FUDMA Journal of Engineering and Technology
ISSN: 3092-9385
https://fjet.fudutsinma.edu.ng/index.php/fjet/
Volume 2, Issue 2, 2026, 447-458

Artificial Intelligence-Enabled Essay Grading System Using NLP Via Semantic Similarity Analysis and Supervised Machine Learning Techniques

Jide E. T. AKINSOLA[1], John A. OLADITI[2], Ifeoluwa M. OLANIYI[3*], Emmanuel A. OLAJUBU[4], Ayomide O. EMMANUEL[5], Ganiyu A. ADEROUNMU[6]

[1,2,3*,5] Department of Computer Sciences, Abiola Ajimobi Technical University, 200255, Ibadan, Nigeria.
[4,6]Department of Computer Science and Engineering, Obafemi Awolowo University, Ile–Ife, 220282, Nigeria

[1]akinsolajet@gmail.com, [2]adewale.oladiti28@gmail.com, [3*]olaniyiifeoluwa12@gmail.com, [4]emmolajubu@oauife.edu.ng, [5]emmanuelayomide173@yahoo.com, [6]gaderoun@oauife.edu.ng

Abstract

A manual grading system can be affected by several factors such as subjective judgment of the reviewer, tiredness, sensitivity to handwriting, as well as the reviewer's emotions, physical and mental state, which may lead to unfairness and inconsistency in student grades. In order to ensure fairness in computing the grades of students, it is good to embrace the use of an automated grading system which ensures consistent and objective grading. This study employed a numerical representation of semantic similarity by comparing the student's answer with both the examiner's answer and the comprehension passage, which was later combined using weights of 0.05 and 0.95. Random Forest (RF), Decision Tree (DT), and Gradient Boosting Regressor algorithms were trained and evaluated using R-squared value, MAE, MSE, and RMSE. The performance evaluation results show that the Decision Tree model gave the best results, achieving the highest R-squared score of 0.9717 and the lowest prediction errors of 0.1030 MAE, 0.0703 MSE, and 0.2651 RMSE, which suggests that the combination of semantic similarity and question score provides useful information for estimating student performance. However, similarity alone may not fully capture the overall quality and correctness of an answer. The study, therefore, suggests that future improvements could incorporate features such as grammatical correctness, relevance, completeness, and factual accuracy.

Keywords: *Decision Tree, Essay Grading, Gradient Boosting Regressor, Natural Language Processing, Random Forest, Semantic Similarity Analysis, Supervised Machine Learning.*

1.0 Introduction

In recent years, the integration of automated systems has risen in the education sector, as most institutions require more robust systems to enhance the learning process, especially in the grading of students. Automation of the evaluation process [1] of student answers has become a workable substitute to the manual process being adopted by many institutions due to the daily improvement in the field of Artificial Intelligence and Natural Language Processing. This advancement in technology provides a solution to academic sectors, especially in an environment of an increasing number of learners with growing demands for prompt feedback.

The grading system is one of the important parts of academics that evaluates students' performance and monitors the growth of students. With a grading system, students are able to get feedback from the previous examination in order to make adjustments where necessary for subsequent learning to ensure positive advancement in their studies. It has been proven that manual computation of results is time-consuming [2] and prone to human bias. Therefore, it is important to adopt the use of computers to process and compute students' grades. The traditional manual way of grading students brings about unfairness and inconsistency in grading a larger number of students. In order to ensure fairness in computing the grades of students, it is good to embrace the use of an automated grading system which ensures consistent and objective grading [3].

The traditional method of grading students has been proven to be influenced by factors such as mood [4], weather, noise, time constraints [5], and subjective judgment of the reviewer [6]. The large number of students in an institution creates significant setbacks for educators when assessing the students. The implementation of an intelligent system not only ensures consistency and fairness but also ensures prompt feedback [7].

Machine Learning (ML) and Natural Language Processing (NLP) provide effective tools to solve the challenges of the traditional way of grading in academics, leveraging methods such as Random Forest Regressor, Decision Tree, and Gradient Boosting Regressor. These algorithms learn from provided information, extract useful details, and use that to carry out effective and cost-effective grading [8]. These methods repeatedly mine data from the dataset provided in order to provide automation of scoring essays, and also ensuring high level of accuracy [9]. This study focuses on developing an intelligent system that accurately grades student essays leveraging

Received: 22/08/2026; Revised: 31/08/2026; Accepted: 01/09/2026; Published: 07/09/2026

Machine Learning, Natural Language Processing, and Semantic Similarity Analysis, which examines sentence resemblances based on their meaning and not word structures alone.

Therefore, this study aims to fill this knowledge gap by providing a machine learning approach to predict students' and examiners' scores. The definite ideas of this study are to assess student performance by employing NLP techniques, create a predictive model by training a generated dataset with three different ML algorithms, and likewise carry out performance evaluation to determine the model that yields the best result. Also, the study checks similarities by making use of different semantic similarity analysis approaches. This study is expected to enhance the grading system, hence providing a seamless automated essay grading system. Below are the contributions of this research to the scientific body of knowledge:

1. Generation of a dataset for accurate prediction of students' scores vs examiners' scores.
2. Implementation of different semantic similarity analysis methods to determine the best one.
3. Execution of semantic weight ratio to determine the relationship among essay, comprehension, examination, and students' answers.
4. Implementation of several semantic weight ratios to determine the optimal one for combining comprehension and students' answers.
5. Combination of semantic similarity analysis with target data frame for predictive analytics.

This study is divided into five sections. The general overview of the study is highlighted in this section of the study, which is the introduction. Section two of this study entails the extracts of related literature reviewed. Section three gives the breakdown of the approach used to accomplish the stated objectives, explaining the process, tools, and techniques employed to achieve the goal of the study. Section four discusses the results achieved from the methodology. Section five gives the overall conclusion.

2.0 Review of Literature

The rapid increase in the rate of learning both offline and online has led to a high demand for effective methods of evaluating learners' performance. Therefore, there is a need to transform from the traditional way of grading students' performance, which is prone to subjectivity and time-consuming, to a more effective and stress-free automated method that ensures accuracy and objective nature in scoring students [10]. NLP could be used together with ML algorithms to effectively understand and interpret human language [2] for accurate essay grading.

