

Engineering Resilience: A Distributed-Systems Approach to AI-Based Flood Early Warning for Bangladesh's River Basins

Kawser Ahmed Shovon

ahmedshovoncse13@gmail.com

Penta Global Limited

Anika Anjum Arin

Ahsanullah University of Science and Technology

Research Article

Keywords: flood forecasting, early warning systems, deep learning, LSTM, distributed systems, disaster resilience, Bangladesh, last-mile dissemination

Posted Date: September 24th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-11139505/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: The authors declare no competing interests.

Engineering Resilience: A Distributed-Systems Approach to AI-Based Flood Early Warning for Bangladesh's River Basins

Kawser Ahmed Shovon

Dept. of Computer Science and Engineering

Daffodil International University

Software Engineer, Penta Global Limited

Dhaka, Bangladesh

ahmedshovoncse13@gmail.com

Anika Anjum Arin

Dept. of Computer Science and Engineering

Ahsanullah University of Science and Technology

Dhaka, Bangladesh

anika.anjum.arin@gmail.com

Abstract—Bangladesh's Flood Forecasting and Warning Centre (FFWC) keeps watch over 109 water-level stations and 61 forecasting points across the Ganges-Brahmaputra-Meghna basin, yet operational lead times still rarely stretch past 24 to 48 hours. Most warnings still reach rural households the way they always have: phone calls, SMS, flags, and volunteers going door to door. The bulk of AI-based flood forecasting research chases accuracy gains and says comparatively little about the reliability engineering that a life-critical pipeline needs once it's actually running at national scale. We don't think these are separate problems. This paper sets out a system architecture and methodology for an AI-driven flood early-warning pipeline that pairs an LSTM/GRU forecasting layer, built on river-gauge, rainfall, and upstream-basin features, with a fault-tolerant ingestion, validation, and dissemination design borrowed from production distributed systems. Dissemination is built around the channels rural households already rely on, SMS and voice messaging on basic phones, rather than assuming smartphone or data-plan access. The framework aligns with UN Sustainable Development Goals 13, 11, and 9. As an initial check on the classification component, we run the proposed risk-tiering scheme against a live snapshot of 115 FFWC stations pulled on 7 September 2026. The result is a coherent picture: five stations fall into the Warning tier, all of them on the Teesta and Brahmaputra systems in Rangpur division, the part of the basin under the most hydrological stress at that moment. We lay out the experimental design for validating the forecasting model against FFWC's own operational baselines, and close with the engineering, ethical, and equity questions that come with deploying a system like this in a resource-constrained setting.

Index Terms—flood forecasting, early warning systems, deep learning, LSTM, distributed systems, disaster resilience, Bangladesh, last-mile dissemination

I. INTRODUCTION

Bangladesh sits at the confluence of the Ganges, Brahmaputra, and Meghna rivers, and most of the country is, geologically speaking, their delta. Flooding here isn't an occasional disaster; it's an annual fact of life, and it lands hardest on rural, low-income communities that have the least room to absorb it. To be fair, the institutional response has genuinely improved. The

Bangladesh Water Development Board's Flood Forecasting and Warning Centre (FFWC), working with partners such as the Regional Integrated Multi-Hazard Early Warning System (RIMES), now issues probabilistic forecasts extending 10 to 15 days out for the major rivers [6], and voice-message programs like SUFAL reportedly reach more than 20,000 households with no smartphone or data connection at all [2].

Two gaps pushed us to write this paper. First, machine-learning work on flood forecasting, in Bangladesh and elsewhere, tends to be judged almost entirely on offline accuracy metrics. Comparatively little gets written about what it actually takes to keep a model alive in production: data feeds that fail, stations that go dark for days at a time, and the uncomfortable fact that a missed warning here isn't just a bad metric, it's a life-safety failure. The second gap is more of a buried assumption than an omission. Most proposed AI flood-warning systems quietly assume the end user has an internet connection. For a large share of rural Bangladesh, that assumption simply doesn't hold, and it leads to systems that look technically impressive on paper but never reach the people who need them most.

Our own background is in distributed, multi-region telecommunications platforms, moving real-time data across teams and infrastructure spanning South Asia and Europe. A lot of that experience carries over directly to what a national flood-forecasting pipeline needs: designing around data feeds that are unreliable at the source, building ingestion that tolerates partial failure, and coordinating systems across regions where connectivity is uneven at best. The case we want to make is that climate-resilient AI, in a developing-country setting, is as much a distributed-systems problem as it is a modeling problem, and getting that framing right is what decides whether a system survives contact with the field or just looks good in a paper.

