IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 15, No. 1, February 2026, pp. 257∼268
ISSN: 2252-8938, DOI: 10.11591/ijai.v15.i1.pp257-268

Benchmarking machine learning models for natural disaster prediction with synthetic IoT data

Moath Alsafasfeh[1,2], Abdullah Alhasanat[1], Atheer Bassel[3], Mohanad Alhasanat[1]
[1]Department of Computer Engineering, College of Engineering, Al-Hussein Bin Talal University, Ma'an, Jordan
[2]Department of Electrical and Computer Engineering, College of Engineering, Tuskegee University, Tuskegee, United States
[3]Department of Artificial Intelligence, College of Computer Science and Technology, University of Anbar, Anbar, Iraq

Article Info

Article history:

Received Oct 2, 2025
Revised Dec 29, 2025
Accepted Jan 22, 2026

Keywords:

Disaster resilience
Early warning systems
Ensemble learning
Extreme weather events
Internet of things
Natural disaster prediction
Synthetic data

ABSTRACT

Natural disasters pose severe threats to human life and infrastructure, demanding robust early warning systems (EWS) supported by machine learning (ML) and internet of things (IoT)-based sensing. This study benchmarks ML models for predicting floods and earthquakes using synthetic IoT sensor data. A dataset comprising nine environmental and seismic parameters was generated and labeled into three classes: no disaster, flood, and earthquake, where the feature preprocessing was applied during model training. Logistic regression (LR), random forest (RF), and extreme gradient boosting (XGBoost) models were trained and evaluated using accuracy, precision, recall, and F1-score. Experimental results on the World-A test set show that ensemble models consistently outperform LR, with XGBoost and RF achieving F1-scores of up to 97% and 99%, respectively, compared to 79% for LR. An independent test on the separately generated World-B dataset revealed that ensemble models maintained higher generalization capability with F1-scores of 80% for XGBoost and 78% for RF. In contrast, LR showed substantial degradation with an F1-score of 54%. Stress testing on the World-B dataset under simulated situations, such as sensor failures, noise injection, and extreme weather events, confirms the resilience performance of ensemble models in comparison to LR. These results demonstrate the usefulness of ensemble learning in handling unpredictable IoT data for disaster prediction and highlight their potential integration into intelligent EWS. Future work will focus on expanding the framework to include cross-time prediction, incorporating additional environmental features, and deploying the models in real-time IoT systems for field validation.

Corresponding Author:

Moath Alsafasfeh
Department of Computer Engineering, College of Engineering, Al-Hussein Bin Talal University
Ma'an, Jordan
Email: malsafasfeh@tsukegee.edu

1. INTRODUCTION

Natural disasters are serious adverse events of geophysical, hydrological, climatological, or meteorological origin that threaten human life, infrastructure, and the environment [1], [2]. Floods and earthquakes are two of the most damaging hazards, accounting for a large proportion of disaster-related deaths and economic losses worldwide [1]. The frequency of such incidents has increased dramatically in recent decades, emphasizing the need for reliable prediction and response systems. Early warning systems (EWS) are designed to deliver timely alerts through hazard monitoring, forecasting, and risk assessment [3]. However, by 2020, only a small minority of countries had operable multi-hazard EWS [4]. Improving these systems remains a global goal. Artificial intelligence (AI) provides promising solutions for disaster prediction by analyzing large, complex datasets and detecting patterns that precede extreme events [5]. Machine learning (ML) models, in particular, have the potential to increase disaster prediction accuracy and facilitate proactive decision-making. Despite these advances, the dependability of ML and deep learning (DL) algorithms must be extensively tested, especially under conditions of data scarcity, noise, or sensor failure. Furthermore, evaluation of both the datasets and the prediction models is critical to ensuring reliable outcomes. This study addresses these gaps by benchmarking ML models for flood and earthquake prediction using synthetic internet of things (IoT) sensor data.

A wide range of IoT-based systems have been developed for natural disaster prediction and management, typically combining sensor networks with ML models for early warning. An IoT-based framework integrating multi-sensor data with neural networks, decision trees (DTs), and random forest (RF) demonstrated strong decision-making capabilities but faced scalability and interoperability challenges [6]. Several papers highlight the importance of applying ML and DL for disaster detection and outline key research directions [7]–[9]. As such, a comparative study in [10] benchmarked ML and DL models, showing convolutional neural networks (CNNs) and hybrid deep networks outperform others, while RF and extreme gradient boosting (XGBoost) remained competitive for smaller datasets; no single algorithm proved universally optimal. For flood prediction, IoT and ML approaches have used water-level, rainfall, and humidity sensors with models such as RF, long short-term memory (LSTM), and CNN, achieving accuracies between 80-95% [11]–[13]. Ensemble RF-LSTM hybrids reached 81% accuracy on the tested data [14]. Rezvani *et al.* [15] applies geospatial AI to flood hotspot detection in Portugal, producing susceptibility maps with 96% accuracy, though limited by reliance on historical datasets. According to Anbarasan *et al.* [16], a convolutional deep neural network (CDNN)-based system combining IoT sensors and big data achieved 93.23% accuracy, outperforming artificial neural network (ANN) and deep neural network (DNN) baselines, but was validated mainly on simulated data. For earthquake prediction, accelerometer-based IoT frameworks analyzed seismic vibrations using support vector machine (SVM) and DTs, with SVM reaching 95% accuracy [17]. Mukherjee *et al.* [18] extracted 61 seismic features from the Himalayan Belt, finding ANN and XGBoost most accurate across longer prediction windows.