2.1 Intelligent Grading System

With the advances in technology, the automation of the grading process of students' examinations has become feasible with the use of supervised ML algorithms such as Random Forest, Decision Tree, and Gradient Boosting Regressor, together with NLP and Semantic Similarity to help clarify the similarities between sentences, laying more emphasis on the meaning of the sentences. This approach disrupts the manual evaluation of students' examinations, reducing drawbacks such as time consumption, subjectivity in scoring, and human error [11] . Therefore, ensuring accuracy in grading. NLP models are used to analyze large volumes of linguistic datasets to study patterns and trends, thereby carrying out grading that is unbiased and dependable [12].

[2] work on mitigating several challenges encountered by academic staff, such as reducing the time spent on grading a large number of students' essays. They developed an automated grading system using Java as the programming language and MySQL for database development. The developed system's attention was on short essay grading. They were able to achieve a 0.4 Exact Agreement rate and a 0.93 Pearson Correlation with a human grader. [13] developed a system to deliver fast and consistent grading of student essays, employing Natural Language Processing techniques and SpaCy for syntactic and semantic analysis. They trained a labeled essay dataset using ML models. The system is designed to evaluate the structures of sentences, offering real-time response via a responsive web application interface.

Another researcher explored the development of an intelligent Grading System which main target was college students. The major area was on grading translation assignments [12]. They implemented two algorithms, which are gradient boosting decision tree and backpropagation neural network, learning from big data, categorizing patterns, and mining meaningful features. The study was able to achieve 98% accuracy. [10] also carried out diverse research methods, such as literature review, comparative studies, and professional consultation, to gain insights into the development of an intelligent grading system for short essays. This study focuses on developing intelligent grading for all essay-related examinations utilizing Machine Learning Algorithms, Natural Language Processing, and Semantic Similarity Analysis.

The present study focuses on providing a robust system for automated grading system, despite the importance didn't focus on solving the issue of bias in the training dataset, subjectivity. Some of the studies also lacked in-depth interpretation of meaning, especially in the case of paraphrased answers. Some of the studies only focus on short essay grading, neglecting longer and more complex responses. Also, some of the research only developed the system using programming languages, leaving behind the application of artificial intelligence. This affects the flexibility of the system.

Received: 22/08/2026; Revised: 31/08/2026; Accepted: 01/09/2026; Published: 07/09/2026

2.1.1 Semantic Similarity Analysis

The quality of answers is very important; therefore, the use of semantic similarity analysis in NLP is very crucial. Semantic similarity analysis helps assess the closeness in context of meaning among students' answers, instead of focusing on word matching. Therefore, providing more precise evaluation for students. [14] worked on the development of an automated grading system implementing NLP and semantic similarity measures to detect closeness of sentences. The study implemented some similarity measures such as edit similarity, Jaccard similarity, normalized word counts, and semantic similarity. [15] implemented two semantic similarity analysis methods, which are corpus-based and knowledge-based, together with deep learning to evaluate against the conventional methods in a short answer grading system to identify gaps.

Semantic Similarity Analysis can be categorized into two major types. These types are based on the basis of the quantity of words they focus on. The two types are lexical semantic and corpus-based styles [15].

i. **Lexical Semantic:** This type of Semantic Similarity Analysis discovers some English notions such as antonyms, hyponyms, hypernyms, and synonyms to detect the meaning of a word and its relationship with other words. Lexical semantics is based on the meaning of a distinct word. Lexical semantics can also be referred to as Lexicosemantic.
ii. **Corpus-based Style:** Corpus-based style, in contrast to lexical semantics, addresses and analyzes a group of text, which is known as corpora, to get the relationship between concepts and words [10]. This is done based on the corresponding patterns. This approach does not focus only on the definition of words, but on how words are used in a sentence.

2.1.2 Natural Language Processing

Natural Language Processing is a branch of Artificial Intelligence that combines computer science, linguistics, and machine learning to allow computers to analyze, understand, and interpret human language. To achieve an effective automated grading system, NLP is required to interpret human-written language [8]. NLP helps in the aspect of textual software, enabling accurate grammar and spelling checks. NLP is a great transformation in the field of artificial intelligence, which eases human-computer interaction. [16] developed a machine learning-based model to perform an automated essay system. He implemented NLP to extract some features from a publicly available dataset. These models reliability were compared with that of a human grader. [17] carried out a survey on some existing automated grading systems, marking out their limitations. The study developed an automated essay scoring system, implementing deep learning in conjunction with Natural Language Processing techniques, focusing more on content-based approaches. Different NLP approaches can be used to analyze and infer human language. Some of the techniques are tokenization, lemmatization, stop word removal, text normalization, sentiment analysis, and named entity recognition, among others. This study implements three of these approaches, which are tokenization, lemmatization, and stop word analysis.

i. **Tokenization:** Tokenization helps to convert text which are shapeless text into text that can be processed and analyzed by models. It is the process of splitting down streams of words into smaller units known as tokens. The outcome of tokenization comes in two components, which are the word index and tokenized text. The word index gives the identifier of each word in a list format. Each word is replaced by a tokenized text with a corresponding numerical token [18]. Tokenization implements different ways of word splitting. The approaches are whitespace tokenization, punctuation-based tokenization, regular expression tokenization, and library-based tokenization.
ii. **Lemmatization:** Lemmatization is one of the important processes in NLP that traces back words to their root word, giving much attention to meaning and part of speech. The resulting word in lemmatization is known as a lemma. Lemmatization is an accurate approach to classify related words in order to improve NLP tasks. Lemmatization is the best approach to deal with systems that deal with understanding the meaning of words [19].
iii. **Stop Word Analysis:** Stop word analysis is a process in NLP that ensures noise reduction by eliminating words that convey less semantic meaning. Words with less semantic meaning are known as stop words. They are words like "is,"" at" etc. The text's most instructive terms are highlighted in this model.

2.2 Machine Learning

Machine learning makes it possible for computers to create patterns from previous experience in order to create new situations through the analysis of a dataset. Machine learning learns from data to improve performance without explicit programming for each task [20]. There are diverse algorithms for machine learning, such as Gradient Boosting Regressor, Decision Tree, Random Forest Regressor, etc. These algorithms are selected based on their features and strengths. The most widely used form of machine learning classification is supervised machine learning due to its various applications in diverse areas such as decision support, inventory systems, intelligent grading systems, intelligent data analytic software, among others [20].

