KhetAI

[Github Repo]

Team – NPnotP
Sri Sai Shanmukh raj Malireddi
IV Year CS Undergrad, IIT Madras

KhetAI – Smart Farming Assistant

1. Introduction

Agriculture remains one of the most vital sectors worldwide, yet farmers often face challenges in accessing timely, location-specific information.

KhetAI is a smart farming assistant designed to provide accurate, real-time advisory tailored to farmers' local contexts. It leverages AI-driven **natural language understanding, speech recognition, and geospatial data** to deliver actionable insights across multiple domains, including weather, irrigation, seed varieties, market prices, government schemes, and more.

Our solution is implemented as a web-based interactive platform using **Streamlit for frontend presentation and a local LLaMA-based language model backend** for AI-driven recommendations. The platform provides both text and voice interfaces, making it accessible for users with limited literacy.

2. Proposed Implementation

2.1 System Overview

KhetAI integrates several components:

- **Frontend Interface**
  Implemented using Streamlit with a **dark-themed, modern UI** that supports chat-like interactions. Users can select their location, ask questions via text or voice, and receive detailed responses.

- **Geospatial Module**
  A CSV-based database of villages, including **state, district, mandal, pincode, latitude, and longitude**, allows location-aware responses. This ensures advice is contextualized for local climatic and soil conditions.

- **Voice Interface**
  Using **Streamlit Mic Recorder** and **SpeechRecognition**, KhetAI can transcribe spoken questions into text, enabling a hands-free experience.

- **AI Backend**
  Leveraging **LLaMA via Ollama CLI**, the system processes user queries, classifies them into categories, and generates precise, actionable responses. Queries are dynamically augmented with location metadata for accurate contextual understanding.

- **Conversation Management**
  The application maintains a **session-based chat history**, allowing users to view past interactions. The interface is designed to replicate modern AI assistants with **speech and text integration**, categorized chat bubbles, and scrolling history.

- **Data Integration Layer**
  Aggregates multiple agricultural datasets including soil health, crop calendars, irrigation advice, market prices, government schemes, pest and disease data, and weather normals. These datasets are filtered by user location to supply comprehensive background for personalized responses.

- **Contextual Prompt Builder**
  Synthesizes geospatial, environmental, and agronomic data to construct rich context prompts, enabling the AI backend to deliver relevant, location-aware advice tailored to the farmer's specific conditions and crops.

- **Real-time Streaming Responses**
  Implements streaming of AI-generated answers to the user interface, providing immediate feedback through incremental updates that mimic natural conversational flow and reduce perceived latency.

- **Accessibility and Usability Features**
  The system incorporates visual accessibility considerations, responsive layout, meaningful input labels, and keyboard navigation support, enhancing user experience across diverse user groups.

- **Extensibility and Modular Architecture**
  Designed with modular components facilitating easy integration of new data sources,

alternative AI models, and additional user interface features, ensuring scalability and adaptability for evolving agricultural needs.

- **Offline Capability and Data Privacy**
  By running AI models locally through Ollama CLI, KhetAI supports environments with limited connectivity and bolsters data privacy by retaining user data and queries on-device.

- **Robust Error Handling and User Feedback**
  Features comprehensive exception handling and user notifications for issues such as data unavailability or speech recognition errors, maintaining transparency and user trust.

- **Continuous Update Mechanism**
  Supports periodic updates of integrated datasets and AI models to incorporate the latest agricultural research, policy changes, and market conditions, keeping the advisory current and reliable.

2.2 Technical Architecture

The KhetAI architecture consists of:

Frontend Layer (Streamlit)

- Utilizes custom HTML and CSS to deliver a sleek, dark-themed, modern user interface focused on usability.
- Interactive widgets include a select box for precise location selection, a chat input box for text queries, and a microphone button enabling voice input.
- Supports real-time response display with visually distinct, categorized chat bubbles that enhance conversational clarity and user experience.

Processing Layer

- Voice-to-Text Conversion: Employs Google's Speech Recognition API wrapped in the SpeechRecognition package to transcribe spoken questions into text, enabling hands-free interaction.