According to Rosca and Stancu [19], an IoT–cloud system using 8,766 seismic and meteorological records achieved 99.84% accuracy, though limited to the Vrancea region. Kubo *et al.* [20] review ML applications in seismology, noting strong advances in detection and catalog completion but persistent issues with generalizability and interpretability. For landslides, IoT sensors monitoring soil moisture, slope, and rainfall combined with ML models such as XGBoost and LSTM achieved over 95% accuracy [21]–[23], while U-Net improved susceptibility mapping with satellite imagery [12]. The study in [24] proposed a low-cost IoT-ML framework in Vietnam, where RF achieved strong predictive performance and enabled real-time alerts, though generalizability and device maintenance remain concerns. On the other hand, smart city-oriented systems have integrated IoT, ML, and cloud computing for multi-hazard monitoring, though challenges persist in scalability, data transmission, and privacy [25]. Overall, prior work demonstrates the potential of IoT-ML integration for disaster prediction, but most studies rely on limited real-world datasets and focus on single hazards. Few have systematically benchmarked multiple models under controlled conditions. This paper addresses that gap by generating a synthetic IoT dataset for flood and earthquake scenarios and conducting a comparative robustness evaluation.

2. METHOD

The paper introduces the development and evaluation of a ML-based system for natural disaster prediction. The approach involves generating two synthetic datasets, World-A and World-B, labeling disaster events, and training multiple classification models to detect floods and earthquakes based on environmental and seismic features.

2.1. Dataset-a preparation

We generate a synthetic environmental time series to train models for predicting floods and earthquakes. The data span 01-Jan-2024 to 01-Jan-2025 at hourly resolution (8,784 timestamps) and include three feature groups: i) meteorology: temperature, humidity, wind speed, rainfall; ii) hydrology: water level, flow rate, and iii) seismology: magnitude, depth, frequency. Table 1 summarizes the distribution type and the physical interpretation for each feature. In natural disaster prediction tasks, model robustness is a critical concern due to the inherent rarity and imbalance of high-severity events. To address this, the data-generation process explicitly simulates rare extreme conditions rather than relying on post hoc data augmentation.

Table 1. Summary of feature generation for synthetic dataset

Feature	Symbol	Distribution / model	Physical interpretation
		Meteorological variables	
Temperature	T_i	Gaussian ($\mu = 25$, $\sigma = 5$), clipped $[-5, 45]$	Represents air temperature with daily and seasonal variation.
Humidity	H_i	Derived from T_i with inverse relation; Gaussian noise $\sigma = 3$	Relative humidity inversely related to temperature, bounded between 20–100%.
Wind speed	W_i	Rayleigh ($\sigma = 3.5$) with added Gamma bursts during storms; capped at 28 m/s	Background wind speed with convective gusts during storm episodes.
Rainfall	R_i	Gamma (shape=1.8, scale=4.0) mixed with Bernoulli rain occurrence; seasonal and diurnal modulation	Hourly precipitation influenced by time of year and time of day.
		Hydrological variables	
Water level	L_i	Recursive (AR-1) with $\alpha = 0.965$ and rainfall-dependent response	Accumulates rainfall and decays slowly over time, capturing antecedent wetness.
Flow rate	Q_i	Power-law $Q \propto (L - H_0)^{1.6}$ with Gaussian noise $\sigma = 6$	River discharge rises nonlinearly with water level, representing runoff response.
		Seismic variables	
Magnitude	M_i	Exponential ($\lambda = 1.0$), bounded [0, 6.5]	Earthquake magnitude approximates a Gutenberg–Richter–like exponential decay (small frequent, large rare).
Depth	D_i	Normal ($\mu = 12$, $\sigma = 6$), clipped [0.5, 50]	Depth of seismic events spanning shallow to intermediate crustal regions.
Frequency	F_i	Poisson with rate λ decreasing exponentially with M_i and D_i	Expected hourly count of seismic events inversely related to magnitude and depth.

2.2. Dataset labeling

The World-A dataset is split into training and testing subsets, 80% and 20%, respectively. Samples in both subsets are assigned class labels based on predefined threshold values indicating potentially hazardous conditions. The empirical selection method is used to determine threshold values that maintain the physical realism of the synthetic disaster simulation, ensuring that both flood and earthquake scenarios appropriately represent plausible environmental dynamics. To eliminate conflicts, Algorithm 1 assigns labels with earthquake priority. Labels begin with class 0 (no-disaster), and progress to class 2 (earthquake) when low seismic frequency, high magnitude, and shallow depth are all met. Remaining unlabeled samples are classified as class 1 (flood) if flow rate, wind speed, and humidity exceed the threshold values. This sequential rule-based scheme cleanly separates hazards while prioritizing rarer, high-severity earthquakes.

Algorithm 1 Disaster labeling logic based on thresholds

```
Initialize all labels as no disaster $\leftarrow 0$
for each row in the dataset do
    if Seismic frequency $<$ threshold and seismic depth $<$ threshold and seismic magnitude $>$ threshold then
        Assign label 2 (earthquake)
    else if Disaster == 0 and flow rate $>$ threshold and wind speed $>$ threshold and humidity $>$ threshold then
        Assign label 1 (flood)
    else
        Keep label as 0 (no disaster)
    end if
end for
```

Twelve predictors are used: 9 numeric (temperature, humidity, wind speed, seismic magnitude, seismic depth, seismic frequency, water level, rainfall, and flow rate) and 3 categorical (hour, day of week, and month). Numeric features are normalized using min–max scaling, while categorical features are one-hot encoded with the first category dropped. All preprocessing is wrapped in a unified column transformer fit only on a-train, then applied to a-test to prevent leakage.

2.3. Models training

Three ML models, logistic regression (LR), RF, and XGBoost, are subsequently trained for multiclass disaster classification, distinguishing between no-disaster, flood, and earthquake events. All preprocessing steps are integrated within a training-only pipeline, and no imputation is applied unless missing values are intentionally introduced during robustness experiments.

