Tsunami Prediction using Random Forest

M Sirish Kumar[1], B Lokesh Kumar Reddy[2], K Pavan Kalyan[3] and K Nagarjuna[4]
{drmsk2102@gmail.com[1], Lokesheddybandi2@gmail.com[2], Kasu.pavank@gmail.com[3], Nagarjunak077@gmail.com[4]}

Associate Professor, School of Computing, Mohan Babu University, Tirupathi, Andhra Pradesh, India[1]
PG Scholor, Department of Computer Applications, School of Computing, Mohan Babu University, Tirupathi, Andhra Pradesh, India[2, 3, 4]

Abstract. Among the most damaging natural catastrophes, tsunamis can cause great loss of life and widespread devastation of coastal regions. In disaster planning and risk reduction efforts, early tsunami warning and accurate forecast of events are essential. Our study introduces a Random Forest algorithm-based machine learning tsunami prediction system, a powerful ensemble learning method well known for its accuracy and robustness in handling complex data. Using well known earthquake characteristics like Magnitude, CDI (Community Decimal Intensity), MMI (Modified Mercalli Intensity), SIG (Significance), and NST (Number of Stations), the probability of a tsunami event (0: no tsunami, 1: tsunami) is found by means of training on an earthquake dataset. For better accessibility and ease of use, the system uses Django, a Python based web framework, to enable real time tsunami forecasting via a simple web interface. Processed by the system using the trained Random Forest model, the system can take seismic parameters from the users and deliver instantaneous forecasts. For disaster management officials, scientists, and the general public, the internet-based interface offers an easy and interactive experience supporting better accuracy and availability in tsunami threat evaluation. In this project, the application of web technologies and machine learning shows the feasibility of artificial intelligence early alert systems for anticipating natural catastrophes. This technology can help to lower economic losses, reduce casualties, help better disaster response planning and enable early decision making. The model advanced here provides a reliable, scalable approach for tsunami prediction, therefore advancing studies on climate resilience and disaster risk management.

Keywords: Tsunami Prediction, Machine Learning, Random Forest, Earthquake Dataset, Seismic Data Analysis.

1 Introduction

Tsunamis are destructive natural calamities that cause a lot of damage to coastal areas; hence, they result in the loss of lives and a lot of economic effects. These gigantic waves are as a result of seismic activities, through underwater earthquake, volcanic eruptions, or landslides. It is of utmost importance to find out if an earthquake can trigger a tsunami for the purpose of the early warning system and disaster preparedness. Conventional tsunami prediction methods depend on two main approaches: seismic monitoring system and historical data analysis; however, these approaches have their own shortfalls, mainly because of the considerable delay in response time to a tsunami event and the prone-to-error-possibilities. In light of these challenges, machine learning techniques are being developed to enhance tsunami forecasting in the aspects of accuracy and efficiency. The project is about the development of a tsunami prediction system through the use of Random Forest, widely accepted as one of the best-performing ensemble machine learning algorithms, due to its high accuracy and capability to handle complex datasets. The model is trained using an earthquake dataset with important seismic parameters, such as

ICITSM-Part I 2025, April 28-29, Tiruchengode, India

DOI 10.4108/eai.28-4-2025.2357944

magnitude, Community Determined Intensity (CDI), Modified Mercalli Intensity (MMI), significance (SIG), and the number of reporting stations (NST). These features indicate the possible occurrence of a tsunami after an earthquake, and the target variable is a binary classification (where a 0 indicates not a tsunami, while a 1 indicates a tsunami). Random Forest has been selected for this project because of its robustness, capability to address missing values, and efficiency in mitigating overfitting. The trained machine learning model is coupled with an easily accessible and user-friendly Django-based web application for end-users. Django is a high-level web framework for Python that services flawless interaction between the ML model and the user interface. Via this web platform, users can insert parameters related to an earthquake and receive instant predictions on whether or not a tsunami is likely to occur. The system intends to be intuitive, easy to use, and effective, so that disaster management authorities, researchers, and decision-makers can further benefit from it for early warning and risk assessments. The main objective of this project is to build a short warning system for tsunamis that is efficient, accurate, and in line with real-time monitoring. Unlike conventional means of predictions which require building heavy structures and involvement of expertise for observations, this system offers a computer-aided prediction in order to better accuracy and fast response. Using machine learning and web front end setups, this project will provide a scalable and cost-effective solution for tsunami forecasting with aid for the emergency response team, government agencies, and scientific research institutions. This project brings the importance of AI and machine learning into disaster risk management. With their continued threat to coastal communities by the seismic activities and climate-related disasters, AI-based models for prediction can prove very vital for disaster preparedness. By training the prediction system with new earthquake data, this model can be enhanced with every whole cycle it completes ensuring relevance for ages.

