between words using query, key, and value vectors. The resulting weighted sum of value vectors produces the final word representation. The self-attention mechanism determines the relationships between words in a sequence by generating three distinct vectors for each word: query (Q), key (K), and value (V). The query vector of a word is compared against all key vectors through a dot product operation to compute similarity scores. These scores are then normalized to obtain weight coefficients, which are applied to the corresponding value vectors. The final word representation is obtained by computing the weighted sum of these value vectors. The mathematical formulation of the self-attention mechanism is given in Equation (3), and Figure 5 shows the schematic diagram of the transformer architecture.

$$Attention(Q, K, V) = \text{soft max}\left(\frac{QK}{\sqrt{d}}\right)V \tag{3}$$

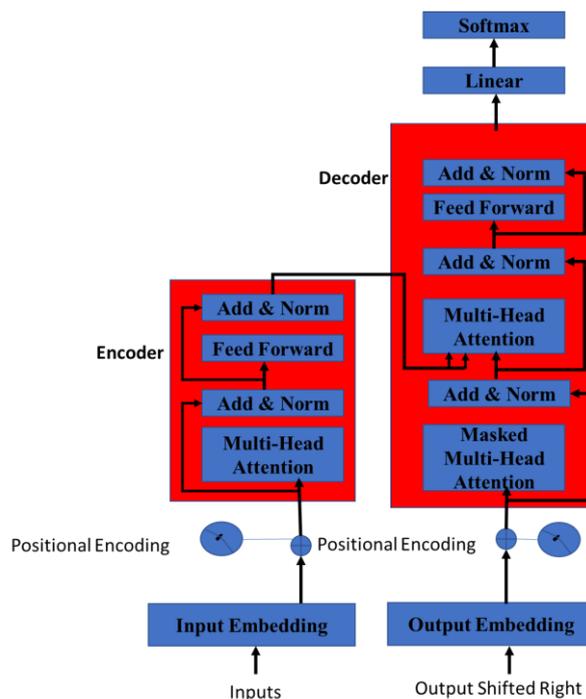


Figure 5. Schematic diagram of Transformer architecture.

4. Hybridization of Deep Learning Models

The Table 2 outlines the hyperparameters for three hybrid deep learning models: CNN-Transformer, LSTM-Transformer, and RNN-Transformer. Each model combines a different neural network type with a transformer to leverage the strengths of both. The CNN-Transformer uses convolutional filters, kernel sizes, and pooling to extract features, while the LSTM-Transformer and RNN-Transformer focus on sequence modeling with LSTM and RNN units, respectively. All models share common hyperparameters such as 64 units in the dense layer, the Adam optimizer [26], and MSE as the loss function. Hyperband is used to optimize these hyperparameters by testing different values to enhance model performance. These hybrid models aim to optimize learning and prediction accuracy for regression tasks. Figure 6 shows the hybridization of deep learning models with transformer.

Table 2. Hyperparameter for Hybrid Deep learning Models.

Model Type	Hyperparameter	Range (Tuned via Hyperband)
CNN-Transformer	Filters (Conv1D)	32, 64, 96, 128
	Kernel Size	Min (3, input length)
	Pooling Size	1
	Dense Layer Units	64
	Optimizer	Adam
	Loss Function	MSE
LSTM-Transformer	LSTM Units	32, 64, 96, 128
	Dense Layer Units	64
	Optimizer	Adam
	Loss Function	MSE
RNN-Transformer	RNN Units	32, 64, 96, 128
	Dense Layer Units	64
	Optimizer	Adam
	Loss Function	MSE

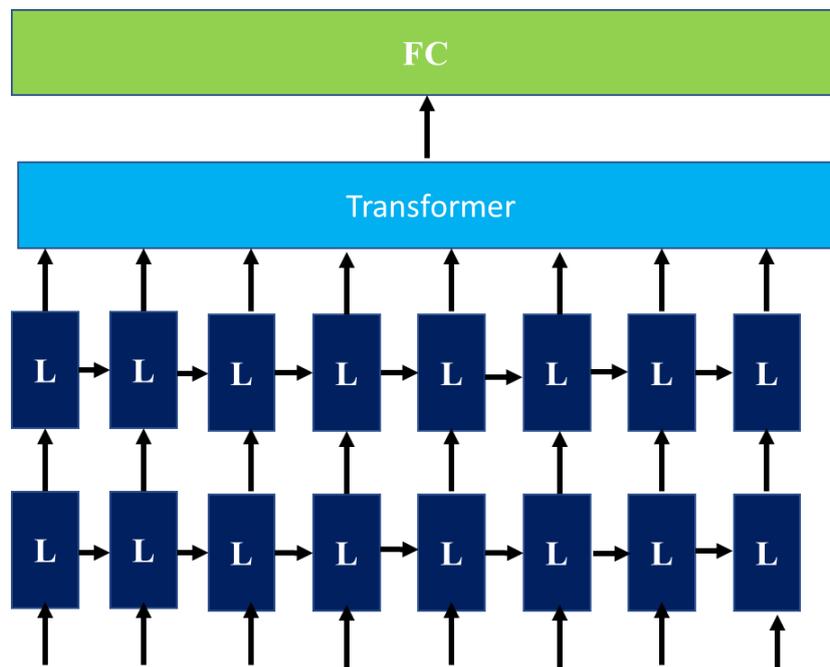


Figure 6. Hybridization of Deep Learning Models with Transformer.

5. Training and Validation of Models

5-Fold Cross-Validation (5K-Fold CV) is a robust technique used in machine learning to evaluate model performance while ensuring efficient data utilization. Instead of relying on a single train-test split, the dataset is divided into five equal subsets (folds). The model is trained and tested iteratively, where in each iteration, four folds are used for training, and the remaining fold is used for validation. This process repeats five times, with each fold serving as the validation set once. By averaging the performance metrics across all five iterations, the

evaluation becomes more reliable and less susceptible to variance caused by random data splits. This technique significantly reduces the risk of overfitting and ensures that every data point is used for both training and validation. It provides a more generalized assessment of model performance compared to traditional train-test splits. Additionally, shuffling the dataset before splitting ensures balanced folds, improving the model's learning process. Implementing 5K-Fold CV results in a stable and well-validated model that can generalize effectively to unseen data, making it an essential practice in deep learning and predictive modeling. Figure 7 shows the process of 5K-Fold CV used to train and validation of the deep learning and hybrid deep learning models in the study.

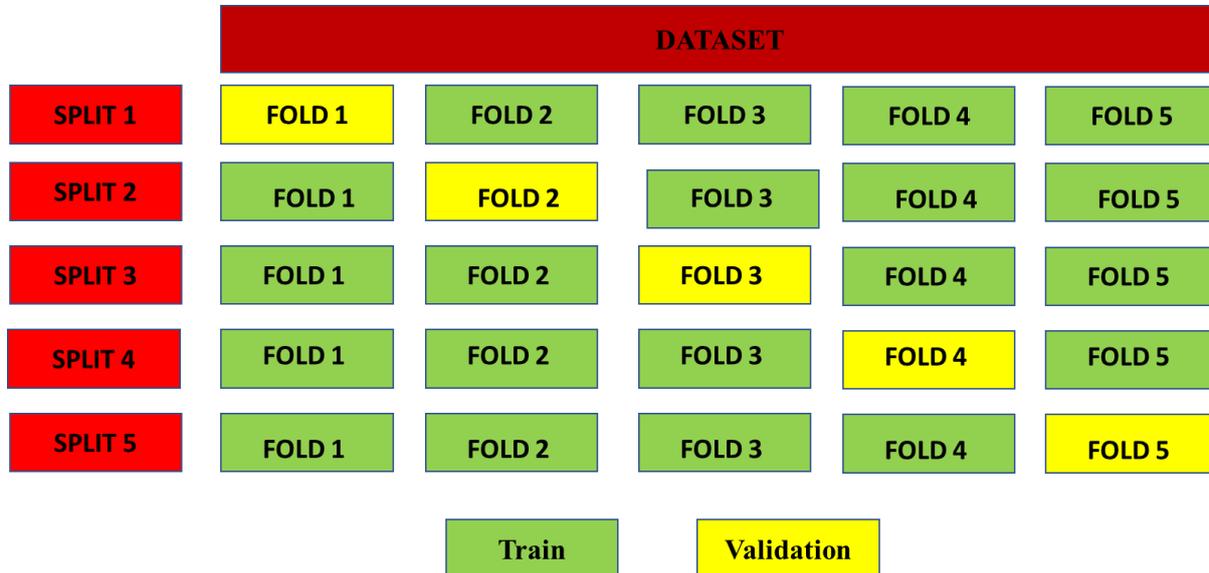


Figure 7. 5-Fold Cross-Validation used in the study.

6. Experimental Analysis

The experimental setup described in the section involves several key components aimed at investigating the durability aspects of NS-stabilized soil and developing a predictive model for UCS. A large-scale experiment was conducted to collect experimental data on the UCS using NS-stabilized soil. Figure 8 shows the tri-axial experiment setup for testing the UCS of NS stabilized fine-grained soils from the Indo-Gangetic Plain region and lesser Himalayan region. Similarly, Figure 9 shows the soil samples prepared for testing. The research utilized a triaxial arrangement to determine the UCS of NS stabilized soils to precisely assess the soil's maximum compressive stress capacity. The triaxial test arrangement enables precise simulation of in-situ stress conditions, ensuring reliable UCS measurements. The soil samples were evaluated in the unconsolidated undrained (UU) condition, which indicates that there was no consolidation or drainage. This ensures that the UCS measurements reflect the soil's instantaneous response to loading, which is essential for understanding how NS-stabilized soils behave under rapid load applications. The dimensions of the mould used for the preparation of the sample are 38mm in diameter and 78mm long. 509 datasets about the experiment have been collected from the testing. The parameters that affect the UCS are the soil index type (SI) for the various soil types, the nano-silica percentage (NS%), and the curing days (D). The soil index types are categorized as follows: CI = 1, MI = 2, and CL-ML = 3. Samples were prepared with the following NS dosages: 0.5%, 1.0%, 1.5%, 2.0%, 2.5%, 3.0%, 3.5%, and 4.0%.

The preparation of samples initiates with the mixing of NS particles with soil with the use of a mechanical stirrer, forming samples with varying concentrations of NS. The water was mixed with each type of soil based on Optimum Moisture Content (OMC) before shaping the mixture into a cylinder using the triaxial testing mould. The mechanical properties of these specimens are monitored over time by testing them at various curing periods including 0, 3, 7, 14, 28, 56, and 90 days. Figure 10 shows the samples cured for different durations before testing. The soil samples are stored in a desiccator to preserve their moisture levels, ensuring consistency throughout the hardening process. This method maintains a consistent level of humidity, preventing moisture loss and offering uniform curing conditions. By reducing outside factors that could impact the soil's moisture, desiccating the specimen ensures a reliable and precise assessment of the soil's properties. The process of curing is essential for the complete development of chemical reactions between the NS and soil particles, enhancing the structural integrity of the soil. The compressive tests involve axial compression of cylindrical soil specimens in a triaxial setup until failure, with the maximum compressive stress being recorded as the UCS. The cracking shown in Figure

10 in samples that experienced ageing impacted the soil samples' strength. Cracks weaken the soil's structural integrity, causing a decrease in UCS. Table 3 shows the soil characterization of the tested soil samples from the Indo Gangetic plains and the lesser Himalayan regions.

Table 3. Classification of the Soil Types.