II. RELATED WORK

A. Machine Learning for Flood Forecasting

There's a substantial body of work applying deep learning to river-level and inundation forecasting, from LSTM and GRU recurrent architectures to newer graph neural network and transformer-based approaches. Karim et al. [3] survey hydrodynamic and machine-learning methods for flood inundation modeling and conclude that data-driven approaches now match or beat traditional ones on accuracy, though they still lag on incorporating expert knowledge and on shared benchmark data. At a more global scale, Zeng and Bertsimas [8] fold geographic and historical disaster data into a multimodal framework for multi-year flood risk prediction. Nearing et al. [5], meanwhile, show that AI-based forecasting can deliver reliable warnings for extreme riverine floods in ungauged watersheds up to five days ahead. The work closest to ours is Koli et al. [4], who build a 72-hour flash-flood warning pipeline for Bangladesh's haor wetlands entirely from freely available data, issuing warnings in Bengali over SMS. Notably, they treat message delivery and community trust as being just as important as model accuracy, not as an afterthought bolted on at the end.

B. Institutional Flood-Warning Infrastructure in Bangladesh

FFWC runs a public API that exposes real-time and historical water-level data, forecasts, and station metadata across its network [1], and RIMES has since pushed probabilistic forecasting out to an experimental 15-day horizon in partnership with BWDB [6]. Closer to the ground, SUFAL and similar community-based programs are already delivering voice-message flood warnings, days ahead of an event, to rural households with no smartphone or data connectivity [2], [7]. The point worth underlining here is that non-smartphone dissemination isn't a hypothetical constraint that researchers need to design around in the abstract. It's working infrastructure, already reaching people, and a research system should plug into it rather than try to replace it.

C. Gap: Production-Engineering Treatment of AI Flood Systems

What's largely missing from this literature is the software-engineering side of actually running one of these systems: validating data under station outages, handling redundancy and failover when sensor gaps are correlated across a basin, building observability and alerting for a pipeline where downtime has real consequences, and degrading gracefully when an upstream source goes quiet. Closing that gap is what this paper sets out to do, by bringing production-grade distributed-systems practice, the kind used to keep high-availability commercial platforms running, to bear on a climate-resilience problem.

III. RESEARCH CONTRIBUTION

This paper makes three contributions:

- A system architecture for an AI-based flood early-warning pipeline that separates forecasting, risk classification, and dissemination into decoupled services, each able to fail and recover on its own, following a pattern drawn from production telecom and distributed-systems design.
- A deep-learning methodology, comparing persistence and ARIMA baselines against LSTM/GRU sequence models that use upstream-basin lag features, aimed at pushing effective warning lead time past the current 24 to 48 hour operational range, drawing on FFWC's public station network as the primary data source.
- A last-mile dissemination design that treats voice-call and SMS delivery, the channels Bangladesh already relies on and trusts, as an output of the system rather than something bolted on afterward, so the contribution builds on existing infrastructure instead of depending on smartphones or internet access that many households simply don't have.

IV. PROPOSED SYSTEM ARCHITECTURE

The architecture splits into four layers, following the same separation used in production telecom systems that have to stay available even when upstream sources start misbehaving.

A. Data Ingestion and Validation Layer

This layer polls the FFWC data load API for water-level readings across the 109-station network, along with rainfall data and, in areas where ground coverage is thin, satellite-derived precipitation and Synthetic Aperture Radar (SAR) backscatter. Years of working with multi-provider telecom feeds teach you not to trust any single upstream source, and the ingestion layer reflects that lesson in three ways. Every incoming reading goes through schema validation and range checks. Stations that are missing or reporting late get flagged, and their values are imputed from neighboring stations where that makes sense. And ingestion runs through a message queue rather than synchronous calls, so one slow or failing source can't stall the rest of the pipeline.

B. Forecasting Model Layer

At the core is a sequence model, LSTM or GRU, benchmarked against a Transformer-based variant and trained on historical water-level and rainfall series. Engineered features bring in upstream station readings at lag intervals tuned to known basin travel times, so the model is effectively watching the whole river system rather than staring at one gauge in isolation. A persistence baseline and a classical ARIMA baseline are included for comparison. The model's output is a probabilistic river-level forecast at each forecasting station for 24-, 48-, and 72-hour horizons, explicitly aimed at stretching FFWC's current operational lead time further out.