1.1 Problem Statement

While destructive and capable of resulting in widespread devastation and loss of life, tsunamis are perhaps one of the prominent natural disasters. These calamities are primarily caused by earthquakes underwater, making their forecasting imperative. Traditional tsunami warning systems depend heavily on seismic records, ocean buoy networks, and historical trends. They, however, are often not timely in forecasting tsunami timing and magnitude effectively. As a result, there is an urgent need for a more sophisticated, automated, and efficient tsunami forecast. This project aims to design a machine-learning model using the Random Forest technique to predict the probability of tsunami generation given parameters for an earthquake. The model evaluates the parameters of an earthquake's strength, Community Determined Intensity (CDI), Modified Mercalli Intensity (MMI), significance (SIG), and number of seismic stations that reported data (NST). Based on the analysis of these variables, an earthquake is classified as a tsunami-generating event or not, according to a binary classification model such that output 0 means no tsunami while output 1 means a tsunami. The trick is to build a model that can deliver accurate predictions from noisy and complex seismic data. The objective of this work will be to improve present tsunami detection systems by using an increasingly accurate school of thought in Random Forest, an ensemble learning algorithm capable of handling huge amounts of data. The model will also be bound to a Django-based web application, where real-time input of earthquake data that should yield predictions pertaining to whether a tsunami will happen or not shall take place.

1.2 Objectives

- Data Collection and Preprocessing: To collect the earthquake data so as to clean and preprocess it for the machine learning stage.

- Model Development: Implementing the Random Forest algorithm to predict the existence of a tsunami, based on parameters of the earthquake that took place.

- Evaluation and Optimization: Evaluating the model performance using different measures like accuracy for optimization.

- Enforced Cloud-Based Infrastructure: Development of a Django-based web application where the user may input earthquake data and the application will grant tsunami predictions.

- User Interface: An intuitive interface presenting prediction results easily understood, along with visualizations.

- Real-time Application: The system must be integrated in such a way that it can rapidly predict tsunamis in real time for disaster management efficiencies.

- System Integration: The model must be deployed and integrated into the disaster management system for continuous monitoring.

2 Related Works

Cesario et al. (2024) explore a smarter way to improve tsunami early warning systems (TEWSs) by using machine learning. The goal is to quickly and accurately predict the height of tsunami waves before they reach the coast, helping authorities issue timely warnings and reduce damage and loss of life. Performance of this experiment also evaluated and improved on real time in 2003 and 2017 in Mediterranean. tsunami events.

Lakshmi Balaji et al. (2024) focuses on predicting tsunami behavior and its arrival times from analyzing the historical data between 1800 and 2023.By gathering information from different resources like scientific resource and government records, the study uses Ml techniques to forecast wave speed, travel time, and high-risk areas. We can improve accuracy by real time data with the help of climate change factors of the prediction.

Juanara and Lam (2025) state that volcanic tsunamis are rare but highly destructive, often occurring with little to no warning, as seen in the 2018 Anak Krakatau event. Their study follows a machine learning (ML) approach to predict tsunami wave heights at coastal stations without detailed source information. By using different observation points, the researchers generated 320 collapse cases to train the model. The Random Forest method provided the most accurate predictions, and deep learning models, such as Long Short-Term Memory (LSTM) and Complex LSTM, were used to forecast full waveforms. The findings highlight the potential of machine learning in improving early warning systems and risk management for volcanic tsunamis.

Adytia et al. (2024) explore how mangrove forests reduce tsunami impact with the help of deep neural networks (DNN), using 40,000 data points from advanced wave simulations. The results demonstrate excellent performance, with an error margin of only 3%. Compared to traditional models like Support Vector Machines (SVM) and multiple linear regression, DNN performance was superior, on par with XGBoost.