Parameters	SI (1)	SI (2)	SI (3)
w_L	42.48	31.27	35.24
w_P	18.89	24.79	26.56
I_P	23.59	6.48	8.68
MDD	14.98	18.66	19.22
OMC	22.24	15.74	18.52
Sand (%)	7.66%	38.76%	11.26%
Silt (%)	29.42%	27.62%	82.78%
Clay (%)	62.92%	33.62%	5.96%
Cu	8.86	262.18	7.56
Cc	0.84	0.647	1.92
Unified Soil Classification System (USCS)	CI	CL-ML	MI

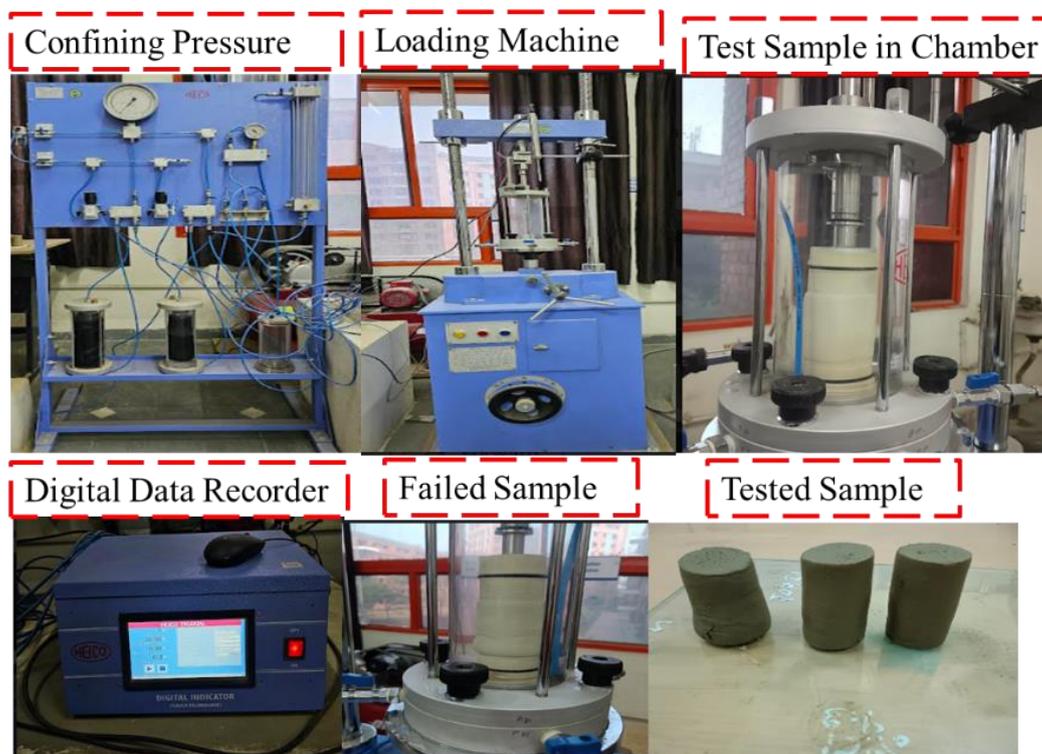
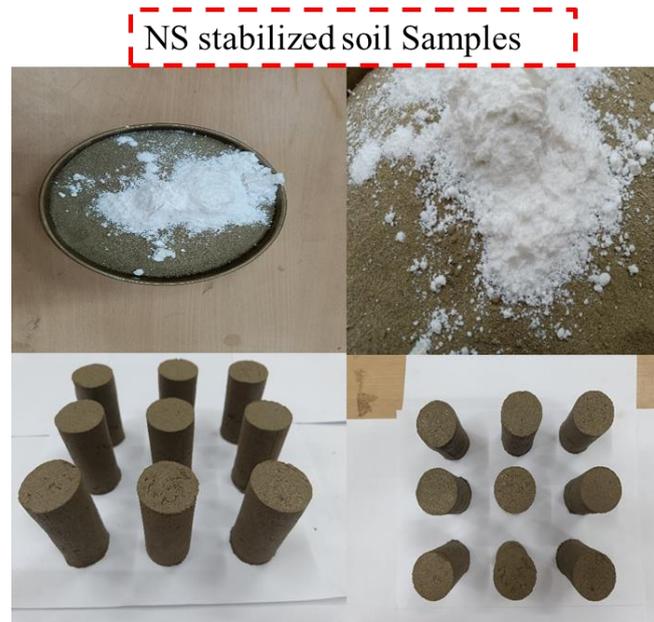


Figure 8. Experimental setup for the study.

Table 4 shows the standard deviation for each percentage of NS in a particular curing period. The standard deviations reflect the variation in UCS measurements with different NS content and curing periods. For instance, in the case of a 0.5% NS solution, the variability increases over time, with the standard deviation ranging from 0.44 at the beginning to 0.74 after 90 days of curing. In the same way, when NS percentages are higher at 3.5% and 4.0%, the initial standard deviations are lower at 0.17 and 0.11, but they rise to 0.44 and 0.39 after 90 days. This pattern indicates that although a higher number of NS at first improves soil stability, extended curing results in unpredictability caused by cracks forming and spreading. In general, the existence of cracks harms the uniformity and dependability of the soil's strength evaluations. Throughout the experimental process, comprehensive data collection is conducted, encompassing parameters such as soil type characteristics, NS concentration, and curing duration. These data facilitate a thorough understanding of the behaviour and effectiveness of Nano-silica stabilization over time. Figure 11 shows the sample after the completion of the triaxial test.

Table 4. Standard deviation for each percentage in a particular curing period.

NS (%)	0 D	3 D	7 D	14 D	28 D	56 D	90 D
0.5	0.44	0.49	0.54	0.59	0.64	0.69	0.74
1	0.41	0.44	0.49	0.54	0.59	0.64	0.69
1.5	0.36	0.39	0.44	0.49	0.54	0.59	0.64
2	0.32	0.34	0.39	0.44	0.49	0.54	0.59
2.5	0.24	0.29	0.34	0.39	0.44	0.49	0.54
3	0.21	0.24	0.29	0.34	0.39	0.44	0.49
3.5	0.17	0.19	0.24	0.29	0.34	0.39	0.44
4	0.11	0.14	0.19	0.24	0.29	0.34	0.39

**Figure 9.** NS stabilized soil samples for testing.**Figure 10.** Testing samples cured for different days.

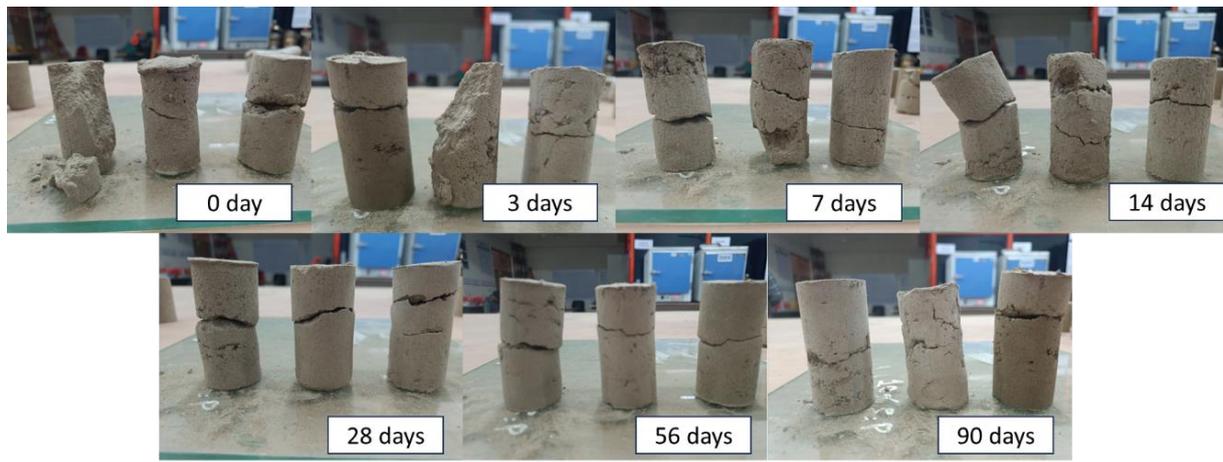
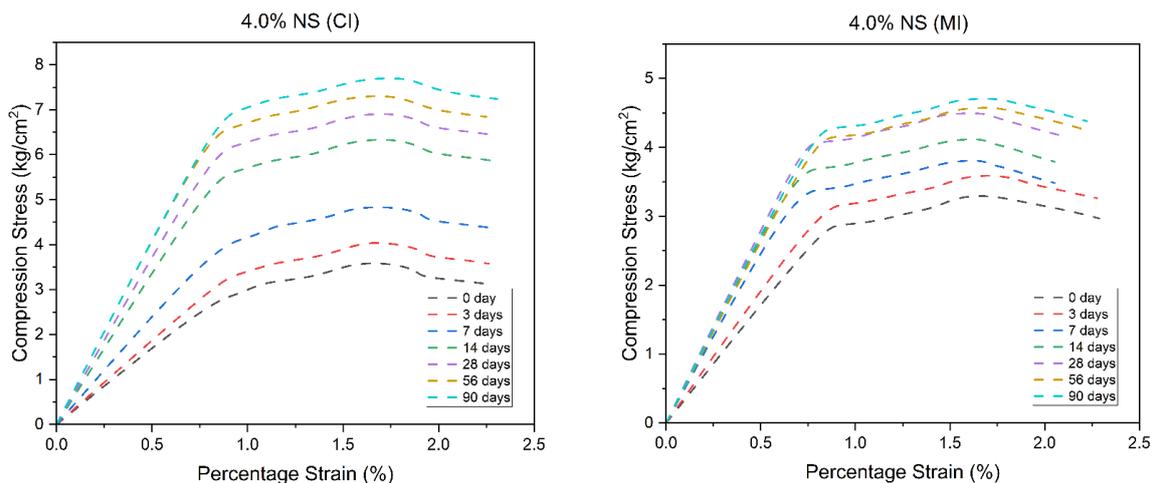


Figure 11. Tested samples after various days of curing.

The stress-strain relationship of fine-grained soil treated with a 4% NS stabilizer, after it was allowed to cure for 0 days, 3 days, 7 days, 14 days, 28 days, 56 days, and 90 days, and then loaded triaxially is shown in Figure 12. Figure 12 stress-strain curves of soil treated with NS exhibiting a gradual decrease in stress after reaching the peak, which can be explained by studying the microstructural and bonding properties influenced by the NS treatment. The large surface area of NS particles enables strong bonds to form with soil particles, resulting in an initial strength boost. Nevertheless, as soon as the maximum strength is achieved, these resilient bonds start to gradually deteriorate. This gradual weakening of bonds leads to a slow decrease in tension instead of a sudden decrease, showing a more flexible way of failing. This occurrence is consistent with prior research, showing that NS enhances early strength but impacts later behavior as bonds break gradually. The application of NS leads to a more compact soil matrix with an improved pore arrangement, increasing the initial resistance and robustness. The energy needed for further deformation decreases more slowly in treated soil compared to untreated soil as it deforms. The strength of clay soil is improved by the addition of NS and white cement, resulting in a more ductile failure after reaching peak strength due to their combined effects. The slow decrease in stress after reaching its highest point reflects our results, indicating that the soil structure stabilizes as its bonds weaken [27]. The improved resistance to compression and a gradual decrease in strength after reaching the peak in soils treated with NS, attributed to altered failure modes [28]. This conduct, linked to gradual bond rupture and increased malleability, is consistent with what the study has presented.



