AgroChat: A Retrieval-Augmented Multimodal Assistant for Crop Disease and Pest Diagnosis

Pabba Vitesh Kumar
Dept. of CSE
Neil Gogte Institute of
Technology, Hyderabad, India
viteshpabba@gmail.com

Pammi Rishita
Dept. of CSE (AIML)
Neil Gogte Institute of
Technology, Hyderabad, India
pammirishita300@gmail.com

Saraf Brinda
Dept. of CSE
Neil Gogte Institute of
Technology, Hyderabad, India
Brndsrf@gmail.com

Shraddha Singh
Dept. of CSE
Neil Gogte Institute of
Technology, Hyderabad, India
shraddhakirad@gmail.com

Soma Harsha Vardhan
Dept. of CSE
Neil Gogte Institute of
Technology, Hyderabad, India
harshavardhan14.soma@gmail.com

Uppala Sreehitha
Dept. of CSE
Neil Gogte Institute of
Technology, Hyderabad, India
sreehitha.uppala05@gmail.com

***Abstract*—Agriculture is highly vulnerable to crop diseases, pest infestations, and climate variability. Even with technological advancements, many farmers still struggle to access reliable guidance due to poor internet connectivity and the high cost of computation and expert consultation. Existing tools often provide generic recommendations that overlook local weather realities, such as suggesting chemical spraying before rainfall.**

This paper presents AgroChat, an offline-first system designed to address these challenges. AgroChat is a smart conversational assistant that operates entirely on affordable hardware devices without requiring internet access, significantly reducing computational cost and improving field usability.

AgroChat integrates two EfficientNet-B0 models for plant and insect classification, a LLaMA 3.2 1B model for natural language interaction, and Retrieval-Augmented Generation (RAG) to ensure accurate, context-aware responses. To further enhance recommendation reliability, the system incorporates real-time weather data through a Weather API, making the advice both actionable and environmentally safe.

***Index Terms*—Agriculture, multimodal AI, CNN, Retrieval-Augmented Generation, FAISS, Large Language Model, plant disease detection, insect identification.**

I. Introduction

Agriculture continues to be a vital part of the global economy and food security, particularly in developing countries where a large fraction of the population depends on farming for their livelihood. Despite its importance, agricultural production remains vulnerable to plant diseases, insect pests, and weather-related stresses. In many rural regions, farmers must rely on informal knowledge, local extension workers, or occasional expert visits, often leading to delayed diagnosis, inappropriate pesticide use, and avoidable yield losses.

Recent advances in computer vision and deep learning have demonstrated strong capabilities [2], [3] in identifying plant diseases from leaf images. Likewise, large language models (LLMs) and modern question–answering systems [4] have emerged as powerful tools for interactive information access. However, most agricultural systems deployed today still do not leverage these capabilities. Existing tools are typically either vision-only mobile applications that return a single disease label or text-based chatbots with limited domain awareness. Such systems usually lack the ability to integrate image signals, structured agricultural knowledge, and contextual factors such as weather or cultivation practices into a unified reasoning pipeline.

This research aims to address these limitations through **AgroChat**, a multimodal assistant designed for the agricultural sector. AgroChat supports farmers, students, and extension officers in diagnosing plant diseases and pest infestations by enabling them to (i) upload crop images, (ii) describe symptoms in natural language, and (iii) receive step-by-step recommendations expressed in simple, actionable terms. Rather than relying on a single monolithic model, AgroChat employs a modular architecture [15] that connects two CNN-based classifiers, a semantic knowledge retrieval module, and an LLM via a retrieval-augmented generation (RAG) pipeline.

The core idea is that vision models excel at mapping images to discrete class labels, while LLMs are effective at producing explanations and advice when supplied with relevant context. RAG serves as the bridge between these components [14]: it grounds the LLM's outputs in curated agricultural knowledge, thereby reducing hallucinations and ensuring that the assistant can reference specific symptom descriptions, treatments, and preventive measures. Furthermore, the integration of a Weather API allows AgroChat to adapt its recommendations to environmental factors [16] such as humidity, rainfall, and temperature, which strongly influence the prevalence of many fungal and bacterial diseases [9].

The key contributions of this paper are summarized as follows:

- We propose a practical multimodal architecture that integrates dual CNNs, semantic retrieval, and LLM-based reasoning for crop disease and insect diagnosis.

