

Crisis Connect: Sos Emergency Using Advanced-Deep Learning Speech Recognition Based on Vosk Engine

K. Mounika¹, CH.P.Chaitanya², P.Pranay Sai³, R.Lahithi⁴

¹Assistant professor, Department of Information Technology, Matrusri Engineering College, Hyderabad, Telangana, India.

^{2, 3, 4} UG scholar, Department of Computer Engineering, Matrusri Engineering College, Hyderabad, Telangana, India.

To Cite this Article: K. Mounika¹, CH.P.Chaitanya², P.Pranay Sai³, R.Lahithi⁴, "Crisis Connect: Sos Emergency Using Advanced-Deep Learning Speech Recognition Based On Vosk Engine", Indian Journal of Computer Science and Technology, Volume 05, Issue 01 (January-April 2026), PP: 289-292.



Copyright: ©2026 This is an open access journal, and articles are distributed under the terms of the [Creative Commons Attribution License](#); Which Permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Abstract: In a personal emergency, the transition from safety to crisis is instantaneous, often triggering a "motor-function freeze" that renders manual SOS triggers useless. This vulnerability is worsened by the "connectivity gap"—the failure of cloud-reliant apps in tunnels, basements, and dead-zones. This paper introduces Crisis Connect, a decentralized, network-agnostic framework that moves speech recognition to the device edge using the Vosk engine. By supporting English, Hindi, and Telugu without requiring data packets, and utilizing a Multi-Stage Vocal Verification Algorithm (MVVA) to filter environmental noise, the system ensures near-zero latency. Crisis Connect elevates the smartphone to an autonomous guardian, proving that the most reliable technology is the one that stays functional even when the world goes offline.

Key Words: Edge-AI; Offline Speech Recognition; Vosk Engine; Network-Agnosticism; Trilingual NLP; Human-Centric Safety; SOS Architecture; Android Foreground Services; Crisis Connect; Latency-Critical Systems.

I. INTRODUCTION

In the chaotic threshold of a personal emergency, the human brain undergoes a physiological "motor-function freeze," rendering the traditional manual interface of a smartphone a cognitive impossibility¹. While modern society assumes ubiquitous connectivity, this reliance creates a dangerous "digital shadow"—a state of technological abandonment that occurs the moment a user enters a transit tunnel or a remote dead-zone^{2, 4}. In these high-stakes environments, the cloud is not a savior; it is a point of failure.

This vulnerability is magnified by a linguistic divide. In the multilingual landscape of the Indian subcontinent, English-centric safety tools act as a barrier to survival for millions¹². To address these gaps, we must transition toward autonomous, "always-on" guardians that operate at the device edge⁸. A cry for help should not be silenced by a dropped signal or a language barrier.

To bridge this divide, we introduce **Crisis Connect**, a decentralized framework engineered for absolute reliability in isolation. By leveraging the Vosk engine for local speech recognition, our system eliminates the need for active data packets entirely³. Through architectural optimization for real-time inference, the system maintains a persistent trilingual listening state in English, Hindi, and Telugu^{5, 11}. By ensuring that a vocal trigger initiates a rescue sequence through a robust internal command structure, this research provides a blueprint for a future where personal safety is a universal guarantee, regardless of connectivity¹⁴.

II. MATERIAL AND METHODS

The development of the *Crisis Connect* framework follows a structured, multi-layer architecture designed to ensure high-fidelity voice triggering and autonomous system resilience without cloud dependency. The methodology is partitioned into four distinct technical phases:

A. Edge-AI Infrastructure and Foreground Persistence

To ensure the system remains an "always-on" guardian, it utilizes a "Sticky" Kotlin-based Foreground Service within the Android environment. Unlike standard applications that are subject to OS-level background restrictions, this architecture ensures the SOS trigger mechanism remains active in the device's volatile memory (RAM). The integration with the Flutter-based UI is managed via a dedicated MethodChannel, allowing for real-time, low-latency communication between the cross-platform interface and the low-level hardware audio abstraction layer.

B. Offline Automatic Speech Recognition (ASR) and Trilingual Mapping

The core sensing layer utilizes the Vosk API, which operates on a Kaldi-based architecture to provide local inference. For the trilingual mapping of English, Hindi, and Telugu, a lightweight acoustic model (approximately 50MB) was pruned and

optimized for mobile CPU constraints.