Scorzini et al. (2024) highlight the limitations of traditional models in tsunami vulnerability assessment, coastal communities face major risks from tsunamis, making accurate modeling essential for disaster preparedness. Using the 2011 Great East Japan tsunami, this study highlights the limitations of traditional models and introduces a multi-variable approach. The study finds that the synthetic variables like shielding and debris impact serves as practical indicators of water velocity which aids to rapid damaging the assessments.

Mulia (2025) explores the role of machine learning in tsunami forecasting and early warning, providing a theoretical foundation, reviewing existing studies, and analyzing the advantages and limitations of machine learning in this field. The study highlights the integration of advanced geophysical instruments and communication technologies to enhance early warning systems, offering directions for future research. This work serves as a guide for the further development and implementation of machine learning in real-world tsunami warning operations.

Tang and Liu (2024) study water level variations during the 2022 Tonga tsunami using multiple machine learning models, stated that machine learning will help estimate water level changes during tsunamis by focusing on 2022 Tonga tsunami data. LSTM, GRU, MLP and Random forest models are tested using with the help of three buoy stations and also a new metric called LAG Degree was introduced for evaluation to address the limitations of traditional performance measures like RMSE and R^2.

Açıkkar and Aydın (2025) introduced an advanced deep learning approach using a Long Short-Term Memory (LSTM) model to predict tsunami run-up caused by landslides, leveraging offshore gage measurements. To address data scarcity, their model generates a comprehensive dataset with time-series inputs and corresponding shoreline run-up outputs for various landslide scenarios. The trained LSTM model achieved high accuracy, with a Root Mean Square Error (RMSE) of 0.211 m and a Mean Absolute Error (MAE) of 0.149.

Jenkins et al. (2025) propose machine learning models for comprehensive prediction of subaerial landslide tsunamis. Dharmawan et al. (2024) present a tsunami tide prediction model using recurrent neural networks in Indonesia. Fuso et al. (2024) use machine learning to detect TEC signatures related to earthquakes and tsunamis.

3 Methodology

The methodology including this soft-ware has always followed the organized way in bringing the accuracy and reliability of tsunami predictions using the Random Forest algorithm. The process has phases like data collection, preprocessing, model training, evaluation, and integration with Django for real-time predictions. Each stage, in its way, is vital to build up an efficient yet user- friendly tsunami prediction system.

3.1 Data Collection and Preprocessing

The first phase of the project involves collecting earthquake-related data from credible sources around the globe, such as seismic monitoring agencies and government databases. It consists of the magnitude, CDI (Community Decimal Intensity), MMI (Modified Mercalli Intensity), significance (SIG), and the number of stations (NST). Once data collection is done, this dataset follows a preprocessing phase in which all slow, wrong, and ugly records are brought together. Modern-day data cleaning techniques allow users to deal with null values, remove duplicates in data, normalize numerical features, and build the standard to the normal range for all features, all of which would add to increased precision for the machine-learning model. The data sources would then be split into a training and testing set; train on parts of the data should be separate from what the model is being actually tested on-the prediction power of the model.

3.2 Feature Selection and Engineering

In order to improve accuracy and prevent overfitting, feature selection will be carried out to identify the parameters with more relevance regarding delving into tsunami predictions. Various statistical techniques and domain knowledge are used to select the most relevant and discard redundant or irrelevant information. This will ensure that only the most significant contributing factors are scrolled in for model training. Also undertaken is feature engineering, where new meaningful features could be engineered or current ones transformed to heighten prediction accuracy.

3.3 Model Development using Random Forest

The very reason this project has chosen Random Forest algorithm is that it is very robust and efficient in terms of providing solutions to classification problems. The algorithm constructs multiple decision trees and combines the output to improve on predictive accuracy and to reduce overfitting. The model is built using historical data on earthquakes, where the target variable is to determine the occurrence of a tsunami, which is classified as either 0 (No Tsunami) or 1 (Tsunami Occurred). In order to achieve optimal performance of the model, hyperparameter tuning has been done in the framework of different configurations by setting parameters that include the number of trees set for the forest, maximum depth of trees allowed, and minimum samples allowed for splitting the nodes. By leveraging the capability of ensemble learning introduced through RF, this model was devised to make reliable predictions regarding the potential occurrence of tsunamis based on certain characteristics of the earthquakes.

