

AgroAskAI: A Multi-Agentic AI Framework for Supporting Smallholder Farmers' Enquiries Globally

Nadine Angela Cantonjos, Arpita Biswas

Rutgers University

npc58@scarletmail.rutgers.edu, a.biswas@rutgers.edu

Abstract

Agricultural regions in rural areas face damage from climate-related risks, including droughts, heavy rainfall, and shifting weather patterns. Prior research calls for adaptive risk-management solutions and decision-making strategies. To this end, artificial intelligence (AI), particularly agentic AI, offers a promising path forward. Agentic AI systems consist of autonomous, specialized agents capable of solving complex, dynamic tasks. While past systems have relied on single-agent models or have used multi-agent frameworks only for static functions, there is a growing need for architectures that support dynamic collaborative reasoning and context-aware outputs. To bridge this gap, we present AgroAskAI, a multi-agent reasoning system for climate adaptation decision support in agriculture, with a focus on vulnerable rural communities. AgroAskAI features a modular, role-specialized architecture that uses a chain-of-responsibility approach to coordinate autonomous agents, integrating real-time tools and datasets. The system has built-in governance mechanisms that mitigate hallucination and enable internal feedback for coherent, locally relevant strategies. The system also supports multilingual interactions, making it accessible to non-English-speaking farmers. Experiments on common agricultural queries related to climate adaptation show that, with additional tools and prompt refinement, AgroAskAI delivers more actionable, grounded, and inclusive outputs. Our experimental results highlight the potential of agentic AI for sustainable and accountable decision support in climate adaptation for agriculture.

Code — <https://github.com/ncantonjos04/AgroAskAI>

1 Introduction

As agriculture faces a growing strain from the environmental and socioeconomic disruptions of climate change, artificial intelligence (AI) enables new ways of providing dynamic decision support to farming communities. In these communities, agriculture is not only a primary livelihood but also a fundamental component of local economies and daily life. However, challenges posed by climate-related risks, such as erratic weather, depleted soil health, and shifted rainfall, have negatively affected farmers in vulnerable communities. For instance, crop production in rural regions of Central America has declined over the past two decades due to

Copyright © 2026, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

more frequent storms and severe flooding (Waddick 2017). In the arid and semi-arid regions of Kenya, 31% of smallholder farmers lack access to climate-related information, leaving them unprepared to adapt to rising temperatures, prolonged droughts, and increasingly unpredictable rainfall patterns (Mutunga, Charles, and Patricia 2017).

The severity of these local agricultural challenges is amplified by global environmental degradation and economic instability. Extreme weather-related events caused nearly \$1.5 trillion in economic loss worldwide in the decade to 2019, a substantial increase from \$184 billion in the 1970s (Charlton 2023; Wang et al. 2025). In 2023, droughts in the Southern and Midwestern regions had caused \$14.5 billion loss, impacting the agriculture sector, including damage to field crops from lack of rainfall and heat (Smith 2024; Silici et al. 2021). These trends underscore the need for improved risk management and decision support to prevent a vast amount of damage from occurring in the future.

Previous studies have examined climate adaptation decisions among smallholder farmers in specific regions, such as in the mountainous areas of Vietnam (Hoa Sen, Bond, and Hoang 2022), far-western Nepal (Lamichhane et al. 2022), and the semi-arid zones of Senegal (Zagre et al. 2024). While these localized insights are valuable, increasing global variability in climate patterns and the rising frequency of extreme weather events underscore the urgent need for effective, scalable adaptation strategies. In particular, smallholder farmers across the globe, who are among the most vulnerable to climate disruptions, require decision support to navigate these complex and evolving challenges.

Effective adaptation measures often include the development of climate-resilient crop varieties, improved cultivation practices, and timely dissemination of climate information (Castellanos, Thomas, and Dunston 2019). However, the successful scaling of such innovative practices across diverse agricultural landscapes demands solutions that are both context-specific and locally adaptable. One-size-fits-all solutions risk not only ineffectiveness but may also lead to maladaptation, compounding existing agricultural vulnerabilities. Therefore, systems that support climate resilience must be not only intelligent but also dynamic—capable of adjusting their recommendations and suggestions based on real-time data and feedback, in order to advance climate resilience among smallholder farmers.

This is where agentic AI offers transformative potential. Unlike conventional AI systems that require frequent human oversight, agentic AI systems exhibit a higher degree of autonomy. They can reason through complex, multi-step objectives, integrate historical and real-time external data, and independently execute decision processes from end to end (Mohammed Salah et al. 2025). Applied to agriculture, agentic AI can generate novel, tailored, and context-aware solutions that respond adaptively to changing environmental and socio-economic conditions. Recent work by Raza et al. (2025) highlights the importance of integrating Trust, Risk, and Security Management (TRiSM) into agentic AI systems, emphasizing transparent reasoning, policy-based oversight, and regulatory traceability as key pillars for ethical and reliable AI deployment. However, as they note, no unified framework currently operationalizes these principles within real-world, autonomous AI systems—especially in high-stakes domains like agriculture.

In this paper, we introduce **AgroAskAI**, a modular, agentic, multi-agent AI framework designed to support smallholder farmers by enhancing agricultural decision-making under uncertainty. AgroAskAI integrates role-specialized autonomous agents, real-time external resources (e.g., weather data, forecast model, geospatial information), and a critique mechanism into a multi-agent architecture in order to ensure traceability and context-aware reasoning. In addition, AgroAskAI implements explicit reasoning logs, decision checkpoints, and external validation to promote auditability. A dedicated critique (reviewer agent) embeds correctness and feasibility as first-class objectives to reduce hallucination while ensuring locally grounded, actionable plans. In addition to decision support, AgroAskAI plays an educational role by making its reasoning process transparent, fostering trust and understanding. By autonomously generating adaptive, context-specific strategies, AgroAskAI equips farmers with the tools and knowledge to build lasting agricultural resilience in the face of climate change.

2 Related Work

Artificial Intelligence (AI) is increasingly being leveraged to support farmers in managing the complex challenges posed by climate change in agriculture. Prior work has explored AI-driven systems for automated sensor monitoring of soil and crop health (Microsoft 2019), often aimed at optimizing yield. While such systems are valuable, they fall outside the scope of this work. In contrast, our focus is on developing a conversational AI system designed to support farmers through natural language interactions, rather than through sensor-based automation.

The Agrifriendly Conversational AI Chatbot (Gujjar and Kumar 2025) uses a single agent, generative system based on the Rasa framework (<https://rasa.com/>) to understand natural human language and assist farmers with queries related to crop maintenance. While effective for basic queries, it relies on a single-agent model that limits adaptability and lacks internal critique or modular reasoning. Farmer.Chat (Singh et al. 2024) introduces a generative chatbot deployed via platforms like WhatsApp and Facebook Messenger, using Retrieval-Augmented Generation (RAG) to deliver in-

formative, multilingual responses. Its governance relies on user feedback loops and data shared through local partners. AgriBuddy (Tonmoy et al. 2025) is an agentic chatbot system, tailored for Bangladeshi smallholder farmers supporting interaction in Bangla, English, and hybrid Bangla-English. The system is composed of specialized agents, such as a query handler, memory agent, and expert advisory agent, and includes a CNN (Convolutional Neural Network) vision model for rice disease detection.

