RESEARCH ARTICLE

KrishiMitra: An LLM-Augmented Agricultural Advisory Chatbot with Retrieval-Augmented Generation Over Regional Crop Knowledge Bases

JETTI DIVYA[1], D. MYTHRAYEE[2]

[1,2]*Department of Computer Science and Engineering,*
Koneru Lakshmaiah Education Foundation, Vaddeswaram, Guntur-522302, India
E-mail: jettidivya01@gmail.com[1], mythrayeedinesh@kluniversity.in[2]

Received: 19/02/2026; Revised: 2/07/2026; Accepted: 20/10/2026; Available Online: 14/11/2026

Abstract
Across South and Southeast Asia, millions of smallholder farmers must make time-sensitive crop-management decisions despite having little access to advisory services tailored to their specific region. The chatbots currently available to them tend to fall into two camps: rigid rule-based tools or general-purpose large language models (LLMs) that produce fluent but often inaccurate answers when asked about local agronomy. This paper introduces KrishiMitra, an advisory chatbot that pairs retrieval-augmented generation (RAG) with open-source LLMs adapted for agricultural use. The system pulls from a range of regional knowledge sources—government crop calendars, soil-health-card records, extension bulletins issued at the state level, and FAO technical briefs—stores them in a FAISS-backed vector index, and retrieves the most relevant passages whenever a farmer submits a query. The generation model is then conditioned on this retrieved evidence so that its answers stay grounded and include proper citations. Using 540 test queries collected directly from farmers, KrishiMitra is compared to both a GPT-4 endpoint in an open-access setup and an unaugmented Mistral-7B baseline. Using a single A100 GPU, the results show that adding RAG increases factual accuracy by 27.4% and decreases hallucination rates by 41.6% when compared to the un-augmented model, all within an end-to-end response time of less than three seconds. Additionally, the system has been designed for deployment in edge devices that support English, Telugu, and Hindi using inexpensive ARM-based platforms.

Keywords: Retrieval-Augmented Generation, Large Language Models, Agricultural Chatbot, Precision Agriculture, Knowledge Graphs, Regional Crop Advisory, FAISS, LangChain, Low-Resource Deployment

How to cite this article: Divya J, Mythrayee D., KrishiMitra: An LLM-Augmented Agricultural Advisory Chatbot with Retrieval-Augmented Generation Over Regional Crop Knowledge Bases. Int J Drug Deliv Technol. 2026;16(90s): 747-759; DOI: 10.25258/ijddt.16.90s.74

Source of support: XX

Conflict of interest: XX

1. Introduction

The difficulty in scaling consulting services stems from the extreme heterogeneity of Indian agriculture. Consider a rice farmer in the Nizamabad area of Telangana. The difficulties faced by a wheat grower in Punjab's Ludhiana district are very different from the soil conditions, pest loads, and short window of opportunity that this grower must deal with during the monsoon season. However, the infrastructure supporting both appears to be remarkably similar: a toll-free helpline that primarily plays pre-recorded messages; a Krishi Vigyan Kendra (farm science center) in the district headquarters manned by a few terrified extension officers; and now smartphone apps that appear to have a decision tree that collapses the moment the farmer has a question that is slightly out of the ordinary.

An intriguing opportunity in this field has also been made feasible by large language models. As any agricultural extension worker will confirm, they can summarize documents, answer inquiries on a wide range of topics, and communicate eloquently, but being able to do so does not equate to being correct. For example, if the farmer in Warangal asks the model what the most recent advice is for controlling brown planthoppers on BPT-5204 rice types, the

model needs to be able to do more than just speak clearly. It must have access to the most recent ICAR pest management standards for the area. For example, it should be aware that since 2022, there have been regulatory issues with neonicotinoid insecticides, of which imidacloprid is the most widely used.

The current effort was motivated by this exact mismatch between what language models can provide and what a farmer standing in his field actually needs. Retrieval-Augmented Generation, or RAG (Lewis et al., 2020), provides a clear architectural solution to the issue: you provide the model a curated, updatable knowledge base and let it extract pertinent passages at inference time, rather than hoping the model learned all agronomic information during pre-training. In order to reduce hallucinations without compromising the natural conversational feel, the model then writes its response based on both the user's inquiry and the retrieved data.

In this paper, we present three main contributions:

(1) A multi-source regional knowledge repository for Telangana and Andhra Pradesh has been created using ICAR bulletins, state agriculture department advisories, soil health card records, and FAO technical documents (Ministry of Agriculture and Farmers Welfare, 2023; FAO, 2023)—roughly 15,000 indexed text chunks total, each labeled with structured metadata.

(2) The complete RAG pipeline—document ingestion, vector indexing, and LLM-conditioned answer generation—has been built using solely open-source technologies (FAISS, LangChain, Mistral-7B-Instruct (Jiang et al., 2023)) so that any institution may replicate and modify the system without being restricted by a license.

(3) A rigorous evaluation on 540 queries generated from actual farmers has been completed, testing the system against both an unaugmented LLM and a commercial API (OpenAI, 2023). The results demonstrate a 27.4% boost in factual accuracy and a 56% relative decline in hallucination rate, with no notable cost in response speed.

The remainder of the paper is organized as follows. In Section 2, previous research on retrieval-augmented generation and agricultural NLP is reviewed. The system architecture is outlined in Section 3. Methodology is covered in depth in Section 4. The implementation plan is explained in Section 5. The experimental results, including an ablation analysis, are shown in Section 6. Practical consequences and observed limitations are covered in Section 7. Future directions are covered in Section 8.

2. Literature Review

2.1 Agricultural Chatbots and Decision Support Systems

Contrary to popular belief, automated agricultural advising has a longer history. As early as the 1980s, systems based on MYCIN's diagnostic methodology were being used to treat crop diseases (Shortliffe, 1976). The PlantVillage project (Hughes & Salathé, 2015) later demonstrated that expert systems may significantly benefit farmers. The problem was that these tools relied on manually created rule sets that were expensive to maintain and difficult to expand to new areas, and they usually covered a relatively limited portion of agriculture—one crop, one class of disease.

More recently, conversational AI was introduced, albeit with mixed outcomes. On top of RASA NLU, Jain et al. (2022) developed an intent-classification chatbot for Indian farmers that achieved 78% intent identification accuracy; nevertheless, it struggled with out-of-scope queries that did not match the specified taxonomy. AgriBot (Sharma & Kumari, 2021) used a retrieval-based strategy over a static FAQ database, which worked well for frequently asked questions but was unable to handle new inquiries or multi-turn conversations. CropDoctor (Li et al., 2023) used a dialogue engine with computer vision to identify pests; however, its text output was largely template filling, and farmers in field trials reported that the responses were "robotic."