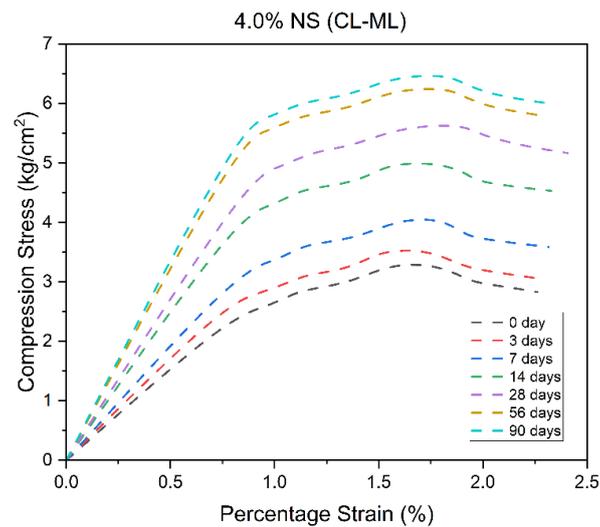


Figure 12. UCS of NS stabilized soil for various soil types.

7. Data Preprocessing

7.1. Data Analysis

This dataset in Table 5 provides a comprehensive statistical summary of four key parameters related to soil stabilization: Soil Index (SI), Nano Silica Content (NS%), Curing Days (D), and Unconfined Compressive Strength (UCS, MPa), based on 507 soil samples. The Soil Index (SI), categorized as CI = 1, CL-ML = 2, and MI = 3. The Nano Silica (NS%) content ranges from 0% to 4%, with an average of 2.24%, suggesting a balanced distribution. The Curing Days (D) exhibit high variability, ranging from 0 to 90 days, with a mean of 28.12 days, highlighting the diversity in sample preparation times. UCS, a crucial measure of soil strength, has a mean of 3.03 kg/cm² but varies significantly, reaching up to 7.97 kg/cm², indicating the potential influence of NS and curing duration on strength enhancement. The dataset's statistical properties further reveal important insights into variability and distribution. SI has low variance (0.67) and no skewness, confirming a uniform distribution across soil types. NS (%) is almost symmetrically distributed (−0.01 skewness), while curing days (D) and UCS exhibit right-skewed distributions (1.01 and 1.09, respectively), indicating that a few samples with high values significantly impact the dataset. The kurtosis values further illustrate distribution characteristics: SI (−1.50) and NS% (−1.22) exhibit flatter distributions, while UCS (1.24) has a more peaked shape, meaning most values cluster around the mean with some extreme cases. The high variance in curing days (954.29) and UCS (1.79) suggests a wide range of curing conditions and soil strengths, emphasizing the role of stabilization techniques. Overall, this dataset provides crucial insights into the behavior of NS-stabilized soil, where curing duration and NS content play a significant role in strength enhancement. These trends indicate the need for predictive modeling to establish relationships between these parameters and optimize soil stabilization strategies.

Table 5. Consistency of the dataset in the study.

Parameters	SI	NS (%)	D (days)	UCS (kg/cm ²)
count	507.00	507.00	507.00	507.00
mean	2.00	2.24	28.12	3.03
std	0.82	1.16	30.89	1.34
min	1.00	0.00	0.00	0.58
25%	1.00	1.00	3.00	2.08
50%	2.00	2.00	14.00	2.74
75%	3.00	3.00	56.00	3.70
max	3.00	4.00	90.00	7.97
variance	0.67	1.34	954.29	1.79
skewness	0.00	−0.01	1.01	1.09
kurtosis	−1.50	−1.22	−0.40	1.24

Figure 13 is a pairplot (scatterplot matrix) that visually represents the relationships between four key geotechnical parameters: Soil Index (SI), Nano Silica Content (NS%), Curing Days (D), and Unconfined Compressive Strength (UCS, kg/cm²). The diagonal plots display the distribution of each parameter through histograms overlaid with kernel density estimation (KDE) curves, while the off-diagonal scatterplots illustrate bivariate relationships, with red ellipses highlighting significant trends and correlations. The distribution of SI shows three distinct peaks, reflecting its categorical nature (CI = 1, CL-ML = 2, MI = 3). Nano-silica (NS%) appears uniformly distributed, indicating controlled experimental conditions. Curing Days (D) follows a right-skewed pattern, with more samples at lower durations but extending up to 90 days. UCS (kg/cm²) exhibits a slightly right-skewed distribution, suggesting that most soil samples achieve strengths between 2–4 kg/cm², with some reaching as high as 7.5 kg/cm². The scatterplots reveal key trends: SI and UCS exhibit a negative correlation, indicating that higher SI values generally lead to lower strength. NS (%) and UCS show a strong positive correlation, confirming that increased NS content enhances soil strength. Similarly, curing duration (D) positively influences UCS, reinforcing the role of curing time in soil stabilization. The relationship between NS (%) and D displays a structured pattern, reflecting a well-designed experimental setup. However, SI vs. NS (%) and SI vs. D show no apparent correlation, suggesting that soil type does not dictate NS content or curing time. Overall, this pairplot provides crucial insights into the behavior of NS-stabilized soil, confirming that NS content and curing duration significantly influence UCS, while soil index plays an inverse role in strength development. These findings emphasize the potential for predictive modeling to optimize soil stabilization strategies.

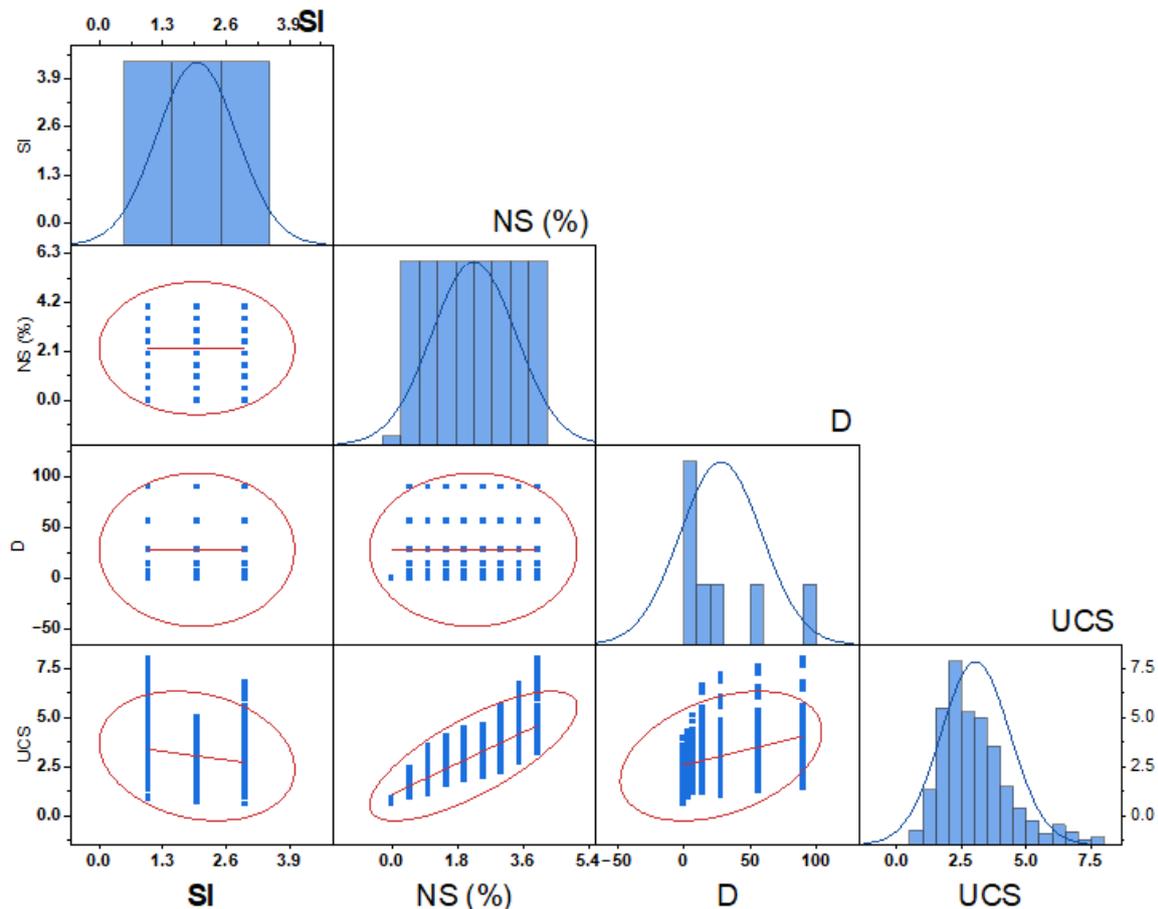


Figure 13. Scatter Matrix Plot of variables.

The correlation heatmap in Figure 14 provides a comprehensive visual representation of the relationships between four key geotechnical parameters: Soil Index (SI), Nano Silica Content (NS%), Curing Days (D), and Unconfined Compressive Strength (UCS, kg/cm²). The color intensity and numerical values indicate the strength and direction of correlations, ranging from -1 (strong negative correlation) to $+1$ (strong positive correlation). This analysis reveals crucial trends influencing soil strength and stabilization. The most significant observation is the strong positive correlation (0.76) between NS% and UCS, indicating that an increase in NS content leads to a substantial improvement in unconfined compressive strength. This confirms the vital role of nano-silica in soil stabilization, making it the most influential factor in enhancing soil strength.

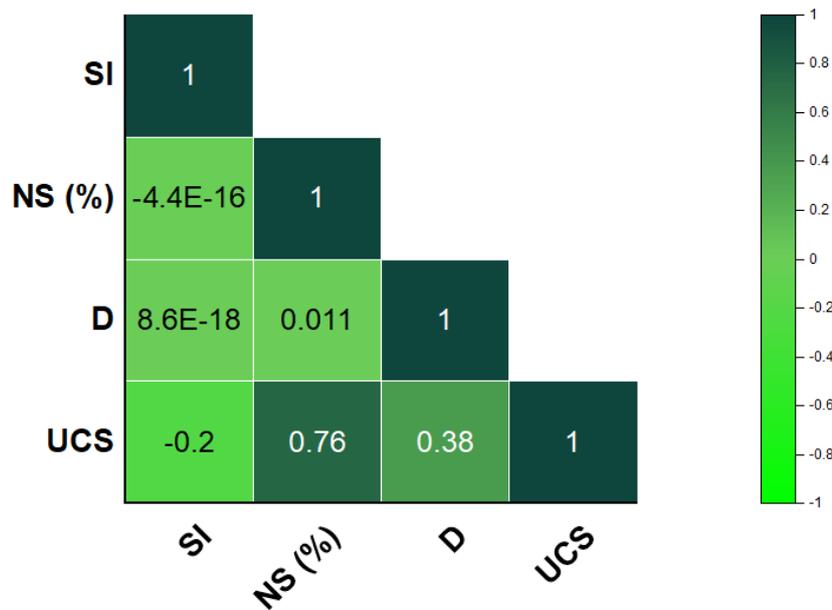


Figure 14. Relationship between the variables.

Additionally, a moderate positive correlation (0.38) exists between curing days (D) and UCS, demonstrating that extended curing periods contribute to strength development, though its effect is secondary compared to NS. Conversely, the negative correlation (-0.2) between SI and UCS suggests that soil types with higher plasticity or fine-grained compositions tend to exhibit lower UCS values. This could be attributed to the inherent mechanical behavior of different soil classifications, where finer soils generally possess lower shear strength. Furthermore, the near-zero correlation (0.011) between NS% and D indicates that NS content and curing time were independently controlled in the experiments, reinforcing the reliability of the dataset for predictive modeling. Similarly, SI and NS% show no correlation (-4.4×10^{-16}), confirming that the soil type does not inherently influence the amount of NS used. In summary, this heatmap highlights NS as the dominant factor driving UCS improvement, followed by curing time, while soil type plays a minor inverse role. These insights provide a strong foundation for optimizing NS dosage and curing conditions to enhance soil stabilization efficiency. The structured nature of the dataset ensures its suitability for advanced predictive modeling, enabling the development of innovative geotechnical solutions.