In contrast to prior research, AgroAskAI provides reliable climate-adaptive solutions to smallholder farmers across the globe by adopting a modular, multi-agent architecture with embedded internal governance mechanisms that proactively mitigate hallucination and ensure output reliability. Instead of relying on external approvals, the system accesses reliable real-time information from trusted historical and forecasted weather data, while also allowing users to contribute verified content through a curated external document repository, enhancing both transparency and adaptability.

3 Design and Methodology

We developed a framework grounded in the agentic AI paradigm to address farmer queries related to agriculture, as illustrated in Figure 1. The framework comprises a collection of specialized agents, each designed to perform distinct tasks within a well-defined logical workflow. In this section, we provide a detailed description of each agent, its role within the overall framework, as well as the inter-agent connectivity and the structured exchange of information between agent pairs. The pseudocode for the agentic components is deferred to Appendix A.

Prompt Agent. Serves as the primary interface with the user, initiating conversations with a greeting and collecting user input. It forwards this input to the Parsing Agent for further processing, while also managing follow-up interactions. This includes handling requests for missing or ambiguous information from other agents, prompting the user for clarification, or notifying them when a query cannot be resolved. A key responsibility of the Prompt Agent is to identify the language used by the user, ensuring all subsequent interactions occur in that same language to maintain a seamless and personalized experience.

Parsing Agent. Responsible for interpreting the user's input received from the Prompt Agent. It extracts essential information, including user intent, geographic location, and relevant time frame, to build a localized and actionable context. When key details are missing or the intent is unclear, the Parsing Agent collaborates with the Prompt Agent to request clarification, ensuring that the system maintains both accuracy and relevance in its responses. Upon complete information, the structured (JSON) data and the user query are then passed to the Agent Manager for further processing.

Agent Manager. Acts as the central coordinator within the system, interpreting the localized context and autonomously selecting the appropriate specialized agents to activate. By invoking only the relevant agents, it ensures efficient and focused task execution. It also plays a critical role in aggre-

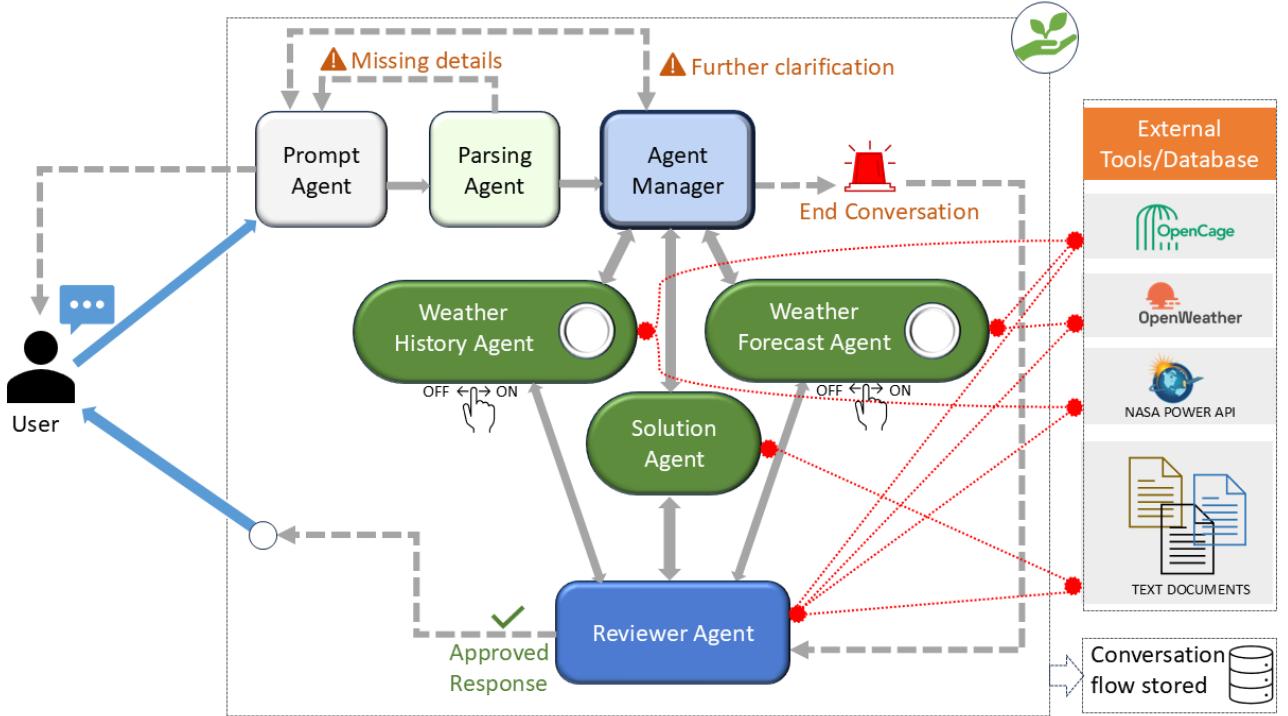


Figure 1: AgroAskAI Multi-Agent Architecture

gating the information needed by the Solution Agent to craft a complete and contextually accurate response. Once a satisfactory response is finalized, typically following reviews from the Reviewer Agent, the Agent Manager terminates the response generation cycle and signals the Reviewer Agent to send the final solution.

Weather Forecast Agent. Specializes in handling queries related to future or predicted weather conditions, utilizing external services such as OpenCage¹ and OpenWeather² to retrieve location-specific forecast data. By integrating these services, the agent provides timely and relevant climate insights to support user decision-making.

Weather History Agent. Designed to handle inquiries about past weather patterns, the Weather History Agent retrieves historical climate data using tools like OpenCage for geospatial resolution and the NASA POWER API³ for reliable weather records. This allows the extraction of historical climate trends to support decision-making.

Solution Agent. Generates context-aware, actionable responses for queries that require recommendations on agricultural practices, specifically in climate-sensitive agricultural settings. It draws upon information provided by the Agent Manager and consults external text documents to supplement domain-specific knowledge.

Reviewer Agent. Plays a critical role in assessing responses generated by the Solution Agent across multiple dimensions, including clarity, technical soundness, fairness, and alignment with user intent. When necessary, it consults external sources to detect and reduce hallucinations in Solution Agent's responses. If the output is found to be incomplete or suboptimal, the Reviewer Agent can trigger a new cycle of agent selection and response generation via the Agent Manager. This iterative feedback mechanism ensures the final output is accurate, contextually appropriate, and socially responsible.

