AgriBuddy Technical Reports
Bangladesh Innovation Conclave

Shaleh Tonmoy*, Rawhatur Rabbii Rafin*, Md. Rezuwan Hassan*,Ruwad Naswan†,Tanmoy Shome‡
School of Data and Sciences, Brac University
Computer Science Engineering, BUET
Software Engineering, Daffodil International University
Email:* [shaleh.tonmoy, rawhatur.rabbii.rafin, md.rezuwan.hassan]@bracu.edu.bd,
† ruwad.naswan@buet.edu.bd,‡ tanmoy10-2099@diu.edu.bd

***Abstract*—Agriculture in Bangladesh faces significant challenges due to the increasing demand for timely, personalized farming advice amidst unpredictable climate conditions and limited access to expert resources, particularly for remote farmers. AgriBuddy, developed by the AgriBRACUtion team, addresses these issues through an LLM-based agentic AI system powered by Retrieval-Augmented Generation (RAG). Designed as an interactive Bangla chatbot, AgriBuddy enables farmers to engage in natural language conversations, share images, request farming plans, and receive real-time expert recommendations tailored to regional agricultural practices. The system integrates multiple AI agents—including a Smart Query Handling Agent, User Context & Memory Agent, and Expert Agricultural RAG Agent—alongside a convolutional neural network (CNN) for rice disease detection from images.**

***Index Terms*—Agent, RAG, LLM, Tool**

I. Introduction

Agriculture remains a cornerstone of Bangladesh's economy, supporting over 40% of its workforce and contributing significantly to its GDP, yet it faces mounting challenges due to climate variability, rising production demands, and limited access to expert guidance [1]. Traditionally, farmers have depended on intergenerational knowledge and local observations, with agricultural offices providing supplementary advice. However, the increasing volume of consultation requests, coupled with the logistical barriers of frequent visits—particularly for remote or hobbyist farmers—has hindered timely, personalized support [2]. As climate change exacerbates weather unpredictability and farming evolves into both a professional livelihood and a recreational pursuit, there is a pressing need for accessible, real-time solutions that deliver region-specific insights and actionable recommendations to diverse farming communities.

To address these challenges, the AgriBRACUtion team introduces AgriBuddy, an advanced LLM-based agentic AI system powered by Retrieval-Augmented Generation (RAG) technology. Tailored to Bangladesh's agricultural context, AgriBuddy functions as an interactive Bangla chatbot, enabling farmers to engage in natural language conversations, share crop images, request tailored farming plans, and receive expert advice on diseases, weather, and more [3]. Built on a modular architecture that integrates AI agents, a convolutional neural network (CNN) for rice disease detection, and a comprehensive knowledge base, AgriBuddy aims to revolutionize agricultural decision-making. This technical report details the design, implementation, and future potential of AgriBuddy, emphasizing its role in fostering sustainable farming practices and supporting farmers across urban and rural landscapes.

II. Problem Statement

Agriculture, a vital profession in regions like Bangladesh, demands continuous adaptation and informed decision-making amidst evolving environmental and economic pressures. Historically, farmers have depended on generational wisdom and local observations, supplemented by agricultural offices that provide expert guidance. However, the escalating volume of consultation requests has overwhelmed these offices, hindering the delivery of timely and personalized support to farmers. With rising production demands and increasingly unpredictable climate conditions, especially in remote areas, farmers require accessible expert advice to refine their strategies, stay abreast of agricultural trends, and adapt to shifting circumstances. For many, farming is a full-time profession necessitating constant support, while for others, it is a hobby seeking general insights to enhance their experience. Yet, frequent visits to agricultural offices are impractical, particularly for those with limited access or resources. This gap leaves farmers—both professional and hobbyist—without reliable, always-available guidance for daily practices and long-term planning, underscoring the need for an innovative solution to empower agricultural decision-making across diverse communities.