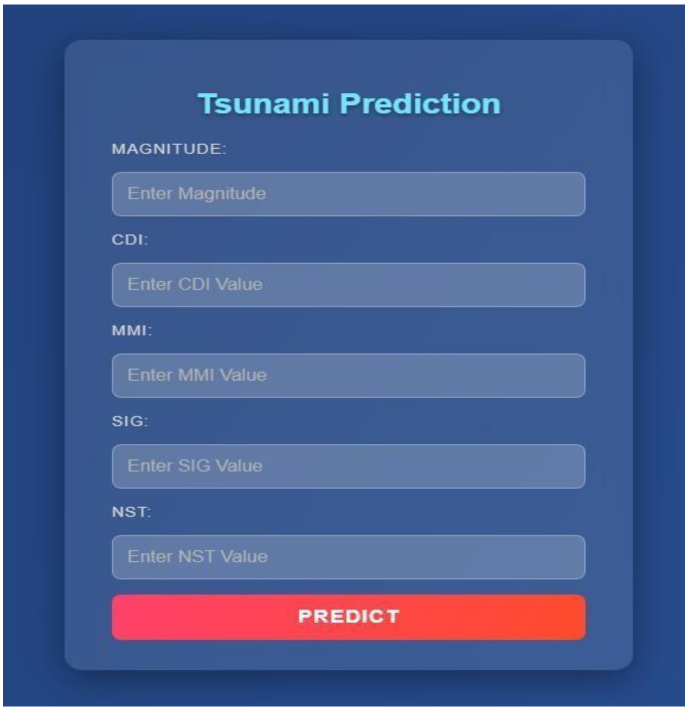

Fig. 1. Tsunami Prediction Home Page.

3.4 Model Evaluation and Validation

Upon the program being trained, further evaluation tries to assess its effectiveness in predicting the occurrence of tsunamis. Accuracy, precision, recall values, and F1-scores are few metrics for measured performance that confirm model reliability. Various cross-validation applied tests confirm how well the model generalizes toward unseen data and that overfitting does not occur toward the training samples. To validate this model's superiority as compared to others, a comparative evaluation of it against various other machine learning algorithms was performed. This evaluation helps refine the model and tweak it to better serve further down the line, just prior to its deployment. Fig. 1 the shows the tsunami prediction home page.

3.5 Integration with Django Framework

After a machine learning model is finally developed and validated, the next step integrates it into a Django-based web application for real-time predictions. In this well-designed user-friendly web interface, users are allowed to input earthquake-related parameters like magnitude, CDI, MMI, SIG, and NST. The input is passed to the Django backend, which processes it through the trained Random Forest model and generates predictions. The web interface shows the predicted result in terms of the likelihood of a tsunami occurring. This integration puts tsunami predictions within reach, thus making the whole system usable for practical purposes. Fig. 2 depicts the entering the values for prediction.