Model Specification

AgroAskAI is built using the Microsoft Semantic Kernel framework and utilizes OpenAI's GPT-4 Learning Language Model (LLM) to understand the natural language of user queries. The architecture of AgroAskAI is inspired by agentic AI design principles, such as perception stage, goal decomposition, and safety guardrails. This system consists of multiple agents whose tasks include interacting with user using the language of interest, parsing user input for necessary information, conducting agent selection, retrieving necessary external data, generating novel solutions and adaptations for user input. AgroAskAI also logs the complete conversation flow between agents for each user query, enabling traceability, transparency, and future system improvements.

Data is sourced through a Retrieval-Augmented Generation (RAG) strategy (Singh et al. 2025), which allows the system to access up-to-date and context-specific data from

¹OpenCage Geocoding API: <https://opencagedata.com>

²OpenWeather Forecasting API: <https://openweathermap.org>

³NASA POWER API: <https://power.larc.nasa.gov>

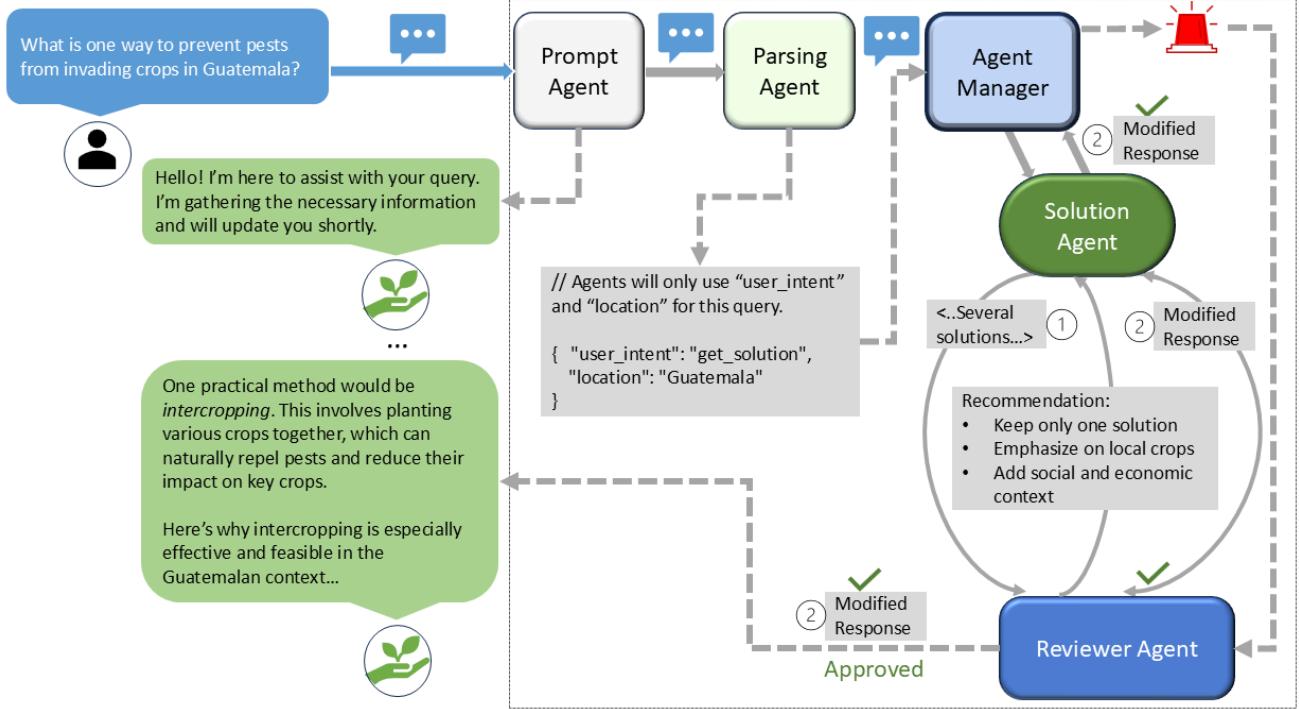


Figure 2: A sample decision-support by **AgroAskAI**.

external APIs. These sources include, but are not limited to, NASA’s POWER Project for historical climate data, OpenWeather for forecasts, and OpenCage for location-based geocoding. These components are orchestrated through a centralized control logic (Agent Manager) that activates the right agents for each task based on user input. This design enables the system to remain flexible, context-aware, and efficient in handling diverse climate-related queries.

Figure 2 illustrates a sample conversation on “What is one way to prevent pests from invading crops in Guatemala?” (Hart 2017). Upon receiving the query, the prompt agent extracts key details (e.g., user_intent, location) and forwards them, along with the original question, to the Agent Manager. Recognizing that no weather data is required, the Agent Manager directs the request to the solution agent. The solution agent generates an initial answer using relevant tools, after which the reviewer agent evaluates its strengths and shortcomings—particularly noting that the user requested a single solution and that local crops and socio-economic context should be specified. The solution agent incorporates these recommendations, and the revised answer is approved. If approval was withheld, the solution agent would continue iterating until a satisfactory solution was achieved. Upon approval, the Agent Manager ends the conversation, and the final response is delivered to the user. The complete conversation and response details are provided in Appendix B.

4 Experimental Results

To evaluate AgroAskAI’s effectiveness, reliability, multi-agent, and multilingual capabilities in supporting climate adaptation decisions for agriculture, we curated region-specific farmer queries and asked AgroAskAI to provide decision-support. Our evaluation focused on queries representative of the challenges faced by rural farming communities, including drought planning, irrigation strategies, crop selection, and pest control. The key highlights are:

- 1. Comparison of AgroAskAI with Other Models** to evaluate AgroAskAI’s response against a single-agent large language model (ChatGPT) and a traditional rule-based decision tool (CROPWAT).
- 2. Importance of Agent Manager:** in selecting and orchestrating the flow of reasoning among specialized agents.
- 3. Significance of the Reviewer Agent:** operating as an internal governance layer within the system.
- 4. Handling Missing Details:** Tackling under-specified queries and missing information.
- 5. Multi-lingual Capabilities** of AgroAskAI enabled by the underlying large language model.

Comparison with Other Models To assess the quality of AgroAskAI’s responses to agriculture-related queries, the responses are compared against (1) ChatGPT and (2) CROPWAT (<https://www.fao.org/land-water/databases-and-software/cropwat/en/>). Interactions with ChatGPT were

	CROPWAT	ChatGPT	AgroAskAI
Weather Information	<ul style="list-style-type: none"> Detailed rainfall metrics No temperature metrics 	<ul style="list-style-type: none"> Listed average rainfall and historical weather conditions during 3 out of 20 trial runs and listed forecasts (temperature and precipitation) during 7 runs. Listed examples of climate issues during 7 out of the 20 runs. 	<ul style="list-style-type: none"> Listed average rainfall and historical weather conditions during 16 of the trial runs and listed forecasts (temperature and precipitation) during 16 trial runs. Listed examples of climate issues during 11 runs.
Water Conservation	Outputs quantified irrigation scheduling and field efficiency metrics	Provides implementation steps during 0 out of 20 runs	Mentions different harvesting system types during 15 runs. Provides implementation steps during 13 runs
Livestock advice	Not addressed	Addressed in 16 trial outputs	Addressed in all 20 trial outputs
Soil Conservation	Not addressed	Names the methods of conservation in 12 outputs	Names the methods of conservation in all 20 outputs. Out of which, 13 outputs had implementation instructions for conservation
Adaptability to Farm Size/Type	Gives quantified data for optimal timing irrigation strategies but does not list them using practical language.	9 outputs provided solutions accounting for farm-specific conditions (e.g., small vs. large-scale water management).	16 outputs offered small and large-scale solutions with step-by-step instructions for varying farm conditions (e.g., barrels vs tanks in water conservation)
Inclusion of Support Programs or Resources	Not addressed	4 outputs mentioned specific projects and institutional support (e.g. KCSAP, NDMA)	16 outputs mentioned specific projects and institutional support
Multilingual Support	Officially provides multilingual support for 4 languages	Officially provides multilingual read/write support for 50+ languages	Officially provides multilingual read/write support for 50+ languages.

