Multimodal LLM for Farmer Assistance: Integrating RAG and Multi-Agent Workflows

1st Sai Tarun Arveity
Computer Science and Engineering
PES University
Bengaluru South, India
saitarun575@gmail.com

2nd Sambhav Birudavolu
Computer Science and Engineering
PES University
Bengaluru South, India
bosssambhav@gmail.com

3rd Sanjana Boggaram J
Computer Science and Engineering
PES University
Bengaluru South, India
sanjana.boggaram28@gmail.com

4th Shraddha
Computer Science and Engineering
PES University
Bengaluru South, India
kumarshraddha12@gmail.com

5th Dr. Uma D
Professor, Computer Science
PES University
Bengaluru South, India
umaprabha@pes.edu

***Abstract*—Our research proposes a Multi-agentic, multimodal RAG based LLM which is multilingual to effectively assist agricultural professionals. Traditional methods fail to be successful due to the complex nature of agricultural data, hence we suggest this hybrid model which handles unstructured data with retrieval augmented generation (RAG) in conjunction with machine learning (ML) models and Convolutional Neural Networks (CNNs) for structured data. Farmers often face several obstacles regarding the accessibility and availability of technological resources. This hinders their ability to implement innovative solutions. Our objective is to bridge this gap by offering them with verified information, analysis, actionable suggestions, and precise predictions in natural language with multilingual communication and integrating speech input functionalities. This ensures that farmers can engage with the technology with ease, irrespective of their linguistic background or technological expertise thus breaking the technology barricade thus providing them with insights which results in lesser resource wastage and increased crop yields.**

***Index Terms*—Large language model, Multi-agentic workflow, Multi-modal, Multi-lingual, Artificial Intelligence, Machine Learning, Agriculture, Convolutional Neural Network, Langgraph, Precision Farming**

I. Introduction

In recent times, various sectors have seen a progressive use of technology to foster innovative methodologies, the agriculture domain is one of them. Multi-modal large language models (LLMs) and machine learning occupy a pivotal role in this paradigm shift. General-purpose language models such as GPT are trained on broad and diverse corpora covering a wide range of general topics. In contrast, a domain-specific agriculture assistant LLM is trained exclusively on curated, agriculture-related materials. This targeted approach allows the model to outperform the former by providing more precise and relevant responses.

The proposed model utilizes a multi-agent workflow using LangGraph, which facilitates the transparent orchestration of multiple agents and states. This approach has an upper hand to traditional single-agent systems because tasks are distributed among agents based on their expertise leading to efficient handling of complex agricultural challenges and delivery of more accurate insights ultimately leading to improved productivity and sustainability in agricultural practices.

In addition to the core LLM-driven dialogue system, we adopt a hybrid AI approach by integrating multiple machine learning models that handle specialized tasks such as crop yield prediction, crop recommendation, fertilizer recommendation, and image-based plant disease detection. The disease detection model leverages computer vision techniques to classify diseases from plant leaf images with high accuracy. To further enhance context-awareness, the system utilizes IP-based geolocation APIs to detect the user's region and fetch real-time weather parameters via weather APIs. Additionally, on-field sensors deployed on the user's farm are used to measure key soil parameters such as nitrogen (N), phosphorus (P), potassium (K), pH, and moisture levels, providing real-time agronomic data.These attributes are essential inputs for the predictive ML models, enabling localized and personalized recommendations tailored to the user's environment.

To maximize accessibility and inclusivity, the assistant supports multilingual interaction, including automatic language detection and support for major Indian regional languages such as Hindi, Telugu, Kannada, Tamil, and Bengali. Users can interact with the assistant not only via text but also through voice input, with speech recognition enabled across multiple languages. This multilingual, speech-enabled interface ensures ease of use for farmers across diverse linguistic backgrounds, significantly expanding the reach and usability of the system.

