ISSN NO: 9726-001X
Volume 13 Issue 02 2025

PREDICTING EARTHQUAKES: A MULTI-CLASS CLASSIFICATION APPROACH WITH MACHINE LEARNING

M. Vignesh Goud,
UG Student,
Department of CSE,
St. Martin's Engineering College,
Secunderabad, Telangana, India
vigneshgoudm@gmail.com

P. Sudharsan,
Assistant Professor,
Department of CSE,
St. Martin's Engineering College,
Secunderabad, Telangana, India
jaisudha.sivaakshith1230@gmail.com

Abstract- The primary objective of this project is to develop a machine learning-based multi-class classification model for predicting the occurrence and impact of earthquakes, classifying them based on parameters such as magnitude, depth, and location for more accurate predictions and preparedness. "Predicting Earthquakes: A Multi-class Classification Approach with Machine Learning" refers to using machine learning algorithms to categorize earthquakes by their characteristics, aiming to predict not just the occurrence but also the severity and impact, thereby enhancing response strategies. Historically, earthquake prediction relied on traditional seismology methods like studying historical patterns, geological surveys, and seismic monitoring, which provided limited accuracy and minimal warning time. The traditional systems faced challenges in offering long-term predictions, highlighting the need for more sophisticated approaches. The motivation for this research is to improve prediction accuracy and minimize the catastrophic impacts on lives and infrastructure by utilizing machine learning to analyze complex seismic data. The proposed system utilizes machine learning algorithms to analyze seismic data, including parameters like magnitude, depth, location, and time of occurrence. The model will classify earthquakes into different categories based on these factors, enabling more accurate predictions. Machine learning, particularly deep learning techniques, can process vast amounts of data and identify patterns that traditional methods might overlook. This system aims to provide timely and accurate predictions, helping authorities and communities take proactive measures.

Keywords: Seismic Data Analysis, Magnitude, Depth, Location, Data Patterns, Catastrophic impacts, Proactive measures.

I. INTRODUCTION

Earthquakes are highly destructive and unpredictable natural disasters, posing significant risks to communities worldwide. According to the USGS, between 2000 and 2021, approximately 18,000 earthquakes of magnitude 5.0 or higher occurred annually. In 2020 alone, 69 major earthquakes were recorded, including the devastating Aegean Sea earthquake in Turkey, which registered a magnitude of 7.0, causing 114 deaths and over 1,000 injuries. The Global Earthquake Model (GEM) estimates that over 1.5 billion people reside in high seismic-risk zones, emphasizing the need for improved prediction and risk management strategies. Traditional earthquake prediction methods, relying on statistical analysis and geological assessments, often lack accuracy and fail to provide timely warnings. With increasing urbanization in seismically active regions, the potential for catastrophic consequences has grown. Between 1990 and 2020, earthquakes were responsible for over 1.2 million deaths and economic damages exceeding $1 trillion worldwide. Machine learning has emerged as a transformative tool in seismic prediction, offering enhanced forecasting capabilities by analyzing vast datasets and identifying complex patterns. Emergency response teams face significant challenges in coordinating rescue efforts, as outdated and fragmented data often delay critical decision-making. The 2010 Haiti earthquake highlighted how data inaccessibility prolonged response times, worsening the disaster's impact. Additionally, high-stress environments and operational inefficiencies hinder the effectiveness of rescue operations. A machine learning-based predictive system could provide real-time insights into earthquake magnitude, epicenter, and aftershocks, enabling emergency teams to allocate resources efficiently and prioritize affected areas. By integrating diverse datasets, such as geological, meteorological, and socio-economic factors, machine learning can improve prediction accuracy, reduce response time, and enhance disaster resilience.

ISSN NO: 9726-001X
Volume 13 Issue 02 2025

This technological advancement promises a more proactive approach to earthquake preparedness, ultimately saving lives and mitigating economic and social damages.

Traditional earthquake prediction and response methods have significant limitations, including slow data collection, delayed processing, and difficulty integrating diverse datasets. Traditional approaches rely on sensor networks that transmit seismic data to centralized systems, often causing delays, as seen in the 2011 Tōhoku earthquake. Additionally, these methods struggle to analyze geological, meteorological, and socio-economic data efficiently, leading to incomplete risk assessments. Machine learning offers a solution by rapidly processing vast amounts of data and identifying complex patterns that human analysts may miss. This automation enhances prediction accuracy, reduces response time, and strengthens disaster resilience, ultimately improving public safety.