Table 1: Comparison of CROPWAT, ChatGPT, and AgroAskAI based on 20 different user queries. The comparison highlights the enhanced capabilities of AgroAskAI in providing detailed responses.

conducted using the OpenAI Chat GPT web interface during June-August 2025. The model used for all experiments was GPT-5.1 (the default model available on the ChatGPT platform at that time). No system prompt modifications, custom instructions, or fine-tuning were applied beyond ChatGPT’s default settings. Experiments using CROPWAT were performed with the latest available version as of July 2025, which was CROPWAT 8.0 (FAO). Agrifriendly could not be included in the comparison due to the lack of access to its source code or API. Also, Farmer.Chat required users to either upload proprietary content or join an existing registered team, restricting independent evaluation of the system.

Table 1 summarizes the comparison of three models evaluated across six dimensions: weather information, water conservation, livestock advice, soil conservation, farm adaptability, support resources, and multi-lingual capabilities, based on 20 trial inputs.

For example, when asked about water management strategies in Kitui, Kenya, ChatGPT briefly mentioned *zai pits* and *mulching* without elaboration, while CROPWAT focused solely on irrigation metrics. In contrast, AgroAskAI detailed

specific water harvesting systems and provided clear instructions on how to apply them.

Regarding weather information, all three systems provided average rainfall data, but only AgroAskAI and ChatGPT included contextual climate risks like dry spells and heat. AgroAskAI additionally offered temperature data (21–27°C), missing in CROPWAT.

Regarding livestock advice, CROPWAT was unable to address any concerns, while AgroAskAI recommended drought-resilient breeds and integrated crop-livestock systems, which were often missing in ChatGPT’s response. Similarly, for soil conservation, AgroAskAI explained practices like composting and cover cropping in practical terms, while ChatGPT listed techniques with minimal detail and CROPWAT lacked relevant content. Regarding adaptability to different farm sizes or types, AgroAskAI tailored advice to both small and large-scale operations. It offered concrete differences in implementation (e.g., tanks vs. barrels), whereas ChatGPT remained general, and CROPWAT provided timing metrics but no practical adaptation suggestions. In terms of institutional support, ChatGPT listed several pro-

grams like the Kenya Climate Smart Agricultural Project (KCSAP) and the National Drought Management Authority Act (NDMA), while AgroAskAI and CROPWAT did not reference any specific support systems. This suggests that integration with external programs remains a gap in the current system. Finally, both ChatGPT and AgroAskAI provide broader multi-lingual support than CROPWAT, which is restricted to four languages only.

Overall, AgroAskAI consistently produced actionable, context-aware responses, outperforming both ChatGPT and CROPWAT across multiple dimensions. Its outputs were structured as step-by-step recommendations written in clear, practical language, enabling farmers and extension workers to directly apply the advice in field conditions. In contrast, ChatGPT often provided generic, surface-level lists with limited contextual agronomic grounding, while CROPWAT, though precise in its numerical calculations, offered quantified results without operational guidance. For smallholder farmers, the gap between information and actionable decision support is often where existing digital tools fall short. AgroAskAI bridges this gap by combining agronomic reasoning, localized context, and practical instructions, transforming analytical outputs into ready-to-use climate-adaptive recommendations. Appendix C contains the outputs of CROPWAT, ChatGPT, and AgroAskAI for a particular query.

Importance of Agent Manager To empirically evaluate the impact of the Agent Manager on task relevance and interaction coordination in a multi-agent system, we generated various queries that may or may not require invoking the weather forecast or the weather history agent. The Agent Manager successfully orchestrated the specialized agents to obtain an appropriate context-relevant response. For instance, in the sample query demonstrated in Figure 2, the Agent Manager activates only the Solution Agent since the other agents (weather history and weather forecast) are not required for the response. We provide a couple of additional examples. In Example 1, the Agent Manager invokes the Weather History Agent only, and in Example 2, only the Weather Forecast Agent is invoked. The Agent Manager successfully terminated the conversation once an appropriate solution has been generated.

Example 1. Agent Manager invokes Weather Forecast

User (Lopez-Ridaura et al. 2019): What is the predicted rainfall in San Marcos, Guatemala, tomorrow?

AgroAskAI: Hello, I'm here to assist with your query. I'm gathering the necessary information and will update you shortly.

AgroAskAI: The approved forecast indicates that in San Marcos, Guatemala, on August 12, 2025, there will be moderate rain with a predicted rainfall of 13.71 mm. The daytime temperature will be approximately 25.19°C, and at night, it will drop to around 19.18°C. Additional details include 74% humidity, wind speeds of 12.7 kph from the southwest (213°C), and 64% cloud cover. Sunrise is at 07:52, and sunset is at 20:32.

Example 2. Agent Manager invokes Weather History

User (Mutunga, Charles, and Patricia 2017): How has the average temperature in Kitui, Kenya, changed over the past 20 years?

AgroAskAI: Hello, I'm here to assist with your query. I'm gathering the necessary information and will update you shortly.

AgroAskAI: From 2005 to 2025 in Kitui, Kenya, the annual average temperature showed a slight increase. The data indicates a gradual fluctuation within annual averages, starting at around 23.62°C in 2003 and reaching 23.83°C in 2023, with some years experiencing notable variations. Generally, this period reflects relatively stable but slightly warming trends in temperature.

Significance of the Reviewer Agent To evaluate the Reviewer Agent, we compared AgroAskAI with a modified version without any Reviewer Agent. AgroAskAI consistently ensures factual accuracy and alignment with user intent in all the queries, which are often missed in the responses by a similar model without any Reviewer Agent. For example, while answering the query: "What crops will survive the upcoming drought in Kitui, Kenya?" (Mutunga, Charles, and Patricia 2017), the model without Reviewer Agent lists adaptations irrelevant to the user prompt, such as solutions to temperature changes and tips for water conservation. Such hallucinations are not found in the responses from AgroAskAI.