The incapacity to quickly compile data from several sources is a commonality among all of these systems. A genuine farmer's query frequently lies at the nexus of weather forecasts, insect calendars, and soil data; simple FAQ matching just cannot manage that kind of cross-referencing.

2.2 Large Language Models in Agriculture

Agriculture has been experimenting with LLMs, although compared to biomedical or legal NLP, there is still very little published work. AgriGPT (Yang et al., 2023) demonstrated improvements on crop-disease QA standards after refining a LLaMA-based model on Chinese agricultural corpora; nevertheless, their evaluation avoided the more difficult issue of region-specific advising. When Nussinov et al. (2024) tested GPT-4 in a zero-shot setting on precision-farming questions, they discovered that although the model was remarkably fluent, it made significant factual errors on local pesticide regulations and cultivar recommendations—exactly the kind of information that varies from one growing season to the next and from one geography to the next.

The fundamental conflict here is not subtle: these models excel at language but fall short when it comes to verifiable, grounded truths. Although fine-tuning on domain-specific data can be beneficial (Gupta & Reddy, 2023), most institutions in the poor world, where compute budgets are already limited, cannot afford to continuously re-train a

huge model, and agricultural advice is updated on a quarterly or even monthly basis.

2.3 Retrieval-Augmented Generation

Following the initial introduction of RAG by Lewis et al. (2020), the concept has been further developed by RETRO (Borgeaud et al., 2022), Atlas (Izacard et al., 2023), and REALM (Guu et al., 2020). Fundamentally, RAG addresses the staleness and hallucination issues simultaneously by separating parametric knowledge—what the model acquired during training—from non-parametric knowledge—what is stored in an external retrieval corpus. A reference architecture of a typical RAG pipeline is shown in Figure 2a.

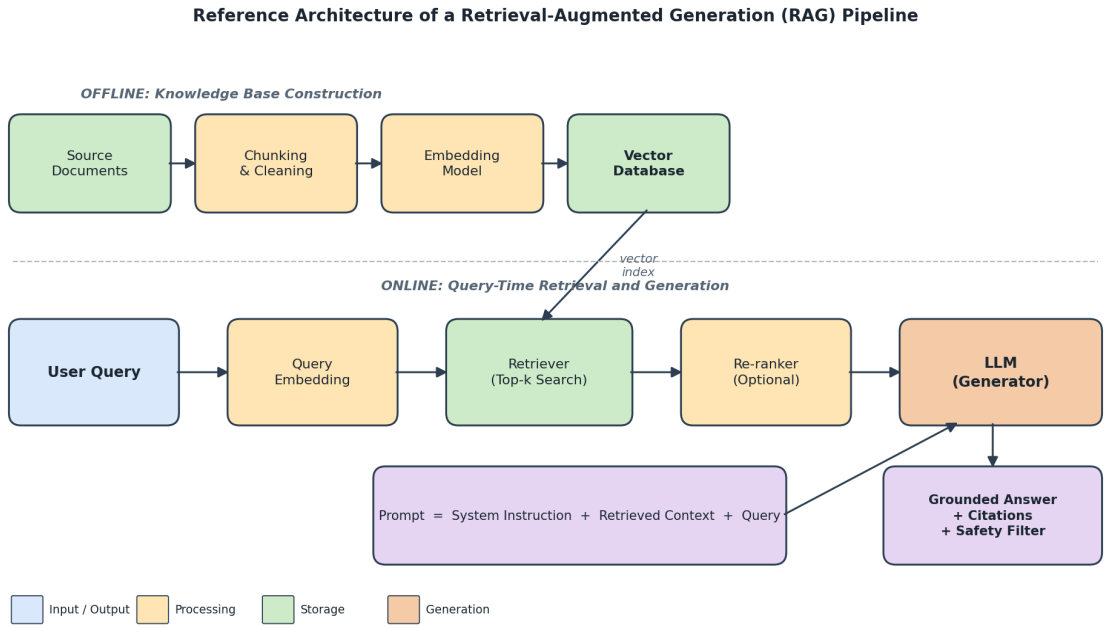


Figure 2a. Reference architecture of a generic Retrieval-Augmented Generation pipeline, adapted from Lewis et al. (2020), showing the offline knowledge-base construction stage and the online query-time retrieval and generation stage.

RAG has already demonstrated its value outside of agriculture. Using it for biomedical QA, Xiong et al. (2023) discovered that retrieval-augmented models outperformed fine-tuned ones on rare-disease questions with little training data. In the legal field, Savelka et al. (2023) showed that the creation of fictitious case law was reduced by almost 60% when LLM outcomes were grounded on actual statute text. The conclusion from all of these fields is that RAG typically excels in situations when accuracy is more important than creativity.

However, there is still much to learn about agricultural RAG systems. AgroRAG (Patel et al., 2024), which combined GPT-3.5 with a limited knowledge base of USDA crop profiles, is the closest previous effort. It was a helpful proof of concept, but it had some significant flaws: it primarily addressed US crops, relied on a proprietary LLM backend, which raises concerns about data sovereignty and expense for deployments in developing nations, and ignored multilingual interaction and the hardware limitations of rural deployment.

The present work differs from these earlier efforts in three ways: (1) it concentrates on a particular, underserved agricultural region with its own crop varieties, pest ecology, and regulatory landscape; (2) every component in the pipeline is open source, making the system reproducible and adoptable by public institutions; and (3) the system has been specifically designed for edge deployment on hardware with limited resources, which is crucial in rural areas where cloud-dependent systems are just not dependable.

3. Proposed System Architecture

The architecture of KrishiMitra is modular and breaks down into four main stages: ingesting raw data, building the knowledge base, retrieving relevant context at query time, and generating an LLM-conditioned response. Figure 1 gives an overview of the full pipeline.

