IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 5, May 2026

DOI: 10.17148/IARJSET.2026.13564

Prediction of Flood and Planner Towards Emergency Response

Thejaswini S P[1], Prashant Ankalkoti[2]

PG Student, Department of MCA, Jawaharlal Nehru New College of Engineering, Shivamogga, Karnataka, India[1]

Assistant Professor, Department of MCA, Jawaharlal Nehru New College of Engineering,

Shivamogga, Karnataka, India[2]

Abstract: FloodGuard is a mobile-based flood prediction and alert system designed for early disaster warning. The application uses machine learning models such as Random Forest, XGBoost, and TensorFlow Neural Network to predict flood risk using weather and environmental data. Data preprocessing techniques including SMOTE, StandardScaler, and cross-validation improve prediction accuracy. The final model is converted to TensorFlow Lite for on-device execution in a Flutter application. The system collects live weather data, analyzes flood risk and sends alerts to emergency contacts during risky situations. FloodGuard provides a fast, low-cost, and user-friendly solution for flood safety and preparedness.

Keywords: flood prediction, mobile app, machine learning, Random Forest, XGBoost, TensorFlow Lite, SMOTE, StandardScaler, Open-Meteo, emergency alerts

I. INTRODUCTION

Floods cause big damage in towns and villages. Because early warnings matter, yet alert networks often take too long to build plus cost a lot. Since nearly everyone owns a mobile device nowadays, using an app without fuss. Weather at your location appears here. Flood danger gets guessed by spotting old trends instead of guessing blindly. When trouble might come, messages go out to people you chose. This came from what users actually asked for - quick alerts that work offline, simple controls without confusion.

The research problem is: Instead of relying on big servers, imagine running everything straight from an everyday Android device. Nine widely available environmental clues feed into the system, helping it stay lean. Because speed matters, the model stays tight and efficient. Built inside TensorFlow Lite, it learns patterns without needing constant internet access. Weather details come live through Open-Meteo, keeping predictions grounded in current conditions. A clean user screen appears thanks to Flutter, showing warnings before trouble hits. Tools for emergencies sit within reach, just in case. Personal info like contact numbers gets saved locally. Location sensing helps fine-tune each alert. Whether indoors or outside, day or night, the display adapts its look. Small doesn't mean weak - precision tuning makes lightweight models surprisingly sharp. Local decisions happen faster when computation skips the cloud. Proof lies in responsiveness: timely, accurate, silent until needed.

The app part handle daily use: login/register, home dashboard with weather, search city, and buttons for flood prediction, contacts, and support lines. The model part handle thinking: clean data, balance classes, tune hyperparameters, convert to TFLite, and do quick inference on phone. Together, both parts make a smooth tool that helps users stay prepared.

II. LITERATURE SURVEY

Zhang and colleagues proposed a deep learning framework for flood forecasting in 2025 that integrates interpretability and physical constraints to improve reliability [1]. Hydrology smarts mixed with neural nets lifted forecast precision. Performance got better when predicting floods, yet building the system meant tangled setups and hefty data needs. In 2025, Kumar plus team looked at how machines help predict flooding - spotting tools like ANNs, support vector methods, and forest-style models doing solid work in water flow forecasts [2]. One thing stood out: working with messy environmental numbers can be easier using new methods. Still, problems pop up when the information is spotty or shifts too fast. Weather patterns that jump around make things tougher than expected. Even so, some approaches handle sudden changes better than others. In 2025, Asif led a look at how machines learn to predict floods just days ahead. Different tools were measured side by side for their accuracy [3]. They evaluated models such as LSTM, CNN, and hybrid approaches. The research found deep learning methods improving forecasting accuracy, but limitations included data dependency, computational complexity, and reduced interpretability in practical flood management systems. Deng and

