Published by :
https://www.ijert.org/
An International Peer-Reviewed Journal

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

AegisCore: AI-Based Flood and Landslide Prediction and Disaster Response Management System Using LSTM and Real-Time Data Integration

Mrs. Deeksha Satish
Asst Proff, Department of CSE
Acharya Institute of Technology
Bengaluru, Karnataka, India

Satvik Nadiger
Department of CSE
Acharya Institute of Technology
Bengaluru, Karnataka, India

Shankar Naik D
Department of CSE
Acharya Institute of Technology
Bengaluru, Karnataka, India

Santosh Neelappa Hulagabal
Department of CSE
Acharya Institute of Technology
Bengaluru, Karnataka, India

Shreyas Mogaveera
Department of CSE
Acharya Institute of Technology
Bengaluru, Karnataka, India

Abstract - AegisCore is a web-based disaster management platform designed to support flood prediction across Kodagu, Uttara Kannada, Belagavi, and Bagalkot, with landslide prediction focused on Kodagu and Uttara Kannada. It integrates geospatial mapping, environmental data processing, and a Flask-based LSTM prediction module with a React.js frontend, Node.js backend, and MongoDB Atlas database. Currently, a controlled pipeline supports mock predictions while the final LSTM models remain under development. The platform also provides real-time alerts, citizen reporting, SOS coordination, shelter management, and evacuation guidance.

Keywords—Landslide Prediction, Flood Prediction, Long ShortTerm Memory, Real Time Data, Disaster Management.

I. INTRODUCTION

Floods and landslides are a recurrent threat to the people and properties of Karnataka state during the monsoon season [1], [2]. The current disaster management practices only take into account broad-scale weather reports and lack a cohesive approach to localized hazard prediction and emergency response coordination at the district level [3], [5]. AegisCore is a proposed disaster management system designed to predict flash floods, landslides, and to coordinate emergency response in the districts of Kodagu, Uttara Kannada, Belagavi, and Bagalkot, Karnataka. While flood prediction features have been implemented for all four districts, the landside-module is limited to the first two, as they are historically more prone to landslides and have sufficient topographical data for accurate modelling. The application also has reporting, emergency response coordination, rescue management, and citizen reporting features [3], [5]. The application has been developed as a full-stack web application, and a working prototype with a backend database, real-time communication system, and general location tracking has been deployed. The system uses a Long Short-Term Memory neural network to predict flood events based on environmental data from the past decade (2015-2024) [1]. However, the data and the final LSTMs model require further refinement and will be implemented in a later stage. This paper discusses the system's architecture, the features that have been implemented, testing procedures and results, and the required future work in developing the application.

II. RELATED WORK

Recent studies have been exploring the ways in which machine learning and deep learning techniques can help predicting floods and landslides. Rasalkar and Surve [1] make a comparison between hydrological, statistical, machine learning, and deep learning approaches in terms of flood prediction, emphasizing the effectiveness of deep learning methods at capturing intricacies of environmental time-series data. Sumalatha et al. [2] utilized neural-network-based research and satellite technology to predict landslide susceptibilities in South India, attempting to identify high-risk areas. Banupriya et al. [3] explored a deep learning-based system for simultaneous prediction of both of the disasters mentioned and real-time alerts. Liu et al. [4] examined the possibility of using remote sensing and crowdsourcing for flood monitoring. Meanwhile, Kavya Sree et al. [5] suggested a multi-hazard prediction and alert framework which also incorporates emergency response management after hazardous events are predicted. While the papers mentioned have contributed greatly to the field in terms of hazard prediction, monitoring, and even alert systems, there was a lack of integrated solutions which would bind the hazard prediction and actual emergency services. AegisCore aims at filling this gap, providing an integrated system which incorporates environmental data processing and visualization, hazard prediction, emergency alerts, SOS in-browser messaging, shelter management, citizen hazard reporting, and other important services. The current prototype implements a controlled mock-up prediction pipeline; the historical database and LSTM-based prediction algorithms are being set up for the final version of the application to appear.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

III. SYSTEM DESIGN AND ARCHITECTURE

AegisCore is organized as a modular full-stack architecture where the user interface, backend, persistent storage, communication, and machine-learning service layers are separated and developed as independent modules. This design decouples the system's different components, enabling them to be built and tested separately without impacting the other parts. Figure 1 below represents the system's architecture and data flow between modules.