2.3.1. Logistic regression

LR can be used to classify input data into multiple classes. In this study, we assign class 0 for non-disaster, class 1 for flood, and class 2 for earthquake, as explained in Algorithm 1. Since there are three possible outcomes, multiclass LR is applied using the softmax function, as shown in Algorithm 2 to calculate the probability that an input corresponds to each class. The probability for class *j* given input *x* is calculated using (1). Where: $x \in \mathbb{R}^d$ is the input feature vector, $\theta_j \in \mathbb{R}^d$ is the weight vector for class j, and $\theta_j^\top x$ represents the dot product between the weights and the input features. The denominator ensures that the sum of probabilities over all classes equals 1.

$$P(y = j \mid x) = \frac{e^{\theta_j^\top x}}{e^{\theta_0^\top x} + e^{\theta_1^\top x} + e^{\theta_2^\top x}} \quad \text{for } j \in \{0, 1, 2\} \tag{1}$$

Algorithm 2 Multiclass logistic regression using softmax

```
Require: Dataset $\{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^{N}$, learning rate $\alpha$, iterations $T$
Initialize weight vectors $\theta_0, \theta_1, \ldots, \theta_{K-1} \in \mathbb{R}^d$
for $t = 1$ to $T$ do
    for each sample $(\mathbf{x}^{(i)}, y^{(i)})$ do
        Compute logits: $z_j = \theta_j^\top \mathbf{x}^{(i)}$
        Compute probabilities: $p_j = \frac{e^{z_j}}{\sum_{k=0}^{K-1} e^{z_k}}$
        for each class $j$ do
            Update weights: $\theta_j \leftarrow \theta_j - \alpha \left(p_j - \nVdash_{\{y^{(i)}=j\}}\right) \mathbf{x}^{(i)}$
        end for
    end for
end for
Prediction: $\hat{y} = \arg\max_j p_j$
```

2.3.2. Random forest

RF enhances prediction accuracy and mitigates overfitting by combining multiple DTs into an ensemble. Initially, multiple random subsets are generated from the training dataset using bootstrap sampling, and each subset is used to independently train a DT. During tree construction, a random selection of features is taken at each node split, and the best feature among them is chosen using impurity reduction criteria. This randomization minimizes correlation between individual trees, enhancing the model's generalization ability. During prediction, each tree votes for a class label, and the final prediction is determined by majority voting across all trees. Algorithm 3 explains the main process of the RF algorithm, where each $h_t(x)$ is a DT trained independently. The bootstrap sampling ensures diversity in data, random feature selection at each node reduces tree correlation, and the final prediction is made by majority vote in classification tasks.

2.3.3. XGBoost

XGBoost is an ensemble learning algorithm that builds a strong classifier by sequentially combining multiple DTs. Unlike RF, where trees are trained independently, XGBoost trains each new tree to correct the errors made by the previous ensemble. XGBoost begins with an initial prediction of 0, then calculates the first derivative (gradient) and the second derivative (Hessian) of the loss function with respect to the predictions.

These are used to fit a new DT that predicts the residual errors. The output of each new tree is scaled by a learning rate η and added to the previous prediction. This process is repeated over multiple boosting rounds. Algorithm 4 outlines the main steps of XGBoost, where: ℓ is the loss function to be minimized, $\Omega(f_t)$ is a regularization term that penalizes tree complexity to prevent overfitting, and $\eta \in (0, 1]$ is the learning rate that controls the contribution of each tree. At each iteration, the model fits a new tree $f_t(x)$ to minimize a regularized objective function using a second-order Taylor approximation. The predictions are updated in step 6.

Algorithm 3 Random forest classifier

```
Require: Training data $\mathcal{D}$, number of trees $T$, number of features per split $m$
Ensure: Final prediction $\hat{y}$
for $t = 1$ to $T$ do
    Draw a bootstrap sample $\mathcal{D}_t$ from $\mathcal{D}$ (sample with replacement)
    Train a DT $h_t(x)$ on $\mathcal{D}_t$
    for each split node in the tree do
        Select $m$ random features from the $d$ total features
        Choose the best feature among $m$ based on impurity reduction
    end for
end for
For an unseen input $x$, aggregate predictions:
        $\hat{y} = \text{mode}\left(\{h_t(x)\}_{t=1}^{T}\right)$
```

Algorithm 4 XGBoost classifier

```
Require: Training data $\mathcal{D}$, number of boosting rounds $T$, learning rate $\eta$
Ensure: Final prediction model $\hat{y}(x)$
1: Initialize predictions: $\hat{y}_i^{(0)} \leftarrow 0$ for all $i$
2: for $t = 1$ to $T$ do
3:     Compute gradients: $g_i = \frac{\partial \ell(y_i, \hat{y}_i^{(t-1)})}{\partial \hat{y}_i^{(t-1)}}$
4:     Compute Hessians: $h_i = \frac{\partial^2 \ell(y_i, \hat{y}_i^{(t-1)})}{\partial (\hat{y}_i^{(t-1)})^2}$
5:     Fit a regression tree $f_t(x)$ to minimize:
           $\sum_{i=1}^{n} \left[ g_i f_t(x_i) + \frac{1}{2} h_i f_t(x_i)^2 \right] + \Omega(f_t)$
6:     Update prediction: $\hat{y}_i^{(t)} \leftarrow \hat{y}_i^{(t-1)} + \eta \cdot f_t(x_i)$
7: end for
8: return Final model: $\hat{y}(x) = \sum_{t=1}^{T} \eta \cdot f_t(x)$
```

2.4. Generating the independent test World-B

To confirm the generalization of the trained models, an independent test is performed to determine their robustness for natural disaster classification. The World-B test set is created synthetically, using the same feature schema and temporal structure as the World-A dataset, but with incorporating new stochastic realizations for all variables. Table 2 shows the key differences between the World-A and World-B datasets. This approach ensures that the models are exposed to unseen environmental and seismic conditions while maintaining consistent physical relationships between features.

2.5. Experimental setup for evaluating machine learning under IoT inefficiencies

A controlled robustness testing approach was used to assess the trained models' stability and generalization capability across different environmental and sensor conditions. The evaluation utilized the independent World-B dataset, maintaining its original labels to ensure independent and consistent ground truth across disruption situations. Three stress scenarios were designed to replicate realistic disturbances in IoT-based monitoring systems: i) sensor failure simulation: a portion of sensor readings was randomly halted, simulating temporary device failures or communication outages, where the absent values were substituted using a hold-last-value (HLV) approach; ii) noisy sensor simulation: additive Gaussian noise and low-frequency drift were injected into continuous features to imitate sensor calibration drift or interference; and iii) extreme weather simulation: meteorological and hydrological variables were scaled to reflect storms, including rainfall, humidity, wind speed, water level, and flow rate. In contrast, seismic features remained unchanged, as these events are primarily environmental rather than tectonic. To ensure statistical reliability, each scenario was repeated at five severity levels (10%-50%) using five random seeds. This examination of multi-seed resilience provides a quantitative assessment of each algorithm's response to sensor degradation and extreme environmental variation. By isolating disruptions from label generation, the framework assesses model-only robustness, ensuring that any reported performance changes are due to instability at the feature level rather than fluctuations in the target.

