Journal of Studies in Science and Engineering

EngiScience Publisher

Research Article

Applying a New Feature Selection Method for Accurate Prediction of Earthquakes Using a Soft Voting Classifier

Oqbah Salim Atiyah[1, *], Mohammed Taher Ahmed [1], Kholood Jamal Mawlood [2], Noor Saud Abd [1]

[1] Department of Computer Science, College of Computer Sciences, University of Tikrit, Tikrit, 34001, Iraq

[2] Department of Mathematics, College of Education for Girls, University of Tikrit, Tikrit, 34001 Iraq.

*Corresponding Author: Oqbah Salim Atiyah, E-mail: oqbah_salim@tu.edu.iq

Article Info

Article History

Received Apr 14, 2024

Revised Sep 20, 2024

Accepted Sep 23, 2024

Keywords

Earthquakes

Voting Classifier

Machine Learning

Hyperparameter Optimisation

Novel Feature Selection Method

Classification Algorithms

Abstract

Earthquakes are among the most hazardous natural disasters, posing significant threats to infrastructure, property and human life. This is primarily due to the sudden nature of earthquakes, which often provide little to no time for preparation. Consequently, the issue of earthquake prediction is crucial for human safety. Developing a reliable and highly accurate earthquake prediction model using machine learning (ML) methods can enhance our understanding of these complex natural phenomena, ultimately aiding in preserving lives and mitigating earthquake-related damage. In this study, we propose a novel feature selection approach that integrates two methods: normalisation based on analysis of variance and the Chi-squared technique, along with correlation based on Logistic Regression (CLR-AVCH). This approach aims to identify the most relevant features to expedite model training, minimise errors and optimise outcomes. We employ three algorithms (Support Vector Machine, Decision Tree and Random Forest) to uncover and identify patterns in the collected data. A soft voting classifier is then constructed, combining the best-performing models (Decision Tree and Random Forest) to create a unified model that leverages both strengths, improving prediction accuracy. The proposed methodology achieves high-performance metrics, including accuracy, F1 score, recall and precision (0.99, 0.98, 0.98 and 0.98, respectively). Future work will focus on implementing new feature selection techniques alongside hybrid algorithms with soft voting classifiers to enhance diagnostic capabilities.

1. Introduction

An earthquake is defined as a mild to severe shaking of the Earth caused by the abrupt displacement of rocks beneath the surface. There are four primary types of earthquakes: volcanic, tectonic, explosion and collapse. Tectonic earthquakes occur at the boundaries of tectonic plates, resulting from geological forces that break the Earth's crust and alter the material and chemical composition. Although tectonic plates move slowly, friction can cause them to become trapped at their edges. An earthquake occurs when the stress at the edge exceeds the friction, releasing energy through waves that travel through Earth's crust, producing

the shaking we experience [1]. Like other natural disasters, earthquakes can cause significant damage, injuries and financial losses. They occur daily worldwide, with countries such as Indonesia, Japan, Turkey, Southern California, Taiwan and Iran particularly vulnerable.

People typically feel earthquakes with a magnitude greater than 2.5, while those below this threshold are not. Highly damaging earthquakes usually have a magnitude greater than 4.5 [2]. Such catastrophic events can lead to massive fatalities, prompting scientists to invest considerable effort into mitigating their negative impacts. Timely notifications are crucial, as inaccurate alerts can result in unnecessary losses. Predicting these events has become one of the most pressing challenges with the increasing frequency and likelihood of natural disasters, particularly earthquakes. Detecting earthquakes and ground noise in the field is difficult due to various factors, but the capabilities of artificial intelligence and advanced sensor technology can help identify subtle signals that may go unnoticed by humans. One advantage of machine learning is its ability to quickly extract signals obscured by noise, thereby reducing human losses and material damage. Machine learning techniques offer numerous capabilities supporting earthquake early warning systems [1], facilitating early detection and prompt responses to minimise damage.

While it is true that humans cannot prevent earthquakes, they can take preventative measures and safeguards to mitigate their negative impacts by employing ML approaches for predicting earthquake magnitudes [3]. Various techniques, including devices, sensors, electrical and magnetic waves, or seismic indicators derived from the analysis of historical earthquake data, can be used to estimate earthquake magnitudes [1]. Although no model can guarantee 100% accuracy, efforts are made to improve predictive accuracy as much as possible [4].

Several factors, such as the number of records in the dataset, the quantity of features (input indicators) and the nature of the problem (regression or classification), influence the suitability of a given ML algorithm. Therefore, we will utilise a variety of ML algorithms and compare their outcomes to identify the most appropriate for the task at hand [5]. Earthquakes are among the most destructive natural disasters globally, often resulting in severe injuries or loss of life. They typically occur suddenly. The goal of earthquake prediction is to use available data to identify three key elements: the location, timing and magnitude of future earthquakes. This approach is an effective means of reducing earthquake-related losses. Accurate earthquake prediction could significantly lessen seismic damage, making it crucial for nations and their citizens. Consequently, there is a growing interest in academic research focused on seismic event prediction [6].

2. Related Work

This section summarises prior research and describes various methods employed to predict and classify earthquakes:

Koehler, et al. [7] utilised a large dataset of earthquakes recorded by well-covered seismic stations in Japan's subduction zone. The study aimed to predict earthquakes using a classification method and applied a deep learning (DL) network to determine whether a time series lasting more than two years would culminate in an earthquake with a magnitude greater than five the following day. The authors developed a unique progressive training approach for the model, which was assessed using data from Japan from 2002 to 2020. The model achieved an overall accuracy of 72.3%. While this classification's accuracy surpassed the baseline, further improvements are necessary with additional data in the future [6].

An, et al. [8] conducted several simulation tests on missing data for earthquake prediction and proposed a recommendation system that combines the DIN model with MLP. This method integrates missing data handling with predictions regarding seismic stations based on the DIN model. The proposed method significantly enhances prediction accuracy compared to the original DIN model. Comparative experiments confirmed the technique's efficacy, demonstrating that the GAUC of the DIN–MLP model reached 0.69, an 11% improvement over the original DIN model. This highlights the algorithm's potential benefits for predicting earthquakes with missing data, although there remains a need to enhance monitoring accuracy and efficiency [8].

Sajan, et al. [9] evaluated the performance of each classifier concerning four popular ML prediction algorithms based on rehabilitation and damage scores. The researchers created and tested ML models using Random Forest, Decision Tree, Logistic Regression and XGBoost methods. The study found that the XGBoost algorithm predicted building collapse and strengthening with approximately 82% greater accuracy than other algorithms. However, there is still a need to develop prediction models to achieve the required accuracy [9].