Handling Missing Details We measure the effectiveness of AgroAskAI in detecting and resolving underspecified or incomplete queries. We found that the Parse Agent is always able to detect missing information related to location and timestamp when those information are relevant for the query. The Parse Agent also notifies the user when the intent is out of scope for AgroAskAI to answer. Moreover, Agent Manager sometimes requires more details and notifies the Prompt Agent to gather further clarification. Example 3 illustrates the effectiveness of AgroAskAI in identifying missing details.

Example 3. Missing location information

User: What are the predicted weather conditions and what can I do to manage my crops?

Prompt Agent to User: Hello! I'm here to assist with your query. I'm gathering the necessary information and will update you shortly.

Parse Agent:

```
{
  "user_intent": "get_solution",
  "location": "null",
  "start_year": "",
  "end_year": "",
  "forecast_date": ""
}
```

Prompt Agent to User: Please enter a location (e.g., city, state, or country):

User: Kitui, Kenya

Prompt Agent to Parse Agent: Updated location to Kitui, Kenya.

<response generation continues>

Multi-lingual Capabilities AgroAskAI supports queries in multiple languages, utilizing the capabilities of the large language

models. AgroAskAI's Prompt Agent identifies the language in which the user asks a query and ensures that subsequent interactions with the user are in the same language. For example, for the query, “*hali ya hewa ikoje leo katika kitui kenya*” (“What is the predicted weather in Kitui, Kenya?”), AgroAskAI correctly identifies the language as Swahili and responds:

- **Muhtasari:** Anga lina mawingu kidogo na vipindi vyauwazi.
- **Joto:** Mchana ni 28.57°C, na usiku ni 21.69°C.
- **Mvua:** Hakuna mvua inayotarajiwa.
- **Unyevu:** 28%.
- **Upepo:** Kasi ya upepo ni 21.6 km/h, ukielekea 138°.
- **Kiwango cha UV:** 11.62 (Juu sana).
- **Jua:** Linachomoza saa 5:30 asubuhi na kuzama saa 5:34 jioni.

Overall, AgroAskAI demonstrated superior performance by delivering localized, adaptable, and step-by-step guidance, bridging the gap between quantitative tools like CROPWAT-8.0 and generic generative models like ChatGPT-5.1.

5 Conclusion and Discussion

In the face of providing climate-adaptive decision support to farmers—ranging from prolonged droughts to unpredictable rainfall—AgroAskAI provides detailed step-by step guidance combining precision of domain-specific tools with the power of generative AI, and leveraging a multi-agent, role-specialized architecture with built-in governance mechanisms. Experimental results affirm AgroAskAI's superior performance in delivering actionable and localized recommendations compared to both static modeling tools and generic single-agent AI models. As agriculture faces an increasingly uncertain climate future, AgroAskAI exemplifies how agentic AI can drive sustainable, scalable, and accountable adaptation strategies, empowering farmers to thrive in a changing world.

Limitations. While AgroAskAI shows promising results in our empirical evaluation, we note that the system's performance is bounded by the quality and scope of external data sources it integrates, including weather databases and local document repositories. Moreover, its multilingual capabilities, while strong, rely on the underlying language model (OpenAI) and may require further fine-tuning for under-resourced dialects. Finally, real-world deployment in low-connectivity environments remains an operational challenge, requiring lightweight interfaces and offline capabilities. One potential solution is the establishment of centralized computing hubs, such as community centers, where a technically proficient individual can operate AgroAskAI on behalf of multiple farmers. This shared-access model can bridge digital gaps while ensuring that even farmers without personal devices or internet connectivity benefit from the system's capabilities.

Future Work. Going ahead, we will focus on extensive user testing with smallholder farmers to evaluate the usability, accessibility, and cultural appropriateness of AgroAskAI. This includes participatory design sessions, where farmers interact and critique system outputs; field trials to assess the clarity, trustworthiness, and impact of recommendations; and iterative refinement based on real-world feedback. Special attention will be given to testing across regions with varying literacy levels, dialects, and technological familiarity, ensuring that the interface, language, and decision-support outputs are truly aligned with farmers' needs.

Acknowledgements

N.C. gratefully acknowledges the support of Douglass WiSE's SUPER Research Program.

References

- Castellanos, E. J.; Thomas, T. S.; and Dunston, S. 2019. Climate change, agriculture, and adaptation options for Guatemala.
- Charlton, E. 2023. This is what the climate crisis is costing economies around the world. Accessed: 2025-06-10.
- Gujjar, J.; and Kumar, H. 2025. Agri Friendly Conversational AI Chatbot Using Open Source Framework. *Indian Journal of Science and Technology*, 18(8): 580–585.
- Hart, N. 2017. At Mexico-Guatemala Border, Sterile Insect Technique has Controlled the Northward March of Medflies Since 1982. Accessed: 2025-06-10.
- Hoa Sen, L. T.; Bond, J.; and Hoang, H. D. T. 2022. Exploring smallholder farmers' climate adaptation decision-making in mountainous areas of Central Vietnam: implications for extension services. *The Journal of Agricultural Education and Extension*, 29(2): 247–268.
- Lamichhane, P.; Miller, K. K.; Hadjikakou, M.; and Bryan, B. A. 2022. What motivates smallholder farmers to adapt to climate change? Insights from smallholder cropping in far-western Nepal. *Anthropocene*, 40: 100355.
- Lopez-Ridaura, S.; Barba-Escoto, L.; Reyna, C.; Hellin, J.; Gerard, B.; and Van Wijk, M. 2019. Food security and agriculture in the Western Highlands of Guatemala. *Food Security*, 11(4): 817–833.
- Microsoft. 2019. FarmBeats: AI, Edge & IoT for Agriculture. Accessed: 2025-06-08.
- Mohammed Salah, A.; Alnoor, A.; Abdelfattah, F.; and Chew, D. X. 2025. Agentic Ai Forging Adaptive, Equity-Driven Governance Pathways for Sustainable Futures. *XinYing, Agentic Ai Forging Adaptive, Equity-Driven Governance Pathways for Sustainable Futures*.
- Mutunga, E. J.; Charles, K.; and Patricia, M. 2017. Smallholder farmers perceptions and adaptations to climate change and variability in Kitui county, Kenya.
- Raza, S.; Sapkota, R.; Karkee, M.; and Emmanouilidis, C. 2025. Trism for agentic ai: A review of trust, risk, and security management in llm-based agentic multi-agent systems. *arXiv preprint arXiv:2506.04133*.
- Silici, L.; Rowe, A.; Suppiramaniam, N.; and Knox, J. W. 2021. Building adaptive capacity of smallholder agriculture to climate change: Evidence synthesis on learning outcomes. *Environmental Research Communications*, 3(12): 122001.
- Singh, A.; Ehtesham, A.; Kumar, S.; and Khoei, T. T. 2025. Agentic retrieval-augmented generation: A survey on agentic rag. *arXiv preprint arXiv:2501.09136*.
- Singh, N.; Wang'ombe, J.; Okanga, N.; Zelenska, T.; Repishti, J.; Mishra, S.; Manokaran, R.; Singh, V.; Rafiq, M. I.; Gandhi, R.; et al. 2024. Farmer. Chat: Scaling AI-Powered Agricultural Services for Smallholder Farmers. *arXiv preprint arXiv:2409.08916*.
- Smith, A. B. 2024. 2023: A historic year of U.S. billion-dollar weather and climate disasters. Accessed: 2025-06-10.
- Tonmoy, M. S. I.; Rabbi, R.; Hassan, M. R.; Naswan, R.; and Shome, T. 2025. AgriBuddy: An Agentic AI System for Bangladeshi Agriculture Using RAG and Vision Models.
- Waddick, K. 2017. Effects of Climate Change on Agriculture in Guatemala and Honduras. *Global Majority E-Journal*, 8(2): 109–120.