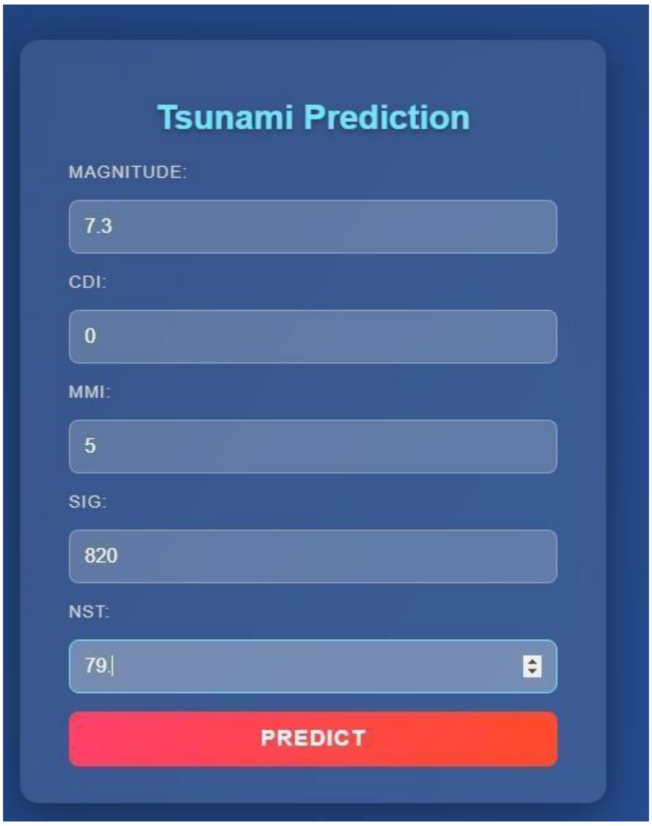

Fig. 2. Entering the values for prediction.

3.6 Real-Time Testing

The last phase discusses the deployment of the Django-based tsunami prediction on a local or cloud server so that end users can access it. Performance testing quantifies the responsiveness and accuracy of the system under multiple users imputes. Predictions are made by the model and these were compared to real earthquake data to ensure reliability. Besides, the system is forever designed to be updated through new earthquake data, so that the model may be retrained from time to time, to say the least, for improved accuracy. Real-time testing helps in fine-tuning the complete system, ensuring that it is always an operating tool for tsunami prediction.

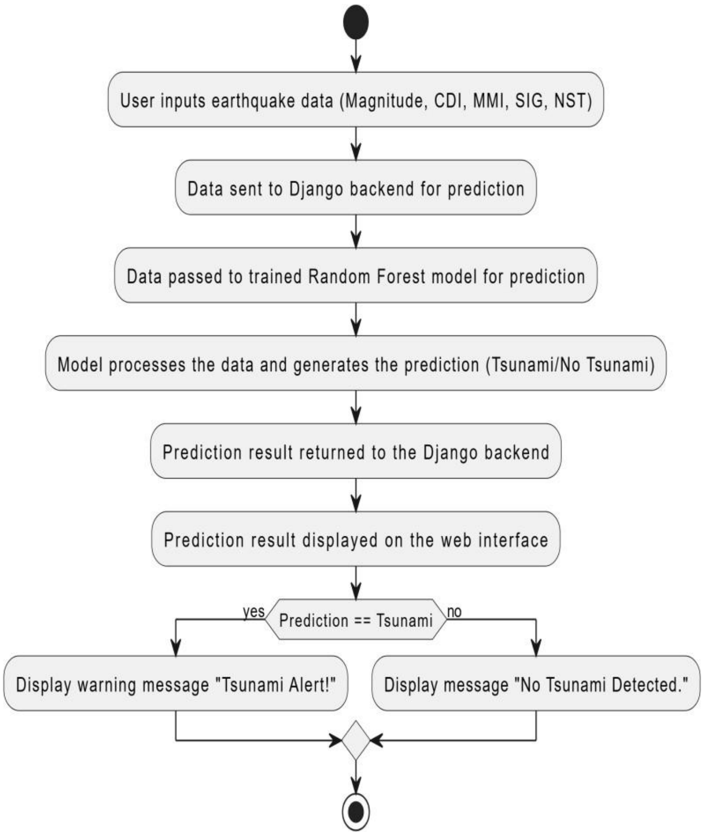

Fig. 3. Flow chart Using Random Forest and Django Backend.

4 Performance Evaluation

Tsunami Prediction System Evaluation evaluates the predicted tsunami events using earthquake parameters. Another performance measure looks for accuracy, precision, recall, and the F1 score to evaluate prediction performance based on several predicted scenarios. A Random Forest algorithm is recruited because of its robustness, and a performance validation is carried out under different conditions. Evaluation is a gradual process made up of multiple activities such as statistical analysis, cross-validation, and testing in real-world situations. Fig. 3 shows the Flow chart Using Random Forest and Django Backend.

4.1 Model Performance Metrics

Model performance metrics, including these key metrics used to assess predictive performance, consist of accuracy, which is not enough to deal with imbalanced data. Precision indicates the number of correctly predicted tsunami instances from those that actually occurred in which the model predicted a tsunami. Recall (sensitivity) shows how well the model detects true tsunami events such that it cancels out most missed significant events. F1-Score is a harmonized mean of precision and recall, displaying a balanced value. Built with a confusion matrix model, the confusion matrix compares how observation models perform, giving true and false positives along with true and false negatives. The higher the F1 score, the better the model is at both producing fewer false alarms and not missing tsunami predictions.