Table 2. Comparison between World-A and World-B datasets

Aspect	World-A dataset	World-B dataset
Purpose	Model training and in-domain evaluation	Independent testing for generalization
Data source	Generated once using fixed random seed (`SEED = 42`)	Regenerated independently using a new random seed
Feature schema	9 numeric + 3 temporal/categorical features	Identical schema replicated for comparability
Data distribution	Shares same stochastic patterns as training data	Different random realizations and temporal sequences
Thresholds for labeling	Fixed physical thresholds (same across world)	Same thresholds reused to preserve label semantics
Evaluation type	In-distribution evaluation	Out-of-sample evaluation
Expected behavior	High performance due to shared distribution	Lower but more realistic performance under new conditions

3. RESULTS AND DISCUSSION

The performance of the trained ML models on the labeled disaster dataset is evaluated. The models are assessed based on their classification accuracy, and their robustness is analyzed under various simulated real-world conditions, including sensor noise and data irregularities.

3.1. Dataset generation

A Pearson correlation heatmap was constructed using all continuous variables to evaluate the internal consistency of the synthetic dataset features. The heatmap shown in Figure 1 shows a physically coherent relationship between the meteorological, hydrological, and seismic features. The meteorological and hydrological features have a moderate positive correlation, where humidity shows a strong negative correlation with temperature. Regarding the seismic subsystem, there is a moderate negative correlation between seismic magnitude and seismic frequency, reflecting the inverse relationship built into the synthetic dataset generation process. The correlation heatmap in Figure 1 validates both the realism and internal consistency of the synthetically generated dataset. These correlation patterns were obtained by a stochastic–empirical generation process designed to mimic the natural behavior of IoT sensors capturing floods and earthquake parameters. This correlation method is considered one of the limitations of this study, where real-world data would give an accurate reading and correlations among the features.

3.2. Evaluation of machine learning models performance

The performance of the trained model is assessed using four classification metrics: accuracy, precision, recall, and F1-score, as shown in Table 3, where the number of testing samples in the World-A dataset is 1,757 for three different classes: 1,690 for the no-disaster class, 34 for the flood class, and 33 for the earthquake class. The accuracy metric is used to measure the proportion of the correctly classified sample among all the predictions. However, precision is used to evaluate the proportion of true positives among all positive predictions for each class. The recall metric is used to show the number of actual events that have been detected successfully. F1-score is used to combine precision and recall in one metric to evaluate the models when dealing with an imbalanced dataset. Since the problem involves multiclass classification (no disaster, flood, and earthquake), the metrics were computed using macro-averaging to treat all classes equally regardless of sample count.

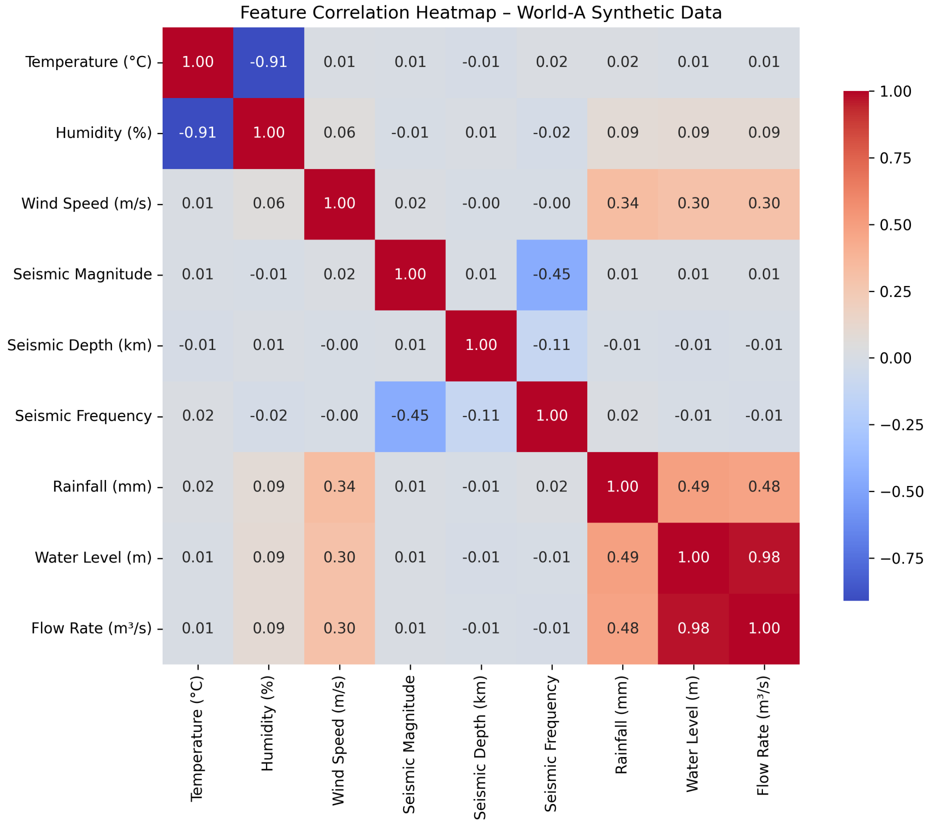


Figure 1. Dataset feature correlation heatmap

Table 3. Classification metrics, formulas, and brief definitions

Metric	Formula	Definition
Accuracy	$\frac{\mathrm{TP+TN}}{\mathrm{TP+TN+FP+FN}}$	Proportion of correctly classified samples among all predictions.
Precision$_i$	$\frac{\mathrm{TP}_i}{\mathrm{TP}_i+\mathrm{FP}_i}$	Share of predicted positives for class i that are truly positive.
Recall$_i$	$\frac{\mathrm{TP}_i}{\mathrm{TP}_i+\mathrm{FN}_i}$	Share of actual positives for class i that are correctly detected.
F1$_i$	$\frac{2\,\mathrm{Precision}_i\,\mathrm{Recall}_i}{\mathrm{Precision}_i+\mathrm{Recall}_i}$	Harmonic mean of precision and recall for class i; robust to imbalance.