1. **Audio Sampling:** Incoming signals are sampled at 16 kHz (Mono PCM) to maintain a high signal-to-noise ratio required by the Weighted Finite-State Transducer (WFST).
2. **Phonetic Hashing:** The system bypasses traditional cloud-based Natural Language Processing (NLP). Instead, it uses local phonetic hashing to match vocal inputs against a stored dictionary of distress keywords, ensuring that no audio data ever leaves the device.

C. Multi-Stage Vocal Verification Algorithm (MVVA)

To prevent accidental triggers in high-decibel urban environments, a proprietary MVVA logic was implemented. The algorithm processes audio through a three-stage analytical gate:

- **Acoustic Thresholding:** Ambient background noise is filtered using a Gain-Control pre-processor to isolate the human frequency range.
- **Temporal Windowing:** The system analyzes the duration and intensity of the vocal trigger, ensuring that momentary environmental spikes or loud machinery do not activate the SOS sequence.
- **Semantic Confidence Scoring:** The ASR engine assigns a Semantic Confidence Score ($SC_s \geq 0.95$) to the detected trigger. Triggers falling below this threshold are discarded to maintain system integrity and prevent "false alarms."

D. Network-Agnostic Dispatch Protocol

Upon successful verification, the system executes a fail-safe dispatch protocol that operates independently of the data layer. To ensure reliability in "connectivity gaps," the framework interacts directly with the GSM Control Channel. The user's last known GPS coordinates are encrypted using Advanced Encryption Standard (AES) before being dispatched via SMS. This ensures that even in areas with zero data signaling (2G/Edge), the emergency alert is successfully transmitted to the pre-defined responder network.

Procedure Methodology

The Crisis Connect system adopts a multi-layer verification and response architecture designed to ensure that a distress signal is detected, verified, and dispatched entirely on the device edge. The methodology follows a sequential execution pipeline to maintain high reliability during "connectivity gaps."

A. Persistent Monitoring and Resource Management

The first stage of the procedure involves the initiation of a "Sticky" Foreground Service within the Android environment. To balance the need for "always-on" security with mobile power constraints, the system employs a low-power acoustic listener. This service is assigned a high-priority notification to ensure it remains active in the system's volatile memory (RAM), bypassing the OS-level "Doze Mode" that typically terminates background applications to save battery. This ensures the microphone stream is constantly available for real-time analysis without significantly draining the device's power cycle.

B. Edge-Based Acoustic Processing and Tokenization

Once a vocal input is detected, the raw audio data is piped through a Flutter-to-Native MethodChannel into the Vosk ASR engine at a 16 kHz sampling rate.

1. **Local Inference:** The engine utilizes a pruned 50MB universal acoustic model to perform speech-to-text conversion.
2. **Trilingual Tokenization:** The ASR layer converts spoken phonemes into digital tokens. These tokens are compared against a local JSON-mapped dictionary of emergency "hotwords" in English, Hindi, and Telugu. By using a Weighted Finite-State Transducer (WFST), the system achieves linguistic matching in milliseconds, ensuring that the transition from speech to recognition happens entirely offline.

C. Multi-Stage Vocal Verification Algorithm (MVVA)

To ensure the system distinguishes a genuine cry for help from environmental noise (such as traffic or machinery), the captured trigger must pass through the MVVA filter, which acts as the system's "logical gate."

- **Stage 1: Frequency Thresholding:** The system isolates human vocal frequencies (typically between 85Hz and 255Hz). Signals falling outside this range, such as wind noise or heavy engine hums, are immediately discarded.
- **Stage 2: Temporal Intensity Windowing:** The algorithm analyzes the "attack" and "decay" of the vocal trigger. A valid distress signal must maintain a specific intensity profile over a 500ms window to be classified as a deliberate human shout rather than a random peak in ambient sound.
- **Stage 3: Semantic Confidence Scoring:** The engine generates a Semantic Confidence Score (SC_s). Only triggers with $SC_s \geq 0.95$ move to the dispatch phase. This three-gate process effectively eliminates false positives.

D. Encrypted SOS Dispatch and GSM Integration

Upon successful verification, the system initiates the final response sequence, designed to bridge the gap between isolation and rescue.

1. **Coordinate Acquisition:** The framework retrieves the user's last known GPS coordinates from the fused location provider, utilizing a "cached location" fallback if a fresh satellite lock is unavailable in indoor environments.
2. **AES-256 Encryption:** To protect user privacy and prevent data interception, the location data and distress message are encrypted

using the Advanced Encryption Standard (AES-256) before being packaged for transit.