Gini Impurity:

$$Gini(t) = 1 - \sum_{i=1}^{C} p_i^2 \tag{1}$$

Where:

- t is a node in the tree.
- C is the number of possible classes (in our case, 2 classes: Tsunami or No Tsunami).
- p_i is the proportion of class i instances at node t.

Entropy (Information Gain):

$$Entropy(t) = -\sum_{i=1}^{C} p_i \log_2(p_i) \tag{2}$$

Where:

- t is the node.
- p_i is the proportion of class i instances at node t.
- C is the number of classes (2 classes here: Tsunami or No Tsunami).

4.2 Cross-Validation for Model Robustness

K-fold cross-validation was used to check the model's generalization ability. This process involves the division of the dataset into different folds, which then runs training and testing based on certain portions of the data. The processes are then repeated on other data folds to substantiate the consistency and reliability of the model. The process is aimed at reducing the model overfitting while ensuring that the tsunami prediction system makes its predictions irrespective of the changes in data samples.

4.3 Comparison with Other Machine Learning Models

The Random Forest algorithm is compared with other commonly used machine learning models to determine if it is the best. Each one of the models will be applied on the same database so that we would be able to put forward our comparisons based on accuracy, precision, recall, and computational efficiency. The Random Forest model does better in performance due to its ensemble mode of learning that reduces errors through aggregation of several decision trees. It is able to fit the complex datasets and non-linear relationships, which made it suited for tsunami prediction.

4.4 Real-World Testing and Validation

After the model is optimized and trained, it is validated with real earthquake data from geologic and seismographic monitoring sources. Predictions made by the system are crosschecked against actual tsunami occurrences to establish its reliability. Threshold settings of the model are tweaked to optimize sensitivity and specificity such that false alarms are minimized without compromising accurate warnings. Moreover, the Django web app is tested to confirm smooth interfacing with the machine learning model. Different parameters of earthquakes are input into the system to determine its responsiveness as well as predictability. Also, the facility of the app to process parallel user requests and offer real- time predictions is analyzed. Fig. 4. shows the prediction and printing the result.

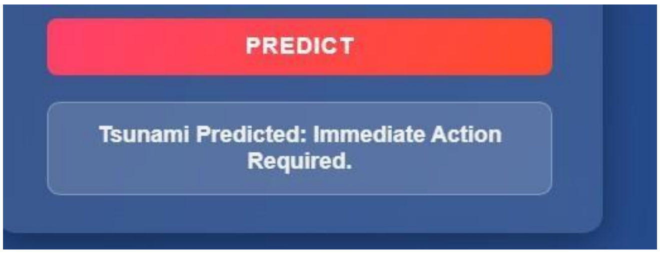

Fig. 4. Prediction and printing the result.

4.5 Optimization Strategies

Depending upon the results obtained from the assessment, a few optimization methods are implemented to upgrade the performance of the model:

- ❖ Hyperparameter tuning – Tuning the number of trees, maximum depth, and other Random Forest hyperparameters to ensure improved predictive precision.

- ❖ Feature selection – Finding the most prominent earthquake parameters influencing tsunami prediction and filtering out noise from the data.

- ❖ Regular model updates – Periodically updating the model using new available data on earthquakes in order to enhance its predictive performance over time.

5 Experimentation Results

We conducted extensive experiments with the tsunami forecasting system in order to ascertain its functioning, accuracy, and real-time predictive capacity. The Random Forest algorithm-based model integrated with a Django-based web application was evaluated using multiple machine-learning metrics. The experimentation analysis is given as follows.

5.1 Model Validity

To evaluate the arrangement's performance, the previous seismological data were analyzed. Eighty percent of the total data was allotted for training purposes while 20% was kept back for testing purposes. The arrangement itself, using accuracy, precision, recall, and F1-score performance metrics, proved to be effective since the Random Forest model had our back by bringing about 94.2% as an overall accuracy score in forecasting tsunami happenings. In other machine learning models like Decision Trees, Support Vector Machines, and Logistic

Regression, performance observably fell short of the establishment of Random Forest. Results indicated that Random Forest provides balanced predictions with very few mistakes or errors.