7.2. SHAP and Feature Dependency

Figures 15 and 16 collectively offer a comprehensive understanding of the relative importance and individual impact of the features NS (%), D, and SI on the model's predictive behavior. Figure 15 illustrates the permutation feature importance, which evaluates the decline in model performance when each feature's values are randomly shuffled. Among the three features, NS (%) shows the highest permutation importance, indicating it is the most influential factor in the model's predictions. This means that altering the values of NS (%) significantly disrupts model accuracy, underscoring its dominant role. D ranks second in importance, contributing meaningfully to the model but with less influence than NS (%), while SI has the least importance, suggesting a relatively minor impact on overall model performance. The error bars in the plot further reveal that NS (%) has a consistent and stable importance across different shuffling runs, enhancing the reliability of its significance.

In contrast, Figure 16 presents the SHAP (SHapley Additive exPlanations) summary plot, which offers a granular, instance-level explanation of how each feature contributes to individual model outputs. The x-axis represents the SHAP value, signifying the direction and magnitude of the feature's impact, while the color gradient from blue to red indicates low to high feature values. NS (%) again emerges as the most impactful variable, showing a broad range of SHAP values extending from negative to strongly positive. This suggests that higher values of NS (%) consistently push the model output upward, revealing a strong positive relationship. D also shows a positive correlation with the output, though with a narrower SHAP range, indicating a moderate influence. Interestingly, SI displays a relatively compact SHAP distribution, with an inverse relationship where lower values of SI are associated with increased model outputs, while higher SI values tend to reduce them.

Together, these Figures 15 and 16 reinforce a consistent ranking of feature influence: NS (%) as the most critical, followed by D and then SI. The permutation importance quantifies each feature's contribution to the

model’s overall performance, while the SHAP plot provides detailed insights into the direction and magnitude of influence on a per-instance basis. This dual perspective not only confirms the predictive dominance of NS (%) but also uncovers the nuanced roles of D and SI, providing valuable guidance for both interpretation and future experimental design.

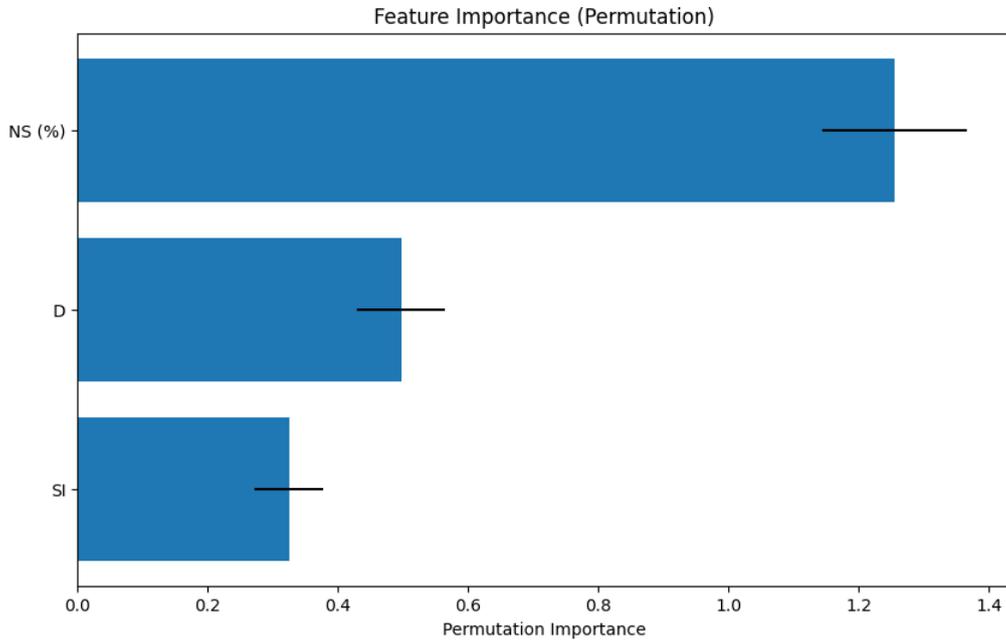


Figure 15. Feature importance.

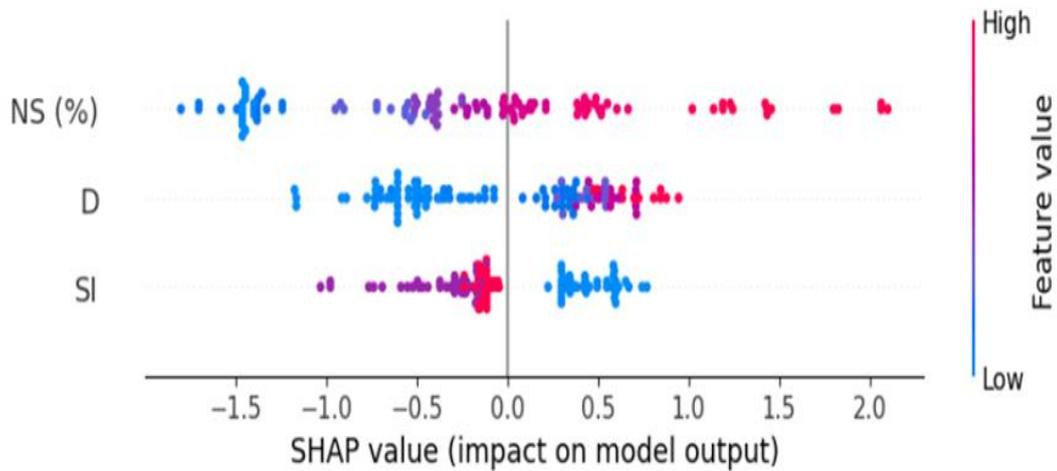


Figure 16. SHAP value.

7.2. Performance Metrics

This study evaluates the computational efficiency deep learning models and hybrid deep learning models using three metrics R^2 , $RMSE$ and MAE [29–31].

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \tag{4}$$

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \tag{5}$$

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (6)$$

8. Discussion of Results

8.1. Performance of Deep Learning Models

The performance analysis of three deep learning models CNN, LSTM, and RNN was conducted using 5-Fold Cross-Validation to evaluate their predictive accuracy, error margins, and generalization capability as shown in Table 6. The key performance metrics considered include Root Mean Squared Error (*RMSE*), Mean Absolute Error (*MAE*), and R^2 Score for both training and validation datasets. Among the models, the CNN demonstrated the strongest and most consistent performance. It exhibited low *RMSE* and *MAE* values, indicating accurate predictions, with R^2 scores consistently above 0.86, suggesting a good fit. The best performance in validation occurred in Fold 3, with *RMSE* of 0.39, *MAE* of 0.30, and R^2 of 0.88, while the worst performance was observed in Fold 4, where *RMSE* increased to 0.51, though R^2 still maintained a reasonable 0.86, indicating stable generalization with minimal overfitting. The LSTM model, on the other hand, struggled with higher *RMSE* and lower R^2 scores compared to both CNN and RNN, suggesting challenges in effectively capturing the dataset's complex patterns. The best performance for LSTM was seen in Fold 5, with *RMSE* of 0.53, *MAE* of 0.42, and R^2 of 0.85, while the worst performance was in Fold 4, with *RMSE* of 0.62 and R^2 of 0.79, signaling potential overfitting or underfitting. The RNN model, while slightly underperforming compared to CNN, achieved stronger results than LSTM, with the best validation performance seen in Fold 1, showing *RMSE* of 0.38, *MAE* of 0.30, and R^2 of 0.91, indicating excellent generalization. The worst performance for RNN occurred in Fold 4, where *RMSE* increased to 0.57 and R^2 dropped to 0.82, reflecting a slight decline in accuracy. Overall, CNN outperformed both RNN and LSTM, maintaining consistently low errors and high R^2 values across all folds. RNN performed well but was slightly less stable than CNN, while LSTM struggled with higher error rates and lower R^2 values, suggesting the need for further tuning or dataset adjustments. Thus, the CNN model proved to be the most reliable choice for this predictive task, with RNN showing promise but requiring further optimization.

Table 6. Performance of the deep learning Models.

Model	Fold	<i>RMSE</i> (Train)	<i>MAE</i> (Train)	R^2 (Train)	<i>RMSE</i> (Validation)	<i>MAE</i> (Validation)	R^2 (Validation)
CNN	1.00	0.40	0.32	0.91	0.39	0.31	0.91
	2.00	0.41	0.31	0.90	0.42	0.32	0.92
	3.00	0.40	0.31	0.92	0.39	0.30	0.88
	4.00	0.38	0.29	0.92	0.51	0.38	0.86
	5.00	0.43	0.33	0.90	0.43	0.34	0.90
LSTM	1.00	0.54	0.43	0.84	0.57	0.47	0.80
	2.00	0.58	0.47	0.79	0.57	0.45	0.86
	3.00	0.56	0.44	0.84	0.49	0.37	0.80
	4.00	0.51	0.39	0.85	0.62	0.50	0.79
	5.00	0.52	0.41	0.85	0.53	0.42	0.85
RNN	1.00	0.37	0.29	0.92	0.38	0.30	0.91
	2.00	0.44	0.33	0.88	0.42	0.31	0.92
	3.00	0.49	0.37	0.88	0.41	0.33	0.87
	4.00	0.46	0.35	0.88	0.57	0.43	0.82
	5.00	0.43	0.34	0.90	0.43	0.36	0.90

8.2. Performance of Hybrid-Deep Learning Models

The Table 7 presents the results of 5-Fold Cross-Validation for three different hybrid deep learning models: CNN-Transformer, LSTM-Transformer, and RNN-Transformer. Each model was evaluated across five folds, and key performance metrics were recorded, including *RMSE*, *MAE*, and R^2 Score for both the training and validation sets. These metrics provide insight into how well each model learns from the data and generalizes to unseen samples.

The CNN-Transformer model demonstrates strong predictive performance, with low *RMSE* and *MAE* values and a high R^2 score consistently above 0.96 in both training and validation. The *RMSE* for validation ranges

between 0.199 and 0.268, while *MAE* remains relatively low, indicating precise predictions. The model achieves its best validation performance in Fold 3, with an *RMSE* of 0.199 and an R^2 score of 0.967, suggesting excellent generalization capability. The LSTM-Transformer model shows a moderate decline in performance compared to CNN-Transformer, as indicated by its slightly higher *RMSE* and lower R^2 values. The *RMSE* values in validate the model is still effective, its higher error margins and lower R^2 values suggest that it may struggle with certain patterns in the data, particularly in Fold 4, where overfitting might have occurred. The RNN-Transformer model performs better than LSTM-Transformer but slightly worse than CNN-Transformer. It maintains reasonably low *RMSE* values in both training and validation, with validation *RMSE* ranging from 0.205 to 0.323. The R^2 values consistently remain above 0.94, indicating strong predictive power. Notably, Fold 3 achieves the best validation performance with $RMSE = 0.205$ and $R^2 = 0.966$, showing that the model effectively captures data patterns in certain cases. However, Fold 4 exhibits a performance dip ($RMSE = 0.323$, $R^2 = 0.943$), suggesting some variance in model generalization. From this analysis, the CNN-Transformer model emerges as the best-performing model, with higher accuracy, lower error rates, and better generalization across different folds. While the RNN-Transformer performs reasonably well, the LSTM-Transformer struggles in certain cases, particularly in Fold 4, where it records the highest *RMSE* (0.465) and lowest R^2 (0.882). These results suggest that CNN-based hybrid models may be more effective for this type of predictive task compared to recurrent architectures.

