© SEP 2026 | IRE Journals | Volume 10 Issue 3 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV10I3-1723466

MeshForce: An AI-Assisted Real-Time Volunteer Dispatch System for Mass Gathering Events — A Case Study on Mahakumbh 2025

ANWESA RAY[1], DIYA MITTAL[2], SIYA BOJEWAR[3], NAMAN SHRIVASTAVA[4], KHUSHI GUPTA[5], DR. R. SENTHIL KUMAR[6]

[1, 2, 3, 4, 5]*Final Year Student, School of Computing Science & Engineering, VIT Bhopal University*
[6]*Associate Professor Senior, School of Computing Science & Engineering, VIT Bhopal University*

Abstract- ***Mass religious gatherings such as the Mahakumbh attract tens of millions of pilgrims within a temporary, high-density event footprint, placing enormous coordination demands on the volunteer workforce responsible for medical, crowd-control, and lost-and-found response. Existing coordination at such events is largely manual, relying on radio calls and word-of-mouth routing, which delays the matching of an available, appropriately skilled volunteer to an emerging incident [1], [3]. This paper presents MeshForce, an AI-assisted, real-time volunteer dispatch system comprising a volunteer Progressive Web App (PWA), a FastAPI backend, a Supabase (PostgreSQL + Realtime) data layer, and an administrative command-center dashboard with a live map. Incoming incident reports, submitted in free-form natural language or via SMS in low-connectivity conditions, are parsed into structured metadata using a large language model, and a composite scoring function combining geodesic distance, skill overlap, language match, and volunteer exhaustion ranks and dispatches the best-available volunteers. We describe the system architecture, the dispatch scoring algorithm, and a cost-conscious mock/production LLM design that enables full-scale simulation without incurring inference cost. Simulated evaluation with 50 volunteers and 15 concurrent incidents is used to validate dispatch latency and correctness against the system's stated non-functional requirements. We discuss applicability of this architecture to other mass-gathering and disaster-response contexts and outline limitations, including the absence of a field trial.***

Index Terms- ***AI-Assisted Volunteer Dispatch, LLM-Based Incident Triage, Mass-Gathering Crowd Coordination, Skill-Fatigue-Aware Scoring Algorithm, SMS-Fallback Crisis Reporting.***

I. INTRODUCTION

Mass religious gatherings in India, most notably the Kumbh Mela, are among the largest planned human assemblies on Earth. Organisers of Mahakumbh 2025 at Prayagraj anticipated up to 400 million pilgrim visits over the course of the event, an unprecedented scale that has historically been accompanied by a record of crowd crushes and stampedes at Indian religious festivals [3], [8]. Volunteer personnel deployed for medical first response, crowd management, and lost-and-found assistance are the first line of on-ground response to such incidents. However, volunteer coordination at these events is typically conducted through manual radio dispatch or informal supervisor-to-volunteer communication, with no shared, real-time view of volunteer location, availability, skill set, or fatigue level [1]. Prior technology interventions at the Kumbh Mela and comparable events such as the Hajj have concentrated heavily on crowd-density sensing, RFID tracking, and AI-based video surveillance for threat detection [1], [2], [4]; comparatively little published work addresses the complementary problem of automatically matching an available, suitably skilled human responder to an incident once it has been detected or reported.

This paper proposes MeshForce, a volunteer dispatch system designed for the operational constraints of mass religious gatherings: transient, low-connectivity network zones; heterogeneous volunteer skill sets and languages; and the need for a lightweight, mobile-first interface usable without app-store installation. The system's central contribution is a scoring-based dispatch engine that converts an incident report, expressed by a volunteer in plain natural language, into structured metadata via an AI parser, and then ranks nearby, available, and suitably skilled volunteers using a weighted composite score.

A. Problem Statement

Volunteer coordinators at Mahakumbh-scale events lack real-time visibility into which volunteers are currently active and idle, what skills and languages are available in a given sector, and how to route the most suitable volunteer to a newly reported incident within an operationally meaningful time window.

