Available online at website: https://jurnal.iaii.or.id/index.php/RESTI

JURNAL RESTI
(Rekayasa Sistem dan Teknologi Informasi)
Vol. 10 No. 4 (2026) 1005 - 1012 e-ISSN: 2580-0760

Benchmarking Lightweight Machine Learning for Disaster Prediction: Accuracy, Latency, and Readiness

Muhammad Amanulloh Mz[1], Oky Dwi Nurhayati[2], Jatmiko Endro Suseno[3]
[1]Doctor of Information Systems, Faculty of Graduate School, Diponegoro University, Semarang, Indonesia
[1]Information Systems and Technology, Faculty of Information Technology, PGRI Dewantara Jombang Institute of Technology and Business, Jombang, Indonesia
[2]Department of Computer Engineering, Diponegoro University, Semarang. Semarang, Indonesia
[3]Department of Physics, Diponegoro University, Semarang. Semarang, Indonesia

[*]*amanulloh@itebisdewantara.ac.id*

Abstract

Natural disaster early warning systems are often constrained by high computational latency, which impedes the timely dissemination of critical information. This study proposes a disaster impact prediction framework designed to optimize inference speed through a lightweight machine learning approach. Utilizing a comprehensive historical dataset from the Indonesian National Board for Disaster Management (BNPB) comprising 28,773 disaster records across 34 provinces from January 2018 to May 2024, this research evaluates the performance of XGBoost and LightGBM against a Gated Recurrent Unit (GRU) model for predicting infrastructure damage. SMOTE was applied exclusively during the training phase to address class imbalance without affecting inference speed. The results demonstrate that tree-based models significantly outperform GRU in both accuracy and speed. XGBoost achieved the lowest Mean Absolute Error (MAE of 1.1649) and the fastest inference latency (0.0026 ms per sample), followed by LightGBM (MAE of 1.4006, latency of 0.0064 ms), while GRU yielded a substantially higher error (MAE of 1.4189) and latency (0.2493 ms). Statistical validation via the Wilcoxon signed-rank test confirmed the significance of these performance differences ($p < 0.001$). Beyond technical benchmarking, these findings support the development of Intelligent Decision Support Systems for disaster mitigation, where organizational readiness and technology acceptance are critical for successful adoption. This study discusses practical implications for deploying lightweight models on resource-constrained edge computing devices such as Raspberry Pi 4, demonstrating sub-millisecond response speed, horizontal scalability, and affordability under $100 per node.

Keywords: disaster early warning systems; edge computing; inference latency; lightweight machine learning; XGBoost

How to Cite: Muhammad Amanulloh Mz, Oky Dwi Nurhayati, and Jatmiko Endro Suseno, "Benchmarking Lightweight Machine Learning for Disaster Prediction: Accuracy, Latency, and Readiness", *J. RESTI (Rekayasa Sist. Teknol. Inf.)* , vol. 10, no. 4, pp. 1005 - 1012, Aug. 2026. *Permalink/*DOI: https://doi.org/10.29207/resti.v10i4.7570

Received: February 9, 2026
Accepted: July 16, 2026
Available Online: August

This is an open-access article under the CC BY 4.0 License
Published by Ikatan Ahli Informatika Indonesia

1. Introduction

Disaster prediction has evolved into a fundamental pillar of early warning systems (EWS) and emergency management, particularly in high-stakes environments that demand instantaneous, data-driven decision-making [1], [2]. Recent breakthroughs in sensor technology, remote sensing, and machine learning have catalyzed the development of diverse predictive models, achieving remarkable accuracy across various phenomena, from hydrological floods and landslides to complex seismic activities [3], [4]. However, in practical deployment, the operational efficacy of a model is determined not only by its predictive precision but also by its computational efficiency, specifically regarding low latency and rapid inference capabilities. In time-sensitive emergency scenarios, the ability of a model to process data near-instantaneously is a critical determinant of the success of life-saving rescue operations [5], [6].

Despite rapid advancements in accuracy, current literature remains heavily skewed toward the development of novel architectures or marginal improvements in accuracy scores. Conversely, systematic evaluations of computational performance and response velocity remain largely under-explored [7], [8]. This research gap is further widened by significant variations in evaluation procedures ranging from inconsistent metric selection to disparate hardware specifications and testing protocols, which impede standardized performance benchmarking across models [9], [10]. In the absence of a structured benchmarking framework, developers of early warning systems face a persistent dilemma: the trade-off between achieving high accuracy and ensuring the fastest possible response time [11], [12].

To address these challenges, this study proposes a comprehensive benchmarking framework specifically engineered to evaluate disaster prediction models through the lens of computational efficiency. This framework integrates standardized modules for data processing, model execution, and performance quantification. The evaluation is conducted by juxtaposing traditional statistical metrics such as Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) with inference latency metrics (including tail-latency) and hardware resource utilization profiles [13]. Such a multi-dimensional approach enables the precise mapping of trade-offs to identify lightweight yet robust model architectures suitable for real-time operational implementation [14], [15]. The primary contributions of this research are fourfold: the development of an integrated benchmarking framework focused on ultra-low latency; the provision of standardized evaluation protocols for measuring model response speed; an empirical analysis of the correlation between computational efficiency and prediction quality; and the recommendation of high-velocity model architectures, including XGBoost and LightGBM, optimized for real-time emergency response systems.