II. Related Works

A. Farmer's assistant using Machine Learning and Deep Learning

[5] introduces "Farmer's Assistant," a user-friendly web application designed to aid farmers in several key areas. It uses machine learning and deep learning techniques to

provide recommendations for crop selection via Artificial Neural Networks and Voting Classifiers, fertilizer use via rule-based systems, and plant disease detection using a Residual Attention model on leaf images. The system aims to address challenges faced by farmers, such as erratic irrigation and lack of knowledge about optimal crop and fertilizer choices . By providing data input forms and immediate responses, the application helps farmers make informed decisions to improve crop productivity and reduce crop damage.

[6] introduces a smart farming assistant that uses the Internet of Things to improve farming practices. It aims to address the challenges of outdated farming technology and the migration of rural populations to urban areas, which impact agricultural productivity . The system incorporates sensors, Google services, and expert advice from experienced farmers and researchers to provide features like humidity and temperature sensing, fertilizer estimation, crop disease detection, and overall crop estimation. By utilizing IoT technologies, the assistant seeks to increase productivity and yield while reducing the need for human labor.

[7] explores using fine-tuned Large Language Models like GPT-2 and LLaMA-2 to automate cotton soil analysis and generate detailed reports with actionable recommendations. The study highlights that traditional machine learning models often lack the ability to provide comprehensive reports.The researchers created a custom dataset from cotton-specific resources, focusing on soil nutrient interpretation and recommendations for different growth stages, particularly in the Punjab region of Pakistan. The fine-tuned GPT-2 model achieved a training loss of 0.093 and an evaluation loss of 0.33.The goal is to address the limitations of existing methods by providing comprehensive soil reports without requiring user queries.

[8] presents "Multi-Modal LLMs in Agriculture," a comprehensive review that explores the integration of Large Language Models (LLMs) in modern farming practices. The paper investigates how LLMs, including multi-modal models combining text, image, and sensor data, can enhance various agricultural domains such as crop monitoring, disease detection, irrigation management, and decision support. It addresses 11 key research questions covering LLM capabilities, challenges, and applications. By analyzing over 200 papers, the study highlights the benefits of LLMs in improving data-driven decision-making and boosting efficiency. It also outlines technical and ethical challenges, and proposes a future roadmap for sustainable and equitable AI use in agriculture.

[9] proposes a comprehensive survey on "Leaf-Based Plant Disease Detection and Explainable AI," focusing on combining deep learning methods with interpretability tools to aid plant disease diagnosis. The paper reviews traditional, machine learning, and state-of-the-art deep learning approaches, including CNNs and Vision Transformers, used for leaf disease classification. It catalogs publicly available datasets such as PlantVillage, Plant Pathology, and Cassava Leaf Disease. To bridge the trust gap between black-box models and users, the study emphasizes Explainable AI (XAI) methods like SHAP, LIME, Grad-CAM, and Grad-CAM++. These tools help visualize and interpret model decisions. The authors also explore real-world use cases and future directions like stage-wise disease detection, multi-disease identification, and quantification of disease spread, aiming to improve model reliability, transparency, and user adoption in agriculture.

III. Dataset Collection

An farmer assistant LLM requires a huge range of data. This includes information for the RAG knowledge-base, structured dataset for the machine learning predictor models, and an image dataset for CNN based plant disease detection. The data is cleansed and thoroughly examined to ensure the quality and correctness of the data.

A. RAG Database

The custom built knowledge base was created by compiling and integrating and organizing crop data from trustworthy and accurate sources. It contains information on the following crops: maize, wheat, cotton and paddy. It contains information regarding general crops, optimal environmental conditions, irrigation techniques, diseases, fertilizers, pests, insecticides, and soil.

B. Image plant disease dataset

Consists of quality plant leaf images for training the disease prediction CNN.Several existing datasets are to make this integrated dataset. [1] for paddy disease dataset,[2] for wheat disease detection,[3] for maize disease detection and [4] for cotton disease detection. The dataset contains the following classes:rice bacterial leaf blight, rice blast, rice brown spot, rice healthy, rice tungro, wheat brown rust, wheat fusarium head blight, wheat healthy, wheat mildew, wheat septoria, wheat tan spot, maize blight, maize common rust, maize gray leaf spot, maize healthy, cotton bacterial blight, cotton curl virus, cotton fussarium wilt and cotton healthy. The dataset is manually cleansed, balanced by resampling and augmented to enable the model to capture essential features more accurately.