Yang, et al. [10] proposed an automated regression model based on ML, called Auto-REP. Their contribution to Auto-REP lies in the automated development of a regression pipeline using laboratory seismic data, which ultimately yields predictions regarding laboratory earthquake occurrences. The automated process also incorporates modelling, optimisation, feature selection and feature extraction algorithms. It utilises a Bayesian approach for optimising the hyperparameters of the model. Previous experimental results indicated that the model achieved MSE and MAE results of 1.48, 1.51 and 1.52, 1.59 on the test and training datasets, respectively. The models require further improvement to enhance predictions of laboratory earthquakes [10].

Berhich, et al. [11] proposed a location-based earthquake prediction model that employs recurrent neural network (RNN) methods. To provide location-based predictions, a K-Means approach was used to cluster the seismic dataset according to geographic parameters (latitude and longitude). Each group was further divided into two distinct subgroups: seismic events with an average magnitude between 2 and 5 constituted the first group, while events with a magnitude greater than 5 formed the second group [11]. The

results of the models used to assess the model's strengths were not disclosed.

3. Material and Method

Many machine learning algorithms are designed for classification. This section presents the most common classification algorithms in machine learning and details relevant to this paper.

3.1. Random Forest (RF)

The Random Forest algorithm is one of the most popular machine learning supervised models, applicable to regression analysis and classification. The Random Forest combines many Decision Trees with the training dataset and employs a bagging technique for regression and classification tasks. A specific Decision Tree represents each class prediction, and the votes from these trees are aggregated. The class with the most votes is selected as the final class [12]. The Random Forest comprises a tree structure, with each tree belonging to the classifier groups contained within it. Let the Random Forest contain k trees of classifiers defined as h(x,Θn) for n = 1,2,...,k, where {Θn}n k = 1 is a set of independent random vectors distributed symmetrically, and the input is x. Each tree votes for the most popular category at input x [13]. Once the RF training is completed using k trees, the testing phase employs shared majority voting among these distinct trees:

$$H(x) = argmaxY \sum k\ i = 1\ I(hi(x) = Y) \quad (1)$$

$H(x)$ is a classification model mixture, with the input variable Y, hi representing the model of one Decision Tree, and (I) being the indicator task [12]. The Random Forest classifier is a collection of methods that trains numerous Decision Trees through parallel bootstrapping followed by aggregation, all defined as bagging. The individual Decision Tree ensemble contributes to the final decision made by the Random Forest classifier.

3.2. Decision Tree (DT)

A Decision Tree is employed to address regression and classification problems and is classified as a supervised algorithm. The Decision Tree aims to create a training model that predicts class outcomes by learning simple decision rules based on previous data. Decision Tree learning is a widely used predictive modelling technique in data mining, machine learning and statistics. It utilises a Decision Tree to derive conclusions about the value of an item based on its attributes [14]. Classification involves a target variable that can assume a discrete set of values within a tree structure. The nodes and leaves of the tree correspond

to class names, while the branches represent the attributes leading to those class labels. Regression trees are Decision Trees where the target variable can take continuous values.

Decision Trees are popular in machine learning due to their simplicity and clarity, enabling well-defined decision-making. The model is trained using the training data, which is then used to predict outcomes for the test data during the prediction phase [15]. Algorithms that create Decision Trees begin from the top down, selecting a variable at each step and partitioning the set of elements accordingly [16]. Different algorithms employ various metrics to determine the best variable for partitioning. These metrics assess the subgroup of the target variable for matching. The Gini impurity (named after Italian mathematician Gini Corrado) is commonly used in these algorithms to determine the frequency of incorrect classifications from the set. If the distribution of labels in the subgroup is randomised, Gini impurity can be calculated for a set of items using the following equation:

$$I_G(p) = 1 - \sum_{i=1}^{j} p_i^2 \tag{2}$$

Decision Tree algorithms utilise the concept of entropy to obtain information content. Entropy is defined as follows:

$$H(T) = -\sum_{i=1}^{j} pi \; log_2 \, pi \tag{3}$$

where *pi* is a fractional number that sums to 1, representing the percentage of each class present in the child node resulting from the split in the tree.

$$= -\sum_{i=1}^{j} pi \; log_2 \, pi - \sum_{i=1}^{j} -pr(i|a) log_2 \, pr(i|a) \tag{4}$$

3.3. Support Vector Machine (SVM)

A support Vector Machine is a supervised machine learning algorithm that determines the new data class for discrete and continuous tag data problems. It retains training data to predict scores by calculating similarities between the training instances and the input data. Introduced in 1995, the SVM algorithm addresses various classification and regression problems. This method is based on statistical learning theory [17], making it one of the most common classification techniques. As previously mentioned, SVM is a supervised classification technique that provides sequences of inputs with their labels, defined by the indicator properties of these inputs [18]. It aims to determine the excess level representing the largest category margin. The vector feature of these inputs defines the structures. This technique generates superior recognition of the two groups to separate categories completely. It comprises two vectors parallel to the classifier, with the distance between these parallel vectors referred to as the margin. The edge vectors are known as

support vectors. SVM seeks to partition the data using the hyperplane and extend this to nonlinear boundaries using the kernel trick [19]. The following equations are used for classification:

$$W = \sum n\, i = 1\, ai\, yi\, xi = 0 \quad (5)$$

$$WT\, x + b = 0; \quad \text{x is on the line} \quad (6)$$

$$WT\, x + b > 0; \quad \text{x is above the line} \quad (7)$$

$$WT\, x + b < 0; \quad \text{x is below the line} \quad (8)$$

where w is the hyperplane vector. The radial basis function of the kernel is calculated as follows:

$$Y = WT\,(x) + b \quad (9)$$

To achieve partitioning, the data must be greater than zero. Among all possible hyperplanes, SVM selects the one with the largest margin. If the training data is extensive and all test vectors are found within a radius r of the training vector, the currently defined hyperplane is positioned as far away from the data as possible [20].

4. Methodology

This study proposes a methodology that employs the voting classifier method, which combines multiple models to achieve optimal prediction accuracy for earthquake diagnosis. The following subsections provide further details on the voting classifier. The proposed methodology consists of three main phases: the preprocessing phase, feature selection and prediction phase, as illustrated in Figure 1. Before detailing these phases, we will describe the dataset used in this study. The earthquake dataset from the Kaggle repository contains two classes: tsunami (1) for events in oceanic regions and otherwise (0). It includes numerical and categorical data stored in a CSV format, comprising 78,219 records of earthquakes from 2001 to 2023 [21]. The dataset includes the following columns:

1. date-time: time and date
2. magnitude: the earthquake's magnitude
3. title: title name assigned to the earthquake
4. alert: the alert level – 'yellow', 'green', 'red' and 'orange'
5. mmi: the maximum estimated instrumental event intensity
6. cdi: the maximum reported intensity regarding the event range
7. tsunami: '1' for events in oceanic regions and '0' otherwise

8. sig: a number that describes the significance of the event. The larger the number, the more significant the event. Multiple factors, including magnitude, determine this value, felt reports, maximum MMI and estimated impact
9. net: data contributor ID, identifying the preferred information source for this event
10. nst: the total number of seismic stations utilised to determine the earthquake's location
11. dmin: horizontal distance from the epicentre to the nearest station
12. gap: the largest azimuthal gap between azimuthally adjacent stations (in degrees). Generally, the smaller this number, the more reliable the estimated horizontal position of the earthquake. Earthquakes with an azimuthal gap greater than 180° typically have significant location and depth uncertainties
13. magType: the algorithm or method used to calculate the preferred magnitude for the event
14. depth: the depth at which the earthquake begins to rupture
15. latitude: the coordinate system used to describe and determine the location of any place on the Earth's surface
16. longitude: also a coordinate system used to describe and determine the location or position of any place on the Earth's surface
17. location: location within the country
18. continent: the continent where the earthquake occurred
19. country: affected country

4.1. Preprocessing Phase

In this phase, we perform a series of initial operations on the dataset to enhance data quality and ensure the classification model functions effectively. The primary operations in this phase include cleaning, balancing and label encoding of the dataset.