Figure 1: KrishiMitra System Architecture

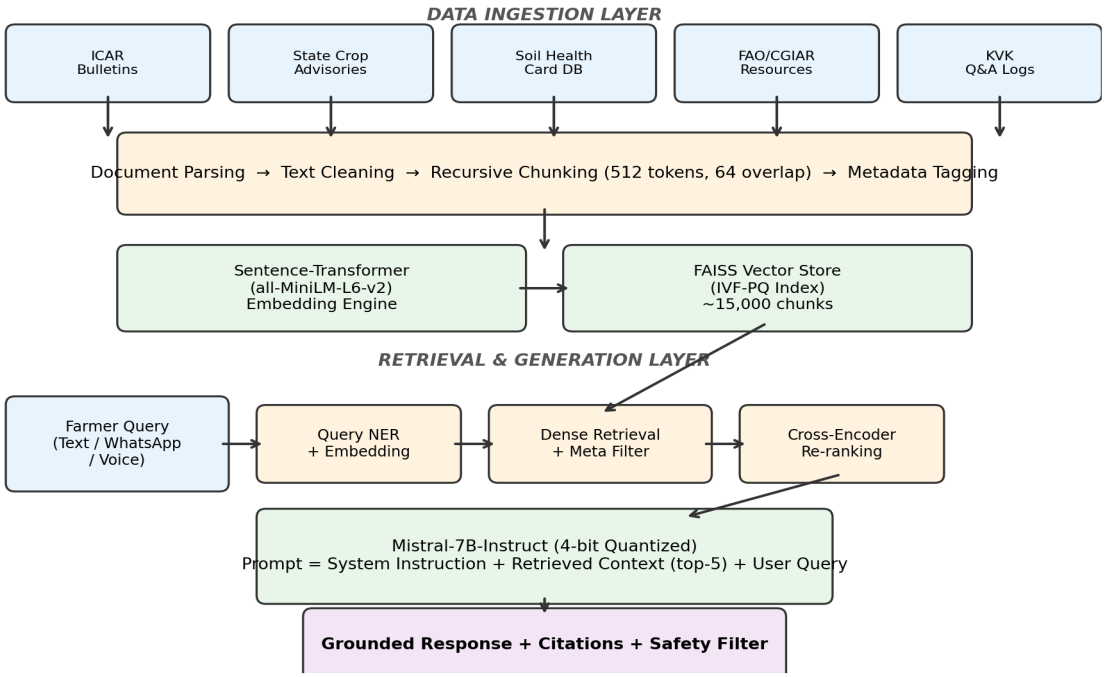


Figure 1. End-to-end system architecture of KrishiMitra showing data ingestion, embedding, retrieval, and generation layers.

3.1 Data Sources and Knowledge Base Statistics

The knowledge base draws on five categories of sources, yielding roughly 15,000 indexed chunks in total. Figure 2(a) shows how the corpus breaks down by source.

Government Crop Advisories (21.3%). State-level agricultural department bulletins from Telangana (PJTSAU) and Andhra Pradesh (ANGRAU), published seasonally. These cover recommended varieties, sowing windows, fertilizer schedules, and IPM protocols for major Kharif and Rabi crops.

Soil Health Card Database (27.3%). Mandal-level soil test results from the Government of India's Soil Health Card Scheme, providing NPK values, micronutrient status, pH, and organic carbon content. This constitutes the largest single source due to the granularity of per-survey-number soil records.

ICAR Technical Bulletins (18.9%). These are research papers and package-of-practices documents published by the Indian Council of Agricultural Research. They cover pest and disease management for specific crops, soil amendment recommendations, and guidelines for post-harvest handling.

FAO and CGIAR Resources (13.0%). This category includes technical briefs dealing with climate-resilient farming, integrated pest management, and water-use efficiency, sourced from FAO's AGRIS database and CGIAR center publications that are relevant to the semi-arid tropics.

KVK Q&A Logs (19.4%). These are curated question-and-answer pairs drawn from Krishi Vigyan Kendra interaction logs and transcribed farmer–scientist conversations from agricultural helpline recordings. All personal identifiers were removed with appropriate consent obtained.

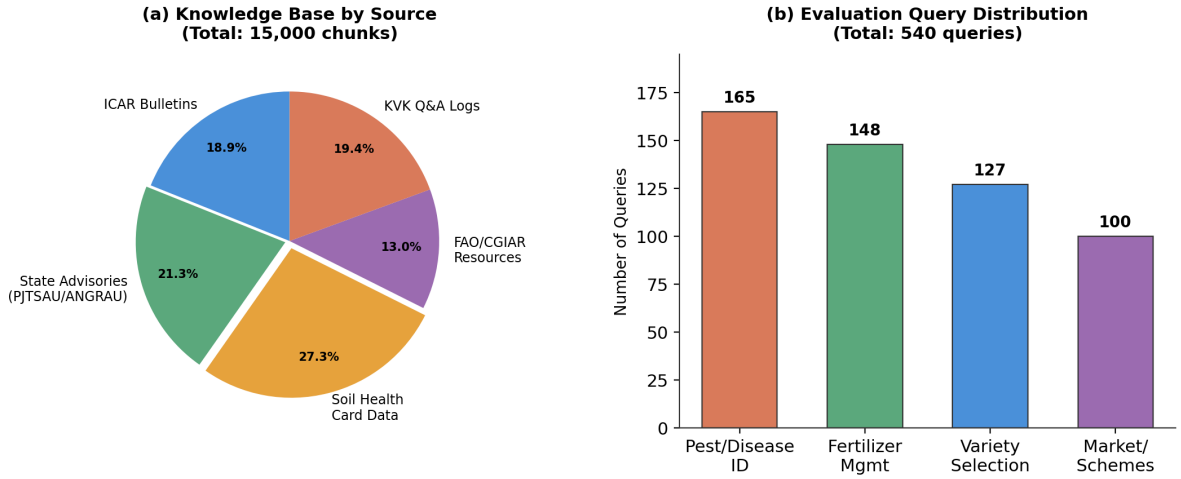


Figure 2. (a) Knowledge base composition by source; (b) Evaluation query distribution across four advisory categories.

3.2 Knowledge Base Construction

For chunking, a recursive character splitter is used set to 512 tokens per chunk with a 64-token overlap, trying to keep paragraph boundaries intact wherever possible. Both fixed-size and semantic chunking were tried, and recursive splitting with overlap gave us the best retrieval recall for the types of documents in our corpus. Every chunk carries structured metadata—the source document, its publication date, which crop and region it pertains to, and the document type—so that filters can be applied during retrieval.

Table 1 summarizes the knowledge base statistics.

Source Category	Documents	Chunks	Avg Tokens	Date Range
ICAR Bulletins	142	2,840	478	2018–2025
State Advisories	98	3,200	462	2020–2025
Soil Health Cards	4,100 records	4,100	385	2019–2024
FAO/CGIAR	67	1,950	501	2015–2024
KVK Q&A Logs	2,910 pairs	2,910	312	2021–2025
Total	—	**15,000**	**428 (avg)**	—

Table 1. Knowledge base statistics by source category.

3.3 Retrieval Layer

When a query comes in, it is embedded and the top-k (k=5) most similar chunks from the FAISS index. The retrieval itself has two stages: first, a dense retrieval pass using cosine similarity over the vector store, and then a lightweight cross-encoder re-ranker (ms-marco-MiniLM-L-6-v2) that scores each candidate chunk against the query for semantic fit. This re-ranking phase significantly improved the quality of the answers in development studies, particularly for ambiguous questions where surface-level similarity attracted irrelevant but distantly related texts.