II. RELATED WORK

Earthquake prediction remains a significant challenge due to the unpredictable nature of seismic events. While risk analysis can forecast some events, natural disasters like earthquakes are difficult to predict accurately. Precautionary measures and rapid response can mitigate human and economic losses, but earthquakes remain one of the most dangerous disasters due to their sudden occurrence and cascading effects like tsunamis, landslides, and industrial disasters, such as the Fukushima Daiichi nuclear disaster triggered by the 2011 Tōhoku earthquake [1].

Since the late 19th century, researchers have explored earthquake precursors, including foreshocks, electromagnetic anomalies, groundwater changes, and unusual animal behavior. Some successful predictions have been based on these precursors [2], but they are unreliable as they can be associated with non-seismic events. The optimism of the 1970s faded due to numerous false predictions [3][4] and the lack of statistically significant precursors [5]. As a result, no general methodology for earthquake prediction currently exists, and scientists remain divided on its feasibility.

Recent advances in machine learning have sparked interest in its application to earthquake science. Some studies focus on precursor analysis, such as using random forest algorithms on acoustic time series data to estimate the time until the next artificial earthquake [6]. Others analyze aftershock patterns; for example, a neural network trained on 130,000 mainshock-aftershock pairs outperformed traditional models in predicting aftershock distributions [7]. However, these studies address related but distinct problems rather than the core challenge of earthquake prediction.

Despite the relevance of earthquake prediction, few studies have systematized knowledge across different fields. A 2016 survey in CRORR Journal reviewed the use of artificial neural networks for short-term forecasting but mainly focused on neural network architectures, limiting its audience [8]. This paper aims to bridge the gap between seismology and computer science by covering all aspects of earthquake prediction, including data collection, feature extraction, and performance evaluation.

In seismology, "prediction" implies greater certainty than "forecasting" [9]. A prediction must specify location, time interval, and magnitude range in a way that allows objective validation [10]. Research in this field spans traditional seismology and modern AI-driven approaches. Allen [11] highlights the ethical and scientific responsibilities in earthquake prediction. The USGS [12] maps global earthquake distributions, emphasizing the need for region-specific models. Richter and Gutenberg [13] laid the foundation for magnitude-energy relationships, further expanded in Richter's *Elementary Seismology* [15]. Alves [14] pioneered the use of neural networks for earthquake forecasting, while Panakkat and Adeli [16][17] developed probabilistic neural networks using seismic indicators. Martínez-Álvarez et al. [18] applied neural networks to predict earthquakes in Chile, demonstrating AI's regional adaptability. Bath [19] and Utsu [20] contributed to understanding mantle inhomogeneities and aftershock statistics, respectively, which are crucial for refining predictive models.

By integrating traditional seismology with machine learning techniques, this research aims to improve earthquake prediction accuracy and reliability. This gap includes the following aspects:

- Limited Accuracy: Traditional methods relied on historical patterns and geological surveys, leading to broad risk assessments with significant uncertainties.
- Inability to Predict Timing: These methods failed to forecast the exact time of earthquakes, making timely warnings and preparedness difficult.
- Minimal Warning Time: Due to timing limitations, warnings were often too short for effective evacuation and emergency response.
- Over-reliance on Experts: Predictions depended heavily on seismologists' interpretations, leading to inconsistencies and reduced reliability.
- Inadequate for Long-term Forecasting: Traditional methods focused on short-term monitoring, limiting long-term disaster preparedness.
- Limited Data Integration: Seismic data, geological surveys, and historical records were analyzed separately, reducing prediction accuracy.

ISSN NO: 9726-001X
Volume 13 Issue 02 2025
JOURNAL OF CURRENT SCIENCE

- Difficulty Predicting Magnitude: While risk zones were identified, predicting the exact magnitude and impact remained challenging.
- Failure to Detect Emerging Threats: The reactive approach overlooked new seismic threats in previously inactive regions.