4.2. Analysing Dataset

Data analysis involves several steps but will focus on the most critical stages. First, we check for missing values: this step is essential to determine if the data contains missing entries, which can be addressed by replacing numeric missing data with the mean or categorical missing data with neighbouring values. Table 1 illustrates the missing data before and after handling. Second, we assess the data types of the features and identify which features (columns) are beneficial for model development, as shown in Table 1. We will drop features that do not contribute to prediction, such as title, location, country or continent.

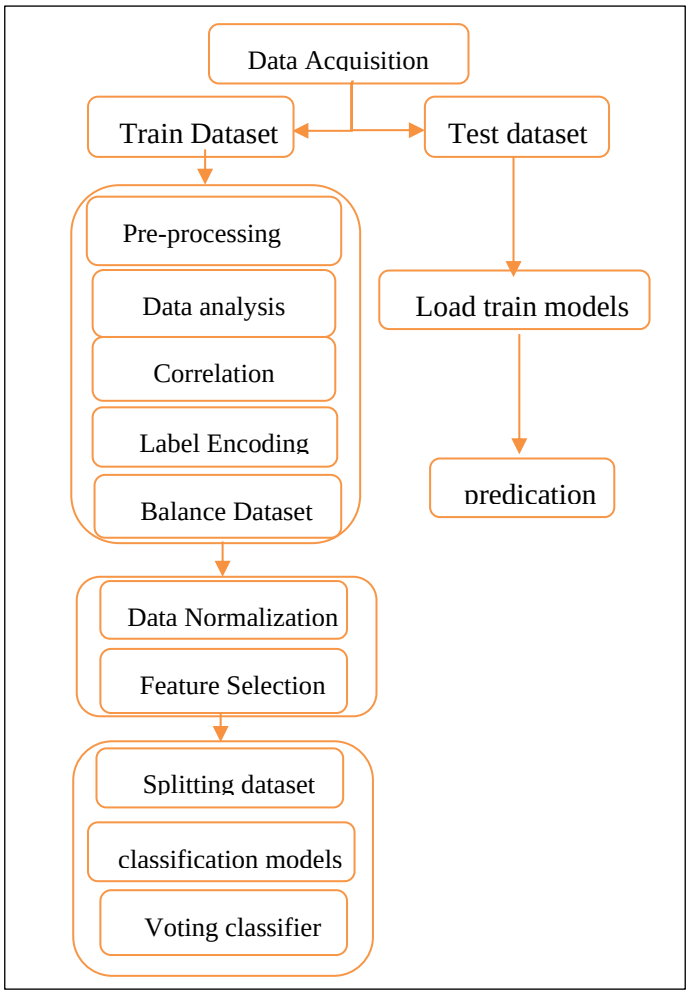


Figure 1. Depicts a system workflow

4.3. Correlation

Correlation indicates the relationship between pairs of variables. A correlation matrix is plotted to illustrate the degree of correlation between variables [22]. Figure 2 displays the correlation heatmap among features, with the scale measuring the correlation degree across all features, where correlation values range from [-1,1]. A score of (1) indicates a perfect positive correlation between two features, a score of (0) indicates no correlation, and a score of (-1) indicates an inversely proportional correlation.

4.4. Data Encoding

Ratio scale variables and qualitative factors in the classification process typically influence the dependent variables. Consequently, these categorical variables must be transformed into numerical values using encoding techniques, as ML algorithms only accept numerical inputs [23, 24]. The categorical columns are transformed to obtain numeric values, as illustrated in Figure 3. In this stage, label encoding will encode the three categorical variables.

Table 1. Summary of the analyzing dataset

Seq.	Features	Missing data	After handling	Types
0	Title	0	0	Object
1	Magnitude	0	0	float64
2	date time	0	0	Object
3	Cdi	0	0	int64
4	Mmi	0	0	int64
5	Alert	367	0	Object
6	Tsunami	0	0	int64
7	Sig	0	0	int64
8	Net	0	0	object
9	Nst	0	0	int64
10	Dmin	0	0	float64
11	Gap	0	0	float64
12	magType	0	0	Object
13	Depth	0	0	float64
14	Latitude	0	0	float64
15	Longitude	0	0	float64
16	Location	0	0	Object
17	Continent	0	0	Object
18	Country	0	0	Object