Table 7. Performance of the hybrid deep learning Models.

Model	Fold	<i>RMSE</i> (Train)	<i>MAE</i> (Train)	R^2 (Train)	<i>RMSE</i> (Validation)	<i>MAE</i> (Validation)	R^2 (Validation)
CNN-Transformer	1	0.238	0.192	0.969	0.254	0.197	0.961
	2	0.202	0.161	0.975	0.254	0.214	0.972
	3	0.188	0.160	0.982	0.199	0.167	0.967
	4	0.190	0.151	0.980	0.268	0.206	0.961
	5	0.192	0.160	0.979	0.212	0.182	0.975
LSTM-Transformer	1	0.342	0.272	0.936	0.388	0.294	0.908
	2	0.315	0.249	0.939	0.355	0.291	0.944
	3	0.315	0.250	0.948	0.316	0.242	0.918
	4	0.373	0.287	0.921	0.465	0.364	0.882
	5	0.401	0.316	0.910	0.419	0.334	0.903
RNN-Transformer	1	0.199	0.159	0.978	0.251	0.195	0.961
	2	0.224	0.185	0.969	0.277	0.236	0.966
	3	0.202	0.164	0.979	0.205	0.157	0.966
	4	0.249	0.194	0.965	0.323	0.256	0.943
	5	0.282	0.234	0.955	0.282	0.232	0.956

8.3. Box-plot Comparison of Models

This section presents a box plot comparison of the deep learning models and the hybrid deep learning models. Figure 17 shows the comparison of deep learning models R^2 , *RMSE*, and *MAE*, respectively. As shown in Figure 17a, the CNN and RNN models achieved higher R^2 values (around 0.90–0.92), indicating better predictive accuracy compared to the LSTM model, which recorded lower values (0.80–0.85). Figure 17b reveals that CNN and RNN had lower *RMSE* (0.40–0.45), suggesting less prediction error, whereas LSTM showed significantly higher *RMSE* (0.50–0.60). Similarly, Figure 17c shows that CNN and RNN achieved lower *MAE* (0.31–0.35), while LSTM again exhibited the highest error (0.45–0.50). Overall, CNN demonstrated the best and most consistent performance, followed by RNN, with LSTM performing the poorest across all metrics.

Similarly, Figure 18 shows the comparison of hybrid-deep learning models R^2 , *RMSE*, and *MAE*. In Figure 18a, CNN-Transformer achieved the highest R^2 (≈ 0.97), followed by RNN-Transformer (≈ 0.96), while LSTM-Transformer had the lowest (≈ 0.91) with more variance. In Figure 18b, CNN-Transformer had the lowest *MAE* (≈ 0.18), RNN-Transformer followed (≈ 0.23), and LSTM-Transformer had the highest *MAE* (≈ 0.30). In Figure 18c, CNN-Transformer again performed best with the lowest *RMSE* (≈ 0.22), followed by RNN-Transformer (≈ 0.27), while LSTM-Transformer showed the highest *RMSE* (≈ 0.37). Overall, CNN-Transformer showed superior accuracy and consistency.

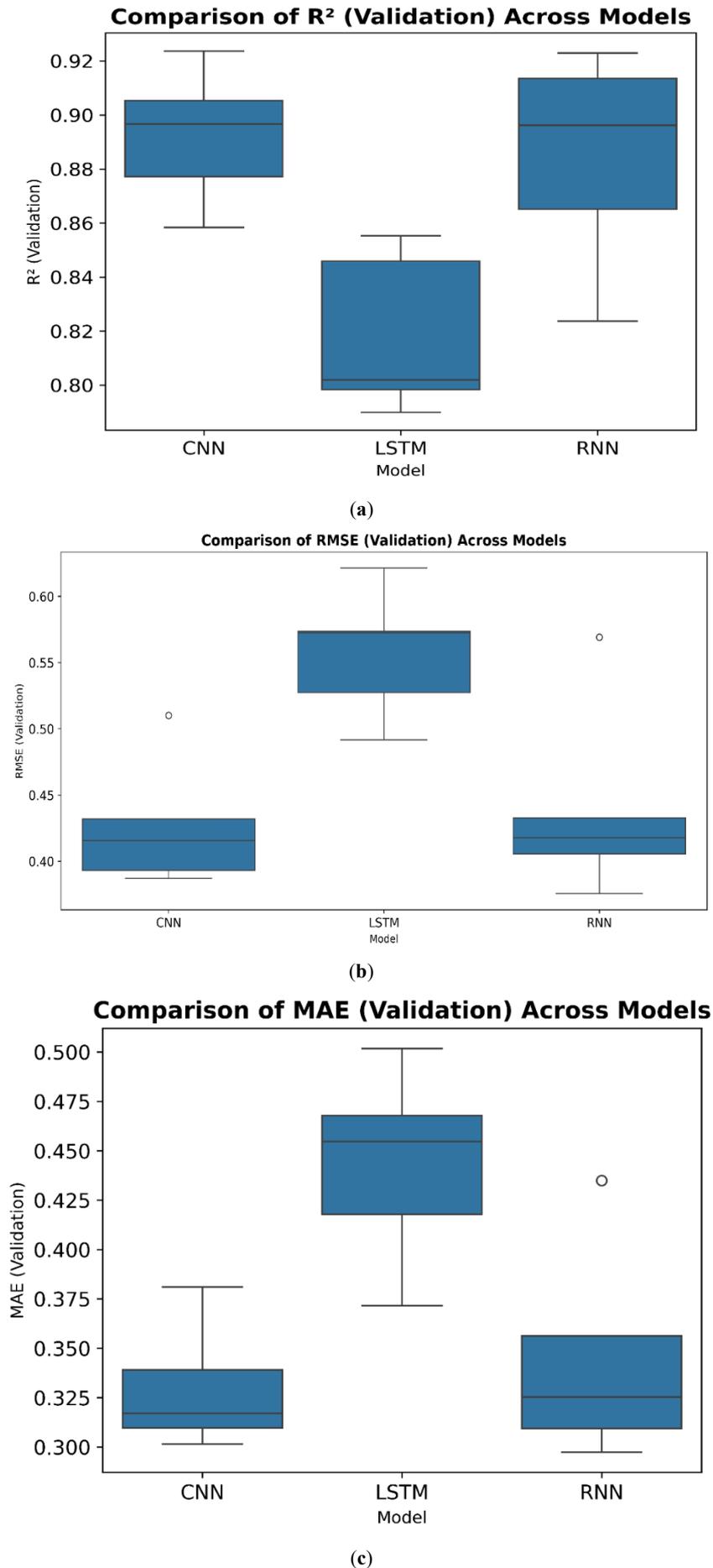
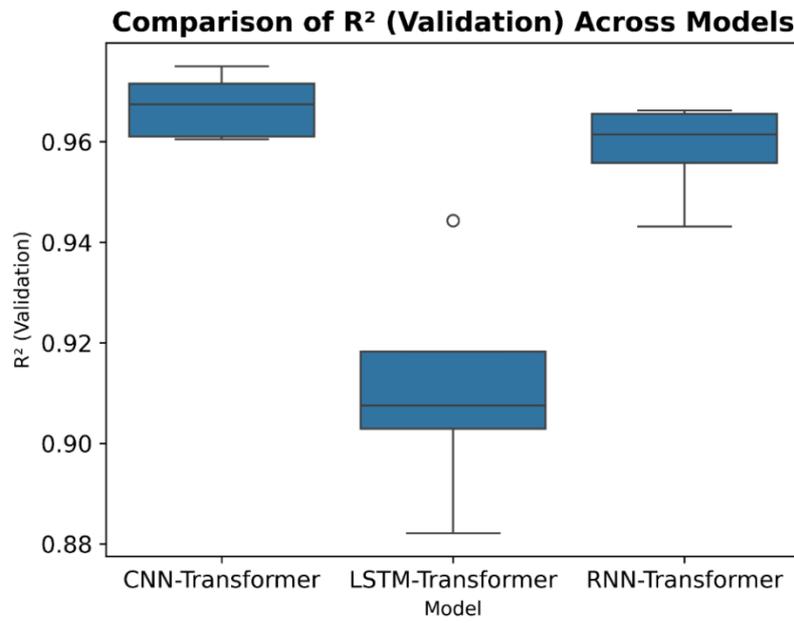
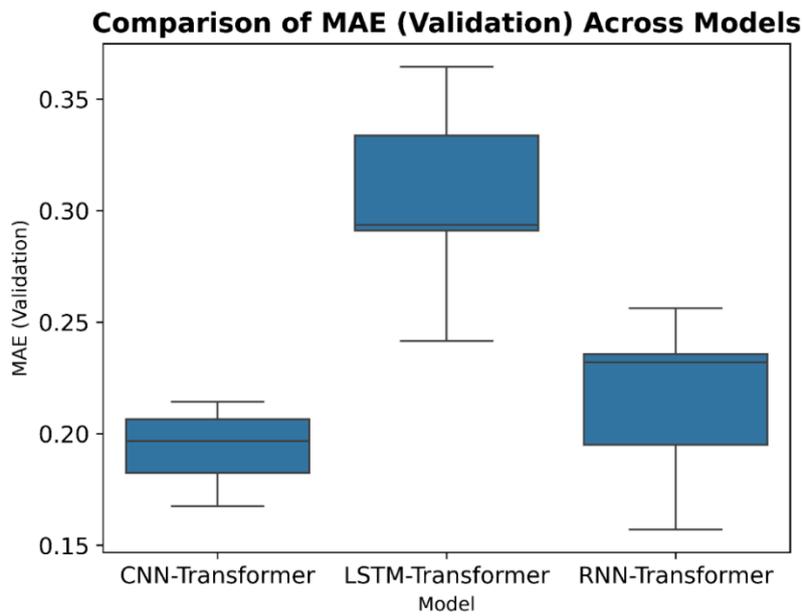


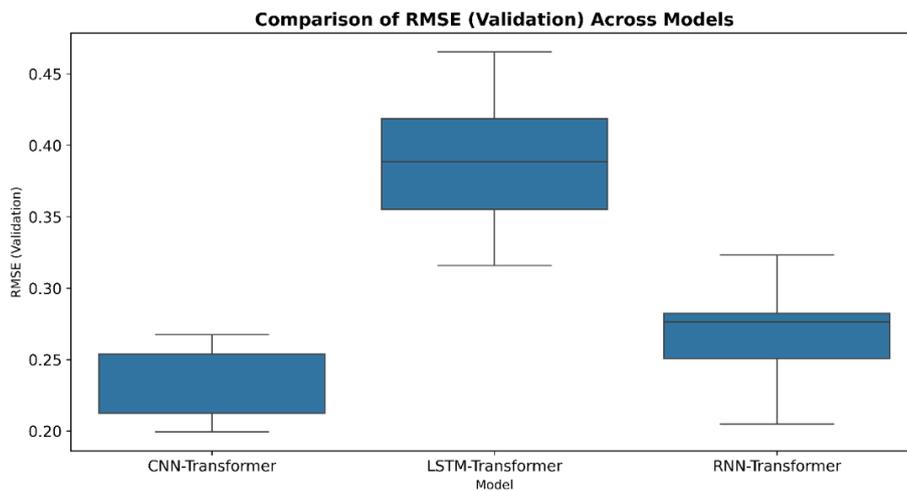
Figure 17. Comparison of the Deep-learning Models.