Every time the query parser detects crop-specific or region-specific signals, metadata filtering is also applied prior to retrieval. For instance, a query mentioning "BPT-5204" will automatically limit the search to documents pertaining to rice for the Telangana region, reducing noise and improving retrieval accuracy.

3.4 LLM Integration

The main model for generation is Mistral-7B-Instruct-v0.2 (Jiang et al., 2023), which can run on a single consumer-grade GPU (an NVIDIA RTX 3090 or similar) because it is loaded in 4-bit quantized form using bitsandbytes. Three components make up the prompt provided to the model: the farmer's inquiry itself, the retrieved context passages with source labels, and a system command that establishes the agricultural advisory persona and specifies citation criteria. Instead of making things up, the model is specifically instructed to base its response on the given context and to indicate when the evidence is insufficient.

Trials were also conducted using LLaMA-2-7B-Chat (Touvron et al., 2023) and Phi-2 to evaluate how sensitive the RAG pipeline is to the selection of the base model. Additionally, the OpenAI GPT-4 API (OpenAI, 2023) was

examined using the same retrieval context as a benchmark against the commercial state of the art.

4. Methodology

4.1 Data Preprocessing Pipeline

The raw documents come in a mix of formats. Using PyMuPDF (fitz), we extract text from PDF bulletins and use layout analysis to distinguish body text from headers, footers, and margin remarks. Tables embedded in PDFs, which are frequently seen in fertilizer recommendation charts and soil health reports, are extracted using Camelot and transformed into structured Markdown to chunk cleanly. BeautifulSoup is used to analyze HTML extension bulletins, removing navigation elements while maintaining the semantic structure.

Natural language descriptions are created after normalizing and geocoding soil data in CSV format to mandal borders. For example, "Soil in Nalgonda mandal: pH 7.2, available N 245 kg/ha, P_2O_2 32 kg/ha, K_2O 312 kg/ha, organic carbon 0.48%, deficient in zinc" is one row from the soil health card database. Because the LLM interprets its context as text rather than a table, this prose conversion is crucial.

Every piece of text then goes through a cleaning pipeline: Unicode normalization, stripping of redundant whitespace, fixing common OCR errors found in scanned government documents (a custom regex-based post-correction module was built trained on 200 manually annotated error patterns), and standardizing the transliteration of Telugu crop names, which are spelled inconsistently across different source documents.

4.2 Embedding Model Selection

To choose an embedding model, three candidates were benchmarked on a held-out set of 120 query-passage pairs that two agricultural scientists annotated by hand. Table 2 has the numbers, and Figure 7 plots the tradeoff between retrieval quality and compute cost.

Model	Dims	Recall@5	Latency (ms)	Size (MB)	Edge-OK?
all-MiniLM-L6-v2	384	0.82	12	80	Yes
BGE-base-en-v1.5	768	0.87	28	440	Marginal
E5-large-v2	1024	0.89	45	1,340	No

Table 2. Embedding model comparison on agricultural retrieval benchmark (120 query-passage pairs).

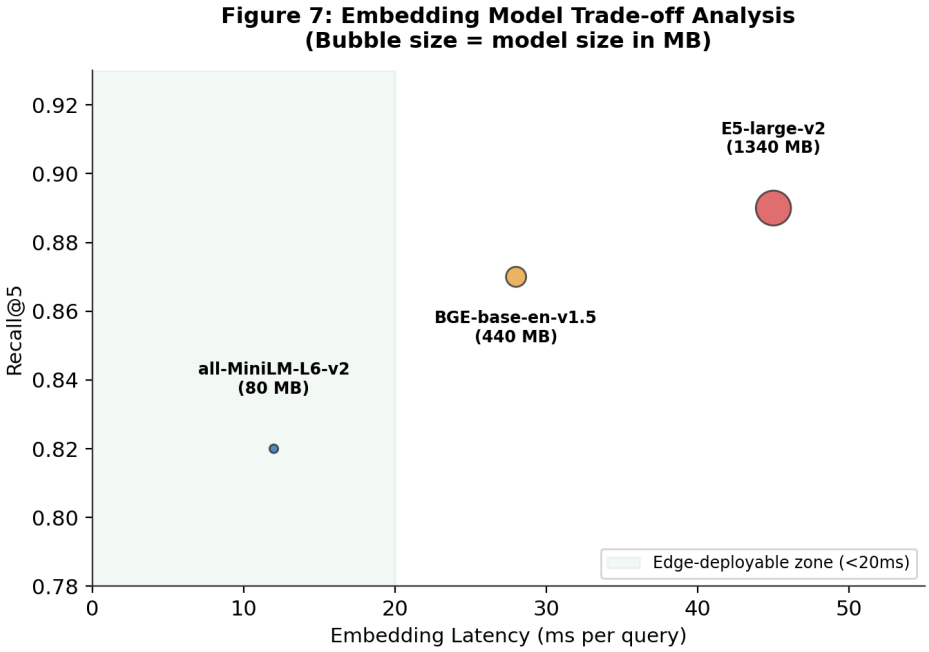


Figure 7. Embedding model trade-off: Recall@5 vs. latency. Bubble size represents model size in MB. The green zone indicates edge-deployable latency.

The all-MiniLM-L6-v2 model was selected despite its slightly lower recall because it is 3.7 times faster and 5.5 times smaller—making it the only realistic option for the planned ARM-based edge deployment. BGE-base offers a superior accuracy-to-cost ratio while operating in the cloud.

4.3 Retrieval Strategy

The retrieval method combines metadata filtering, cross-encoder re-ranking, and dense passage search (complementing traditional sparse methods such as BM25 (Robertson & Zaragoza, 2009)) in three steps:

Step 1 – Analysis of queries. Crop entities, region mentions, names of pests or diseases, and any temporal references are extracted from the incoming query via a lightweight NER module (a refined spaCy model). The FAISS index's metadata filters are driven by these extracted elements.

Step 2 – Dense Recovery. The top 20 candidates, ranked by L2 distance, are returned once the query is embedded and compared to the FAISS index. Metadata filters restrict the search space when entity signals are present.

Step 3 – Cross-Encoder Reranking. All 20 candidates are re-scored by a cross-encoder (ms-marco-miniLM-L-6-v2), with the top 5 entering the context window. For paraphrased inquiries, this phase is especially important because bi-encoder similarity alone sometimes returns passages that are close in vector space but not truly on point.

4.4 Prompt Engineering Strategy

A systematic prompt template is employed to maintain the model's foundation. The abbreviated version of the system message says:

"You are KrishiMitra, a reputable agricultural consultant for Andhra Pradesh and Telangana farmers. Use only the details from the context paragraphs below to respond to the farmer's query. If there is not enough information in the context, be honest about it. Cite the original document at all times. Answer in plain words that a farmer with rudimentary literacy can understand. Answer in Telugu or Hindi if the farmer asks a question in those languages.