C. Prediction datasets

Datasets for building ML models for yield prediction, crop recommendation and fertilizer recommendation.

1) Fertilizer recommendation dataset: Model 1's dataset: Soil color: Visible soil color indicating type (e.g., Black, Red), Nitrogen: Nitrogen content in soil (ppm), Phosphorus: Phosphorus content in soil (ppm), Potassium: Potassium content in soil (ppm), pH: Soil acidity/alkalinity level, Temperature (°C): Ambient temperature, Crop: Crop currently being cultivated, Fertilizer: Recommended fertilizer type.

Model 2's dataset: Temperature (°C): Ambient temperature, Moisture: Soil moisture content (0–1 scale), Rainfall (mm): Average annual rainfall, pH: Soil acidity/alkalinity level, Nitrogen: Nitrogen content in soil (ppm), Phosphorus: Phosphorus content in soil (ppm), Potassium: Potassium content in soil (ppm), Carbon: Organic carbon content in soil (fraction), Soil: Soil type (e.g., Loamy Soil), Crop: Crop currently being grown, Fertilizer: Recommended fertilizer type, Remark: Advisory message explaining the recommendation.

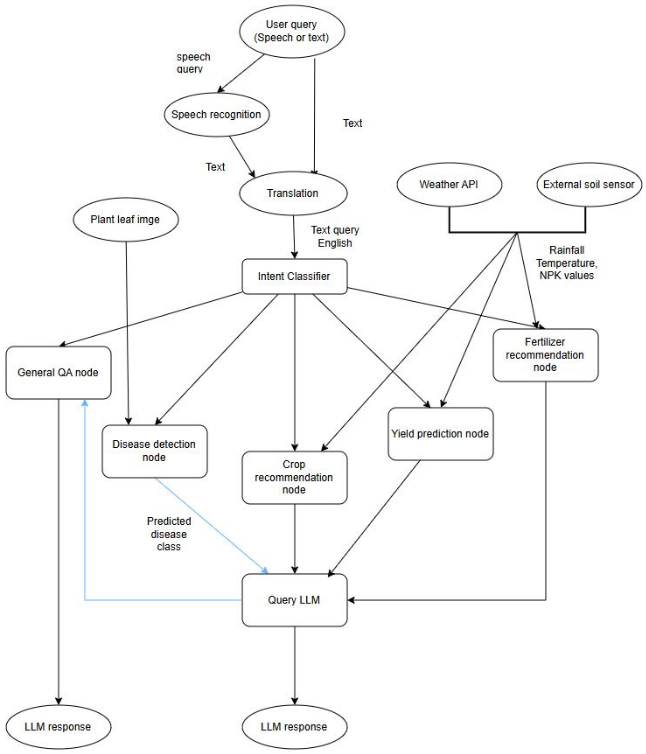

Fig. 1. Enter Caption

2) Crop-recommendation dataset: The dataset includes: N (ppm): Nitrogen content in soil, P (ppm): Phosphorus content in soil, K (ppm): Potassium content in soil, Temperature (°C): Ambient air temperature, Humidity (%): Relative atmospheric humidity, pH: Soil acidity/alkalinity level, Rainfall (mm): Average annual rainfall, Label: Recommended crop type.

3) Yield prediction dataset: The dataset comprises of: Soil Type (categorical): Type of soil (e.g., Sandy, Clay), Crop (categorical): Crop name (e.g., Cotton, Rice), Rainfall mm (continuous): Recorded rainfall in millimeters, Temperature Celsius (continuous): Average temperature in Celsius, Fertilizer Used (binary): Indicator of fertilizer usage, Irrigation Used (binary): Indicator of irrigation application, Weather Condition (categorical): Weather conditions (e.g., Sunny, Cloudy), Days to Harvest (integer): Number of days to harvest, Yield tons per hectare (continuous): Target variable representing yield.