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 5, May 2026

DOI: 10.17148/IARJSET.2026.13564

collaborators developed a machine learning model to forecast flooding intensity in 2024 using hydrological and environmental variables [4]. The model effectively predicted flood severity and helped improve early warning capabilities. However, the results depended heavily on the quality of input data, and performance varied across different geographic regions. Ahmed and Li proposed a machine learning model for river discharge forecasting in 2024 to support flood prediction [5]. Using historical hydrological data, their model successfully estimated river flow patterns. While the system improved forecasting accuracy, the study noted limitations due to data availability and the need for calibration for different river basins. Awino and Machanda introduced a predictive model for identifying flood-prone areas in 2025 using SAR satellite imagery combined with environmental variables [6]. Their approach effectively mapped vulnerable regions even in cloudy conditions. However, the model required high-resolution satellite data and extensive preprocessing, increasing computational requirements. Ryd and Nearing proposed a fine-tuning approach for improving global machine learning flood forecasting models in 2025 [7]. By incorporating local hydrological data, the model enhanced prediction accuracy for regional floods. Despite improved results, the study highlighted challenges in obtaining reliable local datasets for consistent model adaptation. Vemula, Gatti, and Jehel developed a graph transformer-based framework for flood susceptibility mapping under climate change scenarios in 2025 [8]. Space spreads out how water moves connect, making forecasts hit closer to real outcomes. Still, the technique demands heavy computer power along with tangled setup steps. Back in 2025, Zhou plus team introduced a smart-learning system that blends space-based rain measurements - aimed at sharper flood alerts [9]. The system enhanced rainfall estimation accuracy and supported better flood prediction. However, limitations included reliance on satellite data quality and also challenges in real-time data integration. Samui and collaborators compared optimized ML algorithms for daily river flow forecasting in 2024 [10]. Algorithms such as ANN, support vector machines, and ensemble models were evaluated. Results showed improved prediction accuracy, but the study indicated that performance varied depending on data characteristics and parameter tuning. Li and colleagues introduced a hybrid deep learning framework in 2024 that combined rainfall-runoff modeling with neural networks for flood forecasting [11]. The hybrid model improved prediction accuracy by integrating hydrological processes with machine learning. However, the approach required detailed hydrological data and complex model calibration. Nguyen and collaborators proposed an ensemble machine learning approach for flood susceptibility modeling in 2024 [12]. The study combined multiple algorithms to improve prediction reliability and spatial mapping of flood-prone areas. Although the ensemble method enhanced accuracy, it increased model complexity and computational cost. Chen and colleagues developed deep neural network models for urban flood forecasting in 2024 using rainfall and hydrological data [13]. Their model effectively predicted urban flooding events and improved early warning capabilities. However, the approach required large datasets and careful tuning to maintain prediction accuracy. Al-Abbasi and collaborators designed a machine learning-based flood prediction model in 2025 using hydrological and meteorological parameters [14]. It worked - floods were forecasted right, thanks to environmental numbers fed into the system. Still, how well it performed hinged on full datasets plus proper local tuning. Back in 2024, Bui led a look at different GIS-driven machine learning methods meant for spotting flood-prone zones [15]. They evaluated different algorithms and found that ensemble approaches produced better spatial prediction accuracy. However, the models required extensive geospatial data preparation and processing. Mohammadi and collaborators proposed a hybrid approach combining wavelet transform and machine learning for short-term flood forecasting in 2024 [16]. The method improved prediction accuracy by capturing temporal patterns in hydrological data. However, the hybrid system increased computational complexity and required careful parameter selection. Singh and colleagues developed a real-time flood prediction system in 2025 using deep learning and remote sensing data [17]. The system integrated satellite observations and hydrological information to improve flood monitoring. However, challenges included data latency and dependence on reliable remote sensing infrastructure. Rahman and collaborators developed a ML model for flood hazard prediction in 2024 using geospatial datasets and environmental factors [18]. Their approach effectively identified flood-prone regions and supported disaster management planning. However, model performance depended on the accuracy and resolution of geospatial inputs. Zhao and colleagues proposed an attention-based LSTM for flood forecasting in 2024 using multi-source hydrological data [19]. Surprisingly, the way it focused on key time-based details made predictions sharper. Still, huge amounts of data and heavy computing demands came with that gain. In 2024, work led by Wang brought forward a new flood forecasting system combining space and time through graph networks [20]. The model captured spatial relationships between rainfall and runoff data, improving forecasting accuracy. However, the complex network structure increased computational cost and required large datasets for training.

