LLM-Driven First Aid Assistant for Disaster Zones

GROUP 7

Team Members:

- Amal Ajay A - KTE22EC013
- Joshwin Binu - KTE22EC038
- Judin Joe Mathew - KTE22EC039
- Justin Varghese - KTE22EC040

Project Guide:

Asst. Prof. Shinoj K Sukumaran

Introduction: The Challenge

- **Kerala's Vulnerability:** Prone to floods, landslides, and other natural disasters.
- **Critical Infrastructure Failure:** Disasters often knock out power, internet, and mobile networks.
- **The Consequence:** Isolated communities are cut off from vital, life-saving information precisely when they need it most.

Our Solution: The Offline Crisis Assistant

Why *Offline*?

Cloud-Based Assistants (e.g., Siri, Alexa)

- **Require Internet**
- **Fail when networks are down**
- **Not tailored for disaster medicine**

Our Crisis Assistant

- **Fully Offline Operation**
- **Rugged and Portable**
- **Specialized for First Aid & Survival**

A rugged, portable device powered by a specialized offline LLM to provide immediate, expert-level emergency guidance without any connectivity.

Objectives

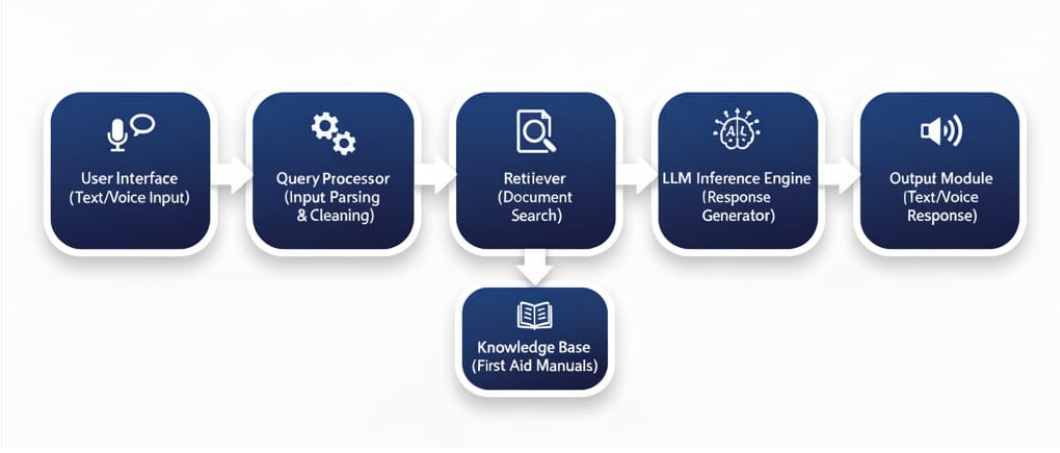
- To design and develop a rugged, portable emergency assistant device.
- To optimize and run a fine-tuned offline LLM on embedded hardware (Raspberry Pi).
- To support intuitive voice and text input for context-aware guidance.
- To provide reliable information on first aid, survival, and emergency protocols.
- To function entirely without internet dependency.

Abstract

Crisis Assistant: Offline AI for Disaster Response

In disaster-stricken areas where internet connectivity is often unavailable, access to timely and reliable emergency information is critical. This project presents the design and development of a rugged, portable device powered by a fine-tuned Large Language Model (LLM) that operates entirely offline. The system provides context-aware assistance on first aid, CPR, wound care, and survival techniques via text or speech input. Trained on verified content from the WHO and Red Cross, the LLM ensures reliable responses. Built on a Raspberry Pi with a microphone, speaker, and battery pack, this project demonstrates how embedded AI can strengthen disaster preparedness and response without relying on cloud services.

BLOCK DIAGRAM



figureProposed System Block Diagram

Literature Review (1/2)

	Paper & Key Points	Relevance to Our Project
	Zheng et al. (2024) *A Review on Edge Large Language Models: Design, Execution, and Applications* Techniques for running LLMs on low-power devices. **DOI:** 10.1145/3719664	Provides the foundational methodology for model optimization (quantization, pruning) crucial for our Raspberry Pi deployment.
	Basit et al. (2024) *MedAide: Leveraging Large Language Models for On-Premise Medical Assistance on Edge Devices* Offline medical chatbot using LoRA fine-tuning and tiny-LLMs for edge deployment. **arXiv:** 10.48550/arXiv.2403.00830	Directly validates our core concept of an offline, reliable medical assistant, informing our fine-tuning strategy.

Literature Review (2/2)

	Paper & Key Points	Relevance to Our Project
	Goecks & Waytowich (2023) *DisasterResponseGPT: Large Language Models for Accelerated Plan of Action Development in Disaster Response Scenarios* Generates disaster response plans from scenarios using LLMs. **arXiv:** 10.48550/arXiv.2306.17271	Informs the design of our system's logic for structured emergency guidance and planning in chaotic environments.
	Soto-Vergel et al. (2023) *Transforming ground disaster response: Recent technological advances, challenges, and future trends for rapid and accurate survivor detection* Review of edge / sensor / TinyML techniques for on-site survivor detection and low-power deployments. **DOI:** 10.1016/j.ijdrr.2023.104094	Strengthens our TinyML/edge-implementation claims by showing real-world approaches to low-power on-site inference in disaster zones.

Phase 1 Work Plan (July 8 – September 30)

Stage	Dates	Key Tasks
1. Ideation	Jul 8 – 21	Brainstorming, topic finalization, preliminary literature survey on edge AI and disaster tech.
2. Scoping	Jul 22 – Aug 4	Finalize objectives, draft hypothesis, initial block-level design.
3. Planning	Aug 5 – 18	Complete system architecture, assign team roles (HW/SW), list requirements.
4. Feasibility	Aug 19 – Sep 12	Resarched about feasability of the project.
5. Refinement	Sep 12 – 22	Iterate on design based on research results intiating the software side of the workflow
6. Reporting	Sep 23 – 30	Prepare and review Phase 1 final report and presentation.

References


World Health Organization, *First Aid Guidelines*, 2023. https://www.who.int


International Federation of Red Cross, *First Aid Manual*, 2022. https://www.ifrc.org


Vicuna Team, "Vicuna: An Open LLM Dialogue Assistant," 2023. https://lmsys.org


ACM TIST / ArXiv, *LLMs for Emergency Response*, 2023. https://arxiv.org/abs/2306.08956


Basit et al., *MedAide*, 2024. https://arxiv.org/abs/2403.00830


Goecks & Waytowich, *DisasterResponseGPT*, 2023. https://arxiv.org/abs/2306.17271

Thank You

Questions?

  github.com/group-7