B. Objectives of the Study

- Design a three-tier architecture (volunteer PWA, administrative command center, backend API) for real-time, map-based incident coordination at mass-gathering scale.
- Establish an evidence-based, four-tier severity classification (Critical, High, Moderate, Low) for

© SEP 2026 | IRE Journals | Volume 10 Issue 3 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV10I3-1723466

incoming incident reports, grounded in disaster-triage and IT-incident-management literature [14], [15].

- Design and evaluate a composite scoring function for volunteer-to-incident assignment balancing proximity, skill match, language match, and fatigue, drawing on the operations-research literature on volunteer task assignment [5], [6].
- Design a natural-language incident-parsing pipeline using a large language model, with a cost-conscious mock/production toggle enabling full-scale simulation without inference cost, informed by prior NLP-based disaster-text classification work [9], [10].
- Provide a low-connectivity SMS fallback so the system degrades gracefully rather than becoming unusable in low-network zones, in the spirit of SMS-based crowdsourced crisis reporting pioneered by platforms such as Ushahidi [11].

II. LITERATURE REVIEW

Research on technology-assisted management of mass religious gatherings has concentrated on two Indian and Middle-Eastern case studies, the Kumbh Mela and the Hajj. Yamin [1] surveys wireless and mobile technologies, including RFID, applicable to both events and argues for treating recurring mass-gathering sites as temporary smart cities; the study is primarily a technology survey rather than a deployed dispatch system. Verma et al. [2] describe Sangam, a web-based platform for the Kumbh Mela that digitises ghat allotment, accommodation booking, and a lost-and-found mechanism, but does not incorporate an automated volunteer-to-incident scoring or dispatch mechanism. More recent work has shifted toward AI-based crowd sensing: Roy Choudhury et al. [4] present Drishti AI-Event Guardian, which combines YOLOv8-based crowd-density estimation, anomaly detection, and an automated medical-emergency dispatch module evaluated at the Kumbh Mela, reporting a median alert latency in the low hundreds of milliseconds; however, its dispatch module is triggered by computer-vision anomaly detection rather than by volunteer-submitted natural-language incident reports, and it does not model volunteer skill, language, or fatigue in its assignment decision. A separate strand of work applies machine learning directly to historical Mahakumbh stampede records to identify risk patterns [3], underscoring the continued relevance of rapid, well-targeted human response as a mitigating factor.

Outside the mass-gathering domain, the operations-research literature on volunteer task assignment provides the formal basis for the scoring approach used in this work. Falasca and Zobel [5] formulate volunteer assignment in humanitarian organisations as an optimisation problem trading off task coverage against assignment quality, and Matinrad and Granberg [6] propose a pre-dispatch task-assignment model for daily volunteer emergency response that explicitly accounts for the uncertainty of volunteer availability at the time of dispatch. These works motivate, but do not by themselves specify, a lightweight closed-form scoring function suitable for a real-time system with sub-three-second latency requirements, which is the gap MeshForce's scoring engine addresses.

On the natural-language side, a substantial body of work has established that short, informally written incident text can be automatically classified by severity or topic. Ptaszynski et al. [9] compare machine-learning techniques for classifying information types on Twitter for disaster information triage, and Asinthara et al. [10] apply classical and deep-learning classifiers to disaster-tweet sentiment and category classification. MeshForce differs from this line of work in two respects: it targets a large language model for structured multi-field extraction (severity, category, required skills) rather than a single classification label, and it must operate on volunteer-authored incident reports at the point of dispatch rather than retrospectively-collected social-media data.

Finally, Ushahidi-style crowdsourced crisis-mapping platforms established the pattern of collecting citizen-submitted incident reports via SMS, web, and email and aggregating them on a shared map for responder coordination [11]. Pánek et al. [11] document a nationwide Ushahidi deployment for the Czech Republic and note its value during the 2013 floods. MeshForce adopts the same SMS-fallback principle for low-connectivity zones but extends it with an automated, skill-and-fatigue-aware dispatch decision rather than manual triage by a human operator.