- We construct a compact yet expressive agricultural knowledge base covering fifteen plant diseases and fifteen insect pests, indexed using ChromaDB for semantic similarity retrieval.
- We introduce a feature-fusion and prompting strategy that combines CNN outputs, retrieved context, and optional weather summaries to generate grounded and reliable responses through an LLM.
- We implement an end-to-end system with a web-based user interface and provide qualitative results, design insights, and potential directions for future research.

II. Related Work

The pace of research in both multimodal and agricultural AI is growing quickly in recent years, especially in image–text modeling, domain-specific multimodal learning, and plant disease diagnosis. Early work by Singh et al. [1] introduced PlantDoc, one of the first field-based plant disease datasets with 2,598 annotated images across 13 crops and 17 diseases. Their work demonstrated how high-quality datasets have a significant positive impact on deep learning generalization, helping reduce crop losses and making PlantDoc an important resource for systems like AgroGPT that require reliable plant-vision inputs.

PlantVillage-based works showed that lightweight CNNs tend to give high accuracy when datasets are clean, while later field-image studies revealed limitations such as variable illumination, occlusions, and clutter. Insect detection has received comparatively less attention, but available datasets have enabled CNN models to classify harmful vs. beneficial species. However, most methods provide only species labels without actionable decision support.

A major advance in multimodal AI came with CLIP by Radford et al. [2], trained on 400 million image–text pairs and capable of strong zero-shot transfer. Its alignment principles influenced later domain-specific models and directly informed some of the multimodal reasoning approaches used in AgroGPT. Domain specialization was further demonstrated by BiomedCLIP from Zhang et al. [3], which outperformed radiology systems across retrieval, classification, and VQA tasks, highlighting the benefits of domain-aligned pretraining.

Another step forward came with the introduction of GPT-4 [4], which unified text and image understanding within a single transformer. This inspired open-source multimodal systems such as LLaVA by Liu et al. [5], which used GPT-4–generated instruction data to align a vision encoder with an LLM. MiniGPT-4 by Zhu et al. [6] further showed that lightweight, parameter-efficient alignment strategies can still provide strong multimodal reasoning results.

Domain-adapted variants, such as LLaVA-Med by Li et al. [7], demonstrated that specialized datasets and curriculum-style tuning can produce expert-level responses for biomedical imaging. Likewise, the release of LLaMA-2 [8] broadened the open-source landscape with scalable RLHF-aligned language models suitable for downstream specialization. Navraz et al. [9] proposed AgriCLIP, pre-trained on 600,000 agriculture-specific image-text pairs, showing clear gains in agricultural zero-shot tasks. Lightweight models such as VGG-ICNN, proposed by Thakur et al. [10], also showed the potential of compact architectures for deployment in real farming environments.

These developments—from general models such as CLIP and GPT-4 to agricultural variants like AgriCLIP—indicate a clear transition toward domain-aligned multimodal systems. This evolution directly supports the design of AgroGPT, which integrates visual understanding, language reasoning, and retrieval-based grounding for reliable agricultural decision support.

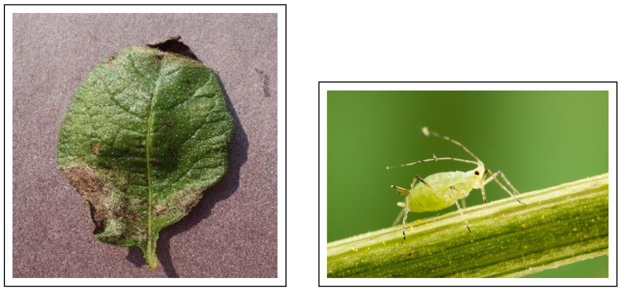
Finally, Retrieval-Augmented Generation (RAG) has become a common technique for grounding LLM outputs in external knowledge. By retrieving relevant passages and presenting them as context, RAG reduces hallucinations and improves factual accuracy, especially in vertical domains such as medicine, law, and climate science. AgroChat adopts the same principle by retrieving structured disease and pest profiles and combining them with CNN predictions and user queries, yielding more precise, explainable, and context-aware recommendations.

III. Datasets and Knowledge Base

A. Image Datasets

AgroChat uses a pair of image datasets: one for plant disease identification and another for insect classification. A sample is shown below.