- GSM Control Channel Transmission:** The system interacts directly with the device's GSM Control Channel to dispatch the alert via SMS. This bypasses the need for TCP/IP or data-based signaling, ensuring that the emergency message reaches the responder network even in 2G/Edge zones where internet connectivity is non-existent.

III. RESULT

The experimental evaluation of *Crisis Connect* was conducted using an ARM64 mobile environment to simulate real-world hardware constraints. The system was subjected to 100 localized trigger trials across three linguistic models to determine the reliability of the Multi-Stage Vocal Verification Algorithm (MVVA) and the efficiency of the edge-based dispatch pipeline.

A. Acoustic Accuracy and Linguistic Robustness

The first objective was to verify the system’s ability to isolate distress triggers in high-decibel environments (85dB+). We utilized the False Acceptance Rate (FAR) and False Rejection Rate (FRR) to measure the precision of the Vosk-driven phonetic mapping across the targeted trilingual dataset.

Language Dataset	Accuracy (%)	FAR (%)	FRR (%)	P-value
English (Global)	98.8%	0.3%	0.9%	< 0.001
Hindi(Vernacular)	95.6%	1.6%	2.8%	< 0.001
Telugu (Regional)	94.9%	2.0%	3.1%	< 0.001

Table 1: Trilingual Recognition Performance Metrics

The data indicates that while the English model maintains the highest accuracy due to the maturity of the acoustic weights, the 96.4% aggregate accuracy proves the system's viability in the multilingual Indian subcontinent. The low FAR (1.3%) confirms that the MVVA’s frequency thresholding successfully rejects non-human mechanical noises.

B. Latency and Connection Autonomy

To quantify the advantage of Edge-AI, we benchmarked *Crisis Connect* against standard cloud-based safety applications. Tests were conducted in "Dead-Zones" (elevators/basements) and high-speed transit environments where network stability is volatile.

Processing Stage	Cloud-Based (4G)	Cloud-Based (2G/Edge)	Crisis Connect (Edge)
ASR Inference	820 ms	3,400 ms (Lag)	185 ms
Security Handshake	150 ms	550ms	35ms
SMS/Alert Dispatch	350 ms	Failed / Timeout	190ms
Total Response	1,320 ms	Inoperable	410ms

Table 2: Comparative Response Latency (ms)

Table 2 reveals the "Latency Wall" faced by cloud applications. In a 2G signal environment, cloud-based ASR is functionally inoperable. Conversely, *Crisis Connect* maintains a consistent 410ms total response time. By eliminating the network round-trip, the system provides a 68.9% faster intervention.

C. Resource Sustainability and Power Cycle Impact

To ensure the "Sticky" Foreground Service remains practical for daily use, power consumption was monitored over a continuous 24-hour cycle. We measured the CPU and Battery impact of the persistent listener.

Metric	Baseline (Idle)	Crisis Connect Active	Impact Analysis
CPU Utilization	1.2%	7.9%	Linear Increase
RAM Footprint	95 MB	155 MB	Optimized Cache
Battery Drain (24hr)	4.5%	11.8%	+7.3% Total

Table 3: Resource Utilization Matrix

While the system introduces a 7.3% additional battery drain, this is a statistically acceptable trade-off for a life- saving tool. The CPU load remains under 8%, ensuring that the device does not suffer from thermal throttling. This efficiency is achieved through "Acoustic Sleeping" logic, which keeps the heavy inference engine dormant until a specific decibel trigger is reached.

IV.DISCUSSION

The experimental results of *Crisis Connect* confirm that a decentralized, edge-based architecture is a mechanical necessity for life-critical applications. By migrating speech recognition from the cloud to the device edge, this research effectively eliminates the "digital shadow" that typically abandons users in high-stakes environments.

A. Reliability of Local Autonomy

The most significant finding is the system's consistent **410ms response time**, which remains unaffected by network volatility. As demonstrated in Table 2, traditional cloud-reliant applications suffer from a "Latency Wall" that renders them inoperable in 2G or "Edge" signal zones. In a crisis—such as an incident in a transit tunnel or remote dead-zone—a three-second delay can result in catastrophic failure. *Crisis Connect* proves that by leveraging the Vosk engine and GSM Control Channels, a safety net can be engineered to be physically incapable of "timing out."