Received: 22/08/2026; Revised: 31/08/2026; Accepted: 01/09/2026; Published: 07/09/2026

2.2.1 Decision Tree

A decision tree is a machine learning algorithm classified under the supervised learning category [21]. The structure of a decision tree is like a tree with different nodes and leaves. It is widely used in the area of numerical value prediction. A decision tree regressor can be used to build a model that predicts a continuous value by undergoing the process of data segmentation recursively, considering the feature value. A decision tree, as the name implies, is structured in a tree-like structure that consists of nodes [20], branches, and leaves. The node represents a decision with respect to the feature; the branch denotes the output of the decision. While the leaf represents the predicted value. Decision Tree Regression can also be called Regression Tree. DT implements a regression model to convert a complex model to a simpler and more understandable model in the form of a graphical display [22]. The benefit of implementing DT is the ability to swiftly predict numerical output in the presence of noise. DT could be used as a regressor that recursively splits the complex task into smaller regions. These are known as nodes. The smaller divided parts are the leaf nodes, while the main task is the root node [23]. Figure 1 shows the structural representation of a Decision Tree. The root node represents the entire dataset, while other nodes are subdivisions of the nodes. The leaf nodes represent the end of the splitting [24]. The dataset is partitioned with the aid of entropy.

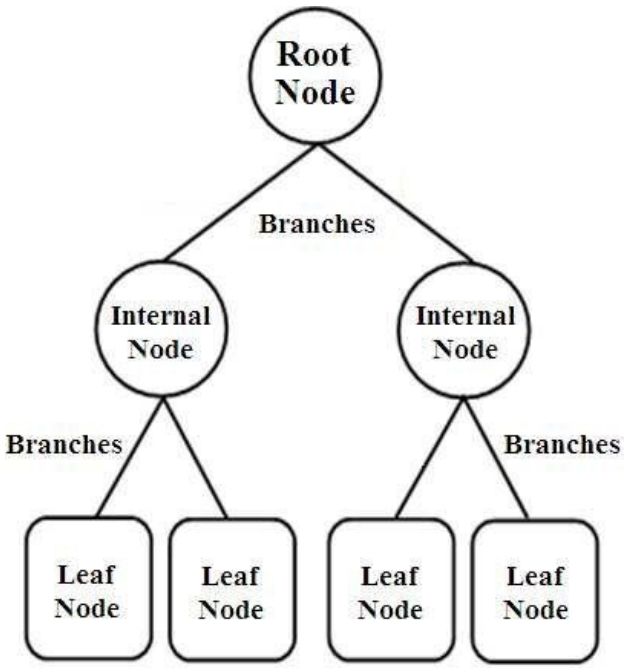

Figure 1: Decision Tree Structure [24]

2.2.2 Random Forest Regressor

Random forest is a machine learning algorithm that can be used to solve regression problems. Random Forest uses a collection of decision trees to carry out prediction on a continuous numerical dataset. Multiple decision trees are trained on different subsets of the training dataset. The final output is obtained through the process of averaging in order to reduce the effect of overfitting. The best among the subset of predictors is randomly selected to split the node [25]. Each predictor depends on the values of a random vector identically distributed across all trees in the forest [26]. A significant volume of high-quality data is essential for efficient data collection to train artificial intelligence models and machine learning algorithms. System performance data is crucial for improving algorithms, hardware and software efficiency, user behavior assessment, pattern recognition, decision-making, predictive modeling, and problem-solving, all of which lead to increased efficacy and accuracy [26]. Random forest ensures reduction in bias.

2.2.3 Gradient Boosting Regressor

In the boosting ensemble learning approach, gradient boosting trees are a potent family of machine learning algorithms that perform gradient descent on decision trees. Iteratively combining many simple models (i.e., weak learners) to create a model with improved prediction accuracy (i.e., strong learners) is the fundamental concept underlying them. Gradient Boosting ensures successive decision trees fit one training example while selecting the best tree based on the loss function. The best is the one with the least loss function [27]. Another ensemble model is the Gradient Boosting Regressor (GBR), which is an iterative collection of tree models stacked sequentially so

Received: 22/08/2026; Revised: 31/08/2026; Accepted: 01/09/2026; Published: 07/09/2026

that the subsequent model learns from the previous model's error. In order to create a more robust model, this machine learning model "boosts" the ensemble of weak prediction models [28], frequently decision trees [28].

Table 1: Summary of Related Works

S/N	Author/Year	Methodology	Dataset Used	Findings	Limitations
1	[1]	Implementation of n-gram and enhanced latent semantic analysis.	Computer science questions and answers from the Department of Computer Science at Babcock University.	Lower difference between human grader and system grade with accuracy of 89.03%	Dataset scope is limited.
2	[2]	Implementation of Java, MySQL, Artificial Neural Network to develop a web application grading system.	Dataset was not specified.	0.4 Exact Agreement rate, 0.93 Pearson Correlation with human grader	The study focused only on short essay grading.
3	[13]	The study used NLP and SpaCy as a syntactic and semantic analysis tool on a labeled essay dataset.	Labeled Essay Dataset	Evaluation of sentence structures, ensuring real-time response	System dependent on pre-defined linguistic patterns.
4	[9]	Analysis of essay dataset using mixed approach	Essay dataset of 5,000 entries	The solution achieved an accuracy of 87%.	The dataset used in training the model is small.
5	[12]	Implementation of gradient boosting decision tree and Back Propagation Neural Network	Large volumes of labeled translation data	Efficient grading system for English students to ensure objectivity.	Focused on college students' English Grading.
6	[10]	Execution of different research methods on the evolution of the grading system.	No dataset was used	The research offered the prospect of an automated grading system in academics.	No system was developed.

3.0 Materials and Methods

This section discusses the techniques, methods, and tools used in the implementation of the Essay Grading System. This section explains the dataset used and the techniques employed for the preprocessing, such as data cleaning, tokenization, lemmatization, and stop word analysis. Semantic Similarity Analysis is performed, and the result is converted to a data frame. The preprocessed dataset is then used in developing the models, which are a random forest regressor, gradient boosting regressor, and decision tree regressor. The developed models are measured based on performance metrics to determine the best model to implement. The backend business logic is developed and integrated with the best model chosen. Figure 2 represents the process flowchart for the development of the AI-Enabled Essay Grading System.