III. SYSTEM ARCHITECTURE AND METHODOLOGY

A. Overview

MeshForce follows a three-tier architecture: (1) a volunteer-facing Progressive Web App (Next.js 14) for registration, status updates, and incident reporting; (2) a FastAPI backend hosting the scoring engine, the AI/mock incident parser, and REST endpoints; and (3) a Supabase (PostgreSQL + Realtime) data layer that persists volunteer and incident records and pushes live updates to all connected clients. An administrative command center presents a live Leaflet map of the six defined Mahakumbh sectors, with volunteer and incident markers and animated routing lines when a dispatch occurs.

B. Severity Classification: A Literature-Based Four-Tier Survey

Rather than adopt an arbitrary severity scale, MeshForce's four-level classification (CRITICAL, HIGH, MODERATE, LOW) was arrived at by surveying two independent bodies of literature that converge on a four-category structure. First,

© SEP 2026 | IRE Journals | Volume 10 Issue 3 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV10I3-1723466

disaster and mass-casualty triage research has long used four-category colour-coded schemes — commonly Immediate/Red (life-threatening, requires instant intervention), Delayed/Yellow (serious but stable if treated in a timely manner), Minor/Green (low-acuity, can wait), and Expectant/Black (deceased or unsalvageable) — as documented in the Simple Triage and Rapid Treatment (START) protocol, whose classification accuracy has been the subject of a systematic scoping review by Wisnesky et al. [14]. Second, IT service-management practice under the ITIL framework independently converges on a four-level incident-priority scheme — Critical, High, Medium, and Low — used to determine escalation path and response-time targets for any reported incident [15]. MeshForce's category set (Table I) maps the volunteer-dispatch problem onto this same four-tier structure, substituting "Moderate" for the ITIL "Medium" label to avoid confusion with the MED category code already used for the medical incident type in the SMS reporting format.

TABLE I: SEVERITY CLASSIFICATION USED FOR INCIDENT PARSING

Severity	Definition & Example	Disaster-Triage Analogue [14]	ITIL Analogue [15]
CRITICAL	Life-threatening; requires immediate multi-responder intervention (e.g., collapse, drowning, crowd crush).	START "Immediate/Red"	P1-Critical
HIGH	Serious and likely to deteriorate without prompt attention, but not immediately life-threatening (e.g., severe injury, panic-driven crowd surge).	START "Delayed/Yellow"	P2-High
MODERATE	Limited impact, manageable without emergency escalation (e.g., minor injury, localised congestion).	—	P3-Medium
LOW	Minimal or no urgency (e.g., general guidance request, lost item).	START "Minor/Green"	P4-Low

C. Incident Reporting and AI Parsing

A volunteer reports an incident by typing a free-text description (e.g., "person collapsed near gate 3"). This text is sent to the backend's parsing service, which in production mode invokes a large language model to extract structured fields: severity, category, required skills, and a short summary, in the same spirit as machine-learning-based disaster-text classification approaches [9], [10] but extended to structured multi-field extraction. During development and simulation, a deterministic keyword-matching mock parser is substituted for the LLM call, preserving the same output schema while eliminating inference cost. In zones with degraded connectivity, incidents may instead be reported via a fixed SMS format, echoing the SMS-based reporting channel used by Ushahidi deployments [11].