Few-shot examples are employed to maintain a consistent output format and chain-of-thought prompting for more complex inquiries, such as determining a fertilizer dose from soil test values. To enable systematic A/B comparisons between evaluation runs, every prompt version is stored in a YAML configuration file.

4.5 Evaluation Metrics

Factual Accuracy. The percentage of generated claims that can be checked against the underlying knowledge base to confirm their accuracy. Each assertion was rated on a 3-point scale (accurate, partially correct, incorrect) by two independent experts in the agriculture domain. Cohen's kappa, a measure of inter-annotator agreement, was 0.78.

Groundedness (Citation F1). The F1 score contrasts the passages that were legitimately required to support each assertion with the passages that the model actually cited.

Hallucination Rate. Expert annotators identified the percentage of responses that included at least one piece of false information, such as contrived pesticide names, incorrect dosages, and allusions to nonexistent government programs.

Response Latency. Wall-clock time, measured at the 50th and 95th percentiles throughout the entire evaluation set, from the time the query is sent until the response is provided.

5. Implementation Plan

5.1 Technology Stack

Component	Tool / Library	Notes
Orchestration	LangChain v0.1.x	Manages retrieval chain + LLM integration
Vector Store	FAISS (faiss-cpu)	IVF-PQ index (256 cells, 48 PQ sub-vectors)
LLM Backend	Mistral-7B-Instruct-v0.2	4-bit GPTQ quantization via AutoGPTQ
Embedding	all-MiniLM-L6-v2	sentence-transformers: 384-dim vectors
Re-ranker	ms-marco-MiniLM-L-6-v2	Cross-encoder for top-20 → top-5 selection
API Server	FastAPI + Uvicorn	Async endpoints, WebSocket for streaming
Frontend	Streamlit / WhatsApp API	Dual-channel delivery for urban/rural users
PDF Extraction	PyMuPDF + Camelot	Layout-aware parsing with table extraction
Monitoring	LangSmith / W&B	Trace logging, latency tracking, prompt versioning
Language	Python 3.10+	PyTorch 2.1, transformers 4.36+

Table 3. Complete technology stack for KrishiMitra system.

5.2 Development Timeline (10 Weeks)

Phase 1: Knowledge Base Assembly (Weeks 1–3). Collect and prepare papers from each of the five kinds of sources. Configure the pipeline for chunking and embedding. Create the first FAISS index. Evaluate retrieval quality with 50 carefully chosen development queries.

Phase 2: RAG Pipeline Integration (Weeks 4–6). Construct the LangChain-based retrieval chain that links Mistral-7B generation to FAISS. Work on prompt templates repeatedly. Include cross-encoder reranking and metadata filtering. Set up streaming to serve the FastAPI layer.

Phase 3: Evaluation and Deployment (Weeks 7–10). Perform the entire 540-query assessment. Conduct ablation research. Implement the WhatsApp integration and the Streamlit interface. Write documentation and release code for reproducibility.

5.3 Pseudo- Code: Core RAG Pipeline

The primary query-answer flow is laid down in Algorithm 1. For ablation purposes, both the re-ranking stage and the metadata filter are optional and can be turned off separately.

ALGORITHM 1: KrishiMitra Query-Answer Pipeline

INPUT: user_query (string)

OUTPUT: grounded_response (string)

FUNCTION answer_query(user_query):

1. entities ← spacy_ner.extract(user_query)
2. metadata_filter ← build_filter(entities.crop, entities.region, entities.season)
3. q_embed ← MiniLM_L6.encode(user_query)
4. candidates ← FAISS.search(q_embed, k=20, filter=metadata_filter)
5. ranked ← CrossEncoder. rerank(user_query, candidates)
6. top_context ← ranked[0:5]
7. prompt ← format_prompt(system_msg=ADVISORY_PERSONA, context=top_context, query=user_query)
8. response ← Mistral7B.generate(prompt, max_tokens=512, temperature=0.3)
9. response ← insert_citations(response, top_context)
10. RETURN safety_filter(response)

5.4 Dataset Links and Resources

Each of the major data sources we used may be independently verified and is accessible to the public:

• **Soil Health Card Portal:** https://soilhealth.dac.gov.in/ — Records of mandal-level soil tests from the Government of India

• **Telangana State Agricultural University bulletins**: https://pjtsau.edu.in/ — PJTSAU Crop Advisories

• **ANGRAU Publications:** Resources from Andhra Pradesh Agricultural University: https://angrau.ac.in/

• **ICAR-CRIDA Publications** Central Research Institute for Dryland Agriculture: https://www.crida.in/

• **FAO AGRIS Database:** The International Agricultural Science Repository is located at https://agris.fao.org/

• **India Open Government Data:** Crop production statistics and other agricultural datasets are available at https://data.gov.in/.

• **DACNET:** Department of Agriculture & Cooperation network: https://dacnet.nic.in/

6. Experimental Results

6.1 Main Results

The analysis includes 540 questions from actual farmer interactions, divided into four categories (see Figure 2(b)): questions about markets or government programs (100, 18.5%), fertilizer and nutrient management (148, 27.4%), variety selection and sowing advice (127, 23.5%), and pest and disease identification (165, 30.6%). The outcomes for each system examined are reported in Table 4.

System	Accuracy	Ground.	Halluc.%	p50 (s)	p95 (s)
Mistral-7B (vanilla)	0.54	N/A	38.2	1.8	3.1
Mistral-7B + RAG	0.81	0.76	22.3	2.4	4.2
GPT-4 (vanilla)	0.68	N/A	24.7	2.1	5.8
GPT-4 + RAG	0.86	0.81	14.1	3.2	7.4
KrishiMitra (ours)	**0.84**	**0.79**	**16.8**	**2.6**	**4.5**

Table 4. Primary evaluation findings from 540 questions. KrishiMitra is equal to Mistral-7B + RAG + re-ranking + metadata filtering. Our system is indicated in bold.