D. Lora dataset

To fine-tune the intent classification component of our system, we constructed a custom dataset tailored for low-rank adaptation (LoRA) training using the DistilBERT model. The dataset comprises six distinct intent categories, with about 1600 labeled instances. Each entry is formatted in JSON and contains a user query along with the corresponding expected output label, supporting effective supervised learning for intent detection in the agricultural domain.

IV. Proposed model

Our proposed model consists of 7 interconnected agents using a langgraph framework designed to enhance farmer assistance through a structured workflow. Here's a breakdown of each node and its function: Each node plays a crucial role in the overall system, ensuring that information flows seamlessly and that farmers receive comprehensive support throughout their agricultural processes. Together, these nodes create a comprehensive system that empowers farmers with actionable insights, promoting productivity and sustainable agricultural practices.

A. Intent Classifier Node

This is the starting point of the model. It receives the user query and classifies the intent behind it. Based on this classification, it forwards the query to the appropriate subsequent node for further processing. It classifies them to one of the following: General question-answering, crop disease detection, crop recommendation, yield prediction, fertilizer prediction, or unsure.

B. General QA Node

B. General QA Node This node is responsible for answering general user questions and providing accurate and insightful information and suggestions tailored to the user's query. This node receives the user query and initiates the processing sequence employing RAG framework which combines retrieval of relevant documents with the generation of natural language responses. This ensures that the answers provided are not only accurate but also contextually relevant.

The data corresponding to a specific crop is chunked and stored as documents in the vector database, with the metadata containing the aforementioned crop's name.Facebook AI similarity search (FAISS) is responsible for the storing and retrieval of related documents from the vector store.

We have used Falcon-10B-instruct as our LLM which generates detailed and accurate responses in natural language, by utilizing the information retrieved from FAISS.

When a General Query is received, the LLM first parses through the query to return a space separated list of crop names that are relevant to the query. This list is parsed through when fetching context in order to fetch context for each relevant crop separately with the help of the metadata filter provided by Langchain FAISS library.These contexts are then passed to the LLM along with the query to generate response for the same.

This metadata filtering technique is advantageous because, if the data of all crops were chunked and stored in the vector database indiscriminately, and if the context were fetched indiscriminately,then the following situation takes place where data for the wrong crop is fetched owing to the fact that in many occasions, the chunks would be missing the crop name to distinguish in the similarity search during retrieval.

1) crop-filter node: A "crop-name" metadata is added to each document in the vector store to ensure that different crop data are not mixed and confused together. The crop-filter LLM upon reviewing the user query, returns a list crop names which are relevant the user query which is then sent to the metadata filter, which filters out the documents and returns all the documents of the required crops. This enables filtering documents by crop for faster retrieval and reduces both computational power used for the similarity search and time needed for retrieval of relevant chunks.

C. Disease Detection Node

This node is specifically designed to detect whether a plant is diseased by processing the inputted image of the leaf of the plant offering crucial detection and disease classification for timely intervention. This detection model employs a CNN based architecture to capture the image features in order to correctly classify the image into the right disease. An pre-trained EfficientNet-V-S neural network is trained on the cured dataset for 15 epochs using (optimizers blah blah) to obtain classification results with an accuracy of 96.15%.

D. ML prediction nodes

To empower farmers with intelligent and actionable guidance, we've seamlessly integrated three core machine learning models—crop yield prediction, crop recommendation, and fertilizer recommendation—into our agricultural chatbot powered by a large language model (LLM). Rather than functioning as separate tools, these models are woven into the LLM's conversational stream, enabling real-time, context-aware predictions during natural dialogue. This integration allows the chatbot to not only understand user queries but also dynamically generate responses enriched with precise, data-driven insights based on inputs like NPK values, weather data, and crop details—bridging the gap between AI predictions and conversational accessibility.