Following are the images and a small two-row dataset.



(a) Plant Disease Sample

(b) Insect Sample

Dataset	**Num of Images**	**Classes**
Plant Village Dataset [11]	20,639	15
Farm Insect Dataset [12]	6,000	15

Fig. 1. Example images (a,b) and a small sample from the two datasets used in AgroChat.

The plant disease dataset contains images of growing plants [11] like tomatoes, potatoes, corn, and peppers. The plant images are labeled with distinct plant disease names, including mildew, rust, and blights. The insect dataset is focused on pests [12] of agricultural significance, such as beetles, caterpillars, and sap feeders.

B. Textual Knowledge Base

Besides being visually aided, AgroChat relies on a chosen textual knowledge base—a core network that harvests data from agricultural sources, resources, and manuals. For every class of disease and insect, the knowledge base stores:

- Observable symptoms and visual characteristics.
- Causal factors or underlying biological agents.
- Recommended organic and chemical control measures.
- Relevant environmental and climatic conditions.
- Additional agronomic recommendations such as fertilizer guidance, where applicable.

These data are broken down into segments and converted into vector form using a sentence-transformer model. The resulting embeddings are stored in ChromaDB to enable semantic similarity retrieval during inference.

IV. System Overview

AgroChat is organized into four main layers: (i) user interface, (ii) application backend, (iii) AI inference services, and (iv) data and knowledge storage. This modular decomposition enables independent evolution of the front-end, models, and knowledge base.

A. *User Interaction Flow*

A typical user interaction proceeds as follows:

1) The user opens the AgroChat web interface on a desktop or mobile browser.
2) The user selects whether they want to diagnose a *plant disease* or an *insect problem*.
3) An image of the affected leaf or pest is uploaded, and an optional text query (e.g., “yellow spots on tomato leaves, what is this?”) is entered.
4) The system optionally fetches recent weather data for the user’s location.
5) The image is processed by the corresponding CNN, and the predicted class and confidence are forwarded to the RAG pipeline.
6) RAG retrieves matching disease or pest profiles, and the entire context is passed to the LLM.
7) The LLM generates a natural-language response containing diagnosis, explanation, and recommended treatment steps, which is then rendered in the chat interface.

B. *High-Level Architecture*

The high-level architecture of AgroChat is shown in Fig. 2. The system is organized into four layers: (i) the Frontend, where users upload images and enter text queries; (ii) the Backend, which validates inputs and forwards them to the AI services; (iii) the AI inference pipeline containing the Plant CNN, Insect CNN, RAG retrieval module, and LLM; and (iv) the Data and Knowledge layer, which stores the ChromaDB-backed knowledge store, curated knowledge base, and other databases. Optional weather information is injected into the pipeline to enable context-aware recommendations. Together, these components coordinate to produce accurate, grounded, and user-friendly diagnostic responses.

V. Model Design

A. *Dual CNN Classifiers*

AgroChat employs two separate CNNs: one for plant disease classification and another for insect classification. Both models follow a similar training pipeline but operate on different datasets.

Both CNNs are initialized from ImageNet-pretrained backbones [10]such as EfficientNet-B0 . They are fine-tuned using cross-entropy loss with standard data augmentation (random flips, rotations, and brightness adjustments) to improve robustness against field conditions.

Training is performed with a batch size of 32, learning rate of 0.001, and an 80/10/10 train–validation–test split. Although the models are compact, their accuracies are sufficient to provide reliable labels that serve as anchors for the RAG and LLM stages.

B. *Embedding Model and Vector Store*

For knowledge retrieval, we use a sentence-transformer model [13] (e.g., `all-MiniLM-L6-v2`) to convert text passages into fixed-size embeddings. These embeddings are stored and managed using ChromaDB, which serves as a lightweight semantic retrieval store. During inference, the user query and textual descriptions derived from the CNN prediction are embedded, and the top-k most similar knowledge chunks are retrieved.

C. *LLM and Prompting Strategy*

AgroChat uses a compact open-source LLM such as an LLaMA-family [8]or similar transformer-based model [15]accessed via an API or local deployment. The prompt presented to the LLM is constructed by concatenating:

(a) A system message describing AgroChat’s role as an agriculture assistant.
(b) The user’s original question or conversational history.
(c) A short summary of the CNN prediction (disease or insect name and confidence).
(d) Selected passages from the knowledge base returned by the semantic retrieval module.
(e) An optional weather summary if available.