III. Data Preparation

The AgriBuddy system relies on a comprehensive dataset tailored for Bangladesh's agricultural landscape, encompassing paddy variants, disease identification, and disease management solutions. This section outlines the data collection and preparation processes, addressing the challenges of extracting Bangla text from ANSI-encoded sources and ensuring data quality through validation.

A. Paddy Variant Data Collection and Processing

Paddy variant data was sourced from the Bangladesh Rice Research Institute (BRRI) website, where agricultural information is published in PDF format, primarily written in Bangla using ANSI encoding. Initial attempts to extract text directly from these PDFs failed due to encoding incompatibilities. To overcome this, we adopted a two-pronged approach based on the PDF type.

For PDFs requiring image-based extraction, we converted each PDF into high-resolution images using standard PDF-to-image conversion tools. The Bangla text was then extracted from these images using Tesseract-OCR, configured with the Bangla language pack to ensure accurate character recognition. For PDFs in direct ANSI format, we manually copied the text and converted it to Unicode using the ANSI2Unicode web portal, ensuring compatibility with modern text processing pipelines. Post-extraction, a manual validation step was conducted to verify the vocabulary against the original PDFs. In cases where extracted text was incorrect or incomplete (e.g., due to OCR errors or diphthong compound letter mismatches), we manually transcribed the affected Bangla words by cross-referencing with the source PDFs, ensuring data integrity.

B. Paddy Disease Dataset

The disease identification component of AgriBuddy leverages the open-source Paddy Doctor dataset, which contains images of rice crops annotated across 10 distinct disease classes. This dataset provides a robust foundation for training the convolutional neural network (CNN) model integrated into AgriBuddy, enabling accurate detection and classification of rice diseases from farmer-submitted images. The dataset was preprocessed to standardize image dimensions to 254x254 pixels, aligning with the input requirements of the CNN model, and augmented to enhance model robustness against varying image conditions such as lighting and orientation.

C. Paddy Disease Solution Dataset

To provide actionable disease management recommendations, we scraped data from the PlantwisePlus Knowledge Bank website, a reputable source for agricultural solutions. This dataset includes detailed information on disease management practices, such as pesticide application, cultural practices, and preventive measures, specific to paddy diseases. The scraped data was cleaned to remove HTML artifacts and irrelevant metadata, then structured into a format compatible with AgriBuddy's Retrieval-Augmented Generation (RAG) framework. Each entry was mapped to corresponding disease classes from the Paddy Doctor dataset, ensuring seamless integration between disease identification and solution retrieval.

D. Data Validation and Integration

To ensure the reliability of the prepared datasets, a rigorous validation process was employed. For the paddy variant data, manual checks confirmed the accuracy of extracted Bangla text, addressing errors in diphthong compound letters by similarity matching with the original PDFs, as specified in the project guidelines. The disease and solution datasets were cross-validated to ensure consistency between identified diseases and recommended solutions, with mappings verified against agricultural literature. The validated data was then integrated into AgriBuddy's knowledge layer, utilizing PostgreSQL for structured data (e.g., disease-solution mappings) and ChromaDB for vector embeddings of unstructured text, facilitating efficient retrieval within the RAG framework.

IV. Method

This section details the methodology behind AgriBuddy, an LLM-based agentic AI system designed to deliver personalized agricultural recommendations to farmers in Bangladesh. The system integrates multiple AI agents, a Retrieval-Augmented Generation (RAG) framework, and a convolutional neural network (CNN) for rice disease detection, all tailored to handle Bangla language queries and regional agricultural data.