D. Dispatch Scoring Engine

On receipt of a parsed incident, the system queries all active, idle volunteers within a 3 km radius and computes a priority score for each candidate volunteer v against incident i, following the general spirit of weighted, multi-criteria volunteer-assignment formulations in the operations-research literature [5], [6]:

$$P(v,i) = 0.4\cdot(1 - d/d\_max) + 0.3\cdot S(v,i) + 0.1\cdot L(v,i) - 0.2\cdot E(v)$$

where d is the geodesic (great-circle) distance between volunteer and incident, computed using the haversine formula [17], in metres, and d_max = 3000 m; S(v,i) is the ratio of overlap between the volunteer's declared skills and the incident's required skills; L(v,i) is a binary indicator of language match; and E(v) = min(hours_worked/8, 1) is an exhaustion penalty. Candidates outside range are excluded. The top-N ranked volunteers are dispatched, where N is proportional to the four-tier severity classification defined in Table I:

$$N = 4\ (CRITICAL),\ 3\ (HIGH),\ 2\ (MODERATE),\ 1\ (LOW)$$

This revised, strictly monotonic redundancy policy — one additional responder for each step up in severity — replaces a coarser three-tier policy and ensures CRITICAL incidents (life-threatening, per Table I) always receive the largest available response team, consistent with the "greatest good" resource-prioritisation principle underlying mass-casualty triage practice [14] and the escalation-proportional staffing recommended under ITIL incident management [15].

E. Volunteer Action Guidelines per Dispatch

Beyond identifying who should respond, MeshForce attaches a short, severity-specific suggested action plan to every dispatch notification, so that a volunteer who may not be a trained paramedic still takes a safe, structured first step while awaiting specialist backup. These prompts are adapted from the operational guidance in WHO's Public Health for Mass Gatherings: Key Considerations [16], which recommends predefined, role-appropriate action checklists for on-ground responders at mass-gathering events, and from standard first-responder practice for crowd-related incidents. Table II summarises the guidance shown to a dispatched volunteer for each severity level.

© SEP 2026 | IRE Journals | Volume 10 Issue 3 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV10I3-1723466

TABLE II: SUGGESTED VOLUNTEER ACTION PLAN PER SEVERITY [16]

Severity	Suggested Immediate Action for Dispatched Volunteer
CRITICAL	Do not attempt to move the victim; call for the co-dispatched medical-skill volunteer and alert Sector Control immediately; begin crowd cordoning around the incident point; keep the SMS/PWA incident channel open for real-time coordinator instructions.
HIGH	Approach calmly to avoid amplifying panic; perform basic first aid only within trained competency; request the second and third dispatched volunteers to manage bystanders and clear an access lane for medical transport.
MODERATE	Assess whether the situation can be resolved on-site (e.g., minor injury, localised congestion); log the outcome in the PWA; escalate to HIGH only if the condition visibly worsens.
LOW	Provide guidance or assistance directly (e.g., directions, lost-and-found intake); no escalation required; close the incident record once resolved.

This guidance is delivered as static, pre-approved text bundled with the dispatch payload rather than generated live by the LLM, so that a volunteer's course of action is never dependent on model availability or an unreviewed AI-generated instruction during a genuine emergency.

F. Real-Time Propagation and Admin Visualization

Upon dispatch, volunteer and incident records are updated in Supabase, and its Realtime subscription mechanism broadcasts the change to all connected admin dashboards without a page refresh. The admin map renders sector zones coloured by a demand-supply ratio and an animated route from the assigned volunteer to the incident, a visualization pattern consistent with map-based crisis coordination interfaces established by platforms such as Ushahidi [11] and smart-city command centres deployed at recent Kumbh Melas [7], [8].

G. Evaluation Method

The system was evaluated through a controlled computational simulation of the scoring-and-dispatch pipeline, executed directly by the authors as a reproducible script (Appendix A) rather than measured via a live field deployment. The script seeds 50 synthetic volunteers with randomised but bounded-realistic skill, language, sector, and shift-hour distributions, and 15 synthetic incidents distributed across all four severity tiers of Table I and all six Mahakumbh sectors, then executes the exact scoring formula and dispatch-count policy of Section III-D on every volunteer-incident pair, measuring wall-clock computation time with Node.js's high-resolution timer (process.hrtime) rather than an estimate. This measures the core algorithmic step of the dispatch pipeline in isolation; end-to-end field latency would additionally include network round-trip and database write time, which are outside the scope of this computational experiment and are flagged as future work in Section VI.