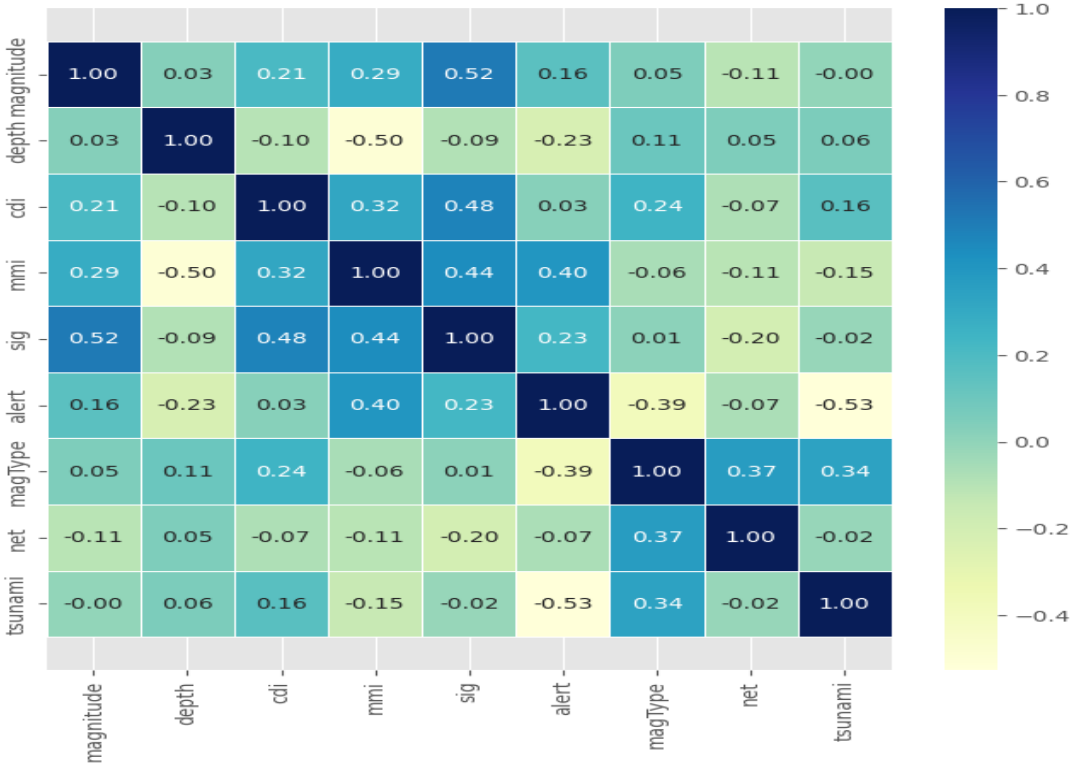


Figure 2. Shows the correlations

4.5. Balancing Dataset

Unequal categories in an unbalanced dataset can lead to bias due to subjective interpretation or underestimation of specific characteristics or categories. Bias refers to the systematic deviation of data from the true value, resulting in distorted results and inaccuracies in measurement tools or techniques used for prediction, ultimately leading to incorrect conclusions. For the classification model, balancing the dataset is crucial to achieving higher accuracy without bias. An imbalanced dataset can hinder the classification and training stages, as classifiers may have insufficient data to understand the features of a particular class. The Synthetic Minority Oversampling (SMOTE) technique is one of the most effective methods for balancing datasets, helping mitigate the overfitting problem of simple oversampling. SMOTE employs nearest neighbour algorithms to generate synthetic new data for training models. Unlike standard up-sampling, this paper utilises SMOTE to generate new data points for minority classes, thereby balancing the dataset and increasing the likelihood of successful learning [25]. Figure 4 illustrates the dataset before and after balancing.

4.6. Data Normalisation

Normalisation is a technique commonly applied during data preparation for machine learning in the preprocessing stage. Normalisation aims to scale features to a similar range, enhancing the model's functionality and training stability while improving data integrity and accuracy [26]. We employed the MinMaxScaler function, which scales each feature individually to a specified maximum and minimum value, with default values of 1 and 0.

4.7. Feature Selection Phase

Before selecting the model that best fits our dataset, choosing the appropriate features for training the model to achieve optimal results is essential. Reducing redundant data enhances modelling accuracy, minimises misleading data and results in faster algorithms. Thus, the primary goal of feature selection is to improve accuracy, reduce training time and decrease overfitting [27]. In this phase, we present a proposed method that combines techniques from the filter method, correlation based on Logistic Regression with normalisation and analysis of variance (ANOVA), and the Chi-squared technique (CL-ANCH).

Figure 5 illustrates the proposed method of our work. First, correlation analysis with Logistic Regression employs the function (logis. coef) to reveal the correlation between features and the target value, producing a distinct set of correlation factors (unrelated to the correlation matrix).

	magnitude	date_time	cdi	mmi	alert	tsunami	sig	net	nst	dmin	gap	magType	depth	latitude	longitude
0	7.0	22-11-2022 02:03	8	7	green	1	768	us	117	0.509	17.0	mww	14.000	-9.7963	159.596
1	6.9	18-11-2022 13:37	4	4	green	0	735	us	99	2.229	34.0	mww	25.000	-4.9559	100.738
2	7.0	12-11-2022 07:09	3	3	green	1	755	us	147	3.125	18.0	mww	579.000	-20.0508	-178.346
3	7.3	11-11-2022 10:48	5	5	green	1	833	us	149	1.865	21.0	mww	37.000	-19.2918	-172.129
4	6.6	09-11-2022 10:14	0	2	green	1	670	us	131	4.998	27.0	mww	624.464	-25.5948	178.278
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
777	7.7	13-01-2001 17:33	0	8	red	0	912	us	427	0.000	0.0	mwc	60.000	13.0490	-88.660
778	6.9	10-01-2001 16:02	5	7	red	0	745	ak	0	0.000	0.0	mw	36.400	56.7744	-153.281
779	7.1	09-01-2001 16:49	0	7	red	0	776	us	372	0.000	0.0	mwb	103.000	-14.9280	167.170
780	6.8	01-01-2001 08:54	0	5	red	0	711	us	64	0.000	0.0	mwc	33.000	6.6310	126.899
781	7.5	01-01-2001 06:57	0	7	red	0	865	us	324	0.000	0.0	mwc	33.000	6.8980	126.579