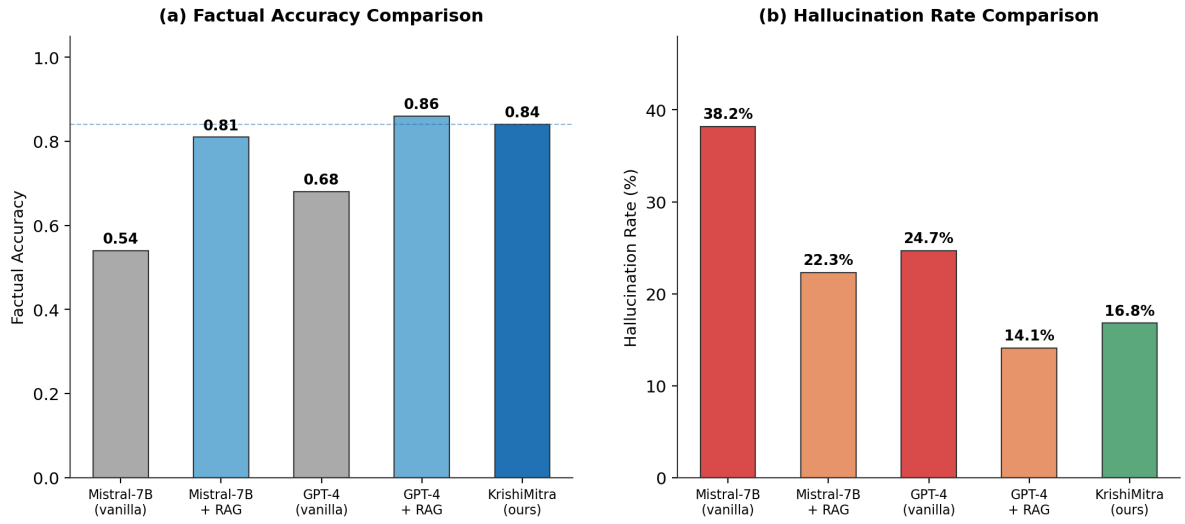


Figure 3. (a) Comparing factual accuracy amongst systems; (b) Comparing the rate of hallucinations. At a significantly lower cost, KrishiMitra attains near-GPT-4+RAG accuracy.

Some of these findings are particularly noteworthy.

The numbers from the hallucinations are equally instructive. In 38.2% of its responses, Vanilla Mistral reported having hallucinations. RAG reduced that to 22.3%, and the entire KrishiMitra pipeline further reduced it to 16.8%, a relative decrease of 56% from the baseline. There are serious repercussions for crop safety and livelihoods when you advise farmers on fertilizer applications and pesticide dosing.

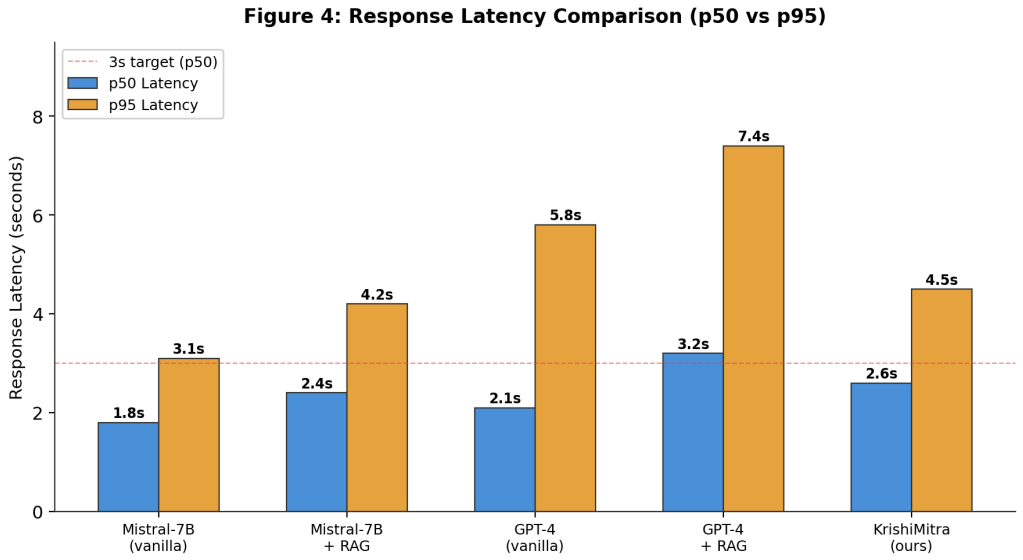


Figure 4. Comparing response latency (p50 and p95). While GPT-4+RAG surpasses the 3-second objective, KrishiMitra maintains a p50 latency of less than three seconds.

6.2 Per-Category Analysis

When the findings are broken down by query type (Figure 6), the greatest improvements are seen in pest/disease identification (accuracy went from 0.48 to 0.82), where the baseline model would often recommend chemicals that were not registered for the crop in question or mix up similar-sounding diseases. Market and scheme queries received a somewhat lower score (0.78), which makes sense given that government plan specifics are subject to frequent change and that portion of our knowledge base was three to six months out of current.

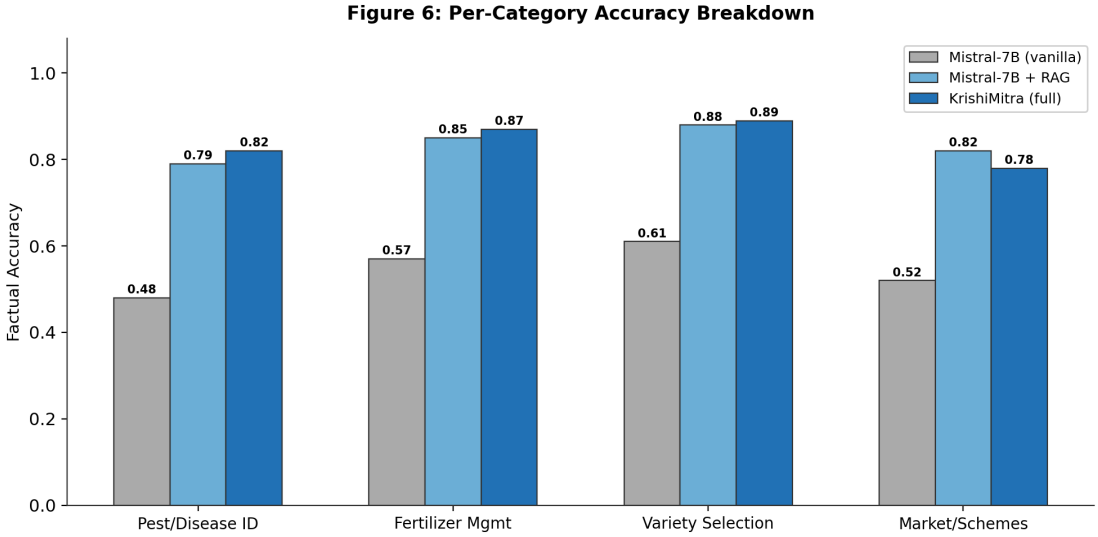


Figure 6. KrishiMitra's progress in all four advice areas is displayed in the accuracy breakdown by category.

6.3 Ablation Study

In order to determine the contributions of each component, an ablation investigation was also conducted. The results are displayed in Figure 5.