3.1 Dataset Description

The dataset used in this study was an original dataset collected from students across different universities. Students were assigned questions and asked to submit their responses, which were subsequently graded using predefined assessment criteria. The collected responses and corresponding grades formed the dataset used in the study. The dataset contains 13,644 records. The columns represent the features of the records in the dataset, such as comprehension, question, examiner score, etc. The collected dataset is used to ensure accurate prediction of examiner's score and student's score.

Received: 22/08/2026; Revised: 31/08/2026; Accepted: 01/09/2026; Published: 07/09/2026

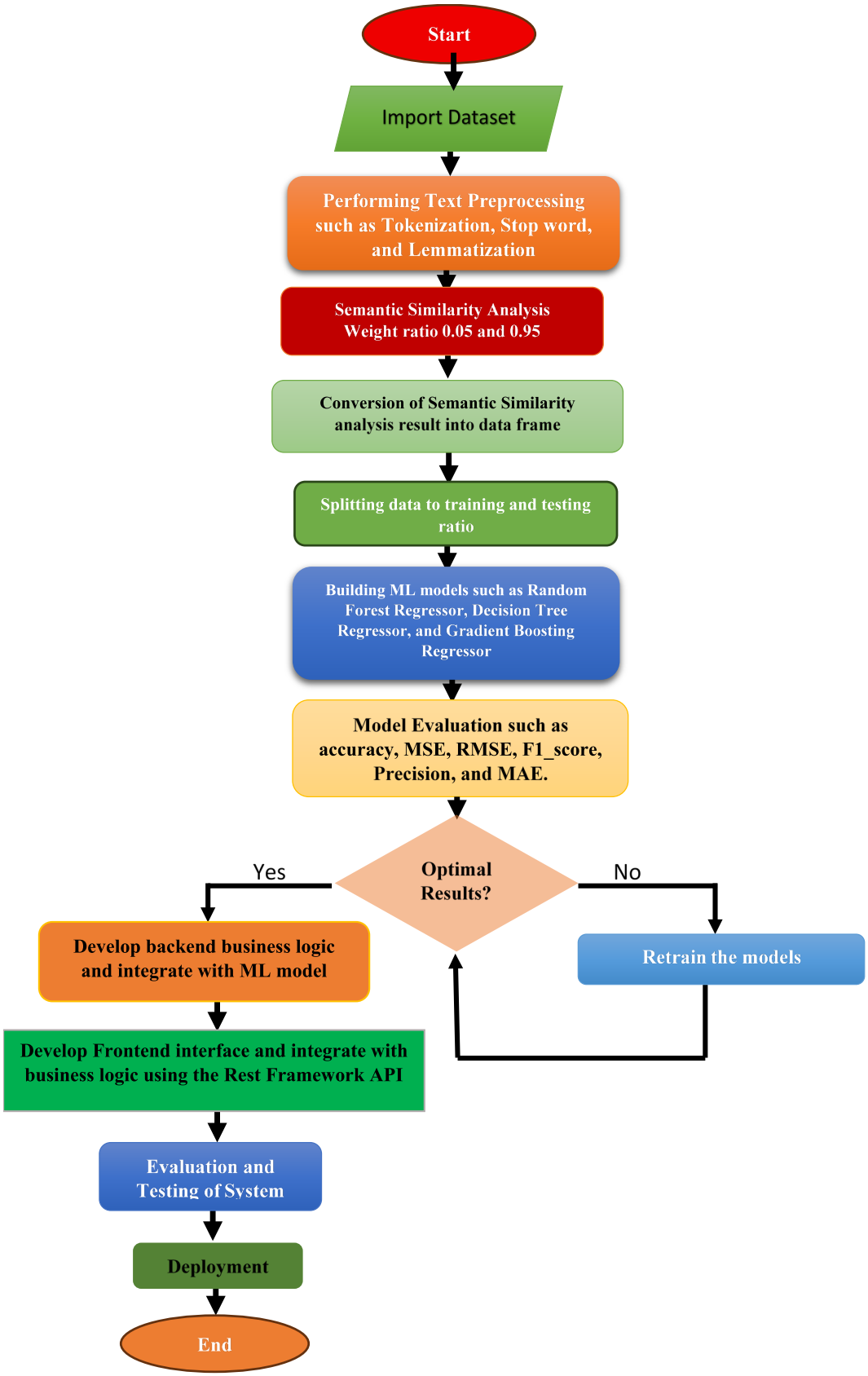

Figure 2: Process Flowchart

3.2 Dataset Processing

It is important to carry out the preprocessing stage when developing machine learning models. This ensures high performance of the model. The preprocessing procedures include data cleaning, data structuring, and NLP procedures.

Received: 22/08/2026; Revised: 31/08/2026; Accepted: 01/09/2026; Published: 07/09/2026

i. **Data Cleaning:** Data cleaning ensures the data is precise, comprehensive, and operational for decision-making. Some of the tasks in data cleaning include elimination of errors, handling missing values, standardizing data, and restructuring data. The key steps include Noise Reduction **(**excess whitespaces, special characters such as #,$%, and HTML tags were removed) and Data Normalization **(**all duplicate entries were removed and missing values were handled).

ii. **NLP Implementation:** NLP is employed to abstract meaningful features from text, ensuring its suitability for developing machine learning. The techniques used in this study include:

a. Tokenization: Tokenization was carried out in this study using the Natural Language Toolkit (NLTK) library. The process of breaking up dense text into smaller, simpler text units called tokens is called tokenization. These tokens are used to represent words or sentences, which makes analysis easier.

b. Lemmatization: Lemmatization ensures consistency across various word forms by reducing words to their original or root form. Words like "working," "worked," and "works," for example, have become synonymous with "work." By treating related terms as the same, this procedure further simplifies the model.

c. Stop Word Analysis: Common words like "as," "is," and "in" that convey less semantic meaning are known as stop words. These words are eliminated to guarantee noise reduction and increased dataset efficiency. This stage ensures that the model will highlight the text's most instructive terms.