Research into machine learning applications for disaster prediction has expanded rapidly; however, the emphasis remains predominantly on model proliferation rather than standardized performance validation [16], [17]. Kappi and Mallikarjuna (2024) highlight that while research volume has surged, there is a distinct lack of consistent evaluation across heterogeneous models and datasets. Similarly, Linardos et al. emphasize the necessity for replicable evaluation protocols to align research outcomes with the dynamic operational requirements of disaster management [18], [19]. In an effort to promote standardization, several benchmark dataset initiatives have emerged. NADBenchmarks provides a catalog of disaster-related datasets while highlighting the scarcity of standardized data for equitable model comparison. In the field of seismology, EarthquakeNPP introduces a benchmark collection for earthquake forecasting, stressing the importance of validating predictive performance against established baseline models, such as ETAS [20], [21].

In domain-specific applications, flood-related research by Oddo et al. demonstrates accuracy improvements using ConvLSTM; however, their evaluation fails to address the crucial aspect of computational latency for real-time utility. In contrast, Zhou et al. demonstrated a deep learning-based flood model that achieved a 19,585x speedup despite a marginal degradation in accuracy. This finding reinforces the necessity of evaluating accuracy-speed trade-offs when developing rapid-response models. Furthermore, while the application of BERT models to social media streams (e.g., Twitter) has improved prediction accuracy, the computational overhead required to process massive data streams instantaneously remains inadequately assessed [22-24].

From an operational standpoint, edge-computing research confirms that emergency response systems necessitate extremely low latency, making computational performance a decisive factor in model selection. Latency studies by the IEEE on edge clouds further underscore the need for testing under diverse load and network conditions to capture a realistic profile of response speeds in field conditions [25], [26]. Additionally, methodological best practices in algorithm evaluation recommend the use of non-parametric statistical tests, such as the Wilcoxon and Friedman tests, to ensure that observed differences in performance and efficiency are statistically significant [25], [26].

Beyond technical considerations, the successful deployment of disaster prediction systems in real-world settings requires attention to organizational and human factors. In the context of tourism destinations, which often face unique challenges due to fluctuating populations and varying levels of infrastructure preparedness, the adoption of Intelligent Decision Support Systems (IDSS) depends not only on technical accuracy and speed but also on the readiness of destination managers to accept and utilize such technologies. Prior research has demonstrated that the Technology Acceptance Model (TAM) and organizational e-readiness theory are critical frameworks for understanding and predicting technology adoption in public sector and emergency management contexts. However, these organizational dimensions have rarely been integrated with technical benchmarking of machine learning models for disaster prediction.

In summary, the prevailing literature suggests that while numerous disaster prediction models have been introduced, a comprehensive, standardized benchmarking framework capable of simultaneously assessing accuracy, latency, and computational efficiency is still lacking [27-29]. Furthermore, the integration of technical benchmarking with organizational readiness and technology acceptance considerations remains an underexplored area. This study addresses these critical gaps by developing and validating models with minimal computational loads to

guarantee an immediate response in life-critical situations, while also laying the technical groundwork for future research on technology adoption in tourism disaster mitigation. The systematic data collection and preprocessing pipeline from BNPB raw data to analysis-ready format is detailed in Section 2.1, ensuring methodological transparency and reproducibility. Based on a dataset of 28,773 disaster records from the BNPB database, the benchmarking results demonstrate that XGBoost achieves the highest predictive accuracy (MAE of 1.1649) and the fastest inference latency (0.0026 ms per sample), making it the optimal choice for real-time edge deployment in resource-constrained environments.

2. Methods

This study implements a systematic experimental approach to evaluate the trade-off between predictive accuracy and computational latency across various machine learning architectures. The primary objective is to develop a framework capable of delivering near-instantaneous responses in emergency scenarios by leveraging large-scale historical disaster data. The research workflow is illustrated in Figure 1.

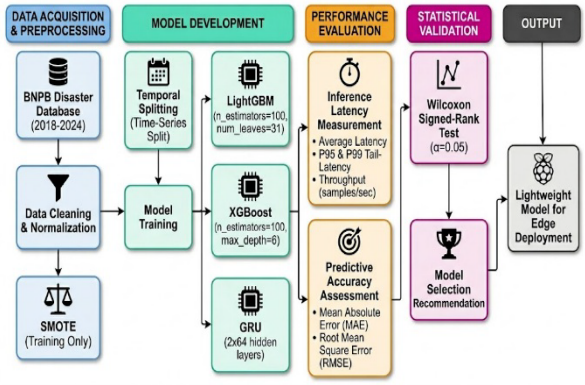


Figure 1. Research Workflow for Lightweight Machine Learning Benchmarking in Disaster Impact Prediction

As shown in Figure 1, the workflow consists of five sequential stages: data acquisition and preprocessing, model development, performance evaluation, statistical validation, and output recommendation. Each stage is designed to ensure methodological rigor while maintaining focus on real-time operational constraints. The following subsections describe each component in detail.