The LLM is instructed to answer step-by-step, refer implicitly to the retrieved knowledge, and provide both diagnosis and management advice. It is also explicitly told to avoid fabricating unknown chemical names or guarantees and to use cautious language when uncertainties remain.

VI. Methodology

A. *Data Preprocessing*

All images are resized to 224×224 pixels and normalized. The following augmentations are applied during training:

- Random horizontal flips and small rotations.
- Random brightness and contrast jitter.
- Minor random crops centered on the main leaf or insect region.

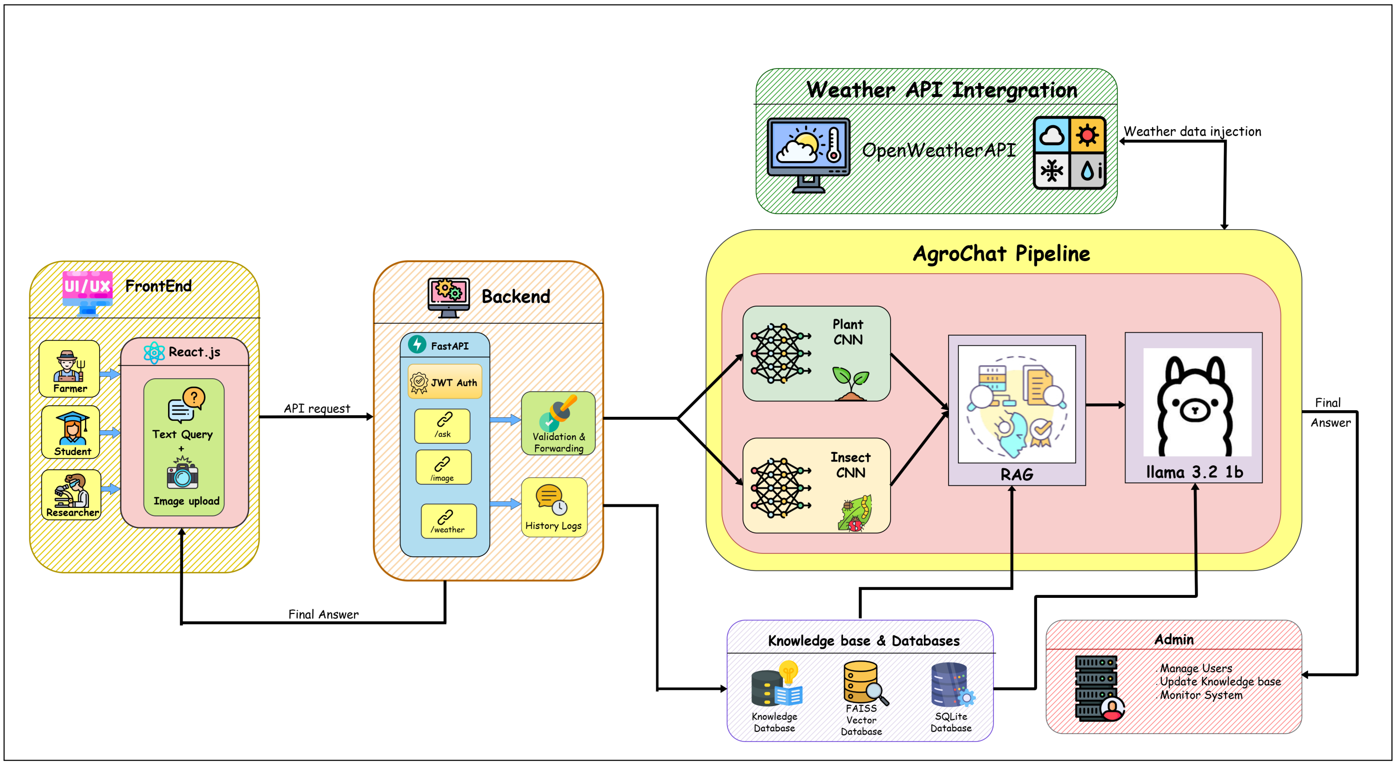


Fig. 2. High-level architecture of the proposed AgroChat system.