3.2.1 Semantic Similarity Analysis

The need for implementing semantic similarity analysis is very crucial when using NLP. This ensures high quality of answer. Semantic Similarity Analysis assesses the closeness in context of meaning among the students' answers, and not focusing on word matching. This study implemented the three major processes of semantic similarity analysis. The processes are feature abstraction, text representation, and similarity computation. The three processes are dependent on one another to ensure accurate computation of similarity.

i. **Feature Abstraction:** This process aims at providing a structured representation of the meaning of each word. This study identifies the semantic features and extracts appropriate semantic features. The extracted features are weighted and represented in vector format. All these are known as Semantic Feature Analysis (SFA).

ii. **Text Representation:** This process utilizes tools such as word2Vec and GloVe to capture the semantic connection between words. Immediately after the features have been extracted from the text, it is necessary to be converted to a format that would be easy for similarity computation, which most of the time is the numerical format. This can either be vectors or matrices.

iii. **Similarity Computation:** This is the final process after the extraction of features and representation of text. The semantic similarity is calculated in this phase using cosine similarity, which measures the angle between two vectors. The process combines different methods, such as cosine similarity and WordNet-based measures. Thereby computing the average of each outcome.

3.3 Model Building

This section discusses the process involved in developing and evaluating the machine learning models to predict examiners' answers and students' answers. A data splitting ratio of 90:10 was used to evaluate the performance of various machine learning models. Splitting of dataset to training and testing ratio was done by means of these ratios. The model with the best performance was selected for additional development. Python libraries such as Scikit-learn, NumPy, pandas, and NLTK were used for the development. Three machine learning models were used in this study, which are Random Forest Regressor, Decision Tree, and Gradient Boosting Regressor.

3.3.1 Decision Tree Regressor

This study develops a Decision Tree Regressor model that predicts continuous values by undergoing the process of data segmentation recursively, considering feature values. A decision tree, as the name implies, is structured in a tree-like structure that consists of nodes. It minimizes the mean squared error by splitting the generated dataset into leaves, respectively. The mathematical model for this is given as:

$$E = -\sum_{i=1}^{c} p_i \, \mathrm{x} \log(p_i) \tag{1}$$

where p_i is the proportion of observations with class label i, which ranges from 1 to c.

The mathematical equation for decision tree regression is given in equation 2 as the average of the target variable in each region. For all training samples, the prediction for a given input is the mean value of the target variable.

Received: 22/08/2026; Revised: 31/08/2026; Accepted: 01/09/2026; Published: 07/09/2026

$$C_m = \frac{1}{|R_m|}\sum_{x_i \in R_m} y_i \quad (2)$$

where $|R_m|$ denotes the quantity of trials in region R_m, y_i is the equivalent target rate, and x is the input.

It is possible in regression to reduce the variance of the target variable. Therefore, splits are selected. Equation 3 gives the scientific modelling for this process:

$$Variance\ Reduction = Var(t)\left(\frac{N_L}{N}Var(R_L) + \frac{N_R}{N}Var(R_R)\right) \quad (3)$$

In equation 3, $Var(t)$ represents the modification of the target variable at the level of the existing node, N denotes the overall count of samples at the existing node, N_L and N_R signifies count of samples in the left and right child nodes, respectively, and $R_L and\ R_R$ represent the left and right regions formed.

3.3.2 Random Forest Regressor

Random Forest Regressor uses a collection of decision trees to carry out prediction on a continuous numerical dataset. Multiple decision trees are trained on different subsets of the training dataset. The final output is obtained through the process of averaging in order to reduce the effect of overfitting. The best among the subset of predictors is randomly selected to split the node. It trains numerous trees on different bootstrap samples. The overfitting of the model is reduced, and randomness is also applied in carrying out the sampling and feature selection at each fragment.

$$y(x) = \frac{1}{N}\sum_{i=1}^{N} h_i(x) \quad (4)$$

where $y(x)$ represents the final prediction for input x, $h_i(x)$ denotes the prediction from the ith decision tree, and N is the total number of trees in the forest.

3.3.3 Gradient Boosting Regressor

Gradient Boosting Regressor is implemented in this study to ensure successive decision trees fit into one training example while selecting the best tree with a low loss function. Another ensemble model is the Gradient Boosting Regressor (GBR), which is an iterative collection of tree models stacked sequentially so that the subsequent model learns from the previous model's error. In order to create a more robust model, this machine learning model "boosts" the ensemble of weak prediction models, frequently decision trees. The mathematical equation for this model is given below:

$$y(x) = \sum_{m=1}^{M} \gamma_m h_m(x) \quad (5)$$

where $y(x)$ is the predicted result, $h_m(x)$ is the prediction from the mth weak learner, which is the decision tree, γ_m is the step magnitude, and M represents the quantity of boosting repetitions.

Gradient Boosting model undergoes updates during each iteration. The model updates are represented as follows:

$$F_m(x) = F_{m-1}(x) + \gamma_m h_m(x) \quad (6)$$

where $h_m(x)$ is trained for the purpose of fitting the negative gradient of the loss function, and F_{m-1} signifies the existing ensemble prediction.

3.4 Evaluation Metrics

Some evaluation metrics were used to evaluate the models' performance. The metrics are as follows:

i. **R-squared:** is a statistical measure used in regression to show how well a model explains the variation in the target variable. It ranges from 0 to 1. The mathematical model to represent accuracy is given below:

$$R^2 = 1 - \frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - \bar{y})^2} \quad (7)$$

where y_i denotes actual value, $\hat{y}_i$ is predicted value and $\bar{y}$ represents the mean of actual values

ii. **Mean Squared Error (MSE):** This metric quantifies the difference between the predicted student score and actual score, then squares it. It is represented mathematically as:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \quad (8)$$

where n denotes the number of fitted points, yi is the prediction, and $\hat{y}i$ represents the true value.

iii. **Mean Absolute Error (MAE):** This metric calculates the average of the differences between the predicted result and the definite result.

$$MAE = \frac{1}{n}\sum_{i-1}^{n}|y_i - \hat{y}_i| \quad (9)$$

Received: 22/08/2026; Revised: 31/08/2026; Accepted: 01/09/2026; Published: 07/09/2026

iv. **Root Mean Squared Error (RMSE):** RMSE measures the prediction error by providing the average scale. RMSE is measured as the square root of MSE.

$$RMSE = \sqrt{MSE} \quad (10)$$

4.0 Results and Discussion

The models developed were analyzed using performance evaluation metrics commonly used for regression algorithms. These metrics include R-squared score, MAE, MSE, and RMSE. The experiment in this study was conducted on a Kaggle Notebook with Keras provision, and the TensorFlow library was employed for building the two deep learning models.