III. PROPOSED WORK

The proposed earthquake prediction system utilizes machine learning to assess earthquake severity based on parameters like magnitude, depth, location, and date-time. It transitions from a regression model, which predicts continuous magnitudes, to a classification model that categorizes earthquakes into severity levels such as 'Strong,' 'Major,' and 'Great.' The preprocessing phase involves handling missing data, encoding categorical variables, and converting magnitude values into severity categories to facilitate classification.

Extra Trees Classifier (ETC) is an ensemble learning method that constructs multiple decision trees using random splits. Unlike Random Forest, which optimizes splits based on metrics like information gain or Gini impurity, ETC selects splits randomly, making it more resilient to overfitting and capable of capturing complex feature interactions. This randomness enhances generalization and computational efficiency, allowing ETC to handle large datasets effectively. Additionally, ETC provides feature importance scores and is generally robust to noise, making it a promising choice for earthquake severity classification.

To evaluate model performance, accuracy, precision, recall, and F1 score are used, with ETC expected to outperform KNN due to its superior handling of large-scale and noisy data. Once trained, ETC is tested on unseen earthquake data to assess its generalization ability. This end-to-end approach—from data preparation to model evaluation—ensures that the most effective model is selected for real-world earthquake severity prediction, offering potential improvements in disaster preparedness and risk assessment.

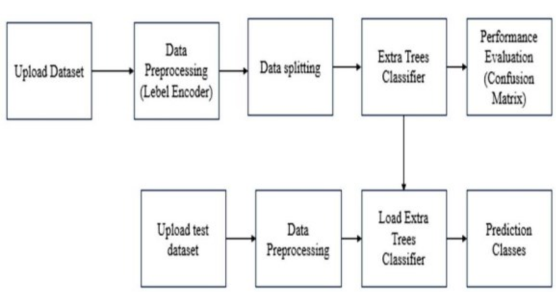

Fig 1: Overall design of proposed methodology.

3.1 Data Preprocessing

Data preprocessing is the process of preparing raw data and making it suitable for machine learning models. This is the first important step when creating a machine learning model. When creating a machine learning project, you can't always find clean, formatted data. Also, when working with data, it is essential to clean it and save it in a formatted format. To do this, use data preprocessing tasks. Real-world data typically contains noise, missing values, and may be in an unusable format that cannot be directly used in machine learning models. Data preprocessing is a necessary task to clean up data and make it suitable for machine learning models, which also improves the accuracy and efficiency of machine learning models.

- Feature Engineering: Raw data is transformed into features suitable for machine learning models. This involves creating new features from existing ones, such as extracting month, day, year, hour, and minute from the date-time information. This step enhances the model's ability to understand temporal patterns in the data.
- Handling Missing Values: Missing values in the dataset are addressed by filling them with appropriate statistics, such as the mode for categorical variables. This ensures that the dataset is complete and usable for model training.
- Removing Unnecessary Columns: Columns that do not contribute to the model's predictive power, such as depth, alert, continent, and country, are removed. This reduces dimensionality and simplifies the dataset.
- Label Encoding: Categorical variables are converted into numerical format using Label Encoding. This step is essential for converting non-numeric data into a format that machine learning algorithms can process.

Regression to Classification Conversion: The original problem involves predicting the magnitude of earthquakes, which is a continuous variable, making it a regression problem. To convert this into a classification problem, the continuous magnitude values are discretized into categorical classes.

Classes:

- Strong: Earthquakes with magnitudes in the range of 6.5 to 6.9.
- Major: Earthquakes with magnitudes in the range of 7.0 to 7.9.
- Great: Earthquakes with magnitudes in the range of 8.0 and above.

This conversion allows the model to classify earthquakes into distinct severity categories rather than predicting a precise magnitude value. The new classification problem involves predicting whether an earthquake falls into one of these categories based on various features like location, date-time, and magnitude.

Importing Libraries: To perform data preprocessing using Python, we need to import some predefined Python libraries. These libraries are used to perform some specific jobs. There are three specific libraries that we will use for data preprocessing, which are:

ISSN NO: 9726-001X
Volume 13 Issue 02 2025

JOURNAL OF CURRENT SCIENCE

NumPy: The NumPy Python library is used for including any type of mathematical operation in the code. It is the fundamental package for scientific calculation in Python. It also supports to addition of large, multidimensional arrays and matrices. So, in Python, we can import it as: import NumPy as nm. Here we have used nm, which is a short name for NumPy, and it will be used in the whole program.