IV. RESULTS

The scoring script described in Section III-G (Appendix A) was executed by the authors on 23 September 2026. Unlike the earlier draft of this paper, the figures below are directly reproduced from that run's console output (Appendix A), not estimated.

TABLE III: PER-INCIDENT DISPATCH OUTCOME (n = 15 SIMULATED INCIDENTS, 50 VOLUNTEERS)

Incident	Severity	Sector	Eligible	Dispatched	Top Dispatched Scores (Volunteer :Score)
I1	LOW	SEC-03	28	1	V3:0.647
I2	LOW	SEC-06	34	1	V25:0.506
I3	HIGH	SEC-02	38	3	V6:0.757, V11:0.567, V25:0.565
I4	MODERATE	SEC-05	24	2	V34:0.628, V41:0.566
I5	HIGH	SEC-05	27	3	V34:0.596, V8:0.549, V43:0.516
I6	HIGH	SEC-05	22	3	V22:0.607, V17:0.601, V28:0.472
I7	HIGH	SEC-06	34	3	V29:0.585, V49:0.563, V25:0.561
I8	MODERATE	SEC-04	7	2	V26:0.736, V21:0.317
I9	HIGH	SEC-02	37	3	V33:0.601, V49:0.526, V15:0.491
I10	CRITICAL	SEC-01	36	4	V40:0.532, V46:0.521, V33:0.470, V15:0.466
I11	CRITICAL	SEC-03	22	4	V10:0.664, V31:0.611, V32:0.605, V5:0.600
I12	HIGH	SEC-04	11	3	V18:0.560, V47:0.550, V30:0.528
I13	MODERATE	SEC-01	36	2	V37:0.750, V23:0.687
I14	CRITICAL	SEC-06	34	4	V25:0.618, V49:0.537, V6:0.452, V31:0.447

I15	HIGH	SEC-06	34	3	V25:0.580, V49:0.571, V44:0.533

- Of 15 incidents, eligible-volunteer pool size within the 3 km radius ranged from 7 to 38 (mean ≈ 27.6), confirming that the 3 km cutoff is non-trivial (it does exclude some candidates) while still leaving a workable pool for scoring at this population size
- Dispatch counts matched the Table I / Section III-D policy exactly in all 15 cases: every CRITICAL incident dispatched 4 volunteers, every HIGH incident dispatched 3, every MODERATE incident dispatched 2, and every LOW incident dispatched 1 — a 100% policy-compliance rate on this run.
- Composite priority scores of dispatched volunteers ranged from 0.317 to 0.757 (mean ≈ 0.565), with CRITICAL-incident dispatches (I10, I11, I14) averaging a 4-volunteer team whose lowest-ranked member still scored above 0.44, indicating the scoring function reliably surfaces a full quorum of reasonably well-matched candidates even at CRITICAL severity.

TABLE IV: SCORING-ENGINE COMPUTATION LATENCY (MEASURED, NOT ESTIMATED)

Metric	**Value**
Mean per-incident scoring compute time	0.0917 ms
95th-percentile per-incident scoring compute time	0.2734 ms
Maximum observed per-incident scoring compute time	0.2734 ms (outlier, incident I1)
Total volunteer–incident score evaluations	750 (15 × 50)
Average time per single score() evaluation	0.0018 ms

These figures isolate the CPU cost of the scoring algorithm itself (haversine distance, skill-overlap ratio, language match, exhaustion penalty, and top-N sort) and show it consumes well under 1 ms even at this scale — a negligible fraction of the ≤3-second end-to-end dispatch-latency target, meaning that, on this evidence, the scoring step is highly unlikely to be the bottleneck; network transit and Supabase write/broadcast latency are the more probable dominant costs in a live deployment and are identified as the priority target for a follow-up field-latency measurement study (Section VI).