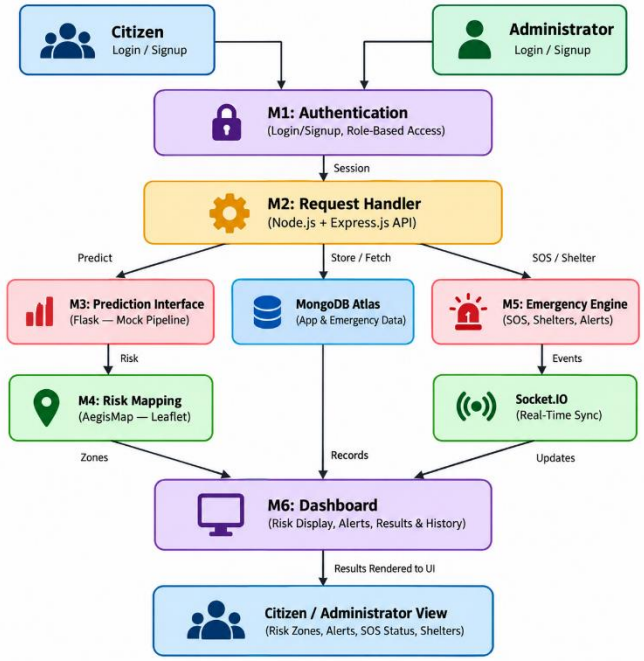


Fig. 1. System-level architecture and data flow diagram of AegisCore.

AegisCore uses a full-stack architecture that includes frontend, backend, database, time communication and prediction components. The frontend of AegisCore is built with React.js and Tailwind CSS. There are screens for administrators and citizens so they can view disaster information and request emergency help. AegisCore uses OpenStreetMap, Leaflet and React-Leaflet to show maps of district borders, disaster events, risk zones, places that have been hit and emergency data [4]. The backend of AegisCore created with Node.js and Express.js handles core services such as user login, disaster records, emergency alerts, shelter details, SOS. Citizen reports. AegisCore stores all application, user, disaster and emergency information in MongoDB Atlas, which lets AegisCore manage data from its modules efficiently. AegisCore uses Socket.IO to deliver real-time updates and instant messages between the platform and its users ensuring that AegisCore can send alerts and emergency notices quickly [3], [5]. AegisCore runs a Python Flask service that hosts Long Short-Term Memory (LSTM) models, for predicting floods and landslides using geographical data [1], [2]. AegisCore plans to cover Kodagu, Uttara Kannada, Belagavi and Bagalkot districts for flood prediction and disaster management while landslide prediction will focus on Kodagu and Uttara Kannada because those areas are especially vulnerable. AegisCore's modular design lets prediction, map visualization, data handling, communication and emergency response work together inside a platform giving AegisCore situational awareness and smoother disaster coordination [5].

IV. METHODOLOGY

The AegisCore methodology consists of a set of sequential stages ranging from receiving information about the environment to presenting risk assessment results and emergency response. Fig. 2 below represents this methodology in the form of a process.