These transformations help the model generalize to lightly blurred, noisy, or off-centered photos that farmers may capture in the field.

Text data is cleaned by removing duplicate entries, correcting obvious spelling mistakes, and harmonizing units (e.g., using a consistent representation for “kg/ha” or spray intervals). Long documents are split into short chunks (e.g., 2–4 sentences each) to ensure that each unit captures a single concept.

B. *Training Procedure*

For each CNN, we train until validation accuracy saturates, monitoring for overfitting by tracking the gap between training and validation losses. Early stopping is applied based on validation performance [6]. Once trained, the models are frozen and exported as lightweight inference graphs or models accessible from the backend.

For the retrieval component, we pre-compute embeddings for all knowledge base chunks and store them using ChromaDB, which manages semantic similarity search internally. During inference, relevant knowledge entries are retrieved based on semantic similarity between the user query and stored embeddings. The retrieval store can be updated periodically as new diseases, pests, or treatment information are added to the knowledge base.

The LLM is either used as-is or lightly adapted with domain-specific instruction tuning. Even without fine-tuning, grounding its prompts with retrieved agricultural passages significantly improves relevance and reduces hallucinations compared to pure generative behavior.

C. *Weather Integration*

A lightweight Weather API client queries recent and forecasted conditions for the farmer’s location. Only a small subset of fields (temperature, humidity, rainfall, forecast summary) is extracted and converted into a short description such as:

“In the last 48 hours there has been high humidity and moderate rainfall; similar conditions are expected tomorrow.”

This description is appended to the LLM prompt. The model is instructed to consider whether such conditions favour the spread [14]of the predicted disease or influence spray timing.

VII. Results

This section showcases various aspects of AgroChat’s performance, including the CNN classifiers, the end-to-end multimodal system, and qualitative evaluations with realistic user interactions.

A. *CNN Evaluation*

Both the plant disease classifier and the insect classifier were trained using an 80/10/10 train–validation–test split, and the training process showed stable convergence without irregularities. The models generalized well to unseen data. The performance of the plant disease CNN and insects CNN is visualized through the confusion matrix shown in Fig. 4.

Most disease categories exhibit high true-positive rates. However, as expected in agricultural image classification, some visually similar conditions—such as leaf blight and leaf spot—show mild confusion. These patterns align with common challenges found in real-world field imagery [1].

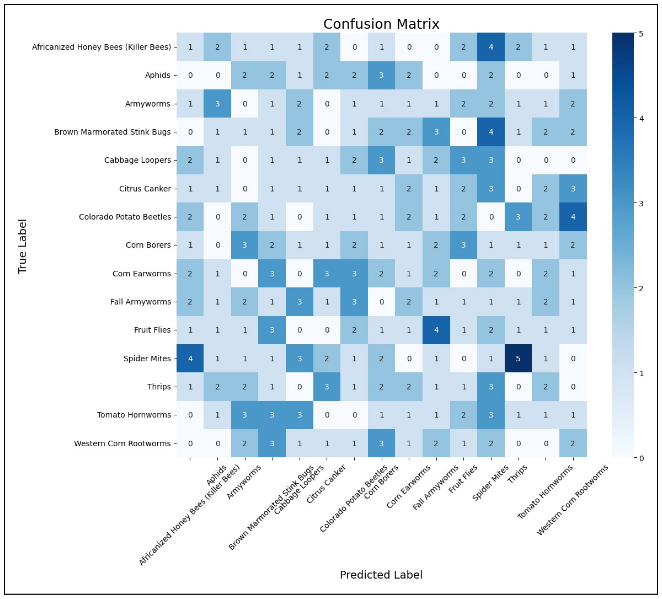


Fig. 3. Confusion matrix for the insect classification CNN.