Wang, F.; Chen, T.; Koh, L.; and Wu, A. 2025. Future-proof finance: Navigating climate risks and ESG goals with agentic AI and collaborative foresight. *FW VISION Futures Thinking Inc.*

Zagre, I.; Akinseye, F. M.; Worou, O. N.; Kone, M.; and Faye, A. 2024. Climate change adaptation strategies among smallholder farmers in Senegal's semi-arid zone: role of socio-economic factors and institutional supports. *Frontiers in Climate*, Volume 6 - 2024.

A Pseudo Code for Agents in AgroAskAI

Algorithm 1: PromptAgent.invoke

Input: User query q and clarification request r

- 1: $kb \leftarrow OpenAIChatCompletion(GPT-4o)$
- 2: **Initiate** language = “English”
- 3: **if** $q \neq \text{null}$ **then**
- 4: language $\leftarrow kb.\text{extract_language}(q)$
- 5: **Display** greeting_message(language)
- 6: ParseAgent.invoke(q)
- 7: **else**
- 8: qr $\leftarrow kb.\text{generate_clarification}(r)$
- 9: **Display** qr to user
- 10: **Wait for** user_input
- 11: ParseAgent.invoke(user_input)
- 12: **end if**

Algorithm 2: ParseAgent.invoke

Input: nl_q : Processed natural language query from the Prompt Agent

- 1: $kb \leftarrow OpenAIChatCompletion(GPT-4o)$
- 2: $i \leftarrow kb.\text{extract_user_intent}(nl_q)$
- 3: **if** $!kb.\text{is_in_scope}(i)$ **then**
- 4: cr $\leftarrow kb.\text{notify_out_of_scope}(i)$
- 5: PromptAgent.invoke(null, cr)
- 6: **return**
- 7: **end if**
- 8: $mv \leftarrow \text{null}$ {missing values}
- 9: **if** $kb.\text{is_location_req}(i)$ **then**
- 10: $l \leftarrow kb.\text{extract_location}(nl_output)$
- 11: **if** $l == \text{null}$ **then**
- 12: $mv \leftarrow mv \cup \{"l"\}$
- 13: **end if**
- 14: **end if**
- 15: **if** $kb.\text{is_timerange_required}(i)$ **then**
- 16: $t = [t_1, t_2] \leftarrow kb.\text{extract_time_range}(nl_output)$
- 17: **if** $t == \text{null}$ **then**
- 18: $mv \leftarrow mv \cup \{"t"\}$
- 19: **end if**
- 20: **end if**
- 21: $\text{parsed} \leftarrow \text{to_JSON}(i, l, t)$
- 22: **while** $mv \neq \text{null}$ **do**
- 23: $cr \leftarrow kb.\text{generate_clarification_request}(mv)$
- 24: user_input $\leftarrow \text{PromptAgent.invoke}(null, cr)$
- 25: $[i, l, t] \leftarrow kb.\text{Extract}(user_input, \text{parsed})$
- 26: $\text{parsed} \leftarrow \text{to_JSON}(i, l, t)$
- 27: $mv \leftarrow \{k \in \{l, t\} : k == \text{null}\}$
- 28: **end while**
- 29: AgentManager.invoke(nl_q, parsed)

Algorithm 3: AgentManager.invoke

Input: Query nl_q and Parsed query parsed

- 1: $kb \leftarrow OpenAIChatCompletion(GPT-4o)$
- 2: **Specify** termination_strategy
- 3: $\text{termination}() = kb.\text{generate}(\text{termination_strategy})$
- 4: **Specify** selection_strategy
- 5: $\text{selection}() = kb.\text{generate}(\text{selection_strategy})$
- 6: **Initialize** chat_history $\leftarrow nl_q$
- 7: **Set** agents = {Prompt, Parse, Solution, WeatherHistory, WeatherForecast, Reviewer}
- 8: **Set** current_agent $\leftarrow \text{null}$
- 9: **Initiate** conversation_active $\leftarrow \text{True}$
- 10: **while** true **do**
- 11: next_agent $\leftarrow \text{selection}()$
- 12: **if** next_agent == None **then**
- 13: ReviewerAgent.invoke(null)
- 14: **break**
- 15: **else**
- 16: current_agent \leftarrow next_agent
- 17: **end if**
- 18: out $\leftarrow current_agent.\text{invoke}(chat_history)$
- 19: chat_history.append(out)
- 20: **if** $\text{termination}(\text{chat_history}, out)$ **then**
- 21: conversation_active $\leftarrow \text{False}$
- 22: **end if**
- 23: **end while**

Algorithm 4: WeatherForecastAgent.invoke

Input: parsed_query, get_forecast_data

Output: forecast_summary

- 1: $l \leftarrow \text{parsed_query.location}$
- 2: $t \leftarrow \text{parsed_query.time}$
- 3: $\text{forecast_data} \leftarrow get_forecast_data(l, t)$
- 4: $kb \leftarrow OpenAIChatCompletion(GPT-4o)$
- 5: $\text{res} \leftarrow kb.\text{GenerateSummary}(\text{forecast_data})$
- 6: **return** res

Algorithm 5: WeatherHistoryAgent.invoke

Input: parsed_query, get_NASA_data

Output: weather_summary

- 1: $l \leftarrow \text{parsed_query.location}$
- 2: $t \leftarrow \text{parsed_query.time}$
- 3: $\text{NASA_data} \leftarrow get_NASA_data(l, t)$
- 4: $kb \leftarrow OpenAIChatCompletion(GPT-4o)$
- 5: $\text{res} \leftarrow kb.\text{GenerateSummary}(\text{NASA_data})$
- 6: **return** res

Algorithm 6: SolutionAgent.invoke

Input: parsed_query p, historical weather data w, forecast data f, *get_adaptations()*

Output: solution_plan

```

1: kb ← OpenAIChatCompletion(GPT-4o LLM)
2: data ← get_adaptations(p.location)
3: solution_plan ← null
4: approved ← False
5: while approved ==False do
6:   solution_plan ← kb.GeneratePlan(data, p,
  w, f)
7:   Send solution_plan to ReviewerAgent
8:   res← ReviewerAgent.response(solution_plan)
9:   Wait for approved signal set by AgentManager
10: end while
```

Algorithm 7: ReviewerAgent.invoke

Input: parsed_query p, previous agent output a

Output: reviewer_output

```

1: kb ← OpenAIChatCompletion(GPT-4o LLM)
2: f ← get_forecast_data(p.location,p.time)
3: n ← get_NASA_data(p.location,p.time)
4: q ← get_adaptations(p.location)
5: review_result ← kb.EvaluateAgentOutput(a, f, n, q, p)
6: if review_result.NotApproved then
7:   reviewer_output ← kb.ConstructImprovement(a, review_result)
8: else
9:   reviewer_output ← "This solution is completely approved."
10: end if
11: Send reviewer_output to AgentManager
```

B Sample Decision-Support by AgroAskAI

We present the complete conversation flow between agents corresponding to Figure 2 below.