C. Risk Classification and Decision Layer

Raw forecasts get converted into one of four discrete risk tiers, Normal, Watch, Warning, or Danger, measured against

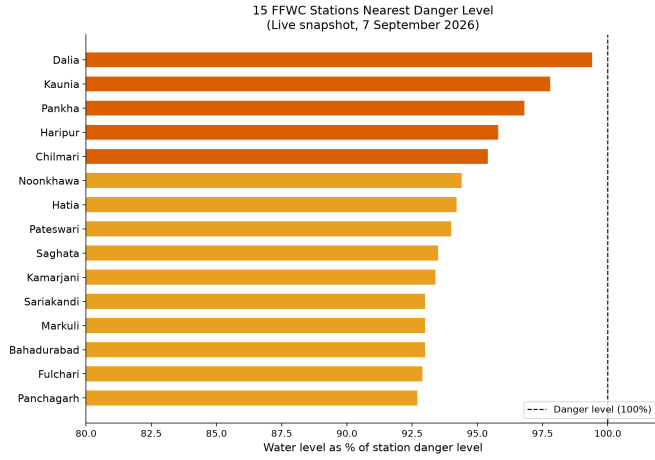


Fig. 1. The 15 FFWC stations nearest their danger level in the 7 September 2026 snapshot, colored by risk tier.

each station’s published danger level. We kept FFWC’s own operational vocabulary here, so the outputs stay interpretable to existing stakeholders without anyone having to retrain their response procedures. Warning and Danger classifications go through a human-in-the-loop review step before anything is disseminated, which is standard practice for automated decisions with this much riding on them.

D. Dissemination Layer

The dissemination layer sits behind an API boundary, decoupled from forecasting, so it can plug into channels that already exist: voice-call gateways of the kind SUFAL uses, SMS gateways, and community-volunteer notification lists. No new smartphone app is required, deliberately. A lightweight monitoring dashboard is proposed too, but as an internal tool for FFWC operators and researchers rather than a citizen-facing interface, since voice and SMS are already doing that job for non-smartphone households.

E. Mapping to Production Distributed-Systems Practice

Table I lays out how each pipeline layer maps onto an established production practice.

V. METHODOLOGY

A. Data Sources

- Primary: the FFWC/BWDB public API, giving real-time and historical water-level readings across 109 monitoring stations and 61 forecasting points [1].
- Secondary: publicly available rainfall records, plus satellite-derived precipitation and SAR imagery for basins where ground coverage is sparse.
- Reference baselines: FFWC’s own published 24 to 48 hour operational forecasts, which we use as the benchmark the proposed model has to beat.

B. Model Design and Evaluation

We compare three model families across held-out historical flood events: a persistence baseline, an ARIMA baseline, and LSTM/GRU sequence models built with upstream-lag features. The metrics are RMSE and MAE on water-level prediction; precision, recall, and F1 on danger-level threshold classification; false-alarm rate; and, since this is ultimately what matters operationally, effective lead-time extension relative to FFWC’s baseline. What this paper reports is the methodology and evaluation protocol itself, not results, since training and reporting are the next phase of the project, once multi-year station-level data has been assembled and cleaned.

C. Reliability and Failure-Mode Testing

Because the core argument here is an engineering one and not just a modeling one, the evaluation plan includes deliberate fault injection: simulating station dropout, delayed data arrival, and partial network outages to see how the pipeline’s forecasts hold up under conditions that are not clean or complete. That’s a different bar than the accuracy-only benchmarking most of this literature settles for, and it’s the one we think actually matters once a system goes live.

VI. PRELIMINARY EMPIRICAL ANALYSIS: A LIVE SNAPSHOT CASE STUDY

Before investing in training a full model, it’s worth checking that the classification logic behaves sensibly on real data. This section runs the proposed risk-tier scheme against a live snapshot of FFWC’s public station network, pulled from the Centre’s water-level reporting page on 7 September 2026. The snapshot covers 115 stations, ones that had a current reading, a historical maximum, and a published danger level all available at once; a handful of others were left out because their danger-level or maximum-level fields were blank at the time we pulled the data.