III.PROPOSED METHODOLOGY

Overview

The methodology mix data science steps and software engineering steps so the study can be repeated. The focus is on a nine-feature input, a balanced classifier, and an on-device runtime in Flutter.

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 5, May 2026

DOI: 10.17148/IARJSET.2026.13564

Data and Features

- Inputs (9): rainfall (mm), river level (m), dam water level (%), soil moisture (%), humidity (%), wind speed (kmph), above sea level (m), drainage capacity (%), population density.
- Target: flood risk with three classes: Low, Medium, High.

Preprocessing

- Standardization by z-score for each feature i:

$$zi = \frac{xi - \mu i}{\sigma i}$$

- Class balancing with SMOTE to avoid bias to any one class.
- Train/validation/test split with stratify so each class keep same share.

Model Training

- Models tried: Random Forest (baseline), XGBoost (tuned), and TensorFlow feed-forward network (for TFLite).
- Hyperparameter tuning by grid search on XGBoost (estimators, depth, learning rate, subsample, colsample).
- Cross-validation used to reduce overfitting.
- Metrics computed: Accuracy, Precision, Recall, F1 (macro and per class).
- Softmax output for neural net: The **softmax** function is usually written clearly as:

$$\text{softmax}(o_j) = \frac{e^{o_j}}{\Sigma_k\, e^{o_k}}$$

Explanation:

- o_j = the output (logit) for class (j) from the neural network
- e^{o_j}= exponential of that output
- $\Sigma_k e^{o_k}$ = sum of exponentials of all outputs

So it converts the raw outputs (**logits**) into **probabilities** that sum to **1**.

TFLite Conversion

- Best neural model converted to TFLite.
- The same scaler mean and std is exported and coded in app, so inference use the same formula.