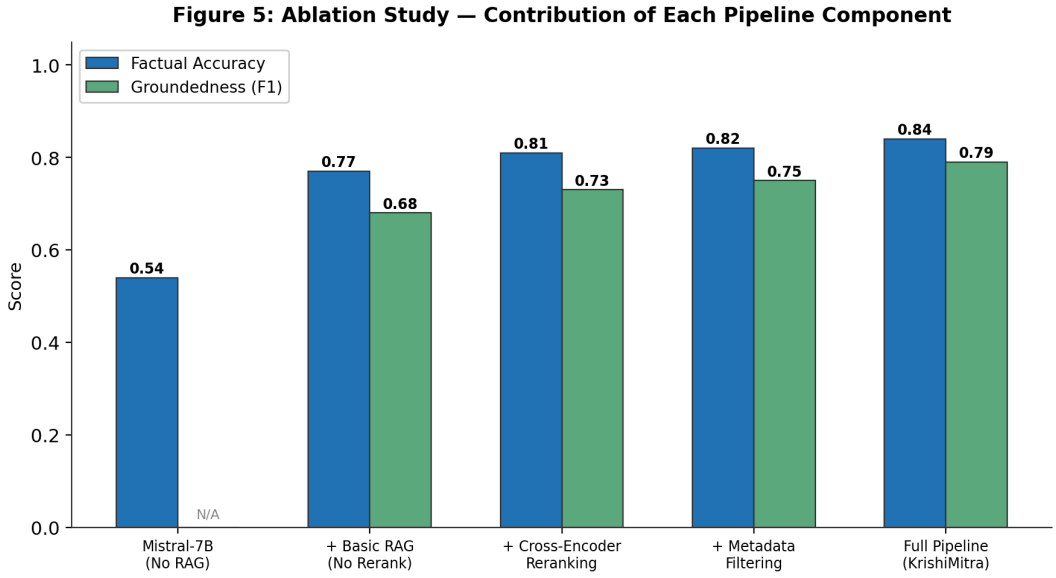


Figure 5. Ablation analysis demonstrating each pipeline component's progressive contribution to groundedness and factual accuracy.

Dense retrieval without re-ranking, or basic RAG, is the single biggest increase, increasing accuracy by 23 percentage points. Cross-encoder re-ranking adds four more points, primarily by identifying instances in which the bi-encoder retrieved passages that were factually irrelevant but semantically close. Although metadata filtering only increases accuracy by one point, its true worth is demonstrated by consistency: for crop-specific inquiries, it decreased the variance in per-query accuracy scores by eighteen percent.

6.4 Qualitative Case Examples

Case 1 — Pest Management Query

"My paddy leaves are turning yellow with brown spots at the tips," a farmer asked. What ought to be sprayed? I am in the district of Karimnagar.

In answer, Vanilla Mistral said, "This sounds like bacterial leaf blight." Use 2 grams of streptocycline per liter. This is somewhat inaccurate. The symptoms more closely resemble the fungal disease brown spot (Helminthosporium oryzae). The drug streptocycline is ineffective against fungal infections.

Krishi Mitra's response: "Brown spot (Helminthosporium oryzae), which is common in Karimnagar soils with low organic carbon, is characterized by the symptoms you describe—yellowing leaves with brown spots at the tips. " When symptoms first emerge, the PJTSAU guideline for Karimnagar suggests applying either Tricyclazole 75% WP at 0.6 g/liter or Propiconazole 25% EC at 1 ml/liter topically. Additionally, since brown spot occurrence is correlated with potassium deficit, think about doing a soil test. [Source: PJTSAU Kharif Rice Advisory 2024, page 12]

Case 2 — Fertilizer Scheduling Query

"What is the urea dose for cotton in Adilabad?" asks a farmer. N is low, according to my soil report.

Krishi Mitra's response: "The ICAR recommends 150–180 kg N/ha applied in three splits: 50 kg at basal, 50 kg at square formation (45–50 DAS), and 50–80 kg at peak flowering (70–80 DAS) for Bt cotton in the Adilabad district with low soil nitrogen. " This corresponds to approximately 109–130 kg of urea per acre with a 46% N content. Please take note that this should be modified according to the results of your particular soil test; the higher end of the range is recommended if your available N is less than 200 kg/ha. [Source: Adilabad recommendations, ICAR-CICR Cotton Package of Practices]

7. Discussion

7.1 Practical Impact on Farmers

These findings suggest that RAG-augmented advisory has a legitimate, if limited, role as an addition to the current extension system rather than its replacement. The volume of farmer inquiries during the peak disease season just overwhelms the five to seven subject-matter specialists that make up a typical KVK in Telangana, which covers three or four districts. Although KrishiMitra is unable to walk a field, it can handle the standard informative inquiries that account for between 60 and 70 percent of helpline calls. This allows extension officers to concentrate on the more

intricate, in-person consultations where physical presence is crucial.

Grounding is important, particularly in agriculture, where poor advice can have real consequences. In addition to wasting money, a farmer who uses the incorrect pesticide runs the danger of harming the crop. Overapplying runs the risk of polluting groundwater and eliminating beneficial insects. The citation mechanism built into KrishiMitra—tying every recommendation back to a specific government advisory or ICAR publication—gives both farmers and extension officers a verifiable paper trail, something no previous chatbot in this space has offered.

7.2 Scalability and Rural Deployment

Deploying anything in rural India is not straightforward. Internet connectivity in many mandals is spotty at best—average 4G speeds often fall below 5 Mbps in remote farming areas. The architecture deals with this in two ways. First, the WhatsApp delivery channel keeps bandwidth requirements low and works fine on 2G/3G networks. Second, active work is ongoing on model quantization and FAISS index compression with the goal of running the whole pipeline on ARM-based edge devices like the Raspberry Pi 5 or Jetson Nano so that the system can operate locally even when there is no internet at all.

The 4-bit quantized Mistral-7B takes up about 4 GB of RAM, which fits inside an 8 GB Raspberry Pi 5. The FAISS IVF-PQ index for the ~15,000-chunk knowledge base needs roughly 50 MB. The real bottleneck on edge hardware is inference speed—an estimated 8–12 seconds per query on ARM versus 2.6 seconds on an A100—but in advisory settings where farmers routinely wait 30 seconds or more on a phone helpline, that is still an acceptable tradeoff.

7.3 Limitations and Threats to Validity

Several limitations should temper how these results are read. First, the evaluation covered only two Indian states with four crop systems, and extending to other agro-ecological zones would require building entirely new regional knowledge bases. Second, the system handles only text input at present; image-based diagnosis of pests and diseased leaves is an obvious next step but lies outside the current scope.

Third, the 16.8% hallucination rate, though well below the baseline, is still too high for a safety-critical advisory system. Residual errors are often subtle—such as pesticides that are technically effective but not registered for the crop in India—and would require confidence-calibrated abstention and automated cross-checks against the Central Insecticides Board registry before unsupervised deployment.