Matplotlib: The second library is matplotlib, which is a Python 2D plotting library, and with this library, we need to import a sub-library pyplot. This library is used to plot any type of charts in Python for the code. we can import it as: import matplotlib.pyplot as mpt. Here we have used mpt as a short name for this library.

Pandas: The last library is the Pandas library, which is one of the most famous Python libraries and used for importing and managing the datasets. It is an open-source data manipulation and analysis library. Here, we have used pd as a short name for this library.

Scikit – learn: Scikit-learn provides a range of supervised and unsupervised learning algorithms via a consistent interface in Python.

3.2 Dataset Description:

Dataset contains detailed information on significant earthquake events, including geographical, seismic, and alert-related parameters. Each row represents an earthquake event with the following attributes:

- Title – Descriptive title of the earthquake event, including magnitude and nearest location (e.g., M 7.0 - 18 km SW of Malango, Solomon Islands).
- Magnitude – Earthquake strength on the Richter scale (e.g., 7.0).
- Date and Time – UTC timestamp of the event (e.g., 22-11-2022 02:03).
- Community Determined Intensity (CDI) – Intensity perceived by people, scaled from 1 (not felt) to 10 (extreme) (e.g., 8).
- Modified Mercalli Intensity (MMI) – Observed earthquake damage level (e.g., 7).
- Alert Level – Impact severity classification (green, yellow, orange, red) (e.g., green).
- Tsunami Indicator – Indicates whether the earthquake triggered a tsunami warning (0 – No, 1 – Yes) (e.g., 1).
- Significance Score – Impact-based significance measure (e.g., 768).
- Network Code – Organization recording the event (e.g., us).
- Number of Stations (NST) – Seismic stations that detected the event (e.g., 117).
- Minimum Distance (Dmin) – Closest station's distance from the epicenter (e.g., 0.509 degrees).
- Gap – Maximum azimuthal gap between recording stations (e.g., 17 degrees).
- Magnitude Type – The scale used to quantify magnitude (mww, mb, ml) (e.g., mww).
- Depth – Epicenter depth in kilometers (e.g., 14 km).
- Latitude & Longitude – Geographic coordinates of the epicenter (e.g., -9.7963, 159.596).
- Location – Nearest named place to the epicenter (e.g., Malango, Solomon Islands).
- Continent & Country – Geographic classification (e.g., Oceania, Solomon Islands).

title	magnitude	date_time	cdi	mmi	alert	tsunami	sig	net	nst	dmin	gap	magType	depth	latitude	longitude	location	continent	country
M 7.0 - 18 km SW of Malango, Solomon Islands	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	Malango, Solomon Islands	Oceania	Solomon Islands
M 6.9 - 204 km SW of Bengkulu, Indonesia	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	Bengkulu, Indonesia	NaN	NaN
M 7.0 -	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	NaN	Oceania	Fiji

Fig 2: Illustration of the sample dataset used for Predicting Earth Quakes: A Multi-Class classification Approach

3.3 Splitting the Dataset

In machine learning data preprocessing, we divide our dataset into a training set and test set. This is one of the crucial steps of data preprocessing as by doing this, we can enhance the performance of our machine learning model. Suppose if we have given training to our machine learning Suppose if we have given training to our machine learning model by a dataset and we test it by a completely different dataset. Then, it will create difficulties for our model to understand the correlations between the models. If we train our model very well and its training accuracy is also very high, but we provide a new dataset to it, then it will decrease the performance. So we always try to make a machine learning model which performs well with the training set and also with the test dataset. Here, we can define these datasets as:

Fig 3: Splitting the dataset.

Training Set: A subset of dataset to train the machine learning model, and we already know the output.

Test set: A subset of dataset to test the machine learning model, and by using the test set, model predicts the output.

For splitting the dataset, we will use the below lines of code:
from sklearn.model_selection import train_test_split

x_train, x_test, y_train, y_test= train_test_split(x, y, test_size= 0.2, random_state=0)

ISSN NO: 9726-001X
Volume 13 Issue 02 2025

• In the above code, the first line is used for splitting arrays of the dataset into random train and test subsets.