Table 4 summarizes the computed evaluation metrics for the three implemented models: LR, RF, and XGBoost. The results clearly demonstrate the superior performance of the ensemble-based models over the linear baseline. RF and XGBoost achieved near-perfect scores across all metrics, showing their robustness in handling the complex and nonlinear relationships present in the disaster dataset and their resilience to class imbalance. In contrast, LR performed significantly worse, particularly in precision, demoenstrating a higher tendency to incorrectly classify non-disaster instances as flood or earthquake events. This underperformance highlights the limitations of linear models in disaster classification scenarios where feature interactions are inherently nonlinear and overlapping.

Table 4. Evaluation metrics for disaster classification models (based on test set)

Model	Accuracy (%)	Precision (%) (Macro)	Recall (%) (Macro)	F1-score (%) (Macro)
LR	96.8	69.7	97.9	79.6
RF	99.9	99.0	99.0	99.0
XGBoost	99.7	96.2	98.0	97.0

Figure 2 shows the confusion matrices for the three models. Ensemble models, RF and XGBoost, show an ideal performance for the classification of the three classes, while LR shows false alarm for the no-disaster class. However, these ideal results showed because of the ability of the models to replicate the deterministic threshold logic used for labeling rather than true generalization. As a result, a further evaluation as an independent dataset World-B is required to test predictive robustness.

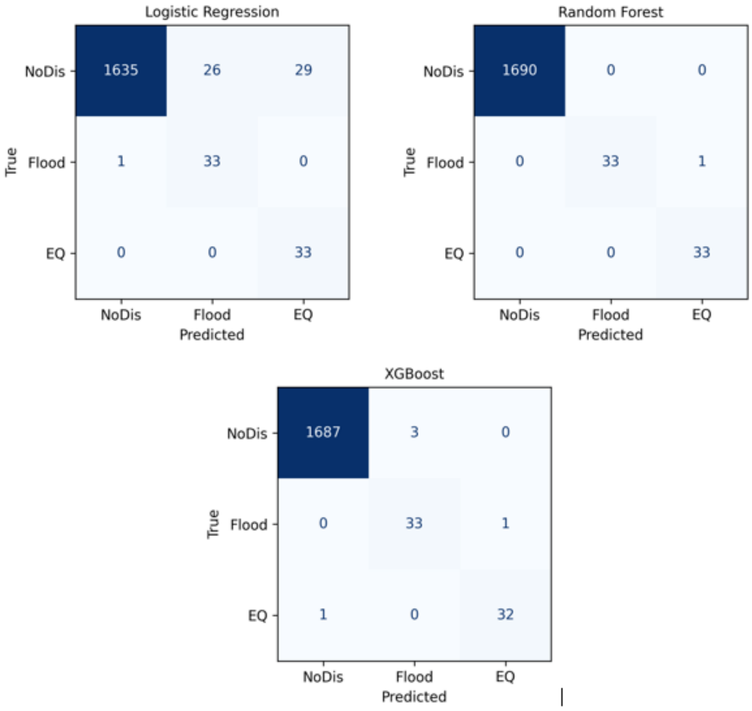


Figure 2. Confusion matrices for the World-A test set

3.3. Validation of the trained models using independent test World-B

The predictive performance of the three trained models was further evaluated using the independently generated World-B dataset. The purpose of this evaluation is to assess the ability of the models to generalize beyond the statistical distributions of the training data in World-A. The World-B test provides a rigorous independent assessment of robustness under distributional shift, where identical feature schema and a fixed labeling thresholds are preserved while modifying the underlying data-generation parameters. The World-B generated dataset consists of 7,830 no-disaster, 870 flood, and 84 earthquake samples. Table 5 illustrates the performance metrics for the three trained models. XGBoost model has the highest F1-score, then RF slightly low, which confirms the ability of the ensemble models to be generalized for natural disaster prediction. On the other hand, LR shows a notable decline in predictive performance, particularly in precision and F1-score, showing its limited capacity to capture the nonlinear feature interactions inherent in environmental and seismic systems. Figure 3 confirms the out-performance of the ensemble models rather than linear models, confirming the ability of XGBoost and RF to be adaptive with complex and nonlinear data distributions.

Table 5. World-B model comparison (macro metrics)

Model	Accuracy (%)	Precision$_{macro}$ (%)	Recall$_{macro}$ (%)	F1$_{macro}$ (%)
LR	70.6	50.6	83.7	54.6
RF	89.0	78.5	78.1	78.3
XGBoost	85.2	78.2	91.1	80.5

Figure 4 shows the confusion matrices to visualize the prediction outcomes of the three models on the independent World-B dataset. The LR model exhibits high false-positive rates, specially it is misclassifying a large number of no disaster samples as flood, indicating it is sensitivity to overlapping meteorological features. LR true positives for flood and earthquake remain acceptable, with substantial confusion across classes. In contrast, the RF model achieves a stronger diagonal dominance, with the majority of no disaster and earthquake samples correctly classified and moderate confusion between flood and no disaster. The XGBoost model shows the most balanced behavior, minimizing false negatives while maintaining high true-positive counts for all classes. Overall, Figure 4 demonstrates that ensemble models XGBoost and RF better capture nonlinear feature dependencies compared to LR. Specifically, XGBoost reduces misclassifications more effectively than RF, which is biased to no-disaster class during flood events. In addition, LR struggles with complex features classification leading to false-alarms.