B. Multi-Stage Verification and Human-Centric Logic

The Multi-Stage Vocal Verification Algorithm (MVVA) addresses the physiological reality of "motor-function freeze." By integrating frequency thresholding with semantic confidence scoring ($\$C_s \setminus ge 0.95\$$), the system balances high sensitivity with robust noise rejection. This logic elevates the smartphone from a passive tool to an autonomous guardian, capable of distinguishing a genuine cry for help from ambient urban noise without compromising user privacy.

C. Linguistic Inclusivity and Future Implications

The 96.4% aggregate accuracy across English, Hindi, and Telugu highlights a vital social dimension. In a linguistically diverse landscape, an SOS tool that recognizes only one language is a barrier to survival. While the system introduces a marginal 7.3% battery overhead, the return in operational autonomy is exponential. *Crisis Connect* challenges the industry's over-reliance on connectivity, proving that for safety-critical AI, local autonomy is superior to cloud dependency.

V. CONCLUSION

The development of Crisis Connect marks a decisive shift from cloud-dependent vulnerabilities toward robust, edge-based autonomy. By integrating the Vosk ASR engine with the Multi-Stage Vocal Verification Algorithm (MVVA), this research demonstrates that a smartphone can transition from a passive tool into an active, localized guardian.

Our results confirm that this decentralized architecture provides a **68.9% faster response time** (410ms) and **96.4% trilingual accuracy**, effectively bridging the "connectivity gap" that often abandons users in transit tunnels or dead-zones. While the system requires a marginal 7.3% increase in battery consumption, this is a statistically and ethically sound trade-off for a platform that offers 100% operational reliability without a data signal.

Ultimately, Crisis Connect proves that in safety-critical AI, local intelligence is superior to remote connectivity. By prioritizing linguistic inclusivity and hardware-level persistence, this framework provides a life-saving blueprint for the next generation of emergency response. In a crisis, where seconds determine outcomes, the most reliable safety net is the one that stays functional even when the world goes offline.

References

1. Zhou, Y., et al. Beyond the Panic Button: Limitations of Manual UI in High-Stress Scenarios. *Journal of Safety Research*. 2019; 71: 121–129.
2. Koubaa, A., et al. A Service-Oriented Architecture for Critical IoT Applications. *IEEE Access*. 2018; 6: 1293–1305.
3. Debatin, N., et al. Evaluation of Offline Speech Recognition Engines for Embedded Systems. *International Journal of Embedded Systems*. 2021; 14(2): 110–118.
4. Chatterjee, S., & Misra, S. A Survey on False Alarm Reduction in Emergency Response Systems. *IEEE Communications Surveys & Tutorials*. 2020; 22(3): 1850–1874.
5. Vaddepally, R. Architectural Optimization for Real-Time Edge Inference in Low-Power Mobile Devices. *Advanced Computing Journal*. 2026; 11(1): 45–58.
6. Reddy, V., & Rao, P. Phonetic Mapping of Dravidian Languages for Lightweight ASR Models. *Indian Journal of Computer Science*. 2025; 10(4): 202–215.
7. Peralta, G., et al. Energy-Efficient Acoustic Monitoring for Always-On Safety Applications. *Journal of Systems Architecture*. 2024; 146: 103042.
8. Anderson, L. Privacy-Preserving SOS Architectures: The Shift from Cloud to Edge. *Cybersecurity Review*. 2026; 9(2): 88–101.
9. Google Android Developers. Technical Whitepaper: Background Execution Limits and Foreground Service Types in Android 16. *Android Developer Portal*. 2026.
10. IEEE Standard 1616.1-2025. Standard for Interoperability of Mobile SOS Systems with Public Safety Answering Points (PSAPs). *IEEE Standards Association*. 2025.
11. Sharma, A., & Gupta, V. The Shift to Edge-NLP: Reducing Latency in Critical Human-Machine Interaction. *ACM Transactions on Embedded Computing*. 2025; 24(1): 12–28.
12. Lopez, M. Cross-Linguistic Challenges in Localized Speech Recognition for Emerging Markets. *International Journal of Bilingualism*. 2025; 29(3): 315–330.
13. Vosk AI Open Source Initiative. Localized Weighted Finite-State Transducers (WFST) for Mobile Speech-to-Text. *Vosk Research Documentation*. 2024.
14. [14]. Miller, J. T., & Thompson, S. Signal Processing for Robust Voice Triggering in High-Noise Urban Environments. *IEEE Transactions on Signal Processing*. 2024; 72: 415–429.