User Input (Hart 2017): “What is one way to prevent pests from invading crops in Guatemala?”

Prompt Agent: “Hello! I’m here to assist with your query. I’m gathering the necessary information and will update you shortly.”

Parse Agent (Raw Response):

```
{
  "user_intent": "get_solution",
  "location": "Guatemala",
  "start_year": "",
  "end_year": "",
  "forecast_date": ""
}

//Agents will only use "user_intent" and "location" for this query.
```

Solution Agent (Response ①):

“One effective way to prevent pests from invading crops in Guatemala is **integrated pest management (IPM)**. This sustainable approach combines various agricultural practices to protect crops while minimizing environmental impact. Considering

Guatemala’s tropical climate and crop practices, here are some practical pest control methods:

- **Crop Diversification:** Plant a variety of crops in the same field to break pest cycles and reduce the risk of widespread infestation.
- **Timely Planting and Harvesting:** Planting just before the beginning of rains or at optimal times for pest avoidance can reduce pest densities and their impact.
- **Rotate Crops:** Change crop types in different seasons. This practice is especially effective in disrupting pests that target specific crops.
- **Use of Biological Controls:** Introduce pest predators (e.g., lady beetles) or use bio-pesticides derived from natural resources, which are safer and environment-friendly.
- **Intercropping:** Planting companion crops that repel pests. For instance, marigolds are known to deter nematodes and other harmful pests.
- **Proper Sanitation:** Regularly remove crop residues and weeds to avoid providing breeding grounds for pests.
- **Planting Pest-resistant or Hybrid Varieties:** Use hybrid crop varieties that are better adapted to resist infestation.
- **Soil Conservation and Fertilizer Application:** Improve soil health with manure or compost, which supports healthier plants that are less vulnerable to pest invasion.

These techniques not only reduce pest risks but also promote sustainable agricultural practices suitable for Guatemala’s ecological conditions. Combining these strategies can provide long-term protection for crops.”

Reviewer Agent (Review of Response ①):

The SolutionAgent proposes an integrated pest management (IPM) strategy and lists several methods (e.g., crop diversification, timely planting, biological controls, intercropping, sanitation, pest-resistant crops, and soil conservation).

Strengths of the Response:

- **Addresses the User’s Context:** The response takes Guatemala’s tropical climate into account, offering regionally relevant and adaptive measures.
- **Presents Practical and Usable Methods:** Diverse techniques like intercropping, use of biological controls, and crop rotation are helpful and widely applicable.
- **Scientifically Valid:** The proposed strategies align with modern sustainable farming practices and IPM principles.

Shortcomings & Missing Aspects:

- **Fails to Focus on “One” Method:** The user explicitly asked for “one way” to prevent pest invasions, but the response offers multiple techniques without clearly identifying or explaining the most effective single solution.
- **Lacks Emphasis on Local Crops:** The response doesn’t highlight specific Guatemalan crops that might benefit from these pest prevention strategies.
- **No Mention of Social or Economic Context:** Solutions should factor in the economic realities of Guatemalan farmers, such as cost-effectiveness and resource availability for small-scale farming.

Recommendations for Improvement:

- **Prioritize a Single Key Solution:** Focus on one standout approach (e.g., intercropping or crop rotation) and provide a detailed explanation of its advantages in Guatemala’s context.

- Incorporate Specific Examples or Local Relevance:** Mention specific pests plaguing crops in Guatemala and how the selected method mitigates those pests most effectively.
- Address Feasibility Concerns:** Include practical aspects like cost, space, and labor efficiency to ensure recommendations are actionable for all farmers.

Unless updated, this response does not fully satisfy the user's request due to its broad focus and lack of prioritization.

Solution Agent (Revised Response ②):

"If we were to focus on one key and practical method to prevent pests from invading crops in Guatemala, an optimal choice would be **intercropping**. This sustainable agricultural practice involves planting various crops together, which can naturally repel pests and reduce their impact on key crops. Here's why intercropping is especially effective and feasible in the Guatemalan context:

Why Intercropping is Recommended:

- Natural Pest Repellence:** Certain companion crops, such as marigolds or garlic, release natural chemicals that deter common agricultural pests. Aromatic herbs like basil or mint can also be intercropped for pest control benefits.
- Disruption of Pest Life Cycles:** By mixing crops in the same area, pests that target specific crop species are less likely to damage extensively. This reduces the risk of significant yield losses.
- Better Resource Use:** For smallholder farmers, intercropping utilizes limited farm space more efficiently and diversifies outputs, which can help mitigate financial risks.
- Environmental and Economic Sustainability:** Reduces the need for chemical pesticides, which not only lowers costs but also minimizes environmental impact and promotes healthier soil.
- Locally Relevant Crops:** For Guatemalan farmers, combining staple crops like maize with beans or chili peppers has both pest management and economic benefits. Beans, for example, can naturally fix nitrogen into the soil while maize provides structural support.

Practical Example:

Farmers in Guatemala could plant maize (a staple crop) alongside squash and beans. This traditional "Three Sisters" intercropping method:

- Naturally repels pests,
- Improves soil fertility, and
- Maximizes yield diversity.

Additional Recommendations:

- Regular Crop Rotation:** Rotate crops with non-host plants annually to further minimize pests' adaptability.
- Training and Awareness:** Farmers could benefit from local workshops or training sessions on effective intercropping techniques and suitable companion crops for their region's pest problems.

In conclusion, intercropping stands out as a simple, cost-effective, and sustainable method for pest management in Guatemala. This approach will not only reduce pest invasions but also align with local farming traditions and resource constraints."

Reviewer Agent (Review of Response ②):

This revisited response provides a precise solution that fully focuses on intercropping as the primary method, as per the user's

request for "one way" to prevent pests. It explains the advantages, contextual relevance to Guatemala, and specific applications such as the local "Three Sisters" cropping strategy. Additionally, it considers implementation feasibility for Guatemalan farmers.

"This solution is completely approved."

C Comparison of CROPWAT, ChatGPT, and AgoAskAI

We provide the response from AgoAskAI, ChatGPT, and CROPWAT when asked "What are the climate issues in Kitui, Kenya and what can I do to protect my crops?" (Mutunga, Charles, and Patricia 2017). Figure 3 shows the response of CROPWAT, where the rainfall metrics are clearly stated and information regarding irrigation scheduling is provided. However, the solution does not include temperature metrics nor livestock and soil conservation advice, and lacks specification of supporting programs.