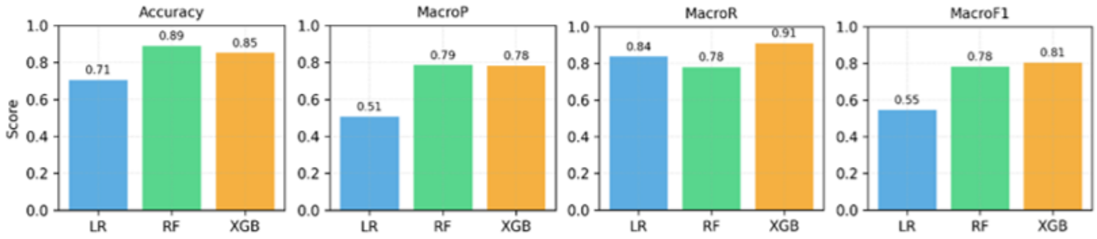


Figure 3. Comparison of model performance metrics on the independent World-B dataset

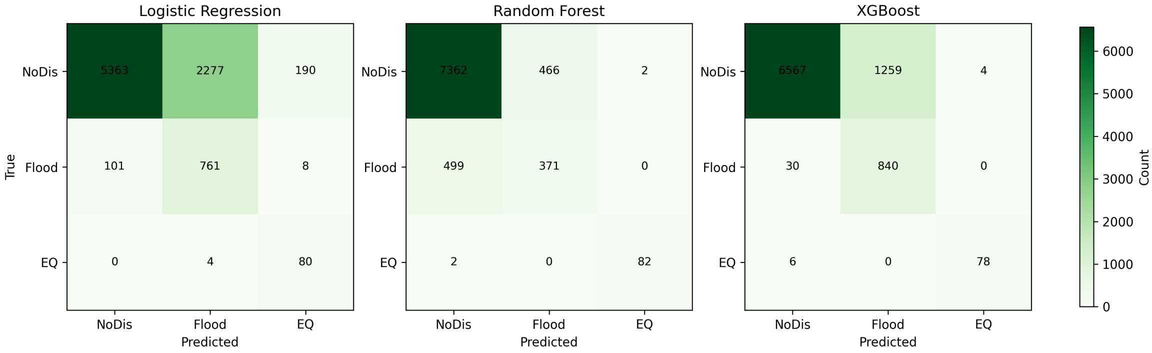


Figure 4. Confusion matrices for the three trained models using the World-B dataset

3.4. Testing of realistic scenarios

The robustness of the three trained models, LR, RF, and XGBoost, under realistic scenarios is essential to assess the use of the ML models to predict the occurrence of natural disasters. To evaluate the model's robustness under such conditions, the three classifiers were tested using three scenarios: sensor failures, noisy sensors, and severe weather conditions. Figure 5 shows the robustness evaluation of the three trained models using the World-B dataset. Across all conditions, a consistent performance degradation is observed as the disruption severity increases from 10% to 50%. The ensemble models, XGBoost and RF, demonstrate substantially greater resilience than the linear baseline, LR. In the sensor failure scenario, XGBoost starts with a high macro-F1-score of 0.76 and RF with 0.74, but both decrease modestly to 0.57 at 50% failure. This shows that tree-based methods preserve prediction fidelity even with partial data loss. In comparison, LR shows a steeper decline from 0.52 to 0.43 F1-score, demonstrating sensitivity to missing inputs. For the noisy sensor scenario, the XGBoost model starts with a high F1-score and maintains a constant performance, reaching 0.73 even at the maximum noise level. The same scenario for the RF model, confirming their tolerance to feature disruption. On the other hand, LR has a stable low F1-score among all the noisy sensor scenarios. In the extreme weather test, which amplifies environmental parameters only, results show that RF starts with a comparatively robust F1-score of 0.80 and declines to 0.75. While XGBoost's boosting framework is inherently more sensitive to environmental feature shifts. The LR model shows a low F1-score for the extreme weather as well, compared with ensemble models. Overall, ensemble-based approaches exhibit strong robustness to sensor failures, noisy sensor and extreme weather scenarios, validating their suitability for IoT-based disaster prediction systems operating under uncertain sensing conditions. The linear LR model, limited by its inability to capture nonlinear dependencies, demonstrates lower adaptability and poorer fault tolerance.

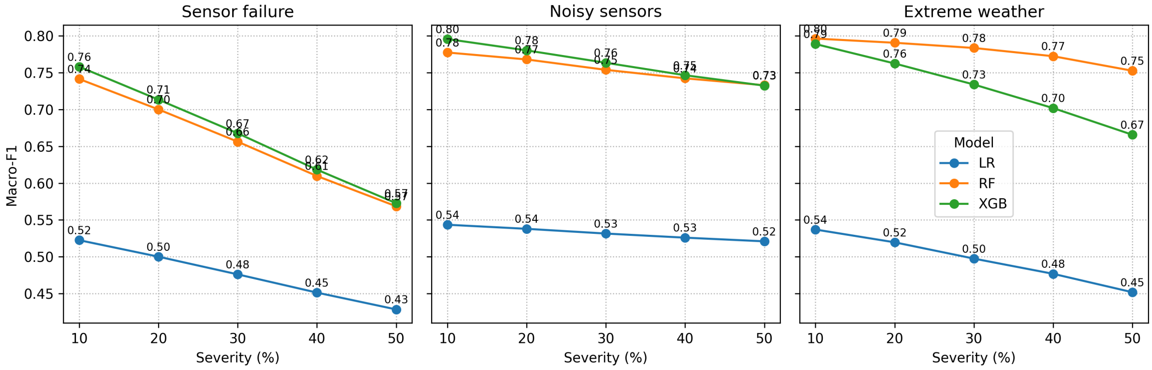


Figure 5. Robustness evaluation of the trained models under three disturbance scenarios

4. CONCLUSION

This study demonstrated that ensemble-based ML models can effectively predict floods and earthquakes from synthetic IoT sensor data. This paper underlined the importance of resilient and accurate disaster prediction systems, and the findings supported this expectation. XGBoost and RF outperformed LR and remained robust under stress conditions such as sensor failure, noisy sensors, and extreme weather scenarios. These findings confirm the potential of ensemble learning as a foundation for intelligent EWS. Future studies should focus on expanding the dataset with additional environmental and seismic parameters, investigating DL approaches for spatiotemporal prediction, and deploying the models in real IoT environments for field validation. Such advancements will enhance the reliability and applicability of disaster prediction systems in practice.