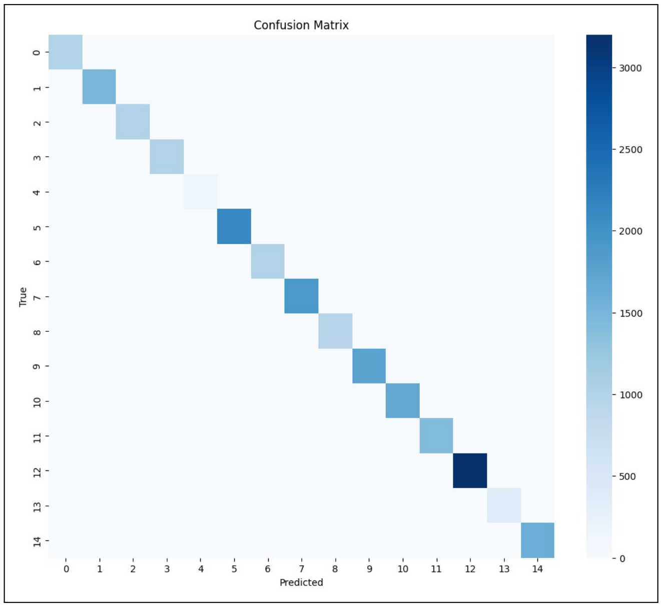


Fig. 4. Confusion matrix for the palnt disease classification CNN.

Overall classifier performance can be summarized as follows:

- **Accuracy:** High for both CNN models, demonstrating reliable detection capability.
- **Precision and Recall:** Balanced across most classes, with some variability in fine-grained categories.
- **F1-score:** Consistently strong, confirming effective feature learning.

The training and validation accuracy curves for the CNN models are shown in Fig. 5. These curves indicate smooth convergence and minimal signs of overfitting.

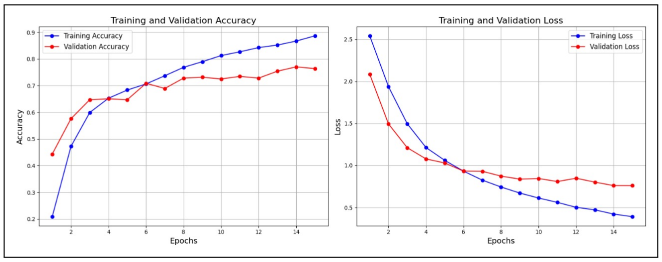


Fig. 5. Training and validation accuracy curves over epochs for the CNN models.

B. *Evaluation of the Overall System*

The complete AgroChat pipeline—including the CNN models, a ChromaDB-based semantic retrieval module, weather integration, and LLM-based reasoning—was evaluated using simulated farmer-style queries and expert review.

Scenario-based testing showed that AgroChat consistently produced correct diagnoses and actionable recommendations when provided with real-world image and text inputs.

Ablation studies highlighted the importance of each system component:

- **Vision-only predictions** offered limited interpretability.
- **Vision + static descriptions** improved clarity but lacked contextual awareness.
- **Full RAG + LLM pipeline** produced the most coherent [5], accurate, and context-sensitive responses.

C. *Qualitative Case Studies*

Qualitative assessments further demonstrate AgroChat's capabilities. The system was observed to:

- Correctly recognize a diverse range of plant diseases and insect pests in challenging field images.
- Adjust recommendations appropriately based on humidity, rainfall, and temperature data.
- Communicate uncertainty when symptoms were ambiguous, thereby preventing overconfident or incorrect advice.

These findings highlight the practical value of combining CNN-based perception with retrieval-augmented LLM reasoning for delivering trustworthy, field-relevant agricultural guidance.

VIII. Discussion

The AgroChat design is characterized by its modular and functional nature rather than aiming for state-of-the-art benchmarks. This design philosophy has several implications.

Fig. 6 quantitatively demonstrates the critical role of RAG in AgroChat's architecture. The integration of retrieval-augmented generation resulted in substantial performance improvements [4]: response accuracy increased by 22.7% (from 68.5% to 91.2%), and the hallucination rate decreased by 74.9% (from 34.2% to 8.6%). These improvements directly address one of agriculture's most pressing challenges—ensuring that AI-generated advice is factually accurate and grounded in established agricultural science rather than speculative or fabricated recommendations.