• In the second line, we have used four variables for our output that are

- x_train: features for the training data
- x_test: features for testing data

- y_train: Dependent variables for training data
- y_test: Independent variable for testing data

• In train_test_split() function, we have passed four parameters in which first two are for arrays of data, and test_size is for specifying the size of the test set. The test_size maybe .5, .3, or .2, which tells the dividing ratio of training and testing sets.

• The last parameter random_state is used to set a seed for a random generator so that you always get the same result, and the most used value for this is 42.

3.4 Extra Trees Classifier

The Extra Trees Classifier (ETC), also known as Extremely Randomized Trees, is an ensemble learning method designed to enhance classification performance by aggregating predictions from multiple decision trees. It extends the Random Forest algorithm but introduces a higher degree of randomness in data splitting and feature selection, leading to improved generalization and reduced overfitting. ETC is widely used in machine learning due to its computational efficiency, robustness to noisy data, and ability to capture complex feature interactions with minimal tuning.

Working of ETC: ETC operates by constructing a large ensemble of decision trees, each trained independently on the dataset. Unlike Random Forest, where tree splits are determined by optimizing criteria such as Gini impurity or information gain, ETC introduces randomness at two key levels:

- Feature Selection: Instead of selecting the best feature based on a statistical criterion, ETC randomly selects a subset of features for each tree.

- Split Point Selection: Within each selected feature, ETC chooses a split point at random rather than computing the optimal cut-off.

This high degree of randomness makes ETC less prone to overfitting and allows it to capture diverse patterns in the dataset. Additionally, in contrast to Random Forest, where trees are trained on bootstrap samples, ETC uses the entire dataset for training, further enhancing its ability to learn from all available data.

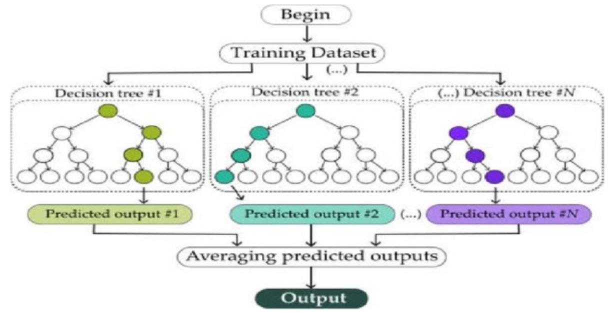

Fig 4: Extra Tree Classifier

Architecture of ETC

The architecture of ETC consists of a **forest of decision trees**, where each tree is trained independently and grown to its full depth without pruning. The key components of this architecture include:

- Randomized Feature Selection: Each tree selects a random subset of features at each split, ensuring high model diversity.

- Randomized Split Point Selection: Instead of choosing the optimal split based on entropy or Gini impurity, split points are determined randomly.

- Full Tree Growth: Each tree is expanded to its maximum depth, ensuring thorough learning of patterns without early stopping or pruning.

- Majority Voting for Prediction: Once all trees make individual predictions, the final output is determined by majority voting in classification tasks or averaging in regression problems.

This architecture leverages the power of bagging (bootstrap aggregating) while incorporating randomness, making ETC an efficient, parallelizable, and highly generalizable classifier.

Advantages of ETC

ETC offers several advantages over traditional decision tree-based classifiers like **Random Forest** and **Gradient Boosting**, making it an effective choice for a variety of machine learning applications.

- Reduced Overfitting: The high level of randomness in feature and split selection ensures that individual trees are highly uncorrelated, preventing the model from overfitting to training data.

- Computational Efficiency: ETC selects splits randomly, making it faster and scalable for large datasets.

ISSN NO: 9726-001X
Volume 13 Issue 02 2025

- Robustness to Noisy Data and Outliers:By averaging predictions across multiple trees, ETC reduces the influence of noise and outliers, making it suitable for real-world, messy datasets.

- Ability to Capture Complex Feature Interactions: The diverse trees in ETC learn different relationships between input features, allowing the model to recognize complex interactions without extensive parameter tuning.

- Feature Importance Analysis: ETC provides feature importance scores, helping analysts identify the most significant predictors and guiding feature selection for other models.

- Scalability and Parallelization: The model's ability to train multiple trees in parallel makes it well-suited for high-dimensional and large-scale datasets.