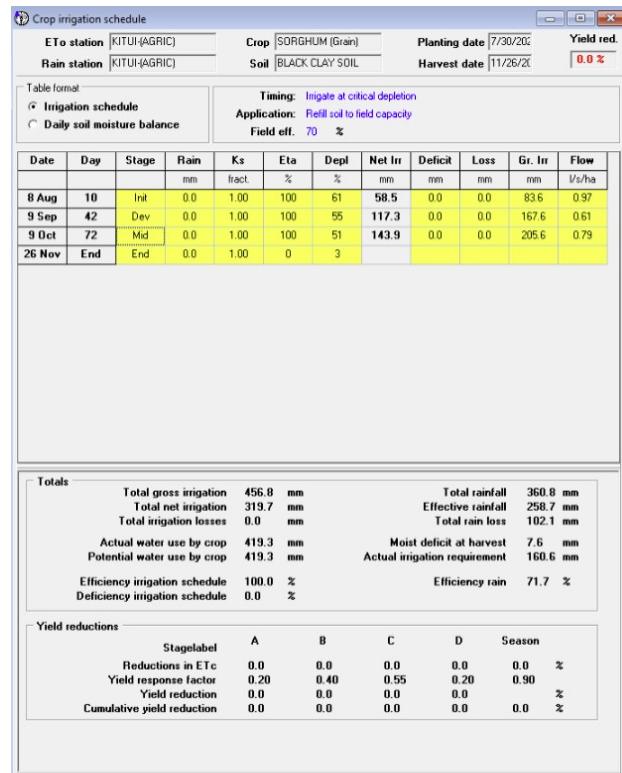


Figure 3: CROPWAT's response.

The following is the response by ChatGPT.

Climate Issues in Kitui, Kenya

1. Unpredictable Rainfall Patterns

- Rainfall is erratic and unevenly distributed.
- The "long rains" (March–May) and "short rains" (October–December) are no longer reliable.
- Leads to poor planting seasons and crop failure.

2. Frequent Droughts

- Kitui experiences recurrent droughts due to low annual rainfall (<700 mm in many areas).

- Droughts stress both crops and livestock, reducing yields and incomes.

3. High Temperatures

- Rising average temperatures increase evapotranspiration.
- Crops require more water while water sources are shrinking.

4. Land Degradation

- Deforestation and overgrazing have led to soil erosion.
- Soils are increasingly infertile and less capable of retaining moisture.

5. Pest and Disease Outbreaks

- Warmer, wetter periods following droughts often bring outbreaks (e.g., fall armyworm).
- Climate change expands the habitat of crop pests.

Adaptation Strategies for Farmers in Kitui

1. Adopt Climate-Resilient Crops

- Grow drought-tolerant and early-maturing varieties like:
 - Sorghum
 - Millet
 - Pigeon peas
 - Improved maize varieties (e.g., KDV4 or DK8031)

2. Use Water Conservation Techniques

- Zai pits or planting basins to trap water at the root zone.
- Mulching to reduce evaporation.
- Roof water harvesting and storage tanks for irrigation.

3. Agroforestry

- Plant trees like Moringa, Neem, and *Melia volkensii* to:
 - Improve soil fertility
 - Provide shade
 - Increase biodiversity
 - Act as windbreaks

4. Soil Conservation

- Use terracing, cover cropping, and composting to restore soil fertility.
- Encourage minimum tillage to preserve soil structure.

5. Improved Farming Practices

- Use intercropping and crop rotation to reduce pest pressure.
- Follow seasonal climate forecasts to decide when and what to plant.
- Join farmer cooperatives or digital platforms for training and updates.

6. Insurance & Risk Management

- Enroll in index-based crop insurance (e.g., via ACRE Africa).
- Use mobile platforms (like iShamba) for weather alerts and advice.

Examples of Projects in Kitui Supporting Farmers

- Kenya Climate Smart Agriculture Project (KCSAP)** – trains farmers on sustainable practices.
- NDMA (National Drought Management Authority)** – offers early warning systems and response plans.
- FAO & World Bank programs** – promote soil restoration and water harvesting.

The following is the response by AgroAskAI.

Based on the data available for Kitui, Kenya, here is a detailed and practical agricultural adaptation strategy tailored to the region:

Observations from Historical Weather Data:

- Rainfall:** Kitui experiences low and irregular rainfall, ranging approximately from 0.08 mm to 7.33 mm monthly during the years analyzed (2010–2022). The rains are bimodal, with most of the heavier precipitation occurring in April, May, October, and November.
- Temperatures:** Average temperatures remain relatively high, ranging from 21°C to 27°C with significant heat stress during certain months.

Recommendations for Farmers in Kitui:

To cope with Kitui's dry climatic conditions and make the most of sporadic rains:

1. Adopt Drought-Resistant Crops

- Grow crops like sorghum, millet, and cowpeas, which are well-adapted to Kitui's arid climate.
- Utilize hybrid and drought-resistant seed varieties to increase tolerance to dry conditions.
- Diversify crop choices to spread risks and improve food security.

2. Plant Timing and Early Maturing Crops

- Plant just before the rains:** Monitor weather forecasts and prepare to sow seeds shortly before expected rainfall.
- Choose early-maturing crops to complete growth cycles during rainy periods and reduce water demand.

3. Irrigation and Water Harvesting

- Irrigation:** Establish small-scale drip or bucket irrigation systems to efficiently use water for crops.
- Water Harvesting:** Construct water pans, tanks, or install gutters to collect rainwater for use during dry spells.
- Explore rock catchment systems or sand dams, which are commonly used in the region to retain water.

4. Soil Health Management and Conservation

- Use manure, compost, and fertilizers to enrich soil and improve water retention.
- Employ soil conservation methods such as terracing, mulching, and planting cover crops to prevent erosion and nutrient depletion.

5. Switching or Supplementing with Livestock Keeping

- Given the arid conditions, consider increasing reliance on livestock farming as an alternative to crop farming.
- Rear drought-tolerant breeds of goats and sheep, which are more suitable for Kitui's conditions.

6. Mixed Cropping and Crop Diversification

- Practice mixed cropping systems (e.g., combining cereals with legumes) to maximize outputs and reduce the risk of complete crop failure.
- Diversify farm production by integrating high-value crops or vegetables into farming cycles.

7. Community and Cooperative Participation

- Join local weather monitoring and farmer groups to get timely weather updates and share resources.

- Engage with agricultural extension officers for technical support and access to drought-resilient seeds.

8. Switch to Water-Efficient Practices in Farming

- For a long-term strategy, lessons like permaculture or agroforestry can be implemented to improve resilience.

By integrating these strategies into farming activities, farmers in Kitui, Kenya, can enhance their adaptability to climate variability, reduce crop failure risks, and improve agricultural resilience over time. If more specific issues arise (e.g., sudden rainfall, locust invasions, etc.), tailored solutions can be provided.