- Query Classification: Uses a lightweight secondary prompt to the LLaMA model to classify user queries into predefined agricultural categories, optimizing subsequent response generation.
- Prompt Construction: Dynamically composes rich prompts by integrating user input with comprehensive geospatial and contextual metadata derived from multiple datasets, ensuring responses are tailored and contextually accurate.

AI Layer (LLaMA via Ollama)

- Executes natural language understanding and response generation via locally hosted LLaMA models accessed through the Ollama CLI interface.
- Utilizes domain-adapted prompt templates and context windows to maximize accuracy and relevance in agricultural advisory.
- Capable of handling diverse question types including factual clarifications, procedural guidance, advisory recommendations, and predictive insights.

Data Layer

- Maintains multiple CSV-based datasets encompassing village-level geospatial and administrative data, soil health, crop calendars, irrigation advisories, market prices, government schemes, pest/disease info, and weather normals.
- Implements caching mechanisms for efficient data loading and retrieval, ensuring high system performance and responsiveness, even with extensive datasets.

2.3 Key Design Decisions

Local AI Model (LLaMA)

- Selected for offline compatibility and fast inference.
- Avoids reliance on external APIs to reduce latency and enhance privacy.

Streamlit as Frontend

- Lightweight, deployable on web servers or cloud notebooks.

- Supports interactive widgets for both text and voice input.

Session-Based Chat History

- Allows persistent context in each user session.
- Categorized messages provide clarity and help users track past advisory.

Voice-First Input

- Prioritized accessibility for rural users with limited literacy.
- Inline microphone button replicates popular AI assistant interfaces for familiarity.

Geospatial Context Injection

- Every user query is augmented with precise location data.
- Ensures recommendations are locally relevant for weather, irrigation, seeds, and market conditions.

3. Limitations and Known Issues

Despite its robust architecture, KhetAI has several known limitations:

Model Limitations

- LLaMA may occasionally produce incorrect or incomplete answers.
- Domain-specific advice may require continuous tuning and prompt engineering.

Data Granularity

- Current geospatial dataset is limited to village-level granularity.
- Microclimatic variations within villages may not be fully captured.

Voice Recognition Accuracy

- Dependent on microphone quality and ambient noise.
- May fail with regional accents or dialects.

Offline Constraints

- While local AI ensures privacy, it is computationally intensive for large queries.
- High-end CPU/GPU resources may be needed for larger models.

Scalability

- Streamlit's single-user session model limits concurrent multi-user deployments.
- Requires additional backend setup for larger-scale adoption.

4. Future Work

Several avenues exist to enhance KhetAI:

Expanded Dataset

- Include soil quality, rainfall, crop history, pest reports, and more.
- Integrate government agricultural databases for richer advisory.

Model Improvement

- Fine-tune LLaMA on agriculture-specific corpora.
- Explore multi-lingual support for regional languages.

Real-Time IoT Integration

- Connect to farm IoT devices for live soil moisture, temperature, and humidity data.
- Enable automated irrigation or fertilization recommendations.

Mobile Deployment

- Build a lightweight mobile app for offline-first usage.
- Push notifications for weather alerts and crop management advice.

Multi-User Support

- Implement a backend with session handling for multiple concurrent users.
- Maintain personalized advisory and history per user account.

5. Novel Aspects of the Implementation

KhetAI differentiates itself through:

Voice-First AI Interaction

- Inline microphone interface similar to modern AI chat applications.
- Supports farmers with literacy or technological barriers.

Contextual Prompt Engineering

- Augments every query with detailed location metadata.
- Provides highly targeted advice beyond generic responses.

Modern, Accessible UI

- Dark-themed, responsive chat interface with categorized message bubbles.
- Scrollable history improves user experience for multi-turn conversations.

Local, Privacy-Preserving AI

- LLaMA runs locally, reducing dependency on internet connectivity.
- Ensures sensitive farm data remains private.

6. Conclusion

KhetAI represents a significant step forward in democratizing **smart farming advisory**.

By combining local AI, geospatial context, voice-first interaction, and a modern, accessible interface, KhetAI empowers farmers with actionable insights tailored to their specific needs.

While the current implementation has some limitations in scalability, voice accuracy, and data granularity, it establishes a strong foundation for future expansion and real-world adoption.

The system exemplifies **practical, context-aware AI deployment**, and serves as a blueprint for intelligent agricultural assistance accessible to rural communities.