- High Generalization Performance: The combination of ensemble learning, randomness, and full tree growth ensures that ETC maintains strong predictive accuracy across various domains.

IV. RESULTS & DISCUSSION

The Extra Trees Classifier (ETC) achieved a remarkable performance with an accuracy of 95.54%, indicating that it correctly classified approximately 96 out of every 100 instances in the test set. It exhibited very high precision at 96.44%, meaning it made few false positive predictions, and a recall of 96.01%, reflecting its strong capability to identify most true positive instances. The F1-Score of 96.22% underscores the model's excellent balance between precision and recall. Overall, the ETC demonstrated superior performance compared to the KNN classifier, with higher precision, recall, and F1 Score, making it a highly reliable and effective model for the classification task.

```
Model loaded successfully.
ExtraTreesClassifier Accuracy    : 95.54140127388536
ExtraTreesClassifier Precision   : 96.43910661609777
ExtraTreesClassifier Recall      : 96.00636766334439
ExtraTreesClassifier FSCORE      : 96.21786492374727

 ExtraTreesClassifier classification report
              precision    recall  f1-score   support

      Strong       1.00      1.00      1.00         2
       Major       0.91      0.93      0.92        42
       Great       0.97      0.96      0.97       113

    accuracy                           0.96       157
   macro avg       0.96      0.96      0.96       157
weighted avg       0.96      0.96      0.96       157
```

Fig 5: Classification report of ETC

Figure 6 shows the confusion matrix that represents the model's performance in predicting three classes: Strong, Major, and Great. To evaluate this performance, we calculated metrics like accuracy, precision, recall, and F1-score. These metrics revealed that the model excels at predicting Strong and Great cases.

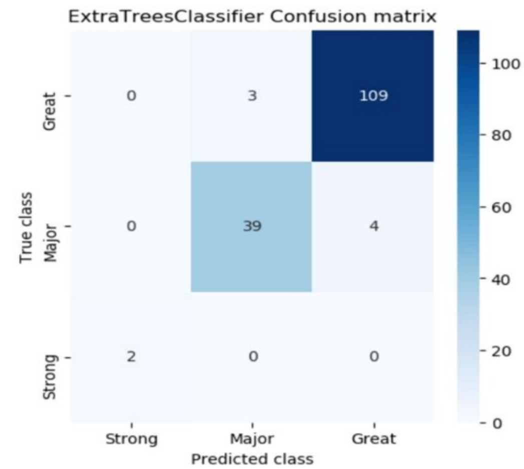

Fig 6: Confusion Matrix of ETC

The below NumPy array is predicted output i.e. Great means 0 and 1 means Major and 2 means Strong.

```
predict

array([2, 1, 2, 2, 2, 2, 1, 2, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 1, 2,
       1, 2, 2, 2, 2, 2, 1, 1, 1, 2, 1, 2, 2, 2, 0, 2, 2, 2, 2, 2, 2, 1,
       2, 2, 1, 2, 2, 2])

labels = ['Strong', 'Major', 'Great']    # Alphabetically
```

Fig 7: predicted output

V. CONCLUSION

The development of a machine learning-based multi-class classification model for earthquake prediction represents a significant advancement in disaster management and public safety. Traditional seismology methods, while foundational, have proven inadequate in providing the accuracy and timeliness required to mitigate the devastating impacts of earthquakes. By leveraging machine learning algorithms, this project aims to improve the accuracy of earthquake predictions, enabling authorities to take proactive measures in disaster preparedness and response. The model's ability to classify earthquakes based on parameters such as magnitude, depth, and location will provide a more nuanced understanding of seismic activity, allowing for more targeted and effective interventions. The potential benefits of this approach extend beyond immediate disaster response, offering valuable insights for urban planning, infrastructure development, and risk assessment. The successful implementation of this model could pave the way for more advanced predictive technologies in other areas of disaster management, contributing to a safer and more resilient society.

ISSN NO: 9726-001X
Volume 13 Issue 02 2025

V.REFERENCES

[1] K. Yasuhara, S. Kawagoe, H. Yokoki, and S. Kazama, "Damage from the Great East Japan Earthquake and Tsunami - A quick report," Mitigation and Adaptation Strategies for Global Change, vol. 16(7), pp. 803–818, 2011.