ACKNOWLEDGMENTS

The authors would like to thank Al-Hussein Bin Talal University for its support.

FUNDING INFORMATION

This research was funded in part by the NATO Science and Peace Programme under grant SPS MYP G5932-RESCUE.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Moath Alsafasfeh	✓	✓	✓	✓	✓	✓		✓	✓			✓		
Abdullah Alhasanat	✓					✓				✓	✓		✓	✓
Atheer Bassel				✓		✓			✓		✓			
Mohanad Alhasanat					✓	✓				✓	✓			

C	: **C**onceptualization	I	: **I**nvestigation	Vi	: **Vi**sualization
M	: **M**ethodology	R	: **R**esources	Su	: **Su**pervision
So	: **So**ftware	D	: **D**ata Curation	P	: **P**roject Administration
Va	: **Va**lidation	O	: Writing - **O**riginal Draft	Fu	: **Fu**nding Acquisition
Fo	: **Fo**rmal Analysis	E	: Writing - Review & **E**diting		

CONFLICT OF INTEREST STATEMENT

The authors state no conflict of interest.

DATA AVAILABILITY

The synthetic IoT sensor dataset generated and analyzed during this study, along with the preprocessing and labeling scripts, is available from the corresponding author, [MA], upon reasonable request.

REFERENCES

[1] A. Ardalan and C. A.- Adegbulu, "Introduction to natural disasters," in *Ciottone's Disaster Medicine*, New Delhi, India: Elsevier, 2024, pp. 594–597, doi: 10.1016/B978-0-323-80932-0.00094-X.

[2] M. Krichen, M. S. Abdalzaher, M. Elwekeil, and M. M. Fouda, "Managing natural disasters: an analysis of technological advancements, opportunities, and challenges," *Internet of Things and Cyber-Physical Systems*, vol. 4, pp. 99–109, 2024, doi: 10.1016/j.iotcps.2023.09.002.

[3] UNDRR, "Sendai framework terminology on disaster risk reduction," *United Nations Office for Disaster Risk Reduction*. Accessed: May 16, 2025. [Online]. Available: https://www.undrr.org/terminology/early-warning-system

[4] M. Esposito, L. Palma, A. Belli, L. Sabbatini, and P. Pierleoni, "Recent advances in internet of things solutions for early warning systems: a review," *Sensors*, vol. 22, no. 6, Mar. 2022, doi: 10.3390/s22062124.

[5] A. S. Albahri *et al.*, "A systematic review of trustworthy artificial intelligence applications in natural disasters," *Computers and Electrical Engineering*, vol. 118, Sep. 2024, doi: 10.1016/j.compeleceng.2024.109409.

[6] V. D. Gowda, A. Sharma, K. D. V. Prasad, R. Saxena, T. Barua, and K. Mohiuddin, "Dynamic disaster management with real-time IoT data analysis and response," *2024 International Conference on Automation and Computation (AUTOCOM)*, 2024, pp. 142–147, doi: 10.1109/AUTOCOM60220.2024.10486101.

[7] P. Purushotham, D. D. Priya, and A. Kiran, "Disaster analysis using machine learning," *2022 International Conference on Advancements in Smart, Secure and Intelligent Computing (ASSIC)*, Bhubaneswar, India, 2022, pp. 1-6, doi: 10.1109/ASSIC55218.2022.10088390..

[8] J. Pwavodi, A. U. Ibrahim, P. C. Pwavodi, F. Al-Turjman, and A. M.- Said, "The role of artificial intelligence and IoT in prediction of earthquakes: review," *Artificial Intelligence in Geosciences*, vol. 5, Dec. 2024, doi: 10.1016/j.aiig.2024.100075.

[9] A. Akhyar *et al.*, "Deep artificial intelligence applications for natural disaster management systems: a methodological review," *Ecological Indicators*, vol. 163, Jun. 2024, doi: 10.1016/j.ecolind.2024.112067.

[10] A. Moghadamnejad, M. A. Moghaddasi, M. Hamidia, R. K. Mohammadi, and M. Zare, "Ranking earthquake prediction algorithms: a comprehensive review of machine learning and deep learning methods," *Soil Dynamics and Earthquake Engineering*, vol. 200, Jan. 2026, doi: 10.1016/j.soildyn.2025.109740.

[11] M. E. Deowan, S. Haque, J. Islam, M. Hanjalayeamin, M. T. Islam, and R. T. Meghla, "Smart early flood monitoring system using IoT," in *2022 14th Seminar on Power Electronics and Control (SEPOC)*, Santa Maria, Brazil, 2022, pp. 1-6, doi: 10.1109/SEPOC54972.2022.9976434.

[12] K. P. I. Karunawardhana, S. P. Munasinghe, J. G. S. Samiru, E. M. H. K. B. Ekanayake, and N. C. Amarasena, "Enhancing human safety through early detection of floods and landslides in Sri Lanka," in *2024 9th International Conference on Information Technology Research (ICITR)*, Dec. 2024, pp. 1–6, doi: 10.1109/ICITR64794.2024.10857761.

[13] N. U. Reddy, P. V. Kumar, N. Kapileswar, J. Simon, and P. P. Kumar, "A prediction model for minimization of flood effects using machine learning algorithms," in *2022 Sixth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, Nov. 2022, pp. 593–597, doi: 10.1109/I-SMAC55078.2022.9987332.

[14] M. Khalaf *et al.*, "IoT-enabled flood severity prediction via ensemble machine learning models," *IEEE Access*, vol. 8, pp. 70375–70386, 2020, doi: 10.1109/ACCESS.2020.2986090.

[15] S. M. Rezvani, A. Gonçalves, M. J. F. Silva, and N. M. de Almeida, "Smart hotspot detection using geospatial artificial intelligence: a machine learning approach to reduce flood risk," *Sustainable Cities and Society*, vol. 115, Nov. 2024, doi: 10.1016/j.scs.2024.105873.