782 rows × 15 columns

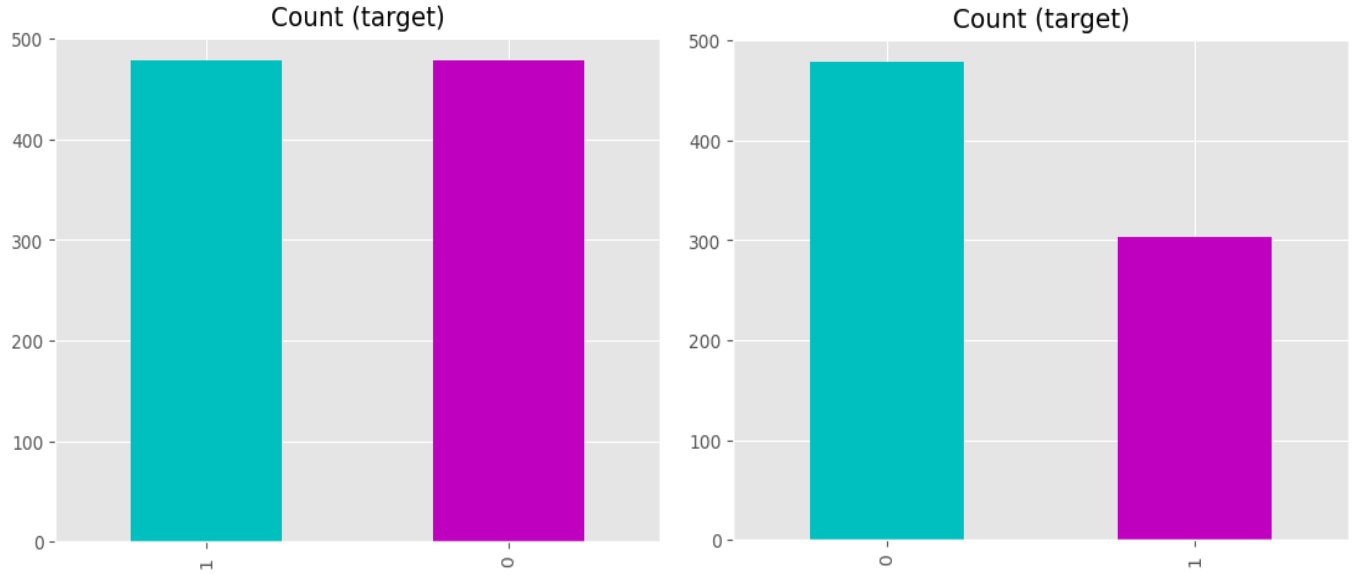
(a) Data before encoding

	magnitude	date_time	cdi	mmi	alert	tsunami	sig	net	nst	dmin	gap	magType	depth	latitude	longitude
0	7.0	11	8	7	0	1	768	9	117	0.509	17.0	8	14.000	-9.7963	159.596
1	6.9	11	4	4	0	0	735	9	99	2.229	34.0	8	25.000	-4.9559	100.738
2	7.0	12	3	3	0	1	755	9	147	3.125	18.0	8	579.000	-20.0508	-178.346
3	7.3	11	5	5	0	1	833	9	149	1.865	21.0	8	37.000	-19.2918	-172.129
4	6.6	9	0	2	0	1	670	9	131	4.998	27.0	8	624.464	-25.5948	178.278
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
777	7.7	1	0	8	2	0	912	9	427	0.000	0.0	7	60.000	13.0490	-88.660
778	6.9	10	5	7	2	0	745	0	0	0.000	0.0	5	36.400	56.7744	-153.281
779	7.1	9	0	7	2	0	776	9	372	0.000	0.0	6	103.000	-14.9280	167.170
780	6.8	1	0	5	2	0	711	9	64	0.000	0.0	7	33.000	6.6310	126.899
781	7.5	1	0	7	2	0	865	9	324	0.000	0.0	7	33.000	6.8980	126.579

782 rows × 15 columns

(b) Data after encoding

Figure 3. Depicts the data encoding



(a)Imbalanced dataset (b) Balanced data

Figure 4. Shows apply SMOTE technique

We collect the highly correlated features that share common elements into one set. Our processing retains the common features with the best values and discards the remaining features in each group. Second, we apply correlation for numerical data using logistic regression with normalisation and ANOVA, a statistical technique for comparing variances through group averages. The P-value resulting from ANOVA indicates the difference between the group variance and within-group variance. The variance between correlated features is calculated using the ANOVA function, which determines the difference between every two interconnected features and selects values between 0.05 and 0, which are significant as they are close to 0 or equality, thus preventing dominance in classification. This ultimately produces a value that allows us to conclude whether the null hypothesis is rejected or supported. A large difference between features results in a larger P-value, leading to the rejection of the null hypothesis. This model enhances data importance, rendering all numerical features significant variables due to the P-value of ANOVA < 0.05.

Table 2 displays the feature selection results for numerical data. For categorical data, correlation is applied using logistic regression with normalisation and the Chi-squared technique, a statistical test used to examine the variance between the correlation of randomly selected categorical features to assess the fit between observed and expected results. Candidate feature variables are removed when they are irrelevant to the problem, meaning correlation between the categorical features is calculated, and values between 0.05 and 0 are selected, indicating ideal correlation. This model demonstrates that all categorical features are significant, as the Chi-squared P-value is < 0.05.

Proposed method first identifies features with correlation values less than 1 by analysing the correlation heatmap of features shown in Figure 2. Second, it merges sets of interrelated features containing common elements into one group. Third, we apply ANOVA and Chi-squared techniques to compare variances through the averages of different groups and identify P-values of features less than 0.05. Finally, the remaining features in each group are removed from the dataset (Table 2). Figure 5 illustrates the proposed method of our work. The outcome of the suggested approach for feature selection is eliminating unimportant features that hinder the ML model's performance while retaining features that facilitate accurate learning and optimal classification accuracy.

4.8. Splitting the Dataset

After preprocessing the dataset and appropriately selecting features, the dataset is ready for predictions using ML algorithms. In this section, we split the dataset (training = 0.8 and testing = 0.2) using k-fold cross-validation (k = 5).

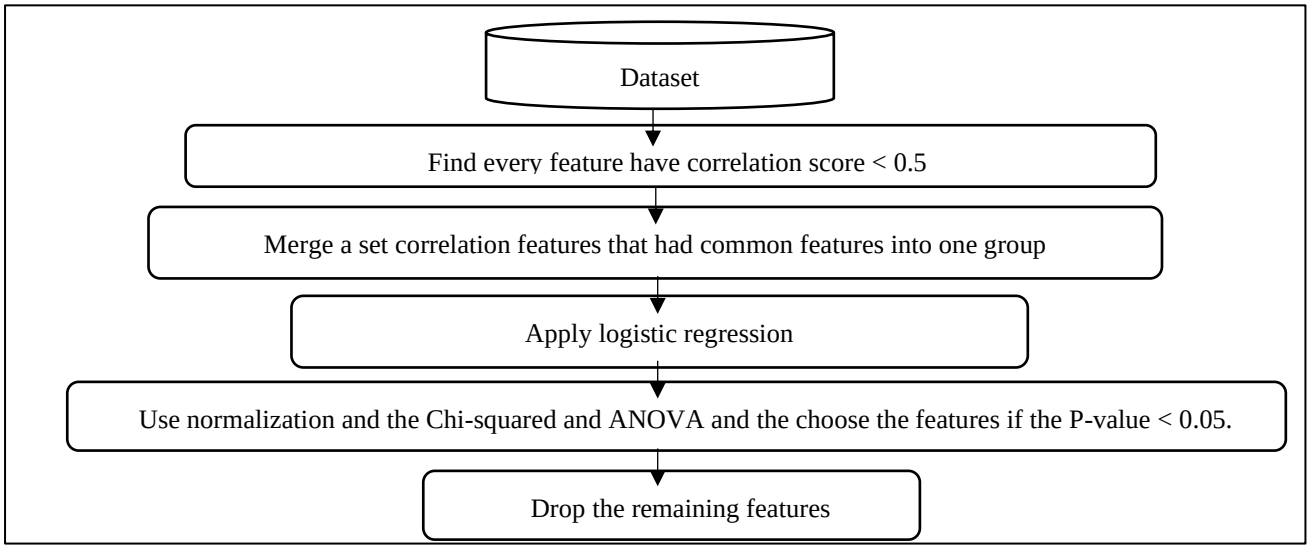


Figure 5. Shows the proposed method

Table 2. Display the Feature Selection Phase

Features	With feature selection	With proposed method
magnitude	1.44	0.022
depth	3.02	0.002
mmi	-0.25	0.003
sig	3.08	0.01
alert	-0.078	0.03
magType	-3.40	0.04
net	-5.8	0.007
tsunami	0.457333	0.004

4.9. Applying Models and Results

The Scikit-Learn library was employed as one of the most important libraries in machine learning, providing simple and effective tools for data analysis and predictive model building, including classification, regression, clustering, dimensionality reduction and model selection. Each algorithm is implemented in a consistent programming format, allowing users to switch between different algorithms and compare their performance. Scikit-Learn supports numerous preprocessing techniques, evaluation metrics and model validation methods, making it a comprehensive toolkit for machine learning [28].