4.1 Data Preprocessing and Exploratory Data Analysis (EDA)

The essay grading dataset comprised 13,644 question-answer instances with seven attributes: Question ID, comprehension passage, question, examiner answer, student answer, student score, and question score. Several data preprocessing techniques were employed. These include missing-value assessment, text preprocessing, semantic feature construction, and feature and target selection.

The dataset was first subjected to a missing-value assessment using Pandas, which confirmed that all seven attributes contained zero missing observations. Consequently, no imputation or record deletion was required. Since the primary data consisted of natural-language responses, a text preprocessing pipeline was subsequently applied to the examiner answers, student answers, and comprehension passages. The procedure involved conversion of text to lowercase, removal of non-alphanumeric characters, word tokenization, English stop-word removal, and WordNet-based lemmatization. The resulting normalized texts were used for semantic feature extraction. Using the spaCy en_core_web_md language model, semantic similarity was calculated between the student response and the examiner answer and between the student response and the comprehension passage. These similarity measures were combined using weights of 0.05 and 0.95, respectively, with greater emphasis placed on the relationship between the student's response and the comprehension context. The resulting semantic similarity score, together with the question score, constituted the predictor variables, while the human-assigned student score served as the target variable. Finally, the data were partitioned into training and testing subsets using a 90:10 ratio with a fixed random state of 42 to ensure reproducibility.

4.2 Performance Evaluation

The three models: Random Forest regressor, Decision Tree regressor, and Gradient Boosting Regressor were evaluated using different performance evaluation metrics. These metrics include the R-squared score and error-based evaluation metrics such as MAE, MSE, and RMSE. Table 2 and Figure 3 show the results of the evaluation using the R-squared score.

Table 2: Performance evaluation using R-squared score

S/N	Model	R-squared score
1	Random Forest Regressor	0.9658
2	Decision Tree Regressor	0.9717
3	Gradient Boosting Regressor	0.8812

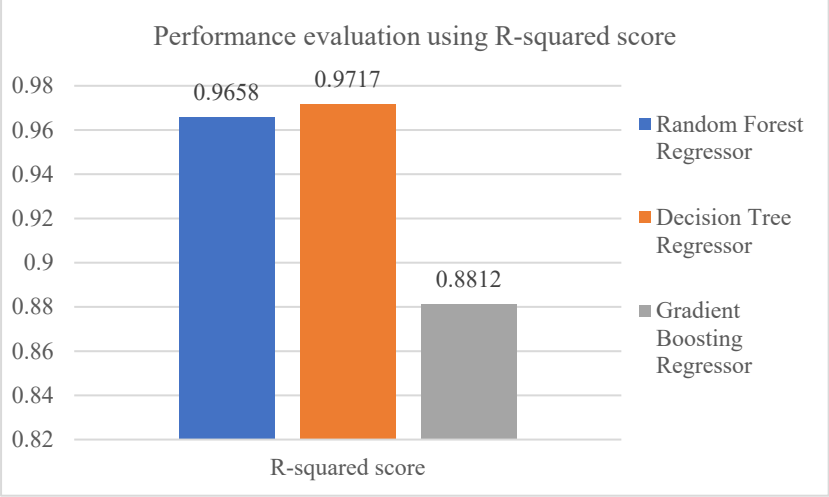

Figure 3: Performance evaluation using R-squared score

Received: 22/08/2026; Revised: 31/08/2026; Accepted: 01/09/2026; Published: 07/09/2026

The results show that the Decision Tree model performed best, achieving an R^2 value of 0.9717. This means that the model was able to explain about 97.17% of the variation in student scores. The Random Forest model also performed very well, with an R^2 of 0.9658, explaining approximately 96.58% of the variation in the scores. In comparison, the Gradient Boosting model recorded an R^2 of 0.8812, meaning that it explained about 88.12% of the variation in student scores. Overall, the results indicate that the Decision Tree provided the best fit among the three models based on the R^2 measure.

The models were further evaluated using three cost metrics. Table 3 and Figure 4 show the performance evaluation using error-based evaluation metrics.

Table 3: Performance evaluation using error-based evaluation metrics

S/N	Model	MAE	MSE	RMSE
1	Random Forest Regressor	0.1339	0.0850	0.2916
2	Decision Tree Regressor	0.1030	0.0703	0.2651
3	Gradient Boosting Regressor	0.2850	0.2956	0.5437