[2] K. Wang, Q. Chen, S. Sun, and A. Wang, "Predicting the 1975 Haicheng earthquake," Bulletin of the Seismological Society of America, vol. 96, pp. 757–795, 2006.

[3] R. Geller, "Earthquake prediction: a critical review," Geophysical Journal International, vol. 131, pp. 425–450, 2007.

[4] R. A. Kerr, "Seismology: Parkfield keeps secrets after a long-awaited quake," Science, vol. 306(5694), pp. 206–207, 2004.

[5] C. Lomnitz, Fundamentals of Earthquake Prediction. Wiley, New York, NY, 1994.

[6] B. Rouet-Leduc, C. L. Hulbert, N. Lubbers, K. M. Barros, C. Humphreys, and P. A. Johnson, "Machine learning predicts laboratory earthquakes," Geophysical Research Letters, vol. 44, pp. 9276–9282, 2017.

[7] P. M. R. DeVries, F. Viégas, M. Wattenberg, and B. Meade, "Deep learning of aftershock patterns following large earthquakes," Nature, vol. 560, pp. 632–634, 2018.

[8] E. Florido, J. L. Aznarte, A. Morales-Esteban, and F. Martínez Álvarez, "Earthquake magnitude prediction based on artificial neural networks: A survey," Croatian Operational Research Review, vol. 7(2), pp. 159- 169, 2016.

[9] W. Marzocchi, and J. D. Zechar, "Earthquake forecasting and earthquake prediction: different approaches for obtaining the best model," Seismological Research Letters, vol. 82(3), pp. 442–448, 2011.

[10] D. D. Jackson, "Hypothesis testing and earthquake prediction," Proc. Natl. Aca. Sci. USA, vol. 93, pp. 3772–3775, 1996.

[11] C. R. Allen, "Responsibilities in earthquake prediction: To the Seismological Society of America," Bulletin of the Seismological Society of America, vol. 66 (6), pp. 2069-2074, 1976.

[12] "Which country has the most earthquakes?" (Last accessed 24 January 2019) [Online] Available: https://www.usgs.gov/faqs/which-countryhas-most earthquakes 65

[13] F. C. Richter, and B. Gutenberg, "Magnitude and energy of earthquakes," Annals of Geophysics, vol. 9, 1956.

[14] E. I. Alves, "Earthquake forecasting using neural networks: results and future work," Nonlinear Dynamics, vol. 44(1-4), pp. 341-349, 2006.

[15] C. F. Richter, Elementary Seismology. W. H. Freeman, 1958.

[16] A. Panakkat, and H. Adeli, "Neural network models for earthquake magnitude prediction using multiple seismicity indicators," International Journal of Neural Systems, vol. 17(1), pp. 13–33, 2007.

[17] H. Adeli, and A. Panakkat, "A probabilistic neural network for earthquake magnitude prediction," Neural Networks, vol. 22(7), pp. 1018–1024, 2009.

[18] F. Martínez-Álvarez, A. Morales-Esteban, and J. Reyes, "Neural networks to predict earthquakes in Chile," Applied Soft Computing, vol. 13, pp. 1314-1328, 2013.

[19] M. Bath, "Lateral inhomogeneities in the upper mantle," Tectonophysics, vol. 2, pp. 483–514, 1965.

[20] T. Utsu, "A statistical study of the occurrence of aftershocks," Geophysical Magazine, vol. 30, pp. 521–605, 1961.

[21] KP Sanal Kumar, S Anu H Nair, Deepsubhra Guha Roy, B Rajalingam, R Santhosh Kumar, 'Security and privacy preservation using Edge intelligence in beyond 5G networks harnessing Federated Machine Learning and Artificial Immune Intrusion Detection System" Computers and Electrical Engineering, Pergamon, Elsevier.

[22] Santhoshkumar, M. Kalaiselvi Geetha, 'Deep Learning Approach for Emotion Recognition from Human Body Movements with Feed forward Deep Convolution Neural Networks', Procedia Computer Science, Vol.152, pp.158-165, May 2019, Elsevier, ISSN 1877-0509.

[23] Rajalingam, B., Al-Turjman, F., Santhoshkumar, R. et al. Intelligent multimodal medical image fusion with deep guided filtering. Multimedia Systems 28, 1449–1463 (2022)