After completing data preprocessing, the dataset was imported into Jupyter Notebook using Python code, and the findings were subsequently reviewed and explained. The imported data was scanned for missing values using the isnull() sum() function, and outliers were identified and addressed appropriately during the modelling stage. The dataset employed for this study is reliable, with tsunami labelled as '1' for

events in oceanic regions and '0' otherwise. Categorical data is encoded, as the selected algorithms perform optimally with scalar values. The dataset was split after these actions, and ML algorithms were applied for predictions. To identify algorithms with high accuracy, random search methods were employed to facilitate rapid training for hyperparameters. Random search selects values randomly from a specified set of numbers, attempting various hyperparameters and measuring the model's performance. Ultimately, it identifies the parameters that yield the best results. This method reduces unnecessary attempts by computing a fixed number of hyperparameters. Random search provides better and faster results than other methods, and hyperparameter tuning was performed using a weights package. Table 3 presents the tuned hyperparameters of the algorithms [28].

Table 3. The hyperparameters were tuned

Algorithm	Hyperparameters	Value
Random forest	n_estimators	100
	max_leaf_nodes	9
	max_features	sqrt
	max_depth	9
Decision Tree	criterion	entropy
	max_features	8
	max_depth	3
	min_samples_leaf	7
Support Vector Machine	kernel	rbf
	gamma	0.1
	C	10

4.10. Voting Classifier

The voting classifier is an ensemble classifier that relies on AI models, combining a specific set of models to produce a single model that incorporates the strengths of the combined models, resulting in optimal prediction accuracy [29].

After applying ML algorithms to the dataset, the algorithms selected for this purpose were those with the highest accuracy, ensuring their results were closely aligned to prevent confusion regarding the strengths and weaknesses of the classifiers. Consequently, a soft voting classifier was generated, incorporating only the strengths of the classifiers. The algorithms that achieved the highest results were chosen (Decision Tree and Random Forest).

Here, we use a soft voting classifier and input two ML models (Decision Tree and Random Forest), which yielded the best results when used with a voting classifier on this dataset based on a series of experiments. This classifier operates probabilistically, with each input model producing a probability value for class 0 and class 1. In the final result, the soft voting classifier utilises the highest probability from all input models, as illustrated in Figure 6. In summary, our proposed methodology involves initial treatments to enhance the dataset, followed by selecting the best features through the proposed method, which is then utilised by the soft voting classifier to achieve the optimal classification of earthquake types, specifically whether they are tsunamis.

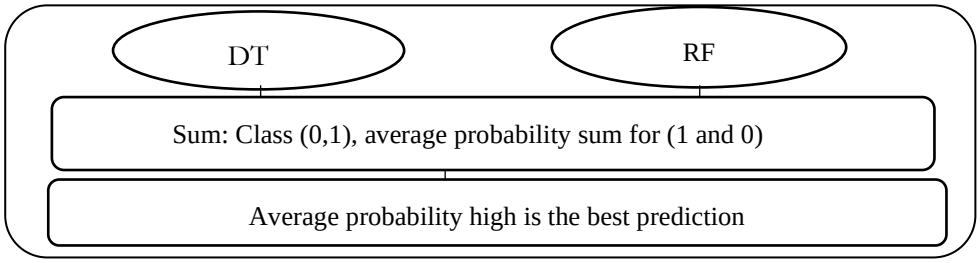


Figure 6. Display the proposed model (soft voting classifier)

4.11. Evaluation

Three commonly utilised algorithms are employed to assist in finding patterns in the acquired data once testing is complete and the trained model has been loaded, as previously described. At this stage of the investigation, the outcomes produced by the algorithms will be presented and discussed. The performance of each algorithm or approach was measured to assess its effectiveness, and the analysis findings were computed for this study based on the confusion matrix. Several metrics were employed, including accuracy, which is significant in cases where incorrect predictions may have serious consequences affecting decision-making, as it measures the proportion of correctly predicted positive cases out of all positive predictions made by the model. This metric helps us understand the model's performance in determining the best model; however, imbalanced datasets can influence accuracy, where the number of positive instances is much smaller than the number of negative instances. In such cases, the model may achieve high accuracy simply by classifying all cases as negative, leading to overfitting. Therefore, accuracy should be used in conjunction with other metrics while checking the balance of the dataset. Precision measures the number of correct predictions made by the model of the target class, while recall assesses the efficiency of the ML model in detecting objects within the target class. These metrics are essential for evaluating the model's

performance. The F1 score is more appropriate, representing the harmonic mean of recall and precision. The F1 score is particularly important for assessing model performance when the dataset is imbalanced, as it considers the number of incorrect predictions and the types of errors (false negatives and false positives). The proposed model (soft voting classifier) was constructed by selecting the two best algorithms based on the results. We compared the performance results of the models used (Support Vector Machine, Decision Tree and Random Forest) with the proposed model (soft voting classifier) using the presented methodology, where the features obtained from the proposed feature selection method (CLR-ANCH) were utilised, and the dataset was split into testing and training sets, with data passed to the classifiers. Table 3 presents the comparison results of these models. The proposed model's soft voting classifier demonstrates the highest accuracy.

Table 4 indicates that the soft voting classifier achieves the highest accuracy (99%), F1 score (0.98), recall (0.98) and precision (0.99) because the voting classifier integrates the three models into one model that harnesses the strengths of these combined models, leading to optimal prediction accuracy. Figure 7 displays the ROC and DET curves for the Support Vector Machine, Decision Tree, Random Forest and the soft voting classifier, including the models it comprises (Decision Tree and Random Forest).

The proposed method contributed to selecting appropriate features that were less redundant and less noisy, which helped improve the results compared to conventional feature selection methods. Selecting the best features for model training is crucial for expediting the model's performance, minimising errors and achieving optimal results. Figure 8 illustrates the accuracy results of models before and after implementing the proposed method. Figure 9 presents the F-Score performance results for the classifiers based on the important performance scaling factors.

Table 4. Displays the results of models

Models	Confusion Matrix		Accuracy	F1_score	Recall	Precision
	TP	FP				
	FN	TN				
DT	63	3	0.97	0.97	0.98	0.95
	1	90				
SVM	55	5	0.90	0.88	0.85	0.92
	9	87				
RF	63	2	0.98	0.98	0.98	0.97
	1	91				
Voting model	63	1	0.99	0.98	0.98	0.98
	1	92				

Table 5 compares the methods applied in related works with our proposed method for earthquake prediction. The proposed method achieved superior results due to its reliance on the new approach for selecting the best features and the soft voting model that combines the most effective algorithms, ultimately leading to improved outcomes.