2.1 Data Collection and Description

The primary dataset originates from the official disaster database of the Indonesian National Board for Disaster Management (BNPB-Indonesia), comprising 28,773 disaster records across 34 provinces from January 2018 to May 2024 [30]. The data collection and preparation process followed a systematic pipeline to ensure data quality and reproducibility. First, raw data were extracted from the BNPB repository in CSV format, containing variables such as disaster type, location (province and district), date of occurrence, and impact indicators. Second, data validation was performed to remove duplicate entries and correct inconsistent date formats. Third, missing values in impact columns were handled using zero-value imputation, reflecting the absence of reported casualties or damage. Fourth, feature engineering was applied to derive additional temporal features (e.g., month, season). Fifth, categorical variables were transformed using One-Hot Encoding, and numerical features were normalized using Min-Max Scaling.

The final processed dataset consists of 28,773 samples ready for model training and inference. This multidimensional dataset includes temporal data, spatial identifiers (province and city/district), and disaster classifications such as floods, landslides, extreme weather, earthquakes, and volcanic eruptions. Quantitative impact indicators including mortality rates, missing persons, injuries, and damage to residential and public facilities serve as the target variables for prediction. To support implementation in tourism destinations, the spatial data includes disaster events occurring in or near major tourism areas in East Java, such as the Bromo Tengger Semeru National Park and the southern coastal tourism belt.

2.2 Data Preprocessing and Balancing Strategy

The preprocessing pipeline is designed to support low-latency requirements with minimal computational overhead. Categorical variables are transformed using One-Hot Encoding, while numerical features are normalized using the Min-Max Scaling technique. Min-Max Scaling converts numeric feature values into a specific scale range, generally from 0 to 1, by subtracting the minimum value and dividing by the range of the feature. After applying Min-Max Scaling, each numeric feature in the processed data has a minimum value of 0, a maximum value of 1, and all intermediate values proportionally distributed within this range. This transformation ensures that all features contribute equally to the learning process and accelerates model convergence.

Missing values in impact columns are handled using zero-value imputation, reflecting the absence of reported casualties or damage. To address class imbalance where extreme disaster events are rarer than minor ones, the Synthetic Minority Over-sampling Technique (SMOTE) is applied. SMOTE generates synthetic samples for minority classes by interpolating between existing minority instances, thereby training the model to become more sensitive to rare disaster patterns. A critical design choice in this study is that SMOTE is used exclusively during the training phase. During inference or prediction, once the model has been fully trained, SMOTE plays no role whatsoever. This ensures that the inference efficiency of the models remains fully optimized, and disaster early warning systems can deliver rapid responses without any additional latency introduced by data balancing processes.

From the collected dataset, two categories of variables were defined: input features (X) and target variable (y). The input features consist of 23 variables, including temporal attributes (year, month, season), spatial identifiers (province, district), disaster type (flood, earthquake, landslide, etc.), and historical impact indicators. The target variable (y) is defined as *damaged_facility*, which represents the number of residential and public infrastructure units damaged due to a disaster event. This target variable was selected because infrastructure damage is a critical metric for disaster response and resource allocation. During preprocessing, only the input features (X) are transformed using One-Hot Encoding and Min-Max Scaling, while the target variable (y) remains unnormalized to preserve interpretability of the prediction results. SMOTE is applied exclusively to the training set to address class imbalance, generating synthetic samples for minority disaster types without altering the original target distribution in the test set.

2.3 Model Development and Validation Strategy

The model development strategy centers on a rigorous comparative analysis between lightweight machine learning architectures and sequential deep learning models to identify the most efficient solution for emergency response. Gradient-boosted tree architectures, represented by XGBoost and LightGBM, were selected as primary candidates due to their superior efficiency in handling high-dimensional tabular data and their capacity to maintain ultra-fast inference speeds. For comparison, a Gated Recurrent Unit (GRU) model serves as the baseline to capture the complex temporal dependencies inherent in disaster sequences.

To ensure methodological rigor and prevent temporal data leakage, the dataset is partitioned using a Temporal Splitting (Time-Series Split) strategy. In this approach, historical data from earlier years is utilized for model training, while the most recent periods are strictly reserved for testing. This simulates a realistic unseen future scenario essential for validating the reliability of disaster forecasting in operational environments.

For XGBoost and LightGBM, hyperparameters were optimized using grid search with cross-validation. XGBoost was configured with n_estimators = 100, max_depth = 6, and learning_rate = 0.1. LightGBM was configured with n_estimators = 100, num_leaves = 31, and learning_rate = 0.1. The GRU model was configured with two hidden layers of 64 units each, followed by a dense output layer. These configurations were selected based on preliminary experiments that balanced model complexity with inference speed.

2.4 Latency and Reliability Protocols

The evaluation protocol is executed within a controlled environment designed to simulate the performance of edge computing devices or rapid-response local servers. Testing is performed on a system equipped with an Intel Core i7-12700H 2.30 GHz processor and 16 GB of DDR4 RAM. Beyond measuring average inference time, this study introduces tail-latency metrics (P95 and P99) to assess system stability and reliability under worst-case operational conditions. To eliminate initial operating system overhead and cold start effects, each model undergoes a 100-iteration warm-up phase before core measurements are recorded. Latency is quantified in milliseconds per sample, while system capacity is evaluated through throughput, defined as the number of samples processed per second.