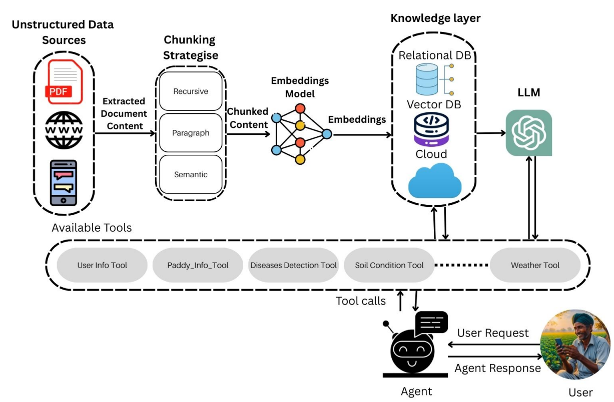

Figure 1: Aggribuddy Agentic Rag Architecture

A. Smart Query Handling Agent

The Smart Query Handling Agent processes farmer inputs, refining queries to ensure accuracy and relevance. When a farmer submits a query (e.g., "What diseases affect my rice crop?"), the agent parses the input using Bangla NLP libraries to extract key entities such as crop type and symptoms. If critical details are missing—such as location or soil condition—the agent generates follow-up questions to gather necessary context (e.g., "What is your farm's location?"). This iterative process ensures well-structured queries for downstream processing, leveraging rule-based heuristics and semantic analysis to disambiguate vague inputs.

B. User Context and Memory Agent

The User Context and Memory Agent maintains a persistent user profile to enable personalized responses. It stores offline data such as land size, livestock details, past queries, and local climate conditions in a PostgreSQL database, ensuring privacy through anonymization and secure data handling protocols. During interactions, the agent retrieves historical data to contextualize responses—for instance, recalling a farmer's previous pest issues to tailor current advice. This memory mechanism enhances response relevance by integrating user-specific insights with real-time query data.

C. Expert Agricultural RAG Agent

The Expert Agricultural RAG Agent forms the core of AgriBuddy's recommendation system, combining retrieval and generation capabilities to provide region-specific advice. The RAG framework operates in two stages: retrieval and generation. First, it retrieves relevant documents from a knowledge base comprising Bangladesh's agricultural databases, research papers, and expert advisories. Documents are chunked using recursive, paragraph-level, and semantic strategies to preserve context, then converted into vector embeddings using OpenAI's `text-embedding-3-small` model, optimized for Bangla language processing. These embeddings are stored in a ChromaDB vector database for efficient similarity search. Given a query, the agent retrieves the top-k relevant documents based on cosine similarity, then uses a fine-tuned large language model (LLM) to generate a response. The LLM, trained on Bangla agricultural terminology, integrates retrieved context with user data from the memory agent to produce fact-based recommendations, such as pest control methods suited to a farmer's region or weather precautions based on real-time forecasts.

D. Rice Disease Detection with CNN

AgriBuddy incorporates a convolutional neural network (CNN) to detect and classify rice diseases from farmer-submitted images, enhancing its diagnostic capabilities. The CNN architecture is designed as follows: an input layer accepts images resized to 254x254 pixels, followed by a 2D convolution layer (`conv2d`) with 128 filters to extract feature maps. Max pooling layers (`max_pooling2d`) reduce spatial dimensions, improving computational efficiency. Subsequent convolutional layers with decreasing filter sizes capture refined features, followed by flattening and dense layers for classification. Dropout layers are interspersed to mitigate overfitting, and the output layer, with 10 units, corresponds to the 10 disease classes in the Paddy Doctor dataset. The model, implemented using PyTorch, was trained on this dataset with data augmentation techniques (e.g., rotation, flipping) to enhance robustness. Upon receiving an image, the CNN predicts the disease class, which is then passed to the RAG agent to retrieve corresponding management solutions.

E. Integration of Specialized Tools

AgriBuddy integrates several specialized tools to augment its functionality. The User Info Tool manages farmer profiles, storing preferences and historical data for personalization. The Paddy Info Tool provides rice cultivation expertise, such as optimal planting schedules, sourced from structured data in PostgreSQL. The Disease Detection Tool interfaces with the CNN to process crop images and suggest treatments, while the Soil Condition Tool analyzes soil data (e.g., pH, nutrient levels) to recommend amendments. Finally, the Weather Tool leverages APIs like PyOWM to deliver forecasts and weather-related guidance, ensuring farmers receive holistic recommendations.