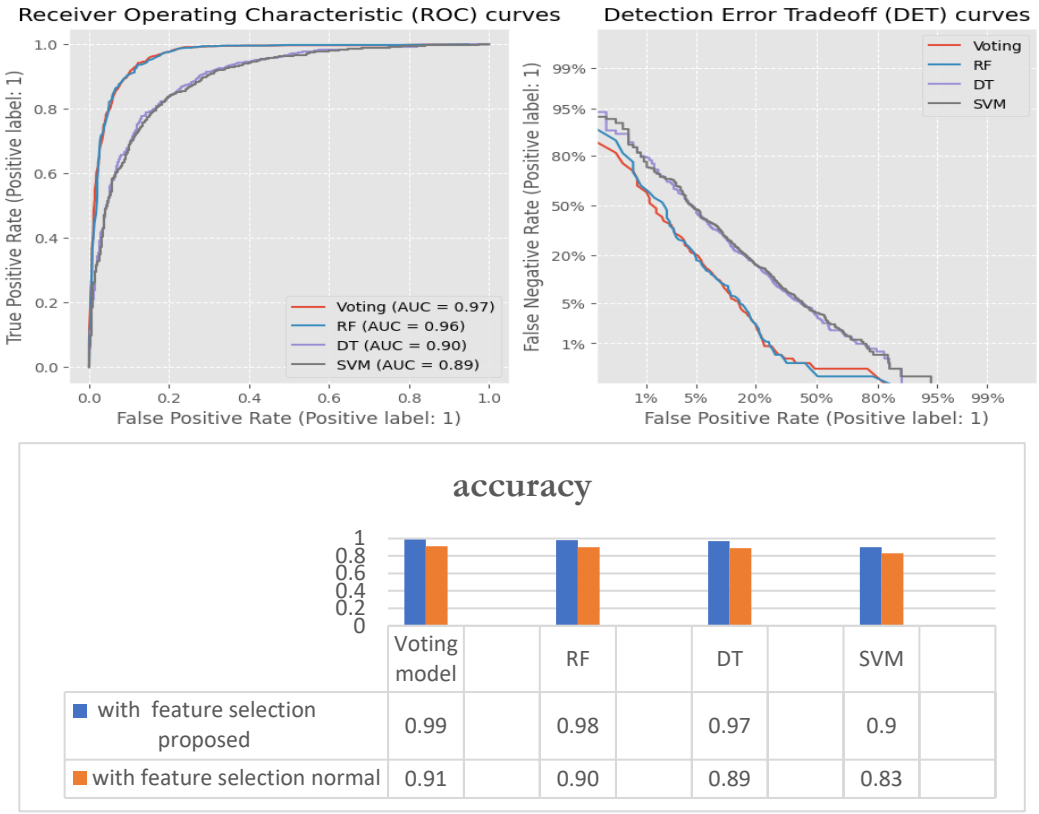


Figure 8. Displays the accuracy result of the models

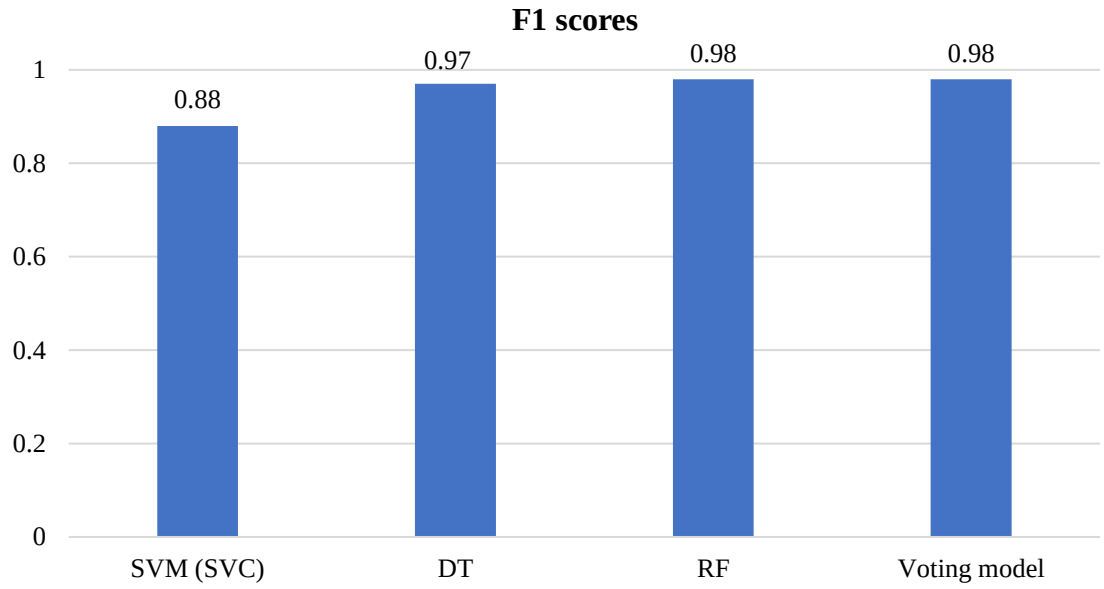


Figure 9. Displays the result F-Score

Table 5. Displays a comparison between the proposed method and related works

Research	Dataset	Methodology	Evaluation Metrics	Performance
Koehler, et al. [7]	Earthquakes	Utilizing a deep Learning network model to forecast earthquakes.	accuracy	72.3%
An, et al. [8]	Earthquakes	They used a model that blends the DIN with MLP to predict earthquakes.	AUC	0.69
Sajan, et al. [9]	Earthquakes	Using ML models such as decision trees, logistic regression, random forest, and XGBoost methods predicts earthquakes and building collapse.	accuracy	82%
Yang, et al. [10]	Earthquakes	The authors suggested an automated regression model that depends on ML, called AutoREP, to predict earthquake occurrence.	MSE, MAE	1.48, 1.51
The proposed model	Earthquakes	a Soft Voting Classifier	accuracy	0.99

5. Conclusion

This section presents the performance results in terms of accuracy, F1 score, recall, precision, AUC and ROC curves. The proposed methodology encompasses dataset preprocessing, data normalisation and feature selection using correlation based on Logistic Regression with normalisation, analysis of variance (ANOVA) and Chi-squared techniques (CL-ANCH) to identify the best features and improvements, balancing the dataset to align with algorithms, and splitting the dataset. The algorithms employed, including the voting classifier, Support Vector Machine, Decision Tree and Random Forest, yielded accuracy results of (0.99, 0.90, 0.97 and 0.98, respectively). The proposed model (soft voting classifier) enhances performance and incorporates two models (Decision Tree and Random Forest). A comparison was made between the performance of this model and the proposed model to demonstrate the efficiency and strength of our proposed model in the prediction process. The proposed methodology outperforms previous work, achieving an accuracy of 0.99, an F1 score of 0.98, a recall of 0.98 and a precision of 0.98. The proposed methodology successfully built a model to select the most influential features for optimal prediction accuracy. We observed that the models achieved better results than before. Finally, the study's future goal is to implement additional feature selection techniques while employing hybrid algorithms with soft voting classifiers to enhance earthquake diagnosis and prediction.