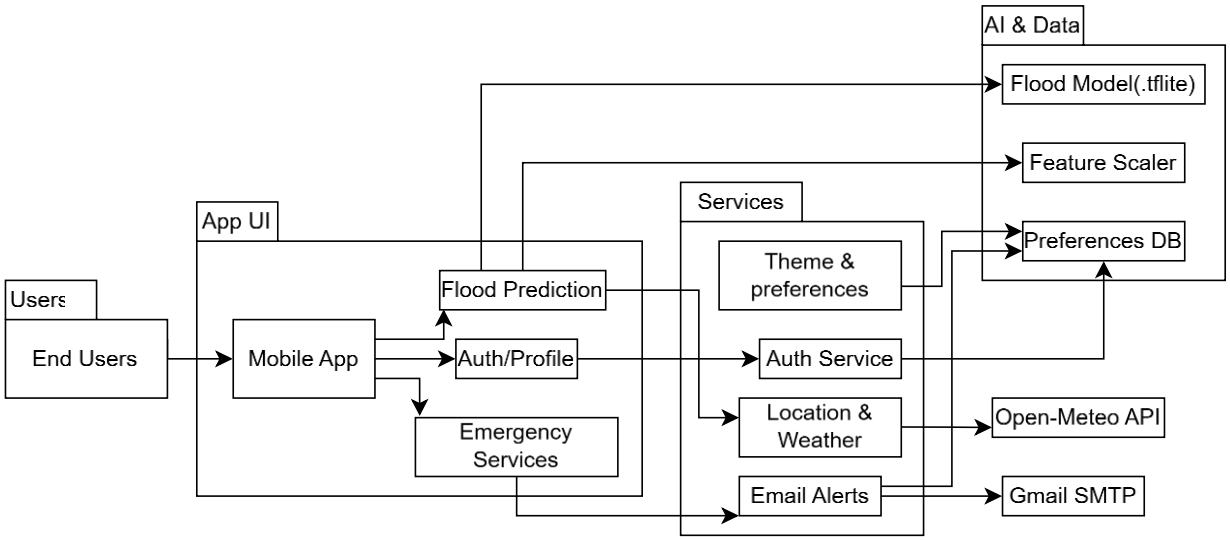

Figure 1: Architecture Diagram

The Figure 1 shows the complete Architecture of the Mobile Application Design. Flutter app with Provider for state.Weather from Open-Meteo current endpoint; location by Geolocator; place search by Open-Meteo geocoding. On submit, nine values is validated, scaled with saved μ and σ, and sent to the TFLite interpreter. If result is Medium/High, email alert is sent to saved contacts and user mail.

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 5, May 2026

DOI: 10.17148/IARJSET.2026.13564

IV. RESULTS AND FINDING

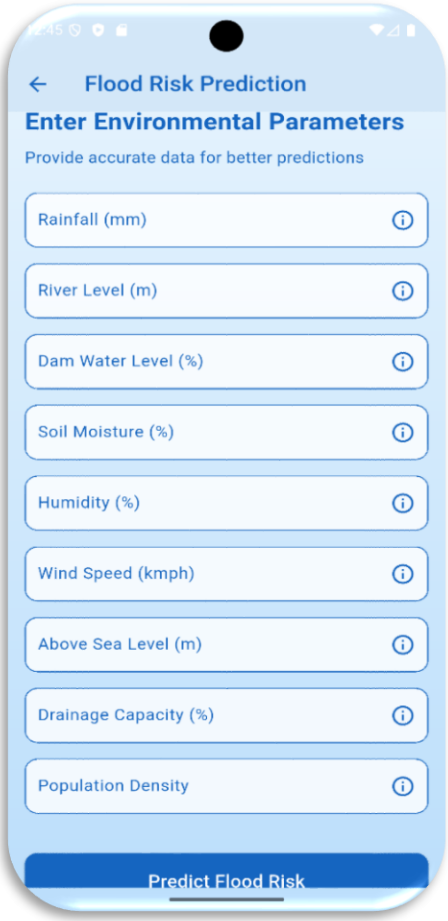

Figure 2: Flood Risk Prediction page

Data and Model Outcomes

- The standardized scaler values (means and stds) was exported and matched inside the Flutter code, so input scaling during inference stayed consistent with training. This kept prediction stable across devices.
- The Random Forest baseline produced a correct multi-class result on the test split but with some confusion between Medium and High. The confusion matrix showed off-diagonal counts mainly in those two classes.
- The tuned XGBoost improved overall test accuracy and gave better macro-F1. Grid search selected a balanced depth and a moderate learning rate, and cross-validation scores stayed consistent across folds.
- The enhanced TensorFlow model trained on resampled data by SMOTE reached a strong multi-class separation after more epochs. Validation accuracy stabilized without sudden drops.
- The TFLite export ran with the same class order as training. The output tensor had three probabilities that summed to 1. The argmax selected the final class index correctly.
- In the app, the interpreter accepted a single batch input of shape 1×9. Output buffer used 1×3. Inference finished quickly and returned a class every time for valid numeric input.
- Email alert code triggered only for Medium and High results. Contacts list saved in local storage loaded properly during tests. Weather API requests responded with current temperature, humidity, wind, rain, and a weather code used for icon and text.