[16] M. Anbarasan *et al.*, "Detection of flood disaster system based on IoT, big data and convolutional deep neural network," *Computer Communications*, vol. 150, pp. 150–157, Jan. 2020, doi: 10.1016/j.comcom.2019.11.022.

[17] K. R. Amat, A. Baitha, B. AdhikariChhetri, N. Bohara, C. Bista, and H. S. Shreenidhi, "QuakeGuard insight: an IoT-enabled machine learning system," in *2024 4th International Conference on Data Engineering and Communication Systems (ICDECS)*, 2024, pp. 1–5, doi: 10.1109/ICDECS59733.2023.10502754.

[18] B. Mukherjee, R. L. Shaw, M. L. Sharma, and K. Sain, "Earthquake prediction using machine learning perspectives in Himalayan seismic belt and its surroundings," *Journal of Asian Earth Sciences*, vol. 293, Nov. 2025, doi: 10.1016/j.jseaes.2025.106764.

[19] C.-M. Rosca and A. Stancu, "Earthquake prediction and alert system using IoT infrastructure and cloud-based environmental data analysis," *Applied Sciences*, vol. 14, no. 22, Nov. 2024, doi: 10.3390/app142210169.

[20] H. Kubo, M. Naoi, and M. Kano, "Recent advances in earthquake seismology using machine learning," *Earth, Planets and Space*, vol. 76, no. 1, Feb. 2024, doi: 10.1186/s40623-024-01982-0.

[21] B. Anuradha, C. Abinaya, M. Bharathi, A. Janani, and A. Khan, "IoT based natural disaster monitoring and prediction analysis for hills area using LSTM network," in *2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS)*, Mar. 2022, pp. 1908–1913, doi: 10.1109/ICACCS54159.2022.9785121.

[22] P. Sreevidya, C. S. Abhilash, J. Paul, and G. Rejithkumar, "A machine learning-based early landslide warning system using IoT," in *2021 4th Biennial International Conference on Nascent Technologies in Engineering (ICNTE)*, 2021, pp. 1–6, doi: 10.1109/ICNTE51185.2021.9487669.

[23] A. Amgain, N. Kumar, S. Bajgain, and H. Rai, "Landslides prediction and detection using IoT system," in *2023 2nd International Conference on Vision Towards Emerging Trends in Communication and Networking Technologies (ViTECoN)*, May 2023, pp. 1–6, doi: 10.1109/ViTECoN58111.2023.10157077.

[24] V.-T. Nguyen, Q.-A. Nguyen, and N.-K. Nguyen, "Method for monitoring and forecasting landslide phenomenon based on machine learning," *MethodsX*, vol. 12, Jun. 2024, doi: 10.1016/j.mex.2024.102797.

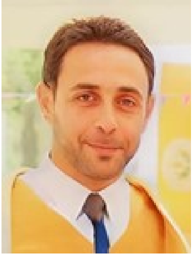
[25] R. K. Singh and O. Prakash, "Disaster detection and alert generation for smart city scenario using deep learning technique," in *2023 2nd International Conference for Innovation in Technology (INOCON)*, 2023, pp. 1–6, doi: 10.1109/INOCON57975.2023.10101261.

BIOGRAPHIES OF AUTHORS

Moath Alsafasfeh is an assistant professor at Tuskegee University, Tuskegee, Alabama, USA. He was an associate professor of Electrical and Computer Engineering at Al-Hussein Bin Talal University, Jordan, from June 2017 until August 2024. He earned his Ph.D. from Western Michigan University (WMU) in 2017, an M.Sc. in Communication and Computer Engineering from the National University of Malaysia (UKM) in 2011, and a B.Sc. degree in Computer Engineering from Mutah University in 2009. His research interests are parallel processing, machine vision, image processing, and machine learning for WSNs and IoT network applications. He has published more than 24 articles in prestigious Scopus-indexed journals. He was awarded several national prizes, and scholarships to get Ph.D. and bachelor's degrees. He has strong relationships with different local, regional, and international researchers. He has been working as a co-PI for 2 internationally funded research projects and 1 project for Capacity Building in Higher Education. He can be contacted at email: malsafasfeh@tuskegee.edu.

Abdullah Alhasanat received a B.Sc. degree in Computer Engineering from the University of Aden in Yemen in 2004, his M.Sc. degree in Computer Engineering from Jordan University of Science and Technology in Jordan in 2007, and his Ph.D. degree in wireless networks from the University of New Castle in the United Kingdom in 2012. In 2012, he joined the Department of Computer Engineering at Al-Hussein Bin Talal University, Jordan. Currently, he is a full professor in Wireless Networking. He published many articles in high-level scientific journals and international conferences. His main research interests include routing and localization in ad-hoc networks, wireless sensor networks, MANET, VANET, delay tolerant networks, parallel computing, and signal and image processing. Currently, he is the dean of the College of Engineering at Al-Hussein Bin Talal University, Jordan. He can be contacted at email: abad@ahu.edu.jo.

Atheer Bassel received the B.Sc. degree in Computer Science from the University of Anbar, Iraq, in 2004, the M.Sc. degree in Computer Science, in 2011, and the Ph.D. degree in Computer Science from Universiti Kabangsaan Malaysia (UKM), in 2018. He is currently an assistant professor with the Department of Artificial Intelligence Science, Faculty of Computer Science and Information Technology, University of Anbar. His main research interests include metaheuristics, optimization algorithms, and image processing and their applications. He can be contacted at email: atheerbassel@uoanbar.edu.iq.



Mohanad Alhasanat received his Ph.D. from the Islamic Science University of Malaysia (USIM) in 2016. Currently, he is an associate professor of networks at the Department of Computer Engineering, Al-Hussein Bin Talal University (Jordan). His research focuses on computer networks including enhancing TCP congestion control mechanisms over wireless networks. In addition, he has research interests in delay tolerant networks, artificial intelligence, computer vision, and deep machine learning. He can be contacted at email: mohanadhasanat@ahu.edu.jo.