Declaration of Competing Interest: The authors declare they have no known competing interests.

References

[1] D. D. Acula, "Detection of Earthquake Damages from Satellite Images using Gradient Boosting Algorithm with Decision Trees as Base Estimator," *Acta Manilan,* vol. 70, no. 2022, pp. 13-28, 2022.

[2] P. Chittora *et al.*, "Experimental analysis of earthquake prediction using machine learning classifiers, curve fitting, and neural modeling," 2022.

[3] M. H. Al Banna *et al.*, "Application of artificial intelligence in predicting earthquakes: state-of-the-art and future challenges," *IEEE Access,* vol. 8, pp. 192880-192923, 2020.

[4] M. A. Salam, L. Ibrahim, and D. S. Abdelminaam, "Earthquake prediction using hybrid machine learning techniques," *International Journal of Advanced Computer Science and Applications,* vol. 12, no. 5, pp. 654-6652021, 2021.

[5] M. Maya and W. Yu, "Short-term prediction of the earthquake through neural networks and meta-learning," in *2019 16th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE)*, 2019, pp. 1-6: IEEE.

[6] J. Li, C. Zhang, X. Chen, Y. Cao, and R. Jia, "Improving abstractive summarization with iterative representation," in *2020 International Joint Conference on Neural Networks (IJCNN)*, 2020, pp. 1-8: IEEE.

[7] J. Koehler, W. Li, J. Faber, G. Ruempker, and N. Srivastava, "Testing the Potential of Deep Learning in Earthquake Forecasting," *arXiv preprint arXiv:2307.01812,* 2023.

[8] Z. An *et al.*, "Research on Earthquake Data Prediction Method Based on DIN–MLP Algorithm," *Electronics,* vol. 12, no. 16, p. 3519, 2023.

[9] K. Sajan, A. Bhusal, D. Gautam, and R. Rupakhety, "Earthquake damage and rehabilitation intervention prediction using machine learning," *Engineering Failure Analysis,* vol. 144, p. 106949, 2023.

[10] F. Yang, M. Kefalas, M. Koch, A. V. Kononova, Y. Qiao, and T. Bäck, "Auto-rep: an automated regression pipeline approach for high-efficiency earthquake prediction using lanl data," in *2022 14th International Conference on Computer and Automation Engineering (ICCAE)*, 2022, pp. 127-134: IEEE.

[11] A. Berhich, F.-Z. Belouadha, and M. I. Kabbaj, "A location-dependent earthquake prediction using recurrent neural network algorithms," *Soil Dynamics and Earthquake Engineering,* vol. 161, p. 107389, 2022.

[12] Y. Wang, Y. Zhang, Y. Lu, and X. Yu, "A Comparative Assessment of Credit Risk Model Based on Machine Learning——a case study of bank loan data," *Procedia Computer Science,* vol. 174, pp. 141-149, 2020.

[13] J. Yoon, "Forecasting of real GDP growth using machine learning models: Gradient boosting and random forest approach," *Computational Economics,* vol. 57, no. 1, pp. 247-265, 2021.

[14] A. Manoharan, K. Begam, V. R. Aparow, and D. Sooriamoorthy, "Artificial Neural Networks, Gradient Boosting and Support Vector Machines for electric vehicle battery state estimation: A review," *Journal of Energy Storage,* vol. 55, p. 105384, 2022.

[15] N. S. Abd, O. S. Atiyah, M. T. Ahmed, and A. Bakhit, "Digital Marketing Data Classification by Using Machine Learning Algorithms," *Iraqi Journal for Electrical And Electronic Engineering,* vol. 20, no. 1, 2024.

[16] L. Rokach and O. Maimon, "Top-down induction of decision trees classifiers-a survey," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews),* vol. 35, no. 4, pp. 476-487, 2005.

[17] O. S. Atiyah and S. H. Thalij, "A comparison of Coivd-19 cases classification based on machine learning approaches," *Iraqi Journal for Electrical and Electronic Engineering,* vol. 18, no. 1, pp. 139-143, 2022.

[18] R. F. Jader, S. Aminifar, and M. H. M. Abd, "Diabetes detection system by mixing supervised and unsupervised algorithms," *Journal of Studies in Science and Engineering,* vol. 2, no. 3, pp. 52-65, 2022.

[19] R. F. Jader, M. H. M. Abd, and I. H. Jumaa, "Signal Modulation Recognition System Based on Different Signal Noise Rate Using Artificial Intelligent Approach," *Journal of Studies in Science and Engineering,* vol. 2, no. 4, pp. 37-49, 2022.

[20] C. J. Burges, "A tutorial on support vector machines for pattern recognition," *Data mining and knowledge discovery,* vol. 2, no. 2, pp. 121-167, 1998.

[21] Warcoder. (2019, August, 01, 2023). *Earthquake dataset*. Available: https://www.kaggle.com/datasets/warcoder/earthquake-dataset

[22] E. Seeram, "An overview of correlational research," *Radiologic technology,* vol. 91, no. 2, pp. 176-179, 2019.

[23] I. Lopez-Arevalo, E. Aldana-Bobadilla, A. Molina-Villegas, H. Galeana-Zapién, V. Muñiz-Sanchez, and S. Gausin-Valle, "A memory-efficient encoding method for processing mixed-type data on machine learning," *Entropy,* vol. 22, no. 12, p. 1391, 2020.

[24] D. N. Sari, D. Kusnadi, R. H. Saputra, and M. U. Khan, "Digital Signal Processing for The Development of Deep Learning-Based Speech Recognition Technology," *International Journal of Electronics and Communications Systems,* vol. 4, no. 1, pp. 27-41, 2024.

[25] D. R. Morrison, E. C. Sewell, and S. H. Jacobson, "An application of the branch, bound, and remember algorithm to a new simple assembly line balancing dataset," *European Journal of Operational Research,* vol. 236, no. 2, pp. 403-409, 2014.

[26] P. J. M. Ali, R. H. Faraj, E. Koya, P. J. M. Ali, and R. H. Faraj, "Data normalization and standardization: a technical report," *Mach Learn Tech Rep,* vol. 1, no. 1, pp. 1-6, 2014.

[27] S. Visalakshi and V. Radha, "A literature review of feature selection techniques and applications: Review of feature selection in data mining," in *2014 IEEE international conference on computational intelligence and computing research*, 2014, pp. 1-6: IEEE.

[28] Scikit-learn. (2010, July, 20, 2024). *Supervised learning*. Available: https://scikit-learn.org/stable/supervised_learning.html

[29] H. Wang, H. Moayedi, and L. Kok Foong, "Genetic algorithm hybridized with multilayer perceptron to have an economical slope stability design," *Engineering with Computers,* vol. 37, pp. 3067-3078, 2021.