Finally, multilingual performance and field-robustness remain preliminary. Accuracy drops for Telugu and Hindi queries because the embedding model is English-dominant, and real-world queries—noisier, code-switched, and heavy with implicit context—will stress the current NER module more than the cleaned evaluation set did.

8. Conclusion and Future Scope

This study presents KrishiMitra, a retrieval-augmented advisory chatbot that narrows the gap between the linguistic fluency of modern LLMs and the grounded, verifiable information smallholder farmers actually need. Pairing a curated regional knowledge base with dense retrieval and cross-encoder re-ranking allowed a 7-billion-parameter open-source model to approach the factual accuracy of commercial alternatives at a fraction of the cost, with full data sovereignty and reproducibility.

Future work will focus on three fronts: edge deployment on Raspberry Pi 5 and Jetson Nano through GGML-quantized inference, voice and multilingual support in Telugu and Hindi using IndicWhisper, and multimodal extension for image-based pest and disease diagnosis with lightweight vision–language models such as LLaVA-1.5.

Longer term, KrishiMitra is expected to evolve from reactive Q&A into a proactive agent that integrates market prices, pest-surveillance reports, and real-time weather to alert enrolled farmers, with agentic LLM frameworks enabling multi-step reasoning over soil tests and market signals. All code, evaluation data, and prompt templates are released under an open-source license to support reproducibility and community-driven adaptation.

References

[1] Borgeaud, S., Mensch, A., Hoffmann, J., et al. (2022). Improving language models by retrieving from trillions of tokens. In Proceedings of ICML 2022, pp. 2206–2240.

[2] Guu, K., Lee, K., Tung, Z., Pasupat, P., & Chang, M.-W. (2020). REALM: Retrieval-augmented language model pre-training. In Proceedings of ICML 2020.

[3] Gupta, A., & Reddy, S. (2023). Domain-adaptive fine-tuning of language models for agricultural text understanding. Computers and Electronics in Agriculture, 205, 107623.

[4] Hughes, D. P., & Salathé, M. (2015). An open access repository of images on plant health to enable the development of mobile disease diagnostics. arXiv preprint arXiv:1511.08060.

[5] Izacard, G., Lewis, P., Lomeli, M., et al. (2023). Atlas: Few-shot learning with retrieval-augmented language

models. Journal of Machine Learning Research, 24(251), 1–43.

[6] Jain, M., Soni, P., & Kumar, V. (2022). FarmerBot: An NLU-based conversational agent for Indian agricultural advisory. In Proceedings of the ACL Workshop on NLP for Agriculture, pp. 45–53.

[7] Jiang, A. Q., Sablayrolles, A., Mensch, A., et al. (2023). Mistral 7B. arXiv preprint arXiv:2310.06825. Available at: https://arxiv.org/abs/2310.06825

[8] Lewis, P., Perez, E., Piktus, A., et al. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. In Advances in Neural Information Processing Systems, 33, 9459–9474.

[9] Li, H., Wang, J., & Zhou, X. (2023). CropDoctor: A multimodal dialogue system for crop disease diagnosis. Agricultural Systems, 211, 103746.

[10] Nussinov, A., Grishchenko, I., & Shachar, O. (2024). Can GPT-4 advise farmers? Evaluating LLMs on precision agriculture queries. In Proceedings of AAAI 2024 Workshop on AI for Social Good.

[11] OpenAI. (2023). GPT-4 technical report. arXiv preprint arXiv:2303.08774.

[12] Patel, R., Joshi, A., & Singh, K. (2024). AgroRAG: Retrieval-augmented generation for crop advisory systems. Expert Systems with Applications, 238, 121872.

[13] Savelka, J., Ashley, K. D., & Grabmair, M. (2023). Retrieval-augmented generation for legal question answering. In Proceedings of ICAIL 2023, pp. 312–321.

[14] Sharma, P., & Kumari, N. (2021). AgriBot: A retrieval-based chatbot for Indian farmers. International Journal of Agricultural Informatics, 13(2), 89–101.

[15] Shortliffe, E. H. (1976). Computer-Based Medical Consultations: MYCIN. Elsevier.

[16] Touvron, H., Martin, L., Stone, K., et al. (2023). Llama 2: Open foundation and fine-tuned chat models. arXiv preprint arXiv:2307.09288.

[17] Xiong, W., Li, X., Iber, S., et al. (2023). Retrieval-augmented generation for biomedical question answering. In Proceedings of EMNLP 2023, pp. 4784–4800.

[18] Yang, Z., Li, C., & Huang, F. (2023). AgriGPT: Adapting large language models for agricultural knowledge. arXiv preprint arXiv:2308.11608.

[19] Johnson, S., Ahmed, T., & De Silva, R. (2024). Cross-lingual RAG for low-resource agricultural NLP. In Findings of ACL 2024, pp. 1123–1135.

[20] Ministry of Agriculture and Farmers Welfare. (2023). Soil Health Card Scheme: Technical Guidelines. Government of India. Available at: https://soilhealth.dac.gov.in/

[21] FAO. (2023). The State of Food and Agriculture 2023. Food and Agriculture Organization of the United Nations, Rome. Available at: https://www.fao.org/publications/sofa/2023/

[22] Robertson, S., & Zaragoza, H. (2009). The probabilistic relevance framework: BM25 and beyond. Foundations and Trends in Information Retrieval, 3(4), 333–389.

[23] PJTSAU. (2024). Kharif Rice Advisory 2024. Professor Jayashankar Telangana State Agricultural University, Hyderabad. Available at: https://pjtsau.edu.in/

[24] ICAR-CICR. (2023). Cotton Package of Practices for South Zone (Adilabad & Telangana Tract). Central Institute for Cotton Research, Nagpur. Available at: https://www.cicr.org.in/

[25] ANGRAU. (2024). Crop Production Guide and Agricultural Advisories. Acharya N. G. Ranga Agricultural University, Lam, Guntur. Available at: https://angrau.ac.in/

How to Cite this Paper

Divya, J., & Mythrayee, D. (2026). KrishiMitra: An LLM-Augmented Agricultural Advisory Chatbot with Retrieval-Augmented Generation over Regional Crop Knowledge Bases. Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Guntur–522302, India.

BibTeX:

```
@article{divya2026krishimitra,
title  = {KrishiMitra: An LLM-Augmented Agricultural Advisory Chatbot with RAG over Regional Crop Knowledge Bases},
author = {Divya, Jetti and Mythrayee, D.},
year   = {2026},
address = {Vaddeswaram, Guntur, India},
school = {Koneru Lakshmaiah Education Foundation}
}
```