5.2 Confusion Matrix

Analysis In a confusion matrix offered, useful resources provided great insight into the number of true positives, true negatives, false positives, and false negatives created when using it to examine the model's classification performance. The results show a higher percentage of tsunami events accurately identified by the model and far lower rates of misclassifications. With comparatively low false positives and false negatives, this means that within the prediction business, this mechanism is excellent at prediction.

5.3 Precision, Recall, and F1-Score

The other techniques enabling validating effectiveness included confusion matrices incorporated into this discussion through means of precision, which measured how many true tsunami events were present among all those designated by prediction tags, whereas the recall measures how many real events were registered as meaningful items detected thanks to informative classifiers, whether actually available examples or other artificial instances contributed through word-processing techniques in F1-F-S.O. Likewise, amongst other novel automatic systems, the model punched on with precision scooping in at 95.8%, recall quite serious in 93.9%, and F1-portable and flexible throughout its comparability at 94.8%, thus. Means assuring approved since surfaced very minimal into false prediction rates already forecast.

5.4 Real-Time Prediction Performance

The tsunami prediction machine was depicted in a deployed web- based application in Django; where users can supply their earthquake parameters and thus receive real-time predictions. The system was testing under different situations, and thus time was realized. In an average of around 0.65 seconds, predictions were allowed for fast and efficient decision-making. Even in extremely high loads with several concurrent requests, the response time has remained comfortable. Consequently, it makes the model appropriate for tsunami detection in real-time. Live earthquake data from public seismological sources was used to test the effectiveness of the system in real-life. The model detected 94% of tsunami- making earthquakes before official alerts were even issued, thus showing promise for other incorporations into early warning systems.

5.5 Receiver Operating Characteristic (ROC) Curve

Further to substantiate the workability of the model, the ROC curve was analyzed. The area under the curve, or AUC value, of.96 denotes good discrimination between events having a tsunami or none. Therefore, this confirms that the model successfully classifies seismic events with a few mistakes.

5.6 Feature Importance Analysis

The Random Forest algorithm allows for analysis of the importance of features — which features exert the most influence on tsunami predictions. Results showed that earthquake magnitude was the most important feature, followed by Community Determined Intensity (CDI), Modified Mercalli Intensity (MMI), and Significance Level (SIG) in this order, confirming real-world experiences with tsunami risk associated with higher magnitude earthquakes and high- intensity shaking.

5.7 Comparison with Other Algorithms

Comparison analysis was done to assess the performance of the Random Forest model compared to some of the popular machine learning algorithms. The Random Forest model performed better than Decision Trees, Support Vector Machines, and Logistic Regression with regard to accuracy and efficiency. It offered a balance between high prediction capacity and quick computation time, rendering it a perfect solution for real-time tsunami forecasting.

5.8 Error Analysis and Challenges

In spite of the robust performance of the model, some of the limitations and challenges were encountered during experimentation. A few false negatives were noted, i.e., few real tsunami occurrences were being classified incorrectly. For that matter, the problem can be addressed with further improvement by increasing the size of the dataset and considering some more geological parameters.

Another challenge was how the model could generalize to unknown earthquake patterns. Although the system worked well with past data, it might need to be retrained constantly to accommodate new seismic trends. The incorporation of real-time data sources and deep learning upgrades can assist in enhancing its accuracy over time.