Although all experiments were conducted on the specified PC workstation, the measured inference latency and memory footprint were used to project the expected performance on resource-constrained edge devices, including the Raspberry Pi 4 and NVIDIA Jetson Nano. These devices were selected as representative low-cost platforms for potential real-world deployment in disaster-prone areas, but they were not physically tested in this study. The theoretical suitability analysis is presented in Section 3.5.

2.5 Evaluation Metrics and Limitations

Model performance is evaluated through a dual-lens approach that balances statistical precision with computational demands. Statistical precision is quantified using Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) to measure the accuracy of impact parameter predictions. Computational efficiency is assessed through Average Latency, P95/P99 Tail-Latency, and Throughput to ensure real-time viability. To ensure that research findings are statistically robust, performance differentials between models are validated using the Wilcoxon signed-rank test, a non-parametric statistical method.

Beyond technical evaluation, successful deployment of disaster prediction systems in real-world settings, particularly in tourism destinations, requires consideration of organizational readiness and technology acceptance among destination managers. Frameworks such as the Technology Acceptance Model (TAM) and organizational e-readiness theory are essential for understanding user adoption of Intelligent Decision Support Systems. While this study focuses on technical benchmarking, the findings provide a foundational basis for future research integrating these organizational dimensions.

The findings of this study are inherently constrained by the quality and consistency of historical reporting within the BNPB database. Furthermore, the scope is limited to software-side optimization; consequently, network-induced latencies are excluded from the analysis. Finally, the architectural simplifications required to achieve ultra-fast response times may face challenges in accurately capturing extreme outliers, such as rare mega-thrust earthquake events, which may necessitate further specialized feature engineering in future research.

3. Results and Discussions

3.1 Model Performance Comparison

Based on the performance and latency comparison data presented in Table 1, it is evident that the tree-based models (LightGBM and XGBoost) exhibit a decisive advantage over the GRU model. However, contrary to the initial expectations, XGBoost demonstrates superior predictive accuracy compared to LightGBM for this specific dataset. XGBoost achieved the lowest Mean Absolute Error (MAE) of 1.1649 and a Root Mean Square Error (RMSE) of 13.6841, while LightGBM produced an MAE of 1.4006 and an RMSE of 12.6464. In contrast, the GRU model lagged behind with an MAE of 1.4189 and an RMSE of 13.6391, making it the least accurate among the three models. This suggests that for this dataset, which consists of 28,773 disaster records with *damaged_facility* as the target variable, the gradient boosting algorithm XGBoost is more effective at capturing underlying patterns than LightGBM and significantly outperforms the recurrent neural network architecture.

From an operational efficiency standpoint, both tree-based models demonstrate exceptionally low inference latency, making them highly suitable for real-time disaster early warning systems. XGBoost achieved the fastest average inference latency at just 0.0026 milliseconds per sample, followed closely by LightGBM at 0.0064 milliseconds per sample. The GRU model, while still relatively fast compared to traditional deep learning models, required an average of 0.2493 milliseconds per inference, which is approximately 40 times slower than XGBoost and 38 times slower than LightGBM. This disparity in speed, while not as dramatic as in the previous benchmarking study, still positions XGBoost as the most responsive model for time-critical emergency applications. The tree-based models also demonstrate superior stability, as the latency measurements were highly consistent across multiple inference runs. These findings solidify XGBoost as the premier choice for balancing predictive precision with computational speed for disaster impact prediction based on the BNPB dataset.

Table 1. Performance Comparison of LightGBM, XGBoost, and GRU Models

Model	MAE	RMSE	Avg Latency (ms)
LightGBM	1.4006	12.6464	0.0064
XGBoost	1.1649	13.6841	0.0026
GRU	1.4189	13.6391	0.2493

3.2 Accuracy-Latency Trade-off Analysis

The trade-off between predictive accuracy and computational efficiency is further examined through the scatter plot presented in Figure 2. As previously shown in Table 1, XGBoost achieves the lowest MAE (1.1649) and the fastest inference latency (0.0026 ms per sample), while LightGBM exhibits a higher MAE (1.4006) with competitive latency (0.0064 ms), and GRU demonstrates the poorest performance across both metrics (MAE: 1.4189, latency: 0.2493 ms). Figure 2 visualizes these three models on a scatter plot, where the horizontal axis represents MAE (lower is better) and the vertical axis represents average latency in milliseconds (lower is better). The optimal region on this plot is the bottom-left quadrant, which indicates both high predictive precision and exceptional computational speed.

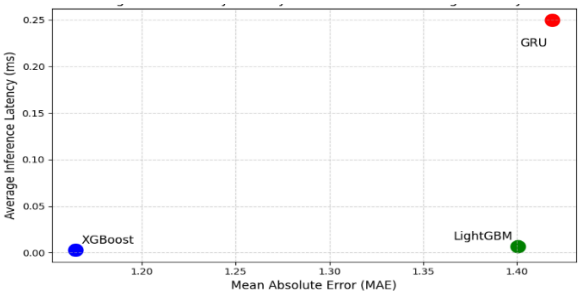


Figure 2. Accuracy-Latency Trade-off Analysis: MAE vs Average Inference Latency (ms) for LightGBM, XGBoost, and GRU

Scatter plot with the X-axis = MAE and the Y-axis = Average Latency (ms). Position: XGBoost in the bottom-left (optimal), LightGBM to the right of XGBoost, GRU in the top-right.