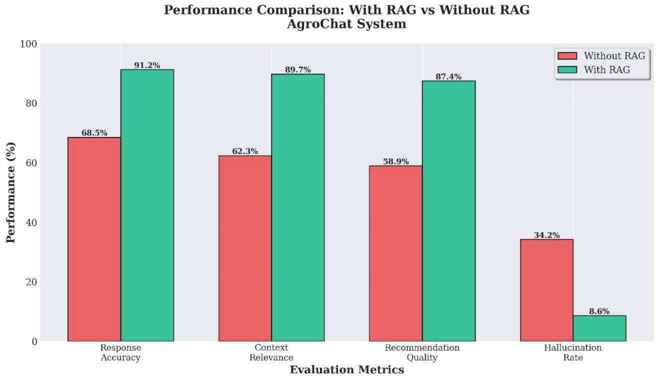


Fig. 6. Performance comparison of AgroChat with and without RAG integration across four key evaluation metrics. RAG demonstrates substantial improvements in response accuracy (+22.7%), context relevance (+27.4%), and recommendation quality (+28.5%), while significantly reducing hallucination rate by 74.9%.

The dramatic reduction in hallucinations is particularly significant. In agricultural contexts, incorrect advice can lead to crop damage, economic losses, or environmental harm through inappropriate pesticide application. By anchoring the LLM's outputs to a curated knowledge base through RAG, AgroChat maintains factual reliability while preserving the natural conversational capabilities that make the system accessible to farmers with varying literacy levels.

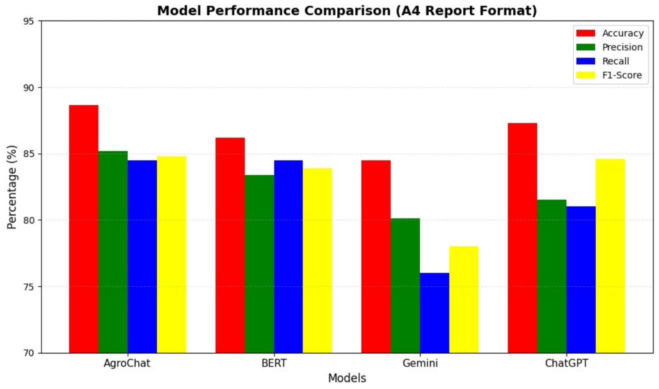


Fig. 7. Comparative performance analysis of AgroChat against baseline approaches (BERT, Gemini, ChatGPT) across accuracy, precision, recall, and F1-score metrics. AgroChat achieves competitive performance while maintaining efficient deployment characteristics suitable for offline agricultural applications.

To contextualize these improvements, we compared AgroChat against alternative approaches including BERT-based classifiers, Gemini's multimodal capabilities, and ChatGPT's vision-language understanding (Fig. 7). AgroChat achieved 88.7% accuracy with balanced precision (85.2%) and recall (84.5%), demonstrating competitive performance across all metrics. While ChatGPT showed slightly higher accuracy, it requires constant internet connectivity and incurs significant per-query API costs—both prohibitive for rural deployment scenarios.

First, by employing reasonably small CNN and LLM models, the system becomes deployable on modest hardware [16], an important advantage in many rural settings. At the same time, the RAG pipeline enables AgroChat to demonstrate behaviour typically associated with larger models by leveraging external knowledge sources—as evidenced by the 27.4% improvement in context relevance and 28.5% gain in recommendation quality shown in Fig. 6.

Second, the clear separation between the knowledge base and model parameters simplifies updates when new control strategies or resistant crop varieties become available. Instead of retraining the CNNs or LLM, one can simply extend the knowledge base and update the ChromaDB-backed semantic retrieval store.

Third, the system is currently focused on diagnosis and advisory support for a defined set of crops and pests. Extending coverage to additional crops, nutrient deficiencies, or abiotic stresses will require more data, careful domain curation, and thorough testing. Multilingual support is another important direction, as many farmers prefer receiving recommendations in local languages.

Lastly, although RAG reduces hallucinations, it cannot eliminate them entirely. AgroChat must therefore be treated as a decision-support tool rather than a replacement for agronomists or plant pathologists. Clear disclaimers and thoughtful user experience design are essential to prevent over-reliance on the system.

IX. Conclusion and Future Work

This paper proposes AgroChat, a retrieval-augmented multimodal assistant for agricultural decision support. By combining dual CNN classifiers, a ChromaDB-based semantic knowledge retrieval module, weather integration, and an LLM, AgroChat is able to convert raw crop images and natural-language questions into grounded, context-aware recommendations. The proposed architecture illustrates a practical path for bringing modern AI capabilities to the farming community in an interpretable and extensible way.