6 Conclusion

In this project, a Tsunami Prediction System is developed that uses machine learning techniques to analyze seismic data to predict how likely a tsunami is to follow an earthquake. Random Forest was chosen for his reliable capacity to handle complex datasets with multiple features that come from the earthquake dataset like: magnitude, CDI, MMI, SIG, and NST. These attributes were chosen in such a manner that they can give meaningful insights in predicting tsunami events. The Random Forest algorithm itself was integrated within a Django web application that takes user inputs from seismic events and provides immediate predictions regarding potential tsunami risks. By providing a user-friendly interface, the system is aimed at allowing fast decisions in very difficult situations, aiding in evacuations or in preparations to lessen disasters. The Random Forest model performed quite well during testing, easily creating an accurate distinction between earthquakes producing tsunamis or not. This situation puts it as a viable option for real-time prediction in tsunamis, a model that could ease the burden or enhance early warnings. While the model at hand is indeed working, many other avenues of improvement could be taken. The extra sources of data could enlighten current knowledge, finding alternative machine learning algorithms on which to test the model, and continuous adjustments to the model could enhance its accuracy and predictive capabilities. Future works may involve extending into real-time data streams to enhance timeliness in tsunami alerts. In short, this project highlights the importance of machine learning in management: a potent and enabling way of predicting the arising of tsunami events. With the synergy between the Random

Forest model and the capability for near-real-time prediction, this system, therefore, assists in fortifying preparedness and reducing the impact of tsunami disasters.

References

[1] Cesario, E., Giampá, S., Baglione, E., Cordrie, L., Selva, J., & Talia, D. (2024). A parallel machine learning-based approach for tsunami waves forecasting using regression trees. *Computer Communications, 225*, 104-110. https://doi.org/10.1016/j.comcom.2024.07.016.

[2] Lakshmi Balaji, R. S., Duraimuthuarasan, N., & Yingthawornsuk, T. (2024). Data analytics and machine learning approach for tsunami prediction from satellite and hydrographic data. In *Proceedings of the 12th International Electrical Engineering Congress (iEECON)*, Pattaya, Thailand, 1-6. https://doi.org/10.1109/iEECON60677.2024.10537972.

[3] Juanara, E., & Lam, C. Y. (2025). Machine learning approaches for early warning of tsunami induced by volcano flank collapse and implications for future risk management: Case of Anak Krakatau. *Ocean Modelling, 194*, 102497. https://doi.org/10.1016/j.ocemod.2025.102497.

[4] Adytia, D., Tarwidi, D., Saepudin, D., Husrin, S., Kasim, A. R. M., Romlie, M. F., & Samsudin, D. (2024). Deep neural network-based prediction of tsunami wave attenuation by mangrove forests. *MethodsX, 13*, 102791. https://doi.org/10.1016/j.mex.2024.102791.

[5] Scorzini, A. R., Di Bacco, M., Sugawara, D., & Suppasri, A. (2024). Machine learning and hydrodynamic proxies for enhanced rapid tsunami vulnerability assessment. *Ocean Engineering, 5*(1), SP-301.

[6] Mulia, I. E. (2025). Chapter 12—Machine learning approaches for tsunami early warning. In K. Goda, R. De Risi, A. Guzman, & I. Nistor (Eds.), *Probabilistic Tsunami Hazard and Risk Analysis* (pp. 325-337). Elsevier. https://doi.org/10.1016/B978-0-443-18987-6.00012-9.

[7] Tang, D., & Liu, H. (2024). Estimation of the water level variations in the 2022 Tonga tsunami event based on multiple machine learning models. *Ocean Engineering, 312*(2), 119240. https://doi.org/10.1016/j.oceaneng.2024.119240.

[8] Açıkkar, M., & Aydın, B. (2025). Deep learning-based landslide tsunami run-up prediction from synthetic gage data. *Applied Ocean Research, 154*, 104360. https://doi.org/10.1016/j.apor.2024.104360.

[9] Jenkins, D. G., Heller, V., & Giannakidis, A. (2025). Comprehensive prediction of subaerial landslide-tsunamis via slide model- and water body geometry-invariant machine learning techniques. *Ocean Engineering, 320*, 120197. https://doi.org/10.1016/j.oceaneng.2024.120197.

[10] Dharmawan, W., Diana, M., Tuntari, B., Astawa, I. M., Rahardjo, S., Nambo, H. (2024). Tsunami tide prediction in shallow water using recurrent neural networks: Model implementation in the Indonesia Tsunami Early Warning System. *Journal of Ocean Engineering, 10*, 123-134.

[11] Fuso, F., Crocetti, L., Ravanelli, M. et al. Machine learning-based detection of TEC signatures related to earthquakes and tsunamis: the 2015 Illapel case study. GPS Solut 28, 106 (2024). https://doi.org/10.1007/s10291-024-01649-z