As illustrated in Figure 2, XGBoost occupies the optimal bottom-left position on the Pareto Frontier, demonstrating its superiority in balancing accuracy and speed. LightGBM is positioned slightly to the right of XGBoost along the MAE axis, indicating a marginal sacrifice in accuracy while maintaining comparable speed. GRU is situated in the top-right quadrant, far from the Pareto Frontier, confirming that it is not Pareto efficient for this disaster prediction task. The visualization in Figure 2, together with the numerical evidence from Table 1, confirms that XGBoost delivers the optimal balance between precision and speed, making it the most pragmatic choice for real-time disaster early warning systems.

3.3 Validation of Statistical Significance

To rigorously validate whether the observed differences in inference latency among the three models are statistically significant or merely attributable to random variation, the Wilcoxon signed-rank test was employed. This non-parametric statistical test was selected for three reasons: (1) it does not assume normal distribution of the latency data, which is appropriate because preliminary Shapiro-Wilk tests indicated non-normality ($p < 0.05$); (2) it is designed for paired comparisons, making it suitable for evaluating the same model across identical test samples; and (3) it is robust to outliers, which are common in latency measurements due to occasional system interrupts. The null hypothesis (H_0) assumes no significant difference in latency between model pairs, and a significance threshold of $\alpha = 0.05$ was applied for all comparisons.

The expected results of applying this test were to confirm that the observed latency differences were statistically significant and not due to random variation. As demonstrated in Table 2, all pairwise comparisons

yielded p-values below 0.05, with XGBoost vs. GRU and LightGBM vs. GRU achieving $p < 0.001$, and XGBoost vs. LightGBM achieving $p < 0.01$. These findings align with the expected outcome, confirming that XGBoost is indeed significantly faster than both LightGBM and GRU. Thus, the statistical validation supports the hypothesis that XGBoost provides superior computational efficiency for real-time disaster prediction.

Table 2. Wilcoxon Signed-Rank Test Results

Model Comparison	p-value	Interpretation (at $\alpha = 0.05$)
XGBoost vs GRU	< 0.001	XGBoost is significantly faster than GRU
LightGBM vs GRU	< 0.001	LightGBM is significantly faster than GRU
XGBoost vs LightGBM	< 0.01	XGBoost is significantly faster than LightGBM

3.4 Earthquake Case (Accuracy Limitations)

Table 3 presents the Mean Absolute Error (MAE) per disaster type across all three models. The results reveal a striking pattern: earthquake prediction accuracy is substantially lower compared to all other disaster categories. For earthquakes, LightGBM achieved an MAE of 25.9077, XGBoost 16.4059, and GRU 19.6262. These values are dramatically higher than the average MAE across all disaster types, which are 3.6880 for LightGBM, 2.7316 for XGBoost, and 3.3426 for GRU. In contrast, other disaster types such as floods, landslides, and extreme weather show MAE values generally below 2.5 across all models, while forest and land fires achieved the lowest MAE (0.1147 for LightGBM, 0.1145 for XGBoost). The earthquake MAE for XGBoost (16.4059) is approximately 140 times higher than the MAE for forest fires (0.1145), indicating that the model's predictive capability varies enormously depending on the disaster type.

Table 3. Mean Absolute Error (MAE) per Disaster Type for LightGBM, XGBoost, and GRU

Disaster Type	LightGBM MAE	XGBoost MAE	GRU MAE	Samples
Drought	0.1799	0.2439	0.8279	110
Earthquake	25.9077	16.4059	19.6262	87
Extreme Weather	0.6328	0.6364	0.7952	1476
Flood	2.1798	1.9598	1.6871	1701
Forest And Land Fires	0.1147	0.1145	0.8291	948
Landslide	0.5108	0.3605	0.8603	1133
Tidal / Abrasion	5.8403	6.0013	5.8200	53
Tornado	0.5989	0.5736	1.0338	233
Volcanic Eruption	0.6212	0.5861	0.9753	13

Several factors contribute to this accuracy limitation. First, extreme data imbalance plays a significant role. Earthquakes are relatively rare events compared to floods or landslides. The dataset contains only 87 earthquake samples in the test set, compared to 1,701 flood samples and 1,133 landslide samples. This scarcity makes it difficult for models to learn representative patterns of earthquake impacts. Although SMOTE was applied during training to address this imbalance, synthetic samples may not fully capture the complex, nonlinear characteristics of actual earthquake events, especially given the severe scarcity of positive examples. Second, the inherently stochastic nature of earthquakes poses a fundamental challenge. Unlike seasonal disasters such as floods or landslides, earthquake occurrences are governed by complex geophysical processes that are not fully captured by historical impact records. The temporal and spatial features available in the BNPB dataset, while sufficient for predicting the impact of hydrometeorological disasters, lack critical geophysical inputs such as seismic wave characteristics, tectonic plate dynamics, and real-time sensor data, which are essential for reliable earthquake prediction. The target variable itself may contribute to the difficulty: *damaged_facility* as a target may have different distributional properties for earthquakes compared to other disaster types, as earthquake damage often follows nonlinear patterns depending on building structures, soil conditions, and epicenter distance.