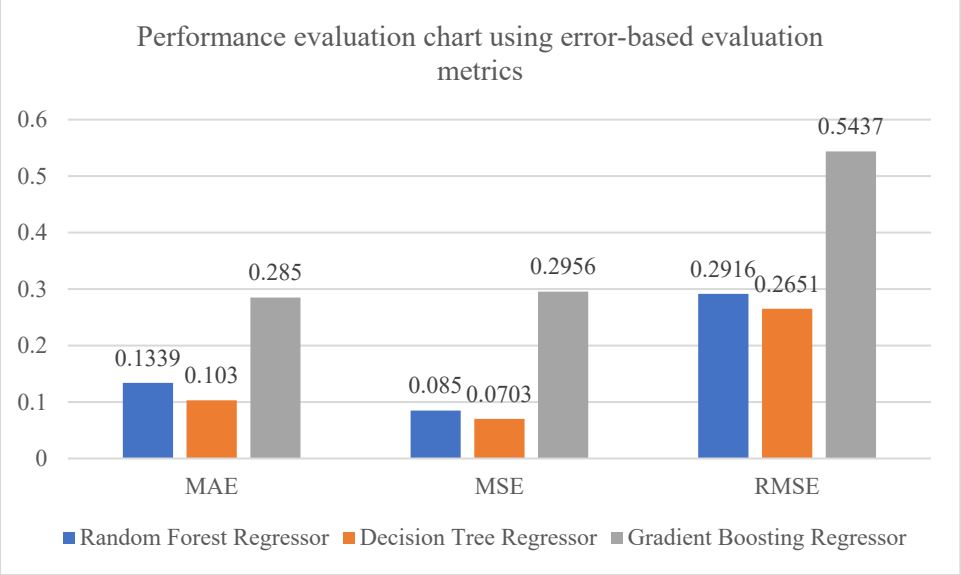

Figure 4: Performance evaluation chart using error-based evaluation metrics

The error-based evaluation further supports the strong performance of the Decision Tree Regressor. The model recorded the lowest MAE (0.1030), MSE (0.0703), and RMSE (0.2651), indicating that its predicted student scores were generally closest to the actual scores. The Random Forest Regressor also performed well, with an MAE of 0.1339, MSE of 0.0850, and RMSE of 0.2916, although its errors were slightly higher than those of the Decision Tree. The Gradient Boosting Regressor produced considerably higher error values, recording an MAE of 0.2850, MSE of 0.2956, and RMSE of 0.5437.

4.3 Discussion of Findings

The findings of this study demonstrate that machine learning models can effectively predict student scores from semantic information derived from student responses. The modelling process used two predictors: Semantic Similarity and Question Score, with Student Score serving as the target variable. The semantic similarity feature was generated by comparing the student's answer with both the examiner's answer and the comprehension passage. Greater emphasis was placed on the comprehension relationship, with weights of 0.05 for examiner-answer similarity and 0.95 for comprehension similarity. This approach provided a numerical representation of the semantic relationship between a student's response and the expected contextual content.

The overall performance evaluation results show that the Decision Tree provided the most consistent predictive performance, achieving both the highest R^2 value and the lowest prediction errors among the evaluated models. This suggests that the Decision Tree was the most suitable of the three models for predicting student scores in this study. The poorer performance of GBR model may indicate that the model did not obtain the same benefit from the two available predictors as the tree-based models under the configuration used in the experiment. The result should not necessarily be interpreted as evidence that GBR model is generally inferior to Decision

Received: 22/08/2026; Revised: 31/08/2026; Accepted: 01/09/2026; Published: 07/09/2026

Trees; rather, it indicates that under the experimental conditions and feature representation used in this study, the Decision Tree achieved better predictive performance.

The results show that the implementation of the model has practical implications for the development of an automated essay-grading system. The strong performance of the Decision Tree model suggests that semantic similarity between a student's response and the reference/contextual information, combined with the maximum score assigned to the question, contains substantial information for estimating the score awarded by a human examiner. The study demonstrates that the system can transform textual student responses into a numerical semantic representation and subsequently use that representation to estimate student scores. This provides evidence that semantic features can serve as useful predictors in an automated grading framework.

5.0 Conclusion

The study demonstrates that machine learning can be used effectively to predict student scores and support automated essay grading. Three machine learning models, including Random Forest, Decision Tree, and Gradient Boosting Regressor, were trained and tested. Among the models tested, the Decision Tree Regressor gave the best results, achieving the highest R^2 score of 0.9717 and the lowest prediction errors of 0.1030 MAE, 0.0703 MSE, and 0.2651 RMSE. This suggests that the combination of semantic similarity and question score provides useful information for estimating student performance.

Although the results obtained in this study are encouraging, the current approach relies mainly on semantic similarity and question score, which provide useful information about how closely a student's response relates to the expected content. However, similarity alone may not fully capture the overall quality and correctness of an answer.

Future improvements could therefore incorporate features such as grammatical correctness, relevance, completeness, and factual accuracy. Grammatical features could help assess the clarity and proper construction of students' responses, while relevance could determine whether the response directly addresses the question. Similarly, assessing completeness would help identify whether students have adequately covered the key points required in an answer. Factual accuracy would also be important in distinguishing between responses that appear semantically similar to the expected answer and those that actually contain correct information. By combining these additional characteristics with semantic similarity, the grading system could develop a more comprehensive understanding of student responses. Such an approach may improve the consistency of automated scoring and bring the system closer to the broader range of factors considered by human examiners when assessing students' answers.

Acknowledgement

The authors gratefully acknowledge the Tertiary Education Trust Fund (TETFund), Nigeria, for the financial support provided for this research under the Institution-Based Research (IBR) Intervention [Project No.: TETF/DR&D/CE/UNI/IBADANIBR/2023/VOL.I]. The authors also appreciate the management of Abiola Ajimobi Technical University, as well as the research assistants whose support and contributions facilitated the successful completion of this work.

References

[1] S. O. Kuyoro, J. M. Eluwa, J. E. T. Akinsola, F. Y. Ayankoya, A. A. Omotunde, and A. A. Adegbenjo, "Intelligent Essay Grading System using Hybrid Text Processing Techniques," *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol.*, vol. 6, no. 5, pp. 229–235, 2020, doi: 10.32628/cseit206547.

[2] A. A. Ibrahim, O. K. Rachael, A. O. Titilayo, G. O. Ajoke, M. B. Oyeladun, and F. A. Samuel, "Artificial Intelligence Based Essay Grading System," *Eng. Technol. J.*, vol. 09, no. 07, pp. 4382–4388, 2024, doi: 10.47191/etj/v9i07.06.

[3] K. Jonäll, S. S. Hashemi, and M. Lundin, "DEPARTMENT OF EDUCATION, COMMUNICATION AND LEARNING Artificial Intelligence in Academic Grading: A Mixed-Methods Study," 2024.

[4] S. R. Morris and M. Townsley, "Traditional vs. Modern-Grading Practices : A Comparative Case Study," *J. Educ. Leadersh. Action Manuscr.*, vol. 9, no. 3, 2025, doi: 10.62608/2164-1102.1169.

[5] W. Tiantian, "An automated english essay scoring system based on deep learning and the internet of things," *Discov. Artif. Intell.*, vol. 6, no. 64, pp. 1–29, 2026, doi: https://doi.org/10.1007/s44163-025-00731-w.

[6] Z. Chen, "Research on Automatic Essay Scoring of Composition Based on CNN and OR," *2019 2nd Int. Conf. Artif. Intell. Big Data*, pp. 13–18, 2019.