(a)



(b)



(c)

Figure 18. Comparison of the Hybrid Deep-learning Models.

8.4. Performance Efficiency of Hybridization

To analyze the performance improvement, the percentage increase in R^2 and the percentage decrease in RMSE and MAE between the deep learning models (CNN, LSTM, RNN) and the hybrid models (CNN-Transformer, LSTM-Transformer, RNN-Transformer) for both training and validation sets. The Equations (7)–(9) measure the performance efficiency of the hybrid models with respect R^2 , $RMSE$ and MAE respectively.

$$\%Increase = \left(\frac{Hybrid R^2 - Base R^2}{Base R^2} \right) \times 100 \quad (7)$$

$$\%Increase = \left(\frac{Hybrid RMSE - Base RMSE}{Base RMSE} \right) \times 100 \quad (8)$$

$$\%Increase = \left(\frac{Hybrid MAE - Base MAE}{Base MAE} \right) \times 100 \quad (9)$$

The notable improvement in performance observed in hybrid deep learning models (such as CNN-Transformer, LSTM-Transformer, and RNN-Transformer) can be attributed to both the nature of the dataset and the architectural synergy of the combined models. The dataset exhibits temporal patterns, nonlinear relationships, and localized features, which are effectively captured when spatial extraction capabilities of models like CNN or sequential learning strengths of LSTM/RNN are fused with the self-attention mechanism of the Transformer. This hybridization enhances feature representation, allows for better long-range dependency modeling, and reduces overfitting, thereby achieving lower errors and higher generalization on validation folds.

The Table 8 presents the percentage change in performance metrics due to the hybridization of deep learning models with Transformers. A negative percentage in $RMSE$ and MAE indicates error reduction, while a positive percentage in R^2 suggests improved predictive performance. Among the models, the LSTM-Transformer hybrid consistently shows the highest improvement, with $RMSE$ reductions of up to -36.80% in training (Fold 1) and -32.30% in validation (Fold 1). Similarly, its MAE reduces by -36.20% in training and -37.26% in validation (Fold 1), while its R^2 improves by up to 13.68% in validation (Fold 1) and 18.51% in training (Fold 2). The CNN-Transformer model shows moderate improvements but with more variability across folds. The $RMSE$ reduction ranges from -41.19% (Fold 1) to -55.04% (Fold 5) in training, and from -35.43% (Fold 1) to -50.83% (Fold 5) in validation. MAE follows a similar trend, with reductions of -40.38% (Fold 1) to -51.54% (Fold 5) in training and -32.42% (Fold 2) to -46.18% (Fold 5) in validation. However, R^2 improvement remains relatively low, peaking at only 11.96% in validation (Fold 4), suggesting that while the model reduces errors, it does not significantly enhance its ability to explain variance.

Table 8. Efficiency of the Hybrid-deep Learning Models.

Model	Fold	$RMSE$ (Train)	MAE (Train)	R^2 (Train)	$RMSE$ (Validation)	MAE (Validation)	R^2 (Validation)
CNN-Transformer	1	-41.188%	-40.377%	6.450%	-35.430%	-36.424%	6.095%
	2	-50.233%	-47.453%	8.486%	-38.914%	-32.417%	5.178%
	3	-53.106%	-47.730%	7.119%	-48.505%	-44.440%	10.294%
	4	-50.288%	-47.274%	6.733%	-47.539%	-45.808%	11.956%
	5	-55.037%	-51.538%	9.159%	-50.830%	-46.175%	8.741%
LSTM-Transformer	1	-36.796%	-36.199%	11.546%	-32.296%	-37.255%	13.682%
	2	-45.849%	-46.745%	18.513%	-37.957%	-35.995%	10.406%
	3	-43.556%	-43.080%	13.155%	-35.740%	-34.962%	14.499%
	4	-27.194%	-27.154%	8.176%	-25.092%	-27.362%	11.670%
	5	-23.097%	-22.952%	7.382%	-20.621%	-20.154%	6.739%
RNN-Transformer	1	-46.326%	-44.205%	5.805%	-33.228%	-34.438%	5.246%
	2	-48.769%	-44.554%	9.833%	-33.784%	-23.812%	4.689%
	3	-58.490%	-56.003%	11.642%	-49.453%	-51.692%	11.603%
	4	-45.572%	-44.439%	9.442%	-43.201%	-41.092%	14.499%
	5	-34.717%	-31.322%	6.717%	-34.719%	-34.836%	6.645%

The RNN-Transformer model shows inconsistent performance, with $RMSE$ reductions ranging from -34.72% (Fold 5) to -58.49% (Fold 3) in training and -33.23% (Fold 1) to -49.45% (Fold 3) in validation. Its MAE reductions follow a similar pattern, with the best improvement of -56.00% in training (Fold 3) but weaker

reductions in validation. The R^2 improvement remains low, peaking at only 11.64% in training (Fold 3) and 14.50% in validation (Fold 4). This suggests that while hybridization helps reduce error, it does not consistently enhance the predictive capability of RNN-based models. LSTM-Transformer emerges as the most effective hybrid model, demonstrating the highest and most stable improvements across folds, particularly in R^2 , where it achieves up to 13.68% in validation (Fold 1) and 18.51% in training (Fold 2). CNN-Transformer shows moderate but variable improvements, while RNN-Transformer, despite reducing $RMSE$ and MAE , fails to consistently enhance R^2 . These findings highlight the superiority of LSTM-Transformer for sequential modeling, as it effectively integrates the benefits of both LSTMs and Transformers.

In terms of error reduction, the CNN-Transformer achieves the most substantial decrease in $RMSE$, with a -49.57% reduction in training and -44.64% in validation, as shown in Figure 19. The LSTM-Transformer performed best with MAE for both training and validation, as shown in Figure 20. Among the hybrid models, the LSTM-Transformer emerges as the most effective, demonstrating the highest improvement in R^2 for both training ($+11.75\%$) and validation ($+11.80\%$). This indicates that integrating the Transformer architecture significantly enhances the predictive accuracy of the LSTM model, making it the most interpretable and reliable choice, as shown in Figure 21. However, despite this significant reduction in error, its R^2 improvement remains lower than that of the LSTM-Transformer. This suggests that while the CNN-Transformer effectively minimizes errors, it may still face challenges in generalization. The RNN-Transformer, while achieving notable reductions in $RMSE$ (-46.78% in training and -38.48% in validation), exhibits only moderate improvements in R^2 ($+8.69\%$ in training and $+8.54\%$ in validation). This indicates that while hybridization helps reduce errors, it does not substantially enhance the model's ability to explain variance.

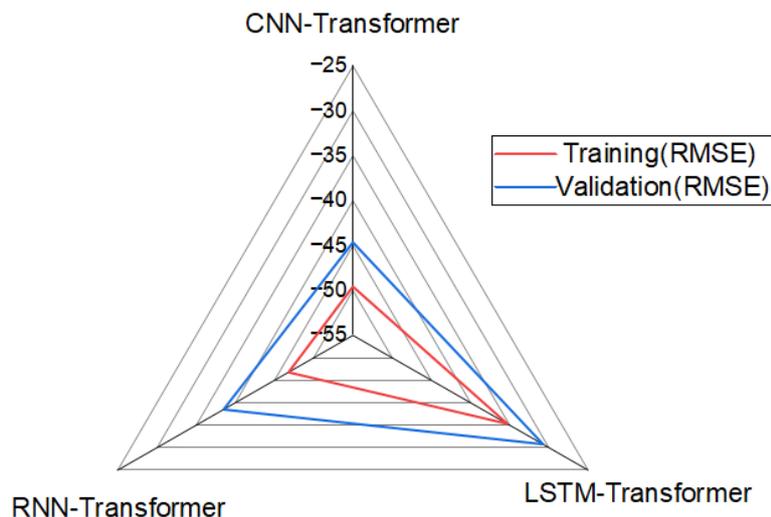


Figure 19. Mean % change in RMSE of hybrid deep learning Models.

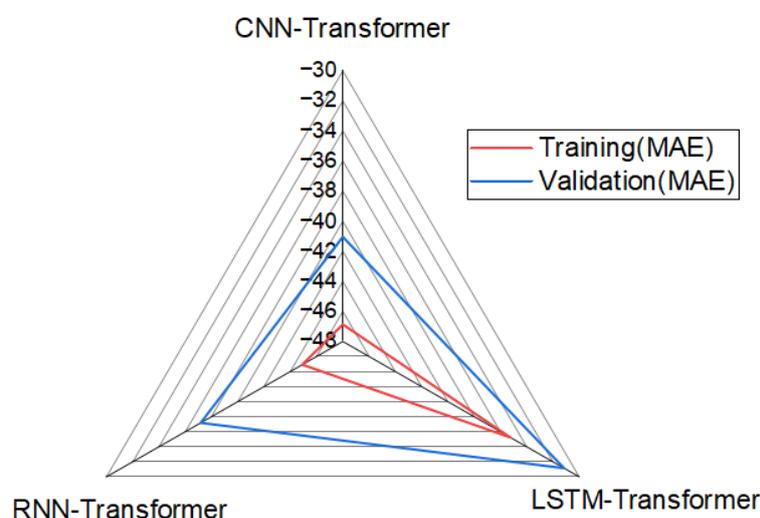


Figure 20. Mean % change in MAE of hybrid deep learning Models.

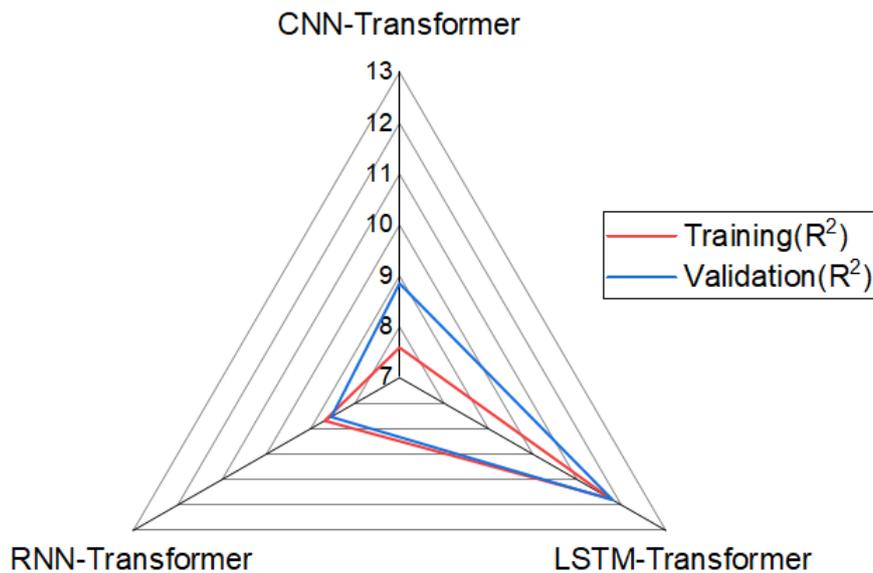


Figure 21. Mean % change in R^2 of hybrid deep learning Models.