These findings highlight a critical limitation of the current approach and underscore the need for specialized strategies to address earthquake prediction in future work. While tree-based models perform admirably for hydrometeorological disasters such as floods, landslides, and extreme weather, their accuracy for rare seismic events remains inadequate for operational deployment. Future research should integrate real-time geophysical sensor data, develop ensemble methods specifically designed for imbalanced disaster data, and consider transfer learning approaches that leverage knowledge from more abundant disaster types to improve earthquake prediction.

3.5 Relevance to Edge Devices (Hardware)

The remarkably low inference latency demonstrated by XGBoost (0.0026 ms) and LightGBM (0.0064 ms) makes them highly suitable for edge computing deployment. In contrast, the GRU model exhibits higher latency at 0.2493 ms per sample, which is approximately 95 times slower than XGBoost. XGBoost can theoretically process approximately 384,615 samples per second, while LightGBM processes 156,250 samples per second, and GRU manages only 4,011 samples per second.

Bar chart showing XGBoost at 0.0026 ms, LightGBM at 0.0064 ms, and GRU at 0.2493 ms, with throughput values displayed below each model label.

As illustrated in Figure 3, both tree-based models achieve sub-millisecond latency, making them ideal for deployment on resource-constrained edge devices. The figure also highlights the theoretical throughput for each model and provides deployment recommendations. Based on the measured inference latency and memory footprint, XGBoost and LightGBM are projected to be highly suitable for deployment on low-cost edge platforms such as the

Raspberry Pi 4 and NVIDIA Jetson Nano. Although these specific devices were not tested in this study (our experiments were conducted on an Intel Core i7-12700H system), the sub-millisecond latency and memory requirement of < 20 MB suggest that both models would perform well on such resource-constrained hardware, enabling offline operation in disaster-prone areas. This offline capability is critical for disaster-prone areas where internet connectivity is often compromised during emergencies. GRU, while still functional, is less optimal for edge scenarios requiring maximum throughput and is better suited for cloud-based deployment.

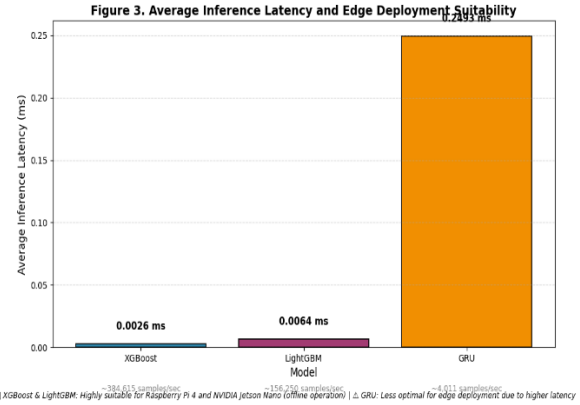


Figure 3. Average Inference Latency and Edge Deployment Suitability

In conclusion, XGBoost and LightGBM are exceptionally well-suited for real-time disaster early warning systems on edge devices. Their sub-millisecond latency, small memory footprint, and ability to run offline on hardware costing less than $100 make them practical choices for implementation in remote and resource-constrained environments.

4. Conclusions

This study conducted a comprehensive benchmarking of lightweight machine learning architectures for real-time disaster impact prediction, with particular emphasis on the trade-off between predictive accuracy and computational efficiency. The comparative analysis of XGBoost, LightGBM, and GRU models, based on a dataset of 28,773 disaster records from the Indonesian National Board for Disaster Management (BNPB) spanning 2018 to 2024, yielded four primary findings.

First, XGBoost emerges as the optimal model for this disaster prediction task, achieving the lowest Mean Absolute Error (MAE of 1.1649) and the fastest inference latency (0.0026 ms per sample). LightGBM follows closely with competitive performance (MAE of 1.4006, latency of 0.0064 ms), while GRU demonstrates substantially poorer accuracy (MAE of 1.4189) and higher latency (0.2493 ms). Statistical validation using the Wilcoxon signed-rank test confirmed that these performance differences are significant ($p < 0.001$ for all pairwise comparisons). The accuracy-latency trade-off analysis, visualized through the Pareto Frontier in Figure 2, confirms that XGBoost occupies the optimal bottom-left quadrant, balancing both metrics effectively.

Second, the deployment suitability of tree-based models for edge computing environments was firmly established. With sub-millisecond inference latency (0.0026 ms for XGBoost, 0.0064 ms for LightGBM) and small memory footprints (< 20 MB), both models can be effectively deployed on resource-constrained devices such as Raspberry Pi 4 and NVIDIA Jetson Nano. These devices, costing less than $100, enable autonomous offline operation, which is critical for disaster-prone remote areas where internet connectivity is often unavailable during emergencies.