F. Knowledge Layer Architecture

The knowledge layer underpins AgriBuddy's data-driven approach, combining relational and vector databases. Structured agricultural data, such as crop calendars and pest guides, is stored in PostgreSQL, enabling efficient querying for static information. Unstructured data, including research papers and web content, is processed through the RAG pipeline: documents are chunked, embedded, and indexed in ChromaDB for fast retrieval. This dual-database architecture ensures scalability and flexibility, supporting both precise data lookups and semantic searches critical for generating context-aware responses.

V. Deployment

The deployment of AgriBuddy ensures robust performance, accessibility, and scalability across diverse agricultural regions in Bangladesh. The system leverages a cloud-native architecture with modern DevOps practices to deliver high availability and minimal latency, even in remote rural areas.

A. Infrastructure Setup

AgriBuddy utilizes a dual-tier deployment architecture. The frontend is hosted on Vercel, benefiting from its serverless infrastructure and global CDN for efficient content delivery. This setup ensures a responsive and mobile-optimized user interface across devices. The backend, containerized using Docker, is deployed on AWS Elastic Beanstalk, offering auto-scaling capabilities and high availability. Databases are distributed across nodes to ensure data reliability and low-latency access. To accommodate bandwidth constraints in rural regions, the system integrates AWS edge computing capabilities. Additionally, a progressive web application (PWA) is used to provide offline functionality and low data consumption, enhancing usability for farmers with intermittent connectivity or limited data plans.

B. Rollout Strategy

AgriBuddy follows a phased deployment model. The initial rollout targets regions with strong internet connectivity, followed by gradual expansion into remote areas. Outreach is supported by collaborations with agricultural cooperatives and extension services to drive user adoption. A comprehensive training program underpins deployment success. This includes farmer workshops, illustrated user manuals in local dialects, and dedicated support channels. Community feedback is continuously gathered to improve system usability and relevance.

C. CI/CD and Containerization

The backend leverages a microservices architecture using Docker for environment consistency and scalability. Multi-stage builds are implemented to optimize container size and performance. A CI/CD pipeline ensures automated testing and deployment. Commits to the main branch trigger automatic frontend deployment to Vercel, while updated Docker images are pushed to a container registry and deployed to Elastic Beanstalk. This process enables zero-downtime deployments, ensuring uninterrupted service delivery to users.

VI. Conclusion

AgriBuddy bridges the gap between farmers and expert agricultural knowledge through AI-driven, personalized support tailored to Bangladesh's needs. With its intelligent agents, localized datasets, and mobile-optimized deployment, it ensures reliable guidance even in remote areas. As it scales, AgriBuddy aims to enhance crop coverage, integrate voice and IoT tools, and further empower farmers with data-driven decisions for a more sustainable and resilient agricultural future. AgriBuddy plans to expand to include soil condition analysis, weather forecasting, and multi-modal input processing, aiming to scale access for diverse farmers through enhanced cloud infrastructure and community-driven features.

Acknowledgment

We would like to express our sincere gratitude to Prof. Satabdha for his invaluable guidance and insightful contributions during the initial idea generation phase of the AgriBuddy project. His support played a crucial role in shaping the foundation of our work.

References

[1] World Bank Group, "Bangladesh Development Update: Navigating climate and economic challenges." 2024, accessed: 2023-10.

[2] FAO, "The State of Food and Agriculture: Leveraging agricultural automation for transforming agrifood systems. Food and Agriculture Organization of the United Nations." 2022.

[3] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, S. Riedel, and D. Kiela, "Retrieval-augmented generation for knowledge-intensive nlp tasks," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020, pp. 9459–9474. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2020/file/6b493230205f780e1bc26945df7481e5-Paper.pdf