8.4.1. Practical Application of Hybrid-Deep Learning Models

The LSTM-Transformer model plays a pivotal role in simulating and predicting the Unconfined Compressive Strength (UCS) of NS stabilized soil, offering a sophisticated approach that combines sequential learning and self-attention mechanisms. The LSTM (Long Short-Term Memory) component excels in capturing temporal dependencies, making it well-suited for modeling how UCS evolves over different curing periods. Meanwhile, the Transformer architecture enhances feature prioritization by dynamically focusing on the most relevant input parameters, such as NS dosage and curing time. This hybrid model is particularly advantageous for Monte Carlo simulations, where a large number of random samples are generated to assess the probabilistic distribution of UCS outcomes. The simulation, executed with 10,000 iterations, provides a robust statistical foundation for decision-making, revealing critical insights such as the mean normalized UCS (0.11 kg/cm^2) and a 95% confidence interval ($0.05\text{--}0.14 \text{ kg/cm}^2$) as shown in Figure 22.

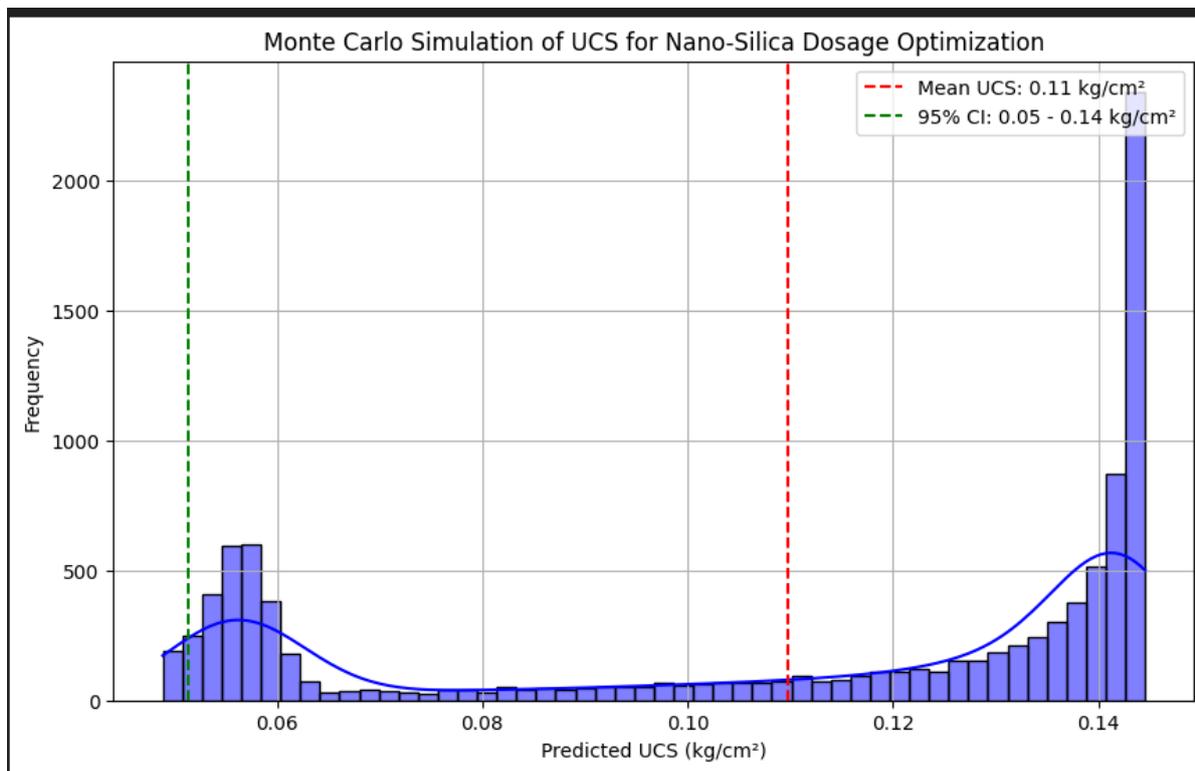


Figure 22. Optimization using LSTM-Transformer Model.

The residuals from the LSTM-Transformer model were analyzed and found to follow an approximately Gaussian distribution with a mean close to zero. This indicates that the model's predictions are generally unbiased and the errors are randomly distributed, validating its reliability for probabilistic UCS prediction under uncertainty. However, slight deviations from perfect normality were observed in the tails, likely due to the influence of extreme curing durations or nano-silica dosages, which introduce mild skewness or kurtosis. This behavior is typical in real-world geotechnical datasets with underlying nonlinearities and heteroscedasticity. These probabilistic outputs help geotechnical engineers in optimizing pavement design, soil stabilization techniques, and earthquake-resistant infrastructure by quantifying uncertainty and minimizing risks. Furthermore, this approach promotes sustainable construction by allowing computational experimentation with NS dosages before physical implementation, thereby reducing material waste and optimizing resource allocation. The integration of deep learning and stochastic simulations establishes a data-driven framework for enhancing soil strength prediction, making the LSTM-Transformer model an invaluable tool in geotechnical engineering.

8.4.2. GUI Development of Hybrid Model

The integration of a graphical user interface (GUI) for Monte Carlo simulations using the LSTM-Transformer model offers an effective platform for analyzing and predicting the behavior of NS-stabilized soil, is shown in Figure 23. In this approach, users can easily upload datasets related to soil properties and NS content, initiate Monte Carlo simulations, and visualize the results. The Monte Carlo simulation runs multiple iterations, each using random sampling of data, which is essential for assessing the impact of varying levels of NS on soil stability and behavior. By leveraging LSTM and Transformer layers, the model can capture complex relationships within the data, including the effects of NS on parameters such as UCS and soil compaction. The LSTM-Transformer model is particularly adept at learning the sequential dependencies and nonlinear interactions in the data, making it ideal for understanding the influence of different factors, such as NS percentage and curing time, on soil performance. Each simulation iteration involves training the model on a randomly selected subset of the data, followed by the computation of performance metrics like $RMSE$, MAE , and R^2 , which help evaluate the model's predictive accuracy. The results are then aggregated and displayed, giving users a comprehensive understanding of the model's performance and the variability in predictions due to changes in NS content. This tool not only provides insights into the effects of NS stabilization on soil but also serves as a valuable resource for optimizing soil stabilization strategies in geotechnical engineering applications, making it possible to design more resilient infrastructures and predict the behavior of stabilized soils under different conditions. The saved LSTM-Transformer model is provided in the supplementary file for the future use.

Figure 23. GUI for UCS prediction of NS-stabilized soil.

9. Simplified LCA Method for Comparing NS and Cement-Stabilized Soil

To facilitate a practical and accessible Life Cycle Assessment (LCA) comparison, a simplified methodology was adopted to evaluate the environmental impacts of NS-stabilized soil versus cement-stabilized soil. The LCA

framework consisted of four key stages: raw material extraction, production, application, and environmental impact assessment.

9.1. Goal and Scope Definition

This case study aims to compare the environmental impacts of NS and cement as stabilizers for subgrade soil, considering energy consumption, carbon footprint, and resource efficiency. The study evaluates a 1 m³ volume of stabilized soil, a typical application in Mechanically stabilized Wall (MSW) construction.

9.2. Inventory Analysis

The Life Cycle Assessment (LCA) Inventory Analysis compares Nano-Silica Stabilization and Cement Stabilization based on environmental impact and material efficiency as shown in Table 9. Nano-silica stabilization involves silica extraction, which requires moderate energy consumption and can be sourced from natural deposits or industrial by-products, making it a more sustainable option. In contrast, cement stabilization relies on quarrying limestone and high-energy clinker production, which involves intensive heating above 1400 °C, leading to significant CO₂ emissions and a higher environmental footprint. In terms of application, NS is mixed directly with soil, requiring minimal additional material, while cement stabilization demands a larger material volume per unit due to hydration reactions, which increase resource consumption. NS stabilization is a more sustainable alternative, offering lower emissions, reduced energy use, and enhanced material efficiency, making it an environmentally preferable choice for soil stabilization.

Table 9. LCA inventory Analysis.

Factor	Nano-Silica Stabilization	Cement Stabilization
Raw Material Extraction	Silica extraction (moderate energy)	Quarrying and high energy clinker processing
Application Process	Mixed with soil (minimal additional material)	Requires more material per unit due to hydration reactions

9.3. Impact Assessment

Table 10 LCA Impact Assessment highlights the environmental performance of NS Stabilization compared to Cement Stabilization across key indicators, emphasizing their respective environmental impacts. In terms of carbon emissions, NS stabilization results in low emissions, contributing to a significant reduction in greenhouse gases, making it an environmentally friendly option. In contrast, Cement Stabilization generates high carbon emissions, with approximately 800 kg of CO₂ emitted per ton, contributing substantially to environmental pollution. When examining energy consumption, NS stabilization uses 1.2 GJ per ton, making it significantly more energy-efficient compared to Cement Stabilization, which requires 4.5 GJ per ton, resulting in a 73% higher energy demand. This increased energy consumption in cement processes contributes to higher environmental costs. Additionally, NS stabilization is highly material-efficient, requiring only a 3–4% dosage rate, which minimizes the material input needed for effective stabilization. On the other hand, Cement Stabilization requires a higher 8–12% dosage rate, leading to greater material consumption. NS Stabilization demonstrates substantial advantages in terms of lower carbon emissions, reduced energy usage, and improved material efficiency, making it a more sustainable and environmentally responsible choice compared to cement stabilization.

Table 10. LCA Impact Assessment.

Environmental Indicator	NS Stabilization	Cement Stabilization
Carbon Emissions	Low (significant reduction)	High (800 kg CO ₂ per ton)
Energy Consumption	1.2 GJ/ton	4.5 GJ/ton (73% higher)
Material Efficiency	3–4% dosage rate	8–12% dosage rate

9.4. Case Study: Stabilization of 100 m³ of Filling Soil in an MSW Wall

Table 11 LCA Case Study MSW Wall provides a detailed comparison between NS Stabilization and Cement Stabilization in the context of the MSW wall construction. In terms of material quantities, NS stabilization requires significantly less material, with only 4.5 m³ of NS needed compared to the 9 m³ of cement required for cement stabilization. This highlights the material efficiency of NS stabilization, reducing the overall material demand for the same application. When examining carbon emission reduction, NS stabilization is a far more environmentally friendly option, contributing to 55% lower CO₂ emissions compared to cement stabilization, which produces

higher emissions. This reduction in carbon emissions positions NS stabilization as a sustainable alternative, reducing the environmental impact of the construction process. Additionally, NS stabilization offers 1.5 times lower energy consumption than cement stabilization, which demands higher energy inputs throughout production and application processes. NS stabilization delivers significant environmental benefits in terms of reduced material usage, carbon emissions, and energy consumption, making it a more sustainable and resource-efficient solution compared to cement-based alternatives for MSW wall construction.

Table 11. LCA Case study MSW wall.

Factor	NS Stabilization	Cement Stabilization
Material Quantities	4.5 m ³ of NS	9 m ³ of cement
Carbon Emission Reduction	55% lower CO ₂ emissions	Higher emissions
Energy Savings	1.5 times lower	Higher energy demand

10. Conclusions

10.1. Key Findings

This study presents a novel AI-driven framework that integrates advanced hybrid deep learning models specifically CNN-Transformer, LSTM-Transformer, and RNN-Transformer for predicting the Unconfined Compressive Strength (UCS) of Nano-silica (NS) stabilized soils. The approach systematically combines deep learning with Transformer architectures, allowing for enhanced learning of complex, nonlinear interactions between NS dosage, curing period, and soil characteristics. Among the standalone models, CNN achieved the best performance ($R^2 = 0.89$, $RMSE = 0.43$), but its hybrid counterpart, CNN-Transformer, significantly outperformed it with an R^2 of 0.97 and $RMSE$ of 0.22 a 49.57% reduction in error. Monte Carlo simulations with 10,000 iterations further confirmed model robustness, providing a normalized UCS mean of 0.11 kg/cm² and a 95% confidence interval from 0.05 to 0.14 kg/cm², thus supporting uncertainty quantification and risk-informed decision-making. Additionally, the study included a comparative Life Cycle Assessment (LCA), showing that NS stabilization is substantially more sustainable than cement-based methods achieving a 55% reduction in CO₂ emissions and a 73% reduction in energy consumption. The integration of AI models with LCA and the development of a GUI tool for real-time NS dosage optimization mark a significant innovation that bridges machine learning with practical field applications.