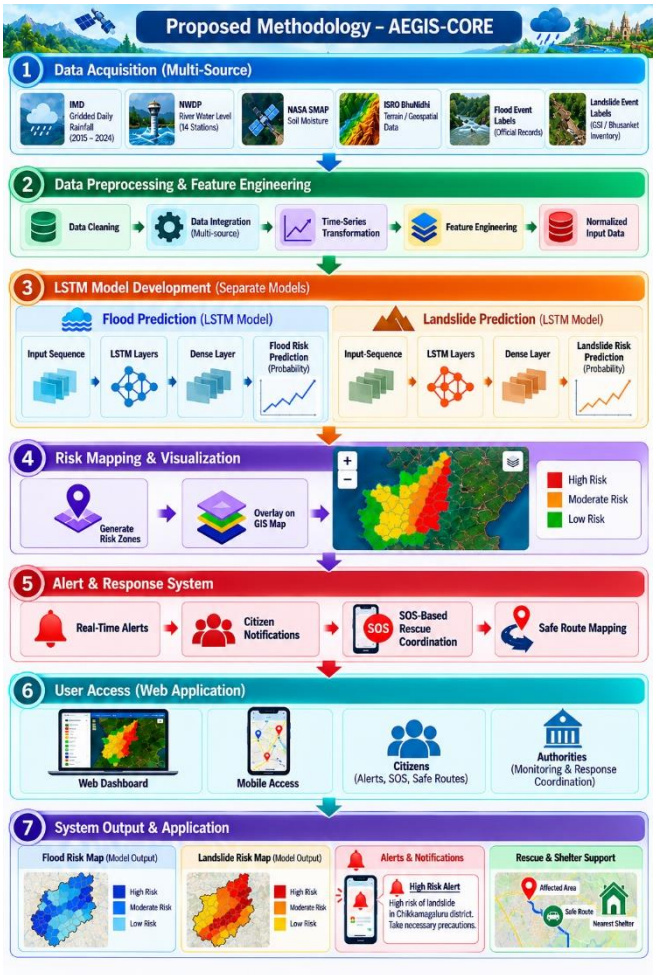


Fig. 2. Stage-wise methodology workflow of AegisCore.

The steps involved in disaster prediction and response in AegisCore include a structured process. First, there is data collection, in which case past data between 2015 and 2024 is collected for the targeted districts. In the data collection process, rainfall and river level data are taken into consideration for all the four districts, while the soil moisture, slope, and terrain are used only for Kodagu and Uttara Kannada districts, as these districts are prone to landslides [2]. The data collected is then pre-processed through the processes of cleaning, feature extraction, normalization, and sequencing for the LSTM models [1], [2]. Predicted flood risk predictions are made for a period of 15 days and landslides risk prediction is made for 30 days. The forecasted risk is then made available visually through AegisMap with the help of district boundaries

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

and severity levels. The system also allows for disaster response through alerts, citizen reporting, SOS coordination, sheltering, and evacuation [3], [5].

V. FEATURES

1. **Administrative Dashboard :** Provides risk information, prediction details, SOS monitoring, shelter administration and emergency alert management [3], [5].
2. **Citizen Interface:** Shows risk zones, active alerts, emergency services and nearby shelters by using the browser's ability to find your location.
3. **SOS Emergency Coordination:** Allows location-based SOS requests to be sent, stored in MongoDB and updated in time using Socket.IO. Status changes are tracked across PENDING, RESPONDING, RESOLVED and CANCELLED states [3], [5].
4. **Shelter Management:** Allows administrators to update shelter capacity and occupancy and keep shelter capacity and occupancy in sync, for citizens in time [5].
5. **Citizen Reporting and Alerts:** It lets citizens send in reports, about disasters and allows administrators to manage emergency alerts. This helps everyone stay better informed during emergencies [3][5].

VI. RESULTS

The AegisCore application has been implemented to support disaster monitoring provide citizen safety information manage SOS alerts and handle shelter operations. Figures 1, through 4 show the disaster monitoring dashboard the citizen safety interface, the historical LSTM evaluation screen and the shelter management dashboard respectively. The frontend production build was completed without any errors. The current system runs with a mock prediction pipeline. Integrating the trained LSTM models is planned for future work.