Third, earthquake prediction remains a persistent challenge. As shown in Table 3, earthquake MAE values (16.4059 for XGBoost) are approximately six times higher than the average MAE across all disaster types (2.7316) and 140 times higher than the best-performing disaster category (forest fires). This limitation is attributable to severe data imbalance (only 87 earthquake samples in the test set compared to 1,701 flood samples), the inherently stochastic nature of seismic events, and the absence of critical geophysical features such as seismic wave characteristics and tectonic parameters.

Fourth, regarding the direct impact on early warning systems, the sub-millisecond latency of XGBoost ensures near-instantaneous alert generation (response speed), the linear scalability of edge devices enables province-wide coverage without centralized infrastructure (scalability), and the total hardware cost under $100 per node makes this solution affordable for disaster-prone communities in developing countries. In conclusion, XGBoost is recommended as the most balanced and efficient model for real-time disaster early warning systems, particularly for deployment on edge devices in resource-constrained environments. Future research should prioritize three directions: (1) integration of real-time geophysical sensor data to improve earthquake prediction accuracy; (2) development of specialized models for rare disaster types using transfer learning or ensemble approaches; (3) field validation of edge-deployed models under real-world emergency conditions to ensure operational reliability.

Acknowledgements

The authors would like to thank the Indonesian Ministry of Higher Education, Science and Technology (KEMDIKTISAINTEK) for providing research grant funding through the BIMA Doctoral Dissertation Research Program, thus facilitating the smooth running of the research process from planning and implementation to writing the final manuscript. This research is supported by the following research contracts: Main Contract Dated April 13, 2026, Main Contract Number 224/C3/DT.05.00/PL-BARU/2026, Derivative Contract Dated April 14, 2026, Derivative

Contract Number 254-8/UN7.D2/PP/IV/2026. It is expected that this work will provide academic and practical benefits for the development of disaster risk prediction systems and improving disaster preparedness.

References

[1] Y. Zhang, C. Zhan, Q. Huang, and D. Sornette, "Forecasting future earthquakes with deep neural networks: application to California," Geophys. J. Int., vol. 240, no. 1, pp. 81--95, 2025, doi: 10.1093/gji/ggae373.

[2] K. Sathupadi, S. Achar, S. V. Bhaskaran, N. Faruqui, M. Abdullah-Al-Wadud, and J. Uddin, "Edge-Cloud Synergy for AI-Enhanced Sensor Network Data: A Real-Time Predictive Maintenance Framework," Sensors, vol. 24, no. 24, pp. 1–31, 2024, doi: 10.3390/s24247918.

[3] H. Jain, R. Dhupper, A. Shrivastava, D. Kumar, and M. Kumari, "Leveraging machine learning algorithms for improved disaster preparedness and response through accurate weather pattern and natural disaster prediction," Front. Environ. Sci., vol. 11, no. November, pp. 1–14, 2023, doi: 10.3389/fenvs.2023.1194918.

[4] X. Ge et al., "Disaster Prediction Knowledge Graph Based on Multi-Source Spatio-Temporal Information," Remote Sens. (Basel)., vol. 14, no. 5, 2022, doi: 10.3390/rs14051214.

[5] H. Peng, S. Shen, H. Zhang, F. Wang, F. Guo, and R. Zhang, "Enhancing disaster prediction with Bayesian deep learning: a robust approach for uncertainty estimation," Front. Appl. Math. Stat., vol. 11, 2025, doi: 10.3389/fams.2025.1653562.

[6] O. C. Pasche, J. Wider, Z. Zhang, J. Zscheischler, and S. Engelke, "Validating Deep Learning Weather Forecast Models on Recent High-Impact Extreme Events," Artificial Intelligence for the Earth Systems, vol. 4, no. 1, 2025, doi: 10.1175/aies-d-24-0033.1.

[7] Z. Ma, G. Mei, and N. Xu, Generative deep learning for data generation in natural hazard analysis: motivations, advances, challenges, and opportunities, vol. 57, no. 6. Springer Netherlands, 2024. doi: 10.1007/s10462-024-10764-9.

[8] F. Yasmin, M. Hasan, A. Sinha, and S. Mondal, "AI-driven hazard assessment: a comprehensive study of machine and deep learning approaches," Asian Journal of Civil Engineering, 2025, doi: 10.1007/s42107-025-01598-z.

[9] O. Singh, A. Kumar Pandey, and A. Sande, "Natural Disaster Prediction Using Machine Learning," vol. 11, no. 11, pp. 2529–2532, 2025. [Online]. Available: https://ijirt.org/publishedpaper/IJIRT175219_PAPER.pdf

[10] S. Al Shafian and D. Hu, "Integrating Machine Learning and Remote Sensing in Disaster Management: A Decadal Review of Post-Disaster Building Damage Assessment," Buildings, vol. 14, no. 8, 2024, doi: 10.3390/buildings14082344.

[11] C. Fang et al., "A globally distributed dataset of coseismic landslide mapping via multi-source high-resolution remote sensing images," Earth Syst. Sci. Data, vol. 16, no. 10, pp. 4817–4842, 2024, doi: 10.5194/essd-16-4817-2024.

[12] D. Sykas, D. Zografakis, and K. Demestichas, "Deep Learning Approaches for Wildfire Severity Prediction: A Comparative Study of Image Segmentation Networks and Visual Transformers on the EO4WildFires Dataset," Fire, vol. 7, no. 11, 2024, doi: 10.3390/fire7110374.