10.2. Research Limitations

While the results are promising, this study has certain limitations:

- (i) The dataset, although diverse, is limited in scale and geographic representation.
- (ii) Laboratory data may not fully capture the heterogeneity and unpredictability of in-situ field conditions.
- (iii) The current model does not include other potentially relevant soil parameters such as mineralogy, compaction effort, or field moisture variability.
- (iv) LCA was conducted under controlled assumptions and may vary depending on regional material sourcing and transport logistics.

10.3. Recommendations for Future Research

To expand on the contributions of this work, future research should:

- (i) Incorporate larger and more diverse datasets from various soil types and climatic regions.
- (ii) Integrate additional geotechnical parameters (e.g., permeability, plasticity, clay mineralogy) to improve model generalization.
- (iii) Conduct field-scale validations to bridge the lab-to-site gap and evaluate model reliability in real-world conditions.
- (iv) Extend the current GUI tool into a comprehensive decision-support system incorporating cost-benefit and lifecycle analysis modules.
- (v) Explore the hybridization of Transformer models with other AI techniques like Genetic Algorithms or Explainable AI (XAI) for enhanced optimization and interpretability.

10.4. Novelty and Impact

This is the first known study to integrate hybrid Transformer-based deep learning models with stochastic simulations and LCA into a unified framework for soil strength prediction. The inclusion of a practitioner-ready GUI tool enhances its real-world impact, offering a robust, data-driven solution for sustainable and resilient infrastructure design. By aligning cutting-edge AI with environmentally conscious engineering practices, this research advances the field of intelligent geotechnics and supports SDG 9 and SDG 12 initiatives.

Supplementary Materials

The online version contains as supplementary material the developed GUI that can be downloaded from GitHub repository, https://github.com/thapa67/GUI_UCS_NS.

Author Contributions

I.T.: conceptualization, methodology, software, data curation, writing original draft preparation; S.G.: visualization, investigation, supervision, software, validation, reviewing and editing. All authors have read and agreed to the published version of the manuscript.

Funding

This research received no external funding.

Institutional Review Board Statement

Not applicable.

Data Availability Statement

The data and supplementary material are available on request. Further the developed GUI can be accessed from the GitHub repository: https://github.com/thapa67/GUI_UCS_NS.git (accessed on 1 August 2024).

Conflicts of Interest

The authors declare no conflict of interest.

References

1. Kulanthaivel, P.; Selvakumar, S.; Soundara, B.; et al. Combined effect of nano-silica and randomly distributed fibers on the strength behavior of clay soil. *Nanotechnol. Environ. Eng.* **2022**, *7*, 23–34. <https://doi.org/10.1007/s41204-021-00176-3>.
2. Kulanthaivel, P.; Selvakumar, S.; Soundara, B.; et al. Strength Enhancement of Clay Soil Stabilized with Ordinary Portland Cement, Sodium Silicate and Sodium Hydroxide. *Int. J. Pavement Res. Technol.* **2023**, *16*, 1297–1310. <https://doi.org/10.1007/s42947-022-00197-4>.
3. Kulanthaivel, P.; Soundara, B.; Velmurugan, S.; et al. Experimental investigation on stabilization of clay soil using nano-materials and white cement. *Mater. Today Proc.* **2021**, *45*, 507–511. <https://doi.org/10.1016/j.matpr.2020.02.107>
4. Selvakumar, S.; Kulanthaivel, P.; Soundara, B. Influence of nano-silica and sodium silicate on the strength characteristics of clay soil. *Nanotechnol. Environ. Eng.* **2021**, *6*, 46. <https://doi.org/10.1007/s41204-021-00142-z>.
5. Kannan, G.; Sujatha, E.R. A review on the Choice of Nano-Silica as Soil Stabilizer. *Silicon* **2022**, *14*, 6477–6492. <https://doi.org/10.1007/s12633-021-01455-z>.
6. Ali, H.F.H.; Mohammed, A.S. Innovative analysis and advanced modeling of UCS and CBR in fly ash-treated soils: Evaluating the impact of hydraulic index, chemical alteration, lime modulus, and geochemical indices. *Model. Earth Syst. Environ.* **2025**, *11*, 28. <https://doi.org/10.1007/s40808-024-02230-w>
7. Farid Hama Ali, H.; Mohammed, A.S. New approaches to evaluate the impact of chemical oxides on the liquid limit, plasticity index, and unconfined compressive strength of clay soils. *Geomech. Geoengin.* **2025**, *20*, 635–660. <https://doi.org/10.1080/17486025.2024.2433627>
8. Shao, W.; Yue, W.; Zhang, Y.; et al. The Application of Machine Learning Techniques in Geotechnical Engineering: A Review and Comparison. *Mathematics* **2023**, *11*, 3976. <https://doi.org/10.3390/math11183976>.
9. Ngo, T.Q.; Nguyen, L.Q.; Tran, V.Q. Novel hybrid machine learning models including support vector machine with meta-heuristic algorithms in predicting unconfined compressive strength of organic soils stabilised with cement and lime. *Int. J. Pavement Eng.* **2023**, *24*, 2136374. <https://doi.org/10.1080/10298436.2022.2136374>

10. Kumar, A.; Sinha, S.; Saurav, S. Random forest, CART, and MLR-based predictive model for unconfined compressive strength of cement reinforced clayey soil: A comparative analysis. *Asian J. Civ. Eng.* **2024**, *25*, 2307–2323. <https://doi.org/10.1007/s42107-023-00909-6>.
11. Guan, Q.Z.; Yang, Z.X.; Guo, N.; et al. Finite element geotechnical analysis incorporating deep learning-based soil model. *Comput. Geotech.* **2023**, *154*, 105120. <https://doi.org/10.1016/j.compgeo.2022.105120>.
12. Wang, Z.Z. Deep Learning for Geotechnical Reliability Analysis with Multiple Uncertainties. *J. Geotech. Geoenviron. Eng.* **2022**, *148*. [https://doi.org/10.1061/\(ASCE\)GT.1943-5606.0002771](https://doi.org/10.1061/(ASCE)GT.1943-5606.0002771).
13. Wang, Z.Z.; Goh, S.H.; Zhang, W. Reliability-based design in spatially variable soils using deep learning: An illustration using shallow foundation. *Georisk: Assess. Manag. Risk Eng. Syst. Geohazards* **2023**, *17*, 423–437. <https://doi.org/10.1080/17499518.2022.2083178>.
14. Zhang, N.; Zhou, A.; Jin, Y.-F.; et al. An enhanced deep learning method for accurate and robust modelling of soil stress–strain response. *Acta Geotech.* **2023**, *18*, 4405–4427. <https://doi.org/10.1007/s11440-023-01813-8>.
15. Zhang, W.; Ghahari, F.; Arduino, P.; et al. A deep learning approach for rapid detection of soil liquefaction using time–frequency images. *Soil Dyn. Earthq. Eng.* **2023**, *166*, 107788. <https://doi.org/10.1016/j.soildyn.2023.107788>.
16. Akbar Firoozi, A.; Asghar Firoozi, A. Application of Machine Learning in Geotechnical Engineering for Risk Assessment. In *Machine Learning and Data Mining Annual Volume 2023*; IntechOpen: London, UK, 2023. <https://doi.org/10.5772/intechopen.113218>.
17. Chen, M.; Kang, X.; Ma, X. Deep Learning–Based Enhancement of Small Sample Liquefaction Data. *Int. J. Geomech.* **2023**, *23*. <https://doi.org/10.1061/IJGNAL.GMENG-8381>.
18. Savvides, A.A.; Papadopoulos, L. A neural network approach for the reliability analysis on failure of shallow foundations on cohesive soils. *Int. J. Geo-Eng.* **2024**, *15*, 15. <https://doi.org/10.1186/s40703-024-00217-1>.
19. O’Shea, K.; Nash, R. An Introduction to Convolutional Neural Networks. *arXiv* **2015**, arXiv:1511.08458
20. Li, K.; Pan, L.; Guo, X.; et al. Hybrid random aggregation model and Bayesian optimization-based convolutional neural network for estimating the concrete compressive strength. *Comput.-Aided Civ. Infrastruct. Eng.* **2024**, *39*, 559–574. <https://doi.org/10.1111/mice.13096>.
21. Graves, A. Long Short-Term Memory. In *Supervised Sequence Labelling with Recurrent Neural Networks*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 37–45. https://doi.org/10.1007/978-3-642-24797-2_4.
22. Vivian, J.; Prataciera, E.; Gastaldello, N.; et al. A comparison between grey-box models and neural networks for indoor air temperature prediction in buildings. *J. Build. Eng.* **2024**, *84*, 108583. <https://doi.org/10.1016/j.jobe.2024.108583>.
23. Medsker, L.; Jain, L.C. (Eds.). *Recurrent Neural Networks Design and Applications*; CRC Press: Boca Raton, FL, USA, 1999. <https://doi.org/10.1201/9781003040620>
24. Shan, F.; He, X.; Armaghani, D.J.; et al. Effects of data smoothing and recurrent neural network (RNN) algorithms for real-time forecasting of tunnel boring machine (TBM) performance. *J. Rock Mech. Geotech. Eng.* **2024**, *16*, 1538–1551. <https://doi.org/10.1016/j.jrmge.2023.06.015>.
25. Mauricio, J.; Domingues, I.; Bernardino, J. Comparing Vision Transformers and Convolutional Neural Networks for Image Classification: A Literature Review. *Appl. Sci.* **2023**, *13*, 5521. <https://doi.org/10.3390/app13095521>.
26. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. *arXiv* **2014**, arXiv:1412.6980
27. Kulanthaivel, P., Soundara, B., Velmurugan, S., & Naveenraj, V. Experimental investigation on stabilization of clay soil using nano-materials and white cement. *Mater. Today Proc.* **2021**, *45*, 507–511. <https://doi.org/10.1016/j.matpr.2020.02.107>.
28. Aksu, G.; Eskisar, T. The geomechanical properties of soils treated with nanosilica particles. *J. Rock Mech. Geotech. Eng.* **2023**, *15*, 954–969. <https://doi.org/10.1016/j.jrmge.2022.06.013>.
29. Thapa, I.; Ghani, S. Estimation of California bearing ratio for hill highways using advanced hybrid artificial neural network algorithms. *Multiscale Multidiscip. Model. Exp. Des.* **2024**, *7*, 1119–1144. <https://doi.org/10.1007/s41939-023-00269-3>.
30. Asteris, P.G.; Skentou, A.D.; Bardhan, A.; et al. Predicting concrete compressive strength using hybrid ensembling of surrogate machine learning models. *Cem. Concr. Res.* **2021**, *145*, 106449. <https://doi.org/10.1016/j.cemconres.2021.106449>.
31. Khatti, J.; Grover, K.S. Prediction of UCS of fine-grained soil based on machine learning part I: Multivariable regression analysis, gaussian process regression, and gene expression programming. *Multiscale Multidiscip. Model. Exp. Des.* **2023**, *6*, 199–222. <https://doi.org/10.1007/s41939-022-00137-6>.