[7] K. Twabu and M. Nakene-Mginqi, "Developing a design thinking artificial intelligence driven auto-marking/grading system for assessments to reduce the workload of lecturers at a higher learning institution in South Africa," *Front. Educ.*, vol. 9, no. December, 2024, doi: 10.3389/feduc.2024.1512569.

[8] V. Suresh, R. Agasthiya, J. Ajay, A. A. Gold, and D. Chandru, "AI based Automated Essay Grading System

using NLP," in *2023 7th International Conference on Intelligent Computing and Control Systems (ICICCS)*, IEEE, May 2023, pp. 547–552. doi: 10.1109/ICICCS56967.2023.10142822.

[9] A. Singh, *Generative AI-powered Automated Essay Scoring System*. 2025.

[10] A. O. Ugbari, U. Chidiebere, and L. N. O, "The Evolution of Short Answer Grading Systems : From Manual Methods to AI-Driven Solutions with GPT-4," vol. 186, no. 30, pp. 33–39, 2024.

[11] J. E. T. Akinsola, M. A. Adeagbo, K. A. Oladapo, S. A. Akinsehinde, and F. O. Onipede, "Artificial Intelligence Emergence in Disruptive Technology," *Comput. Intell. Data Sci.*, pp. 63–90, 2022, doi: 10.1201/9781003224068-4.

[12] X. Li and C. Huang, "Design of an intelligent grading system for college English translation based on big data technology," *Syst. Soft Comput.*, vol. 7, no. November 2024, p. 200205, 2025, doi: 10.1016/j.sasc.2025.200205.

[13] T. Abid, A. Anaiza, A. Muhammadi, and M. F., "EssaySense: Essay Grading System Using NLP," *Int. J. Res. Publ. Rev.*, vol. 6, no. 5, pp. 11143–11146, 2025, doi: 10.2139/ssrn.5134717.

[14] A. Ayaan, "Automated grading using natural language processing and semantic analysis," *MethodsX*, vol. 14, p. 103395, Jun. 2025, doi: 10.1016/j.mex.2025.103395.

[15] Z. H. Amur and Y. K. Hooi, "State-of-the-Art: Assessing Semantic Similarity in Automated Short-Answer Grading Systems," *Inf. Sci. Lett.*, vol. 11, no. 5, pp. 1851–1858, 2022, doi: 10.18576/isl/110540.

[16] H. Ghanta, "Automated Essay Evaluation Using Natural Language Processing and Machine Learning," *Theses Diss.*, pp. 1–46, 2019.

[17] R. Dadi, S. Pasha, M. Sallauddin, C. Sidhardha, and A. Harshavardhan, "An overview of an automated essay grading systems on content and non content based," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 981, no. 2, 2020, doi: 10.1088/1757-899X/981/2/022016.

[18] H. Hallawi, H. Kadhim, Z. Abdullah, N. D. Al-Shakarchy, and D. A. Nasrawi, "User identification based on short text using recurrent deep learning," *IAES Int. J. Artif. Intell.*, vol. 12, no. 4, pp. 1812–1820, 2023, doi: 10.11591/ijai.v12.i4.pp1812-1820.

[19] J. Bolte, S. Sabach, M. Teboulle, and Y. Vaisbourd, "First order methods beyond convexity and lipschitz gradient continuity with applications to quadratic inverse problems," *SIAM J. Optim.*, vol. 28, no. 3, pp. 2131–2151, 2018, doi: 10.1137/17M1138558.

[20] J. E. T. Akinsola, O. Awodele, S. O. Kuyoro, and F. A. Kasali, "Performance Evaluation of Supervised Machine Learning Algorithms Using Multi-Criteria Decision Making Techniques," in *International Conference on Information Technology in Education and Development (ITED)*, Academia In Information Technology, 2019, pp. 17–34. [Online]. Available: https://www.academiainformationtechnology.org/ited2019/

[21] F. Y. Osisanwo, J. E. T. Akinsola, O. Awodele, J. O. Hinmikaiye, O. Olakanmi, and J. Akinjobi, "Supervised Machine Learning Algorithms: Classification and Comparison," *Int. J. Comput. Trends Technol.*, vol. 48, no. 3, pp. 128–138, 2017, doi: 10.14445/22312803/ijctt-v48p126.

[22] B. Olawoye, O. F. Fagbohun, O. Popoola-Akinola, J. E. T. Akinsola, and C. T. Akanbi, "A supervised machine learning approach for the prediction of antioxidant activities of Amaranthus viridis seed," *Heliyon*, vol. 10, no. 3, p. e24506, 2024, doi: 10.1016/j.heliyon.2024.e24506.

[23] M. Shahrizan *et al.*, "The Application of Decision Tree Classification Algorithm on Decision-Making for Upstream Business," no. August 2023, 2024, doi: 10.14569/IJACSA.2023.0140873.

[24] I. Mienye and N. Jere, "A Survey of Decision Trees: Concepts, Algorithms, and Applications," *IEEE Access*, vol. 12, pp. 86716–86727, 2024, doi: 10.1109/ACCESS.2024.3416838.

[25] A. L. and W. Matthew, "Classification and Regression by randomForest," *J. Dent. Res.*, vol. 83, no. 5, pp. 434–438, May 2004, doi: 10.1177/154405910408300516.

[26] M. Reza, S. Miri, and R. Javidan, "Random Forest Algorithm Overview," *Int. J. Adv. Comput. Sci. Appl.*, vol. 7, no. 6, pp. 1–33, 2016, doi: 10.14569/ijacsa.2016.070603.

[27] E. Díaz1 and G. Spagnoli, "Gradient Boosting Trees With Bayesian Optimization to Predict Activity From Other Geotechnical Parameters," 2023.

[28] D. A. Otchere, T. O. A. Ganat, J. O. Ojero, B. N. Tackie-Otoo, and M. Y. Taki, "Application of gradient boosting regression model for the evaluation of feature selection techniques in improving reservoir characterisation predictions," *J. Pet. Sci. Eng.*, vol. 208, p. 109244, Jan. 2022, doi: 10.1016/j.petrol.2021.109244.

Received: 22/08/2026; Revised: 31/08/2026; Accepted: 01/09/2026; Published: 07/09/2026