For each station, we expressed the current level as a percentage of its published danger level and sorted it into one of four tiers, matching FFWC’s own operational vocabulary: Normal below 85%, Watch from 85 to 95%, Warning from 95 to 100%, and Danger at or above 100%. The rule is intentionally simple, almost crude, and that’s the point: we wanted to confirm the scheme produces a picture that makes sense before layering a trained model on top of it.

A. Snapshot Results

Of the 115 stations, 70 came back Normal, 40 Watch, and 5 Warning. Nothing had crossed its danger level, which tracks with the timing, this snapshot falls near the tail end of the 2026 monsoon season. Thirty-six stations, about 31%, showed a rising three-hour trend, and the mean 24-hour change across the whole network was a modest +0.95 cm. That average, though, hides a fair amount of variation underneath it; several individual stations swung 60 to 80 cm in a single day. Table II lists the ten stations sitting closest to their danger threshold.

One pattern jumps out right away. Four of the five Warning-tier stations, Dalia, Kaunia, and Haripur on the Teesta, and

TABLE I
MAPPING BETWEEN FLOOD-WARNING PIPELINE LAYERS AND PRODUCTION DISTRIBUTED-SYSTEMS PRACTICE

Layer	Flood-warning pipeline	Analogous production practice
Ingestion	Poll FFWC/rainfall/satellite feeds; queue-based buffering	Multi-carrier roaming data feeds with async message queues
Validation	Range checks, gap detection, neighbor imputation	Schema validation across heterogeneous regional data sources
Forecasting	LSTM/GRU sequence model, versioned and monitored	Model/service versioning and rollback in production ML systems
Decision	Rule-based risk tiering with human review for high severity	Automated decisioning with human escalation for edge cases
Dissemination	Decoupled API to SMS/voice gateways	Loosely coupled microservices with independent failure domains
Observability	Uptime, latency, and data-freshness monitoring and alerting	SRE-style monitoring across distributed, multi-region teams

TABLE II
THE TEN FFWC STATIONS CLOSEST TO THEIR DANGER LEVEL, LIVE
SNAPSHOT OF 7 SEPTEMBER 2026

Station	River	WL (m)	DL (m)	% DL	24h	Tier
Dalia	Teesta	51.85	52.15	99.4	+5	Warning
Kaunia	Teesta	28.67	29.31	97.8	+2	Warning
Pankha	Ganges	21.35	22.05	96.8	+17	Warning
Haripur	Teesta	21.36	22.30	95.8	+2	Warning
Chilmari	Brahmaputra	22.18	23.25	95.4	+4	Warning
Noonkhawa	Brahmaputra	24.59	26.05	94.4	+7	Watch
Hatia	Brahmaputra	22.98	24.40	94.2	+6	Watch
Pateswari	Dudkumar	27.81	29.60	94.0	+3	Watch
Saghata	Jamuna	17.29	18.50	93.5	-3	Watch
Kamarjani	Brahmaputra	20.26	21.70	93.4	+2	Watch

Chilmari on the Brahmaputra, all sit in Rangpur division in the northwest, with Noonkhawa just barely missing the cut into Watch. A simple station-by-station threshold check catches each of these individually, but it catches them one at a time. A model that's watching the whole river system at once, which is exactly what the LSTM/GRU layer in Section IV-B is designed to do, could in principle catch this kind of clustering earlier. The rule-based tiering used here is reactive by design; a trained sequence model, conditioned on upstream readings, should be able to flag a station on its way toward Warning before it actually gets there, not just report it after the fact. Fig. 1 shows the 15 stations nearest their danger level.

B. What This Does and Does Not Demonstrate

To be clear about what this snapshot is and isn't: it's a sanity check on the classification scheme, not evidence that the forecasting approach itself works. It says nothing about lead time, because it's a single moment in time rather than a model projecting forward. What it does show is that FFWC's live data is actually retrievable, that its structure, station name, river, current level, danger level, historical maximum, short-term trend, fits directly into the pipeline described in Section IV, and that the tiering logic produces results a hydrologist would find reasonable at a glance: stations on the same river system cluster together, and the highest-risk stations line up with the systems that were genuinely under stress at that point in the season. The next step is assembling the multi-year historical

series needed to actually train and backtest the forecasting model.