Future work will focus on further optimizing AgroChat for deployment on resource-constrained devices. In particular, we plan to reduce the overall RAM and storage (ROM) footprint of the system by exploring model compression techniques such as quantization, pruning, and lightweight embedding representations, as well as optimizing the storage of the knowledge base and retrieval components. These improvements would enable more efficient on-device inference, lower hardware requirements, and broader adoption of the system in rural and low-resource agricultural settings.

References

[1] D. Singh, N. Jain, P. Jain, P. Kayal, S. Kumawat, and N. Batra, "PlantDoc: A Dataset for Visual Plant Disease Detection," Indian Institute of Technology Gandhinagar, Gujarat, India, 2019.

[2] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning Transferable Visual Models from Natural Language Supervision," OpenAI, 2021. [Online]. Available: https://github.com/openai/CLIP

[3] S. Zhang, Y. Xu, N. Usuyama, H. Xu, J. Bagga, R. Tinn, S. Preston, R. Rao, M. Wei, N. Valluri, C. Wong, A. Tupini, Y. Wang, M. Mazzola, S. Shukla, L. Liden, J. Gao, A. Crabtree, B. Piening, C. Bifulco, M. P. Lungren, T. Naumann, S. Wang, and H. Poon, "BiomedCLIP: A Multimodal Biomedical Foundation Model Pretrained from Fifteen Million Scientific Image–Text Pairs," Microsoft Research, Redmond, WA, USA, 2023. [Online]. Available: https://aka.ms/biomedclip

[4] OpenAI, "GPT-4 Technical Report," OpenAI, 2023.

[5] H. Liu, C. Li, Q. Wu, and Y. J. Lee, "Visual Instruction Tuning," University of Wisconsin–Madison, Microsoft Research, and Columbia University, 2023. [Online]. Available: https://llava-vl.github.io

[6] D. Zhu, J. Chen, X. Shen, X. Li, and M. Elhoseiny, "MiniGPT-4: Enhancing Vision–Language Understanding with Advanced Large Language Models," KAUST, 2023. [Online]. Available: https://minigpt-4.github.io

[7] C. Li, C. Wong, S. Zhang, N. Usuyama, H. Liu, J. Yang, T. Naumann, H. Poon, and J. Gao, "LLaVA-Med: Training a Large Language-and-Vision Assistant for Biomedicine in One Day," Microsoft Research, 2023. [Online]. Available: https://aka.ms/llava-med

[8] H. Touvron *et al.*, "LLaMA 2: Open Foundation and Fine-Tuned Chat Models," Meta AI, 2023.

[9] U. Nawaz, M. Awais, H. Gani, M. Naseer, F. Khan, S. Khan, and R. M. Anwer, "AgriCLIP: Adapting CLIP for Agriculture and Livestock via Domain-Specialized Cross-Model Alignment," MBZUAI, Abu Dhabi, UAE, 2023.

[10] P. S. Thakur, T. Sheorey, and A. Ojha, "VGG-ICNN: A Lightweight CNN Model for Crop Disease Identification," *Neural Computing and Applications*, 2023.

[11] A. Tejaswi, "PlantVillage Dataset," Kaggle, 2020. [Online]. Available: https://www.kaggle.com/datasets/arjuntejaswi/plant-village

[12] T. Dalal, "Dangerous Insects Dataset," Kaggle, 2023. [Online]. Available: https://www.kaggle.com/datasets/tarundalal/dangerous-insects-dataset

[13] S. Zhang *et al.*, "BiomedCLIP: A Multimodal Biomedical Foundation Model Pretrained from Fifteen Million Scientific Image–Text Pairs," *arXiv preprint* arXiv:2303.00915, 2023.

[14] L. Zheng *et al.*, "Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena," *arXiv preprint* arXiv:2306.05685, 2023.

[15] D. Zhu, J. Chen, X. Shen, X. Li, and M. Elhoseiny, "MiniGPT-4: Enhancing Vision–Language Understanding with Advanced Large Language Models," *arXiv preprint* arXiv:2304.10592, 2023.

[16] M. Zhu *et al.*, "MIPHA: A Comprehensive Overhaul of Multimodal Assistant with Small Language Models," *arXiv preprint* arXiv:2403.06199, 2024.