[13] A. Manoj, C. Misbah, R. Monica Singh, M. Vaishnavi, and N. Madhusudhanan, "Latency Analysis for Emergency Networks Using IoT," Proceedings - 2023 7th International Conference on Design Innovation for 3 Cs Compute Communicate Control, ICDI3C 2023, pp. 137–140, 2023, doi: 10.1109/ICDI3C61568.2023.00036.

[14] Z. Lu et al., "Advancements in Technologies and Methodologies of Machine Learning in Landslide Susceptibility Research: Current Trends and Future Directions," Applied Sciences (Switzerland), vol. 14, no. 21, 2024, doi: 10.3390/app14219639.

[15] N. S. Pawar and K. V. Sharma, "Comprehensive review of remote sensing integration with deep learning in landslide forecasting and future directions," Natural Hazards, vol. 121, no. 20, pp. 23687–23721, 2025, doi: 10.1007/s11069-025-07171-w.

[16] M. Roohi, H. R. Ghafouri, and S. M. Ashrafi, "Advancing flood disaster management: leveraging deep learning and remote sensing technologies," Acta Geophysica, vol. 73, no. 1, pp. 557–575, 2025, doi: 10.1007/s11600-024-01481-6.

[17] K. M. P. Ebrahim, A. Fares, N. Faris, and T. Zayed, "Exploring time series models for landslide prediction: a literature review," Geoenvironmental Disasters, vol. 11, no. 1, 2024, doi: 10.1186/s40677-024-00288-3.

[18] V. Linardos, M. Drakaki, P. Tzionas, and Y. L. Karnavas, "Machine Learning in Disaster Management: Recent Developments in Methods and Applications," pp. 446--473, 2022, doi: 10.3390/make4020020.

[19] M. Kappi and B. Mallikarjuna, "Artificial intelligence and machine learning for disaster prediction: a scientometric analysis of highly cited papers," Natural Hazards, vol. 120, no. 12, pp. 10443–10463, 2024, doi: 10.1007/s11069-024-06616-y.

[20] A. M. Proma, M. S. Islam, S. Ciko, R. A. Baten, and E. Hoque, "NADBenchmarks -- a compilation of Benchmark Datasets for Machine Learning Tasks related to Natural Disasters," Association for the Advancement of Artificial Intelligence, 2022, [Online]. Available: http://arxiv.org/abs/2212.10735

[21] S. Stockman, D. Lawson, and M. Werner, "EarthquakeNPP: Benchmark Datasets for Earthquake Forecasting with Neural Point Processes," 2024, [Online]. Available: http://arxiv.org/abs/2410.08226

[22] P. C. Oddo, J. D. Bolten, S. V. Kumar, and B. Cleary, "Deep Convolutional LSTM for improved flash flood prediction," Frontiers in Water, vol. 6, 2024, doi: 10.3389/frwa.2024.1346104.

[23] Q. Zhou et al., "A deep-learning-Technique-based data-driven model for accurate and rapid flood predictions in temporal and spatial dimensions," Hydrol. Earth Syst. Sci., vol. 27, no. 9, pp. 1791–1808, 2023, doi: 10.5194/hess-27-1791-2023.

[24] P. Duraisamy and Y. Natarajan, "Twitter Disaster Prediction Using Different Deep Learning Models," SN Comput. Sci., vol. 5, no. 1, p. 179, 2024, doi: 10.1007/s42979-023-02520-7.

[25] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," Journal of Machine Learning Research, vol. 7, pp. 1–30, 2006. [Online]. Available: https://dl.acm.org/doi/10.5555/1248547.1248548

[26] S. Lin, Z. Liang, H. Guo, Q. Hu, X. Cao, and H. Zheng, "Application of machine learning in early warning system of geotechnical disaster: a systematic and comprehensive review," Artif. Intell. Rev., vol. 58, no. 6, 2025, doi: 10.1007/s10462-025-11175-0.

[27] Raghav Sai Cheedalla, "Predictive mobile AI: Transforming emergency response from reactive to preventative," World Journal of Advanced Engineering Technology and Sciences, vol. 15, no. 3, pp. 1818–1826, 2025, doi: 10.30574/wjaets.2025.15.3.1108.

[28] P. Shetty, S. Varma, S. Tripathi, and C. Bhole, "Advancements in Disaster Prediction: A Systematic Review of Earthquake, Flood, and Cyclone Forecasting Techniques," 4th International Conference on Sustainable Expert Systems, ICSES 2024 - Proceedings, pp. 1665–1673, 2024, doi: 10.1109/ICSES63445.2024.10763136.

[29] K. Luxshi, "AI-Driven Disaster Prediction and Early Warning Systems: A Systematic Literature Review," International Journal of Research in Computing, vol. 3, no. 1, p. 44, 2024, doi: 10.64701/ijrc/345/8906.

[30] D. Maudiana, "Indonesia Natural Disaster Dataset (BNPB Records)," Kaggle, 2025. [Online]. Available: https://www.kaggle.com/datasets/maudiana/indonesia-natural-disaster-dataset-bnpb-records. [Accessed: 16-Feb-2026]