VII. ALIGNMENT WITH SUSTAINABLE DEVELOPMENT GOALS

This framework supports SDG 13 (Climate Action) by strengthening adaptive capacity and early-warning infrastructure for climate-related hazards; SDG 11 (Sustainable Cities and Communities) through resilient, inclusive disaster-response infrastructure for at-risk rural and peri-urban communities; and SDG 9 (Industry, Innovation and Infrastructure) by designing reliable digital infrastructure with applications well beyond this one use case.

VIII. LAST-MILE CONNECTIVITY AND EQUITY CONSIDERATIONS

A system that improves the forecast but never reaches the people at risk hasn't really accomplished much. So last-mile delivery is treated here as a requirement, not a detail to sort out later. By exposing forecasts through an API that interoperates with existing voice-call and SMS infrastructure, including programs like SUFAL, and with community-volunteer networks, the system's usefulness doesn't hinge on smartphone ownership, mobile data access, or digital literacy, the same constraints that limit purely app-based approaches across rural Bangladesh [7]. That constraint cuts both ways, though: it also forces discipline on the modeling side. Outputs have to be simple enough to fit in a short voice message or SMS, a risk tier plus an expected timing, rather than a dashboard visualization that assumes someone is looking at a screen.

IX. LIMITATIONS AND ETHICAL CONSIDERATIONS

- Historical data quality and completeness across all 109 FFWC stations hasn't been fully assessed yet; whatever results come out of future work will be bounded by how good that data actually turns out to be.
- False alarms carry real social and economic costs, evacuations, lost labor days, on top of the cost of missed warnings, so the evaluation protocol weighs both error types rather than chasing recall alone.

- Any real deployment would need to happen in partnership with FFWC/BWDB and existing last-mile programs, not as a parallel or competing system. That matters both for data legitimacy and for not confusing communities with two warnings that disagree.
- Model outputs that touch life-safety decisions should keep a human in the loop, which is why that review step is built into the decision layer.
- The snapshot analysis here covers a single point in time, and only stations with complete metadata. It should be read as a check that the data and classification logic behave sensibly, not as evidence that the forecasting model works.

X. CONCLUSION AND FUTURE WORK

This paper has laid out a system architecture and evaluation methodology for an AI-based flood early-warning pipeline for Bangladesh, one that treats production-grade software engineering, fault tolerance, data validation, decoupled services, and graceful degradation as inseparable from forecasting quality itself, and that builds dissemination around channels that already work without a smartphone. The snapshot analysis gives us some confidence that the underlying data and classification logic hold up. What comes next is assembling and cleaning multi-year historical FFWC station data for training and backtesting, building the ingestion and forecasting pipeline as a working prototype, and running the fault-injection and lead-time evaluation described in Section V. Future work will report empirical results from this pipeline and look at extending the same architecture to cyclone and flash-flood hazards beyond riverine flooding.

REFERENCES

- [1] Bangladesh Water Development Board, Flood Forecasting and Warning Centre (FFWC), “Real-time and historical hydrological data API.” [Online]. Available: <https://ffwc.bwdb.gov.bd/>
- [2] Care Climate Change, “Early warning systems saving lives and livelihoods in Bangladesh: The SUFAL project voice-message dissemination model,” 2022.
- [3] F. Karim, M. A. Armin, D. Ahmedt-Aristizabal, L. Tychsen-Smith, and L. Petersson, “A review of hydrodynamic and machine learning approaches for flood inundation modeling,” *Water*, vol. 15, no. 3, p. 566, 2023, doi: 10.3390/w15030566.
- [4] S. H. T. Koli, F. H. T. Jely, M. S. Alim, and M. Z. Hossen, “HaorFlood-Alert: A 72-hour machine learning early warning system for flash floods in Bangladesh’s haor wetlands,” 2026, arXiv:2605.20167.
- [5] G. Nearing et al., “Global prediction of extreme floods in ungauged watersheds,” *Nature*, 2024, doi: 10.1038/s41586-024-07145-1.
- [6] Regional Integrated Multi-Hazard Early Warning System (RIMES), “Advanced deterministic, probabilistic and long-range flood forecasting for the Ganges-Brahmaputra-Meghna basin,” 2025.
- [7] United Nations Office for Disaster Risk Reduction (UNDRR), “Bangladesh: Advancing early warnings for all: Last-mile connectivity and Common Alerting Protocol implementation,” 2025.
- [8] C. Zeng and D. Bertsimas, “Global flood prediction: A multimodal machine learning approach,” 2023, arXiv:2301.12548.