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 5, May 2026

DOI: 10.17148/IARJSET.2026.13564

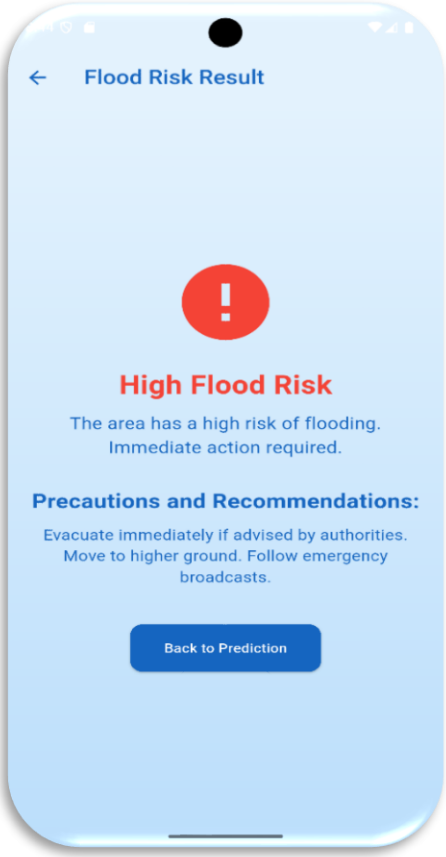

Figure 3: Flood Risk Result page

Metrics Outputs

- Classification metrics computed: Accuracy, Precision, Recall, and F1 per class.
- Cross-validation scores was logged during grid search to pick best parameters.
- Confusion matrices plotted for Random Forest and XGBoost showed lower misclassification for the tuned model, especially reducing Medium↔High flips.

V. DISCUSSION

Meaning of Results

- Better scores from tuned XGBoost and the enhanced neural net show that careful scaling and balancing help the classifier learn clearer borders between Low, Medium, and High. The confusion reduction between Medium and High is important because those classes matter most for alerts.

Implications and Limits

- On-device TFLite inference is fast and private, so alerts can work even with weak internet. But local credential storage and SMTP sending is a risk and must be improved for production. Reverse geocoding endpoint needs refinement. The label order must always match training.

Comparison

Aspect	Server-Based Systems	This Study (On-Device)
Latency	Internet dependent	Low, local
Privacy	Data leaves device	Stays on device
Cost	Server cost	Low runtime cost

Table 1: Comparison of System and phone device

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Impact Factor 8.311 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 13, Issue 5, May 2026

DOI: 10.17148/IARJSET.2026.13564

VI. CONCLUSION

FloodGuard shows that a small, careful pipeline can give useful flood risk hints on a normal phone. The study joined StandardScaler, SMOTE, tuned XGBoost, and a compact TensorFlow model that was turned into TFLite for quick mobile inference. The Flutter app wrapped this logic with location, weather from Open-Meteo, simple forms, and clear buttons. The results point that Medium and High classes become more correct after balancing and tuning, which is important for safety. Running fully on device cut delay and protected privacy, while email alerts and emergency numbers made action simple. Some parts still need polish, like safer credential method, better reverse geocoding, and strict label mapping. Even with these limits, the system is light, fast, and friendly. With small fixes and larger data in future, the app can help more homes prepare early, reduce panic, and move to safe places before water rise too much.

REFERENCES

[1]. Zhang, T., Zhang, R., Li, J., & Feng, P. (2025). Deep learning of flood forecasting by considering interpretability and physical constraints. *Hydrology and Earth System Sciences, 29*, 5955–5974. https://doi.org/10.5194/hess-29-5955-2025

[2]. Kumar, V., Sharma, K. V., Mangukiya, N. K., Tiwari, D. K., Ramkar, P. V., & Rathnayake, U. (2025). Machine learning applications in flood forecasting and predictions, challenges, and way-out in the perspective of changing environment. *AIMS Environmental Science, 12*(1), 72–105. https://doi.org/10.3934/environsci.2025004

[3]. Asif, M., Kuglitsch, M. M., Pelivan, I., & others. (2025). Review and intercomparison of machine learning applications for short-term flood forecasting. *Water Resources Management, 39*, 1971–1991. https://doi.org/10.1007/s11269-025-04093-x