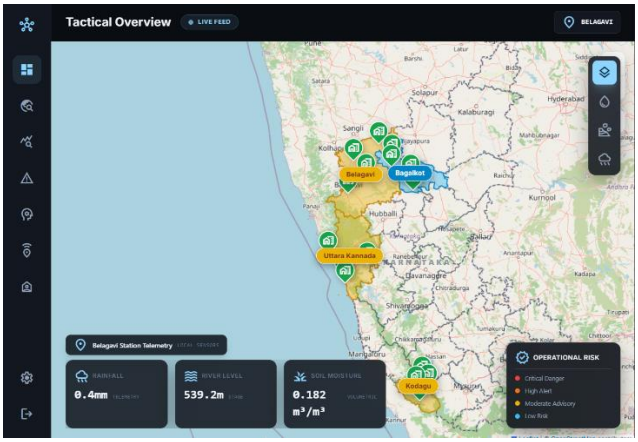


Fig.1 AegisCore's interactive disaster monitoring dashboard.

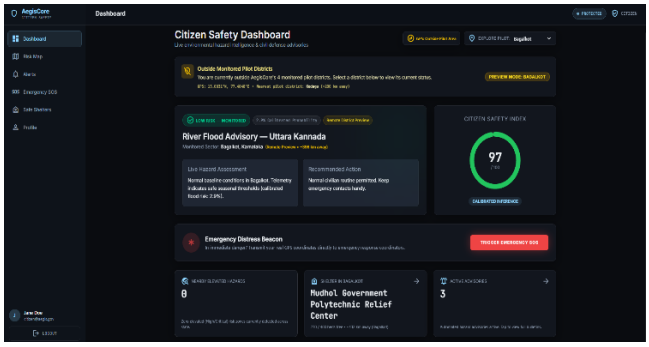


Fig.2 AegisCore's citizen safety dashboard.

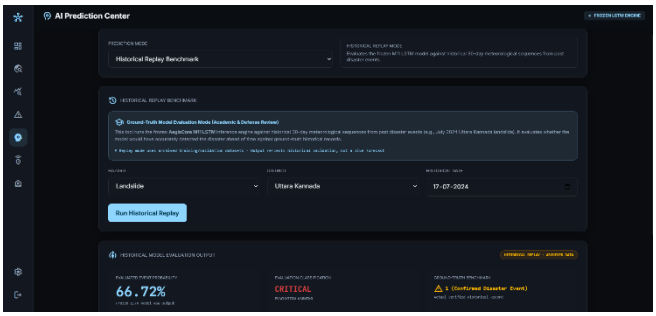


Fig.3 AegisCore's historical LSTM model evaluation dashboard.

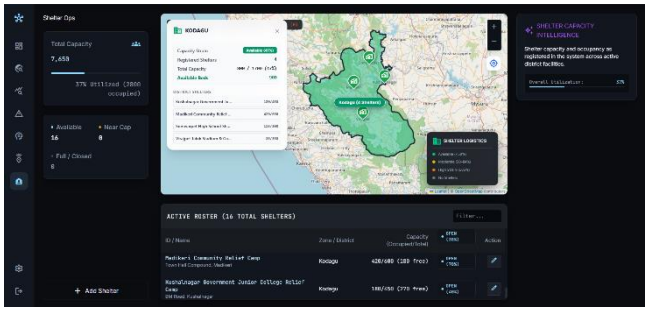


Fig.4 AegisCore's shelter management dashboard.

Metric	Flood				Landslide	
	Bagalkot	Belagavi*	Kodagu	Uttara Kannada	Kodagu	Uttara Kannada
Model / Threshold	LSTM / 0.4915	Random Forest / 0.20	LSTM / 0.5924	LSTM / 0.5854	LSTM / 0.50	LSTM / 0.50
Accuracy (%)	93.60	98.13	96.49	87.23	93.84	85.09
Sensitivity (%)	9.09	81.82	83.33	100.00	83.33	85.71
Specificity (%)	96.15	98.63	96.70	86.81	93.93	85.08
Balanced Accuracy (%)	52.62	90.22	90.02	93.41	88.63	85.40
ROC-AUC	0.5704	0.9790	0.9748	0.9721	0.9731	0.9459

Note: Belagavi flood detection uses a secondary Random Forest classifier with 26 input features. Thresholds for the other flood LSTM results were selected using training-period data through 2022. Landslide results use a fixed threshold of 0.50. All results are evaluated on held-out 2023–2024 data.