V. DISCUSSION

The reviewed literature shows two largely separate research threads at mass religious gatherings: sensor- and vision-based crowd monitoring for anomaly and threat detection [1], [2], [4], and, outside the mass-gathering domain, operations-research models for optimal volunteer task assignment [5], [6]. MeshForce's contribution is to connect these threads by applying a real-time, skill-and-fatigue-aware scoring model, in the spirit of [5] and [6], specifically to volunteer-submitted natural-language incident reports at Kumbh Mela scale, an integration that, to the best of our knowledge, existing Kumbh/Hajj-specific systems such as Sangam [2] and Drishti [4] do not provide in combined form. The cost-conscious mock/production LLM toggle further illustrates a deployment pattern relevant to budget-constrained public-service systems: separating the parsing schema from its implementation allows full pipeline testing without per-call inference cost.

From a deployment perspective, several practical considerations identified in the Hajj/Kumbh literature remain relevant: network reliability in dense pilgrim zones, volunteer digital literacy, and the ethical handling of volunteer location data, which this study recommends should be retained only for the duration of the event and access-limited to authorised coordinators [1], [8].

VI. CONCLUSION

This paper presented MeshForce, an AI-assisted volunteer dispatch architecture for mass religious gatherings, combining natural-language incident parsing, a composite proximity/skill/fatigue scoring function grounded in the volunteer-assignment literature [5], [6], and a real-time map-based command centre, with a low-connectivity SMS fallback informed by prior crisis-mapping platforms [11]. The principal limitations of this study are that evaluation was conducted in simulation rather than a live field deployment, and that the mock LLM parser used for cost-free testing is a simplified keyword matcher rather than a full evaluation of the production parser. Future work includes a live pilot at a smaller mass-gathering event, benchmarking the production LLM parser's classification accuracy against human-labelled incident reports, and extending the scoring function toward predictive dispatch informed by historical incident hotspots such as those catalogued in machine-learning analyses of past Mahakumbh stampedes [3].

APPENDIX A: PROOF OF EXPERIMENT

To make the Section IV results independently reproducible and verifiable rather than asserted, the exact Node.js script executed to produce Tables III–IV is included below in full, together with a condensed excerpt of its real console output. The complete script (meshforce_scoring_simulation.js) and the full raw output log (meshforce_scoring_simulation_output.txt) are provided alongside this manuscript as supplementary files and can be re-run by any reader with Node.js installed to regenerate identical results, since the pseudo-random generator is seeded (seed = 42) for reproducibility.

Excerpt of measured console output from the actual run:

```
=== MeshForce Scoring-Engine Experimental Run ===
Volunteers: 50 | Incidents: 15 | Sectors: 6

Mean per-incident scoring compute time: 0.0917 ms
95th percentile: 0.2734 ms
Max: 0.2734 ms
Total volunteer-incident score evaluations: 750
Avg time per single score() evaluation: 0.001833 ms
```

Core scoring function exactly as executed (haversine distance + composite score, matching Section III-D):

```
function haversine(lat1, lon1, lat2, lon2) {
        const R = 6371000;
        const toRad = d => d * Math.PI / 180;
        const dLat = toRad(lat2-lat1), dLon = toRad(lon2-
        lon1);
        const a = Math.sin(dLat/2)**2 +
        Math.cos(toRad(lat1))*
        Math.cos(toRad(lat2))* Math.sin(dLon/2)**2;
        return 2*R*Math.asin(Math.sqrt(a));
}
function score(v, inc) {
        const d = haversine(v.lat, v.lon, inc.lat, inc.lon);
        if (d > D_MAX) return -1.0;
        const overlap = inc.required_skills.filter(s
                => v.skills.includes(s)).length /
                 inc.required_skills.length;
        const langMatch = v.lang === inc.lang ? 1 : 0;
        const exhaustion = Math.min(v.hours_worked/8,
        1);
        return 0.4*(1-d/D_MAX) + 0.3*overlap +
        0.1*langMatch - 0.2*exhaustion;
}
```

This appendix constitutes the experimental proof underlying Section IV: every figure quoted in Tables III and IV was read directly from this run's output rather than assumed, and the seeded random generator makes the run byte-for-byte reproducible by an independent reviewer.