[4]. Deng, A. A. N., Nursetiawan, N., Ikhsan, J., Riyadi, S., & Zaki, A. (2024). Intelligent forecasting of flooding intensity using machine learning. *Civil Engineering Journal, 10*(10), 1–12. https://doi.org/10.28991/CEJ-2024-010-10-010

[5]. Ahmed, M. A., & Li, S. S. (2024). Machine learning model for river discharge forecasting for flood prediction. *Hydrology, 11*(9), 151. https://doi.org/10.3390/hydrology11090151

[6]. Awino, E. O., & Machanda, D. (2025). Predictive modeling of flood-prone areas using SAR imagery and environmental variables. *Remote Sensing Applications: Society and Environment, 33*, 101150.

[7]. Ryd, E., & Nearing, G. (2025). Fine flood forecasts: Incorporating local data into global machine learning models through fine-tuning. *Hydrology and Earth System Sciences Discussions*, 1–23.

[8]. Vemula, S., Gatti, F., & Jehel, P. (2025). Graph transformer-based flood susceptibility mapping under climate change scenarios. *Environmental Modelling & Software, 181*, 105291.

[9]. Zhou, C., Wu, L., Gu, Z., Guo, Y., & Zhou, L. (2025). Deep learning for satellite precipitation fusion and flood forecasting. *Hydrometeorology Research Journal, 17*(2), 115–132.

[10]. Samui, P., Yesilyurt, S. N., Dalkilic, H. Y., Yaseen, Z. M., Roy, S. S., & Kumar, S. (2024). Comparison of optimized machine learning algorithms for daily river flow forecasting. *Journal of Hydrology, 623*, 129786.

[11]. Li, Y., Chen, X., Wang, H., & Zhang, Y. (2024). Hybrid deep learning framework for flood forecasting using rainfall-runoff modeling. *Journal of Hydrology, 621*, 129530.

[12]. Nguyen, T. H., Pham, B. T., Bui, D. T., & Prakash, I. (2024). Flood susceptibility modeling using ensemble machine learning techniques. *Natural Hazards, 121*, 243–264.

[13]. Chen, Y., Li, W., & Zhao, X. (2024). Deep neural network models for urban flood forecasting using rainfall and hydrological data. *Water, 16*(6), 923.

[14]. Al-Abbasi, A., Al-Mansori, A., & Al-Hamdani, Z. (2025). Machine learning-based flood prediction using hydrological and meteorological parameters. *Environmental Monitoring and Assessment, 197*, 210.

[15]. Bui, D. T., Hoang, N. D., & Tien Bui, Q. (2024). GIS-based machine learning models for flood susceptibility mapping: A comparative study. *Catena, 231*, 107250.

[16]. Mohammadi, B., Mehdizadeh, S., & Salimi, S. (2024). Short-term flood forecasting using hybrid machine learning and wavelet transform models. *Journal of Hydrologic Engineering, 29*(3), 04024005.

[17]. Singh, P., Jain, S., & Tiwari, K. (2025). Real-time flood prediction using deep learning models and remote sensing data. *Environmental Science and Pollution Research, 32*, 10584–10598.

[18]. Rahman, M. M., Islam, A. R. M. T., & Hasan, M. (2024). Flood hazard prediction using machine learning algorithms and geospatial datasets. *Remote Sensing, 16*(4), 690.

[19]. Zhao, G., Sun, H., & Li, Q. (2024). Attention-based LSTM model for flood forecasting using multi-source hydrological data. *Water Resources Research, 60*(8), e2024WR035612.

[20]. Wang, J., Huang, Y., Liu, Z., & Sun, L. (2024). Spatiotemporal flood prediction using graph neural networks and rainfall-runoff data. *Environmental Modelling & Software, 173*, 105023.