(This work is licensed under a Creative Commons Attribution 4.0 International License.)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 11 , November - 2026

VII. ADVANTAGES

AegisCore is an integrated platform that combines disaster monitoring, visualization of risk levels, emergency communication, and shelter management. One of the benefits of the proposed model is the separation of the machine-learning service from the main application; thus, the current mock pipeline can be replaced with trained LSTM models [1], [2]. Moreover, the utilization of OpenStreetMap, Leaflet, React-Leaflet, and MongoDB Atlas suggests the possibility of implementing an open-source solution for rendering districts, risk levels, emergency shelters, and other related information. The use of Socket.IO also allows for the implementation of real-time communication and information-sharing with the application. At the same time, the software's limitation is the concentration on four districts, including Kodagu, Uttara Kannada, Belagavi, and Bagalkot, which may have different peculiarities of landslides and floods, as compared to the entire region.

VIII. CONCLUSION

AegisCore is an integrated disaster management and emergency response platform created for districts in Karnataka. It brings together visualization, environmental data processing, emergency alerts, shelter management, citizen reporting and SOS communication [3], [5]. The frontend is built using React.js and Tailwind CSS while the backend runs on Node.js and Express.js with support from MongoDB Atlas and Socket.IO. For mapping, OpenStreetMap, Leaflet and React-Leaflet are used to create maps. A Python Flask service is set up to run LSTM-based prediction models [1], [2]. The current version has a controlled mock prediction pipeline includes 38 regression test cases and has a working frontend that's ready for production. In the future the plan is to gather validated datasets from 2015 to 2024 and put trained LSTM models into use, for real flood and landslide predictions [1], [2].

IX. FUTURE SCOPE

The main idea behind future development includes the implementation of the proven environmental dataset existing between the years 2015 – 2024 and employing trained LSTM's for flood and landslide prediction [1], [2]. Afterward, the performance of the implemented model can be validated using a relevant performance indicator (i.e. RMSE, MAE, precision, recall) according to the final model specification. The application can be extended to other districts if proven environmental data, GIS boundary information, and validated prediction model are available for the time being. Moreover, it might be relevant to consider future developments involving IoT-based sensors, including rain, river level, and soil moisture sensors, aiming to provide frequent environmental intelligence [5]. Connecting the application with disaster-management authorities and emergency response units can be an option for future development [3], [5]. Additionally, creating hazard-aware evacuation routing could be one of the innovative concepts behind the future development of the application.

X. REFERENCES

[1] P. M. Rasalkar and A. R. Surve, "Evolution of Flood Prediction Methods: Hydrological, Statistical, Machine Learning and Deep Learning Approaches," IEEE ICFT, 2025.
[2] M. R. Sumalatha et al., "Landslide Susceptibility Mapping for South India using Dense Neural Network and Bi-LSTM," IEEE IDCIoT, 2025.
[3] V. Banupriya et al., "Flood and Landslide Prediction and Risk Assessment System using Deep Learning with Real-Time Alerts," IEEE ICISS, 2025.
[4] Z. Liu et al., "Integration of Remote Sensing and Crowdsourced Data for Fine-Grained Urban Flood Detection," IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 2024.
[5] K. S. Kavya Sree et al., "An Adaptive AI-Driven Multi-Hazard Prediction and Early Alert Framework for Real-Time Emergency Response," ICRDICCT, 2025.

Data Sources :

IMD: Gridded daily rainfall data (2015–2024)
NWDP / CWC: River-level observations from 14 stations
NASA SMAP: Soil-moisture data
ISRO BhuNidhi: Terrain and elevation data
GSI / Bhusanket: Landslide inventory data
GIS Boundary Shapefiles: Karnataka district boundaries for spatial mapping

(This work is licensed under a Creative Commons Attribution 4.0 International License.)