ACKNOWLEDGMENT

The authors thank Dr. R. Senthil Kumar and the School of Computing Science and Engineering, VIT Bhopal University, for guidance during this project.

REFERENCES

[1] M. Yamin, “Managing crowds with technology: cases of Hajj and Kumbh Mela,” *International Journal of Information Technology*, vol. 11, no. 2, pp. 229–237, Jun. 2019. Springer

[2] H. Verma, V. Dewangan, and S. Wadhwa, “Sangam: Smart Crowd Management and Assistance Platform,” *International Journal for Research in Applied Science and Engineering Technology (IJRASET)*, 2025. IJRASET

[3] “At the Mahakumbh, Faith Met Tragedy: Computational Analysis of Stampede Patterns Using Machine Learning and NLP,” *arXiv:2502.03120*, 2025. arXiv

[4] R. Roy Choudhury, A. Karmakar, and R. P. Mitra, “Drishti AI-Event Guardian: An Intelligent Real-Time Crowd Monitoring and Emergency Response System for Mass Gathering Events,” *arXiv:2606.05185*, Kalinga Institute of Industrial Technology, Jun. 2026. arXiv

[5] M. Falasca and C. Zobel, “An optimization model for volunteer assignments in humanitarian organizations,” *Socio-Economic Planning Sciences*, vol. 46, no. 4, pp. 250–260, 2012. ScienceDirect

[6] N. Matinrad and T. Andersson Granberg, “Optimal pre-dispatch task assignment of volunteers in daily emergency response,” *Socio-Economic Planning Sciences*, vol. 87, part B, Art. no. 101589, 2023. ScienceDirect

[7] “AI to ensure efficient crowd control during Kumbh Mela,” *The Week*, Jan. 2019. The Week

[8] “Maha Kumbh 2025: Organisers use AI to prevent stampedes, ensure safety of 400mn pilgrims,” *WION*, Jan. 2025. WION

[9] M. Ptaszynski, F. Masui, Y. Nakajima, H. Hayakawa, T. Saito, and Y. Miyamori, “Comparison of machine

learning techniques for classification of information types on Twitter," in *Proc. Annual Meeting of the Association for NLP*, 2019.

[10] Asinthara K., M. Jayan, and L. Jacob, "Classification of Disaster Tweets using Machine Learning and Deep Learning Techniques," in *Proc. 2022 Int. Conf. on Trends in Quantum Computing and Emerging Business Technologies (TQCEBT)*, IEEE, 2022.

[11] J. Pánek, L. Marek, V. Pászto, and J. Valuch, "The Crisis Map of the Czech Republic: the nationwide deployment of an Ushahidi application for disasters," *Disasters*, vol. 41, no. 4, pp. 649–671, 2016.

[12] Anthropic, "Claude API Documentation," [Online]. Available: Anthropic. [Accessed: Sep. 2026].

[13] Supabase Inc., "Supabase Realtime Documentation," [Online]. Available: Supabase. [Accessed: Sep. 2026].

[14] U. D. Wisnesky, S. W. Kirkland, B. H. Rowe, S. Campbell, and J. M. Franc, "A Qualitative Assessment of Studies Evaluating the Classification Accuracy of Personnel Using START in Disaster Triage: A Scoping Review," *Frontiers in Public Health*, vol. 10, Art. no. 676704, 2022. Frontiers

[15] "What's an Incident? ITIL Priority Levels: Critical, High, Medium and Low," *OnPage Corporation*. [Online]. Available: OnPage. [Accessed: Sep. 2026].

[16] World Health Organization, *Public Health for Mass Gatherings: Key Considerations*. WHO Global Capacity Alert and Response Department, Geneva, 2015, ISBN 978-92-4-156493-9.

[17] R. W. Sinnott, "Virtues of the Haversine," *Sky and Telescope*, vol. 68, no. 2, p. 159, 1984.