1) Automatic Weather Data Fetching and Processing: To support accurate predictions, our system includes a weather data module that automatically fetches current and historical weather information—such as temperature, rainfall, and humidity—based on the user's location. First the user's location (latitude, longitude and city)is obtained IP address using api from ip-api.com[] . After the user's location is obtained, the weather data of that place is obtained using Open-Meteo API's weather API []. The daily metrics were fetched, including average temperature, relative humidity, total precipitation, cloud cover, and sunshine duration for specific periods. These daily values are aggregated on a monthly basis—monthly rainfall is accumulated, while temperature and humidity are averaged across valid days. The total rainfall is calculated as the sum of monthly precipitation, representing the total rainfall over the entire queried duration. Additionally, soil moisture is approximated using relative humidity, computed as humidity / 100, providing a lightweight proxy for estimating field moisture without physical sensors. Simultaneously, it classifies the weather stats into weather conditions (such as Rainy, Cloudy, or Sunny) using a rule based classifier based on precipitation, cloud cover, and sunshine duration, and stores it for the yield prediction model. This data is essential for the Crop Recommendation, Yield Prediction, and Fertilizer Recommendation Nodes.The module ensures data is preprocessed and structured for use in the models, enabling context-aware and location-specific recommendations.

2) Crop Recommendation Node: To provide personalized crop recommendations based on soil and climatic conditions, a Random Forest-based model was developed using a labeled dataset []. The dataset includes soil data and climate conditions to recommend suitable crops for the farmer, optimizing their yield potential based on specific agricultural contexts. The target variable is the crop label, representing the optimal crop for the given conditions.

Prior to training, the target variable (crop label) was Label encoded to transform string labels (e.g., "rice", "maize") into numerical format. The dataset was then split into training and testing sets using an 80:20 ratio. A Random Forest Classifier was trained with 200 estimators,min samples split=2, min samples leaf=1, leveraging all CPU cores for faster computation. The model achieved high classification accuracy of 99.32 percent demonstrating its robustness in identifying suitable crops for varying environmental conditions. For deployment, both the trained model and the label encoder were serialized using Python's pickle module. The user's climatic data: rainfall and temperature are obtained by a weather API [5] depending on the user location. The Nitrogen (N), Phosphorus (P) and Potassium (K) values of the soil are obtained by a NPK-sensor precision-farming technique.

3) Yield Prediction Node: Utilizing Random-forest ML model on pre-processed dataset, this node predicts the potential yield of the recommended crops thus enabling the farmers set realistic expectations and plan their resources accordingly. For model consistency, the dataset was filtered to include only records from the eastern region of the dataset, and the Region column was subsequently dropped. Categorical features were label-encoded using LabelEncoder, and binary columns were converted to numerical format. The data was split into training and testing subsets (80:20), and a Random Forest Regressor was employed for regression. Hyperparameters were tuned empirically, with the best configuration comprising 150 estimators, a maximum tree depth of 25, and a minimum split size of 4. Model performance was evaluated using Mean Squared Error (MSE) and the coefficient of determination (R^2). Additionally, feature importance was computed to quantify the relative influence of each input variable on yield prediction.

4) Fertilizer Recommendation Node: This node assesses the nutritional needs of the crops and suggests the most effective fertilizers to enhance growth and productivity, ensuring optimal resource utilization.

Two Random Forest classifiers were developed to predict optimal fertilizers and agronomic recommendations based on soil and environmental attributes. Both models utilized datasets containing numerical (e.g., nitrogen, pH, temperature) and categorical (e.g., soil type, crop) features. Data was pre-processed by removing negative values in numerical columns, label-encoding the categorical variables and splitting the datasets into training and testing sets using an 80:20 ratio.

Model 1: Fertilizer Prediction Model 1, trained on the Crop and fertilizer dataset [] and predicts the most suitable fertilizer. A Random Forest classifier (n estimators=100) was trained using encoded features. The model achieved an accuracy of 93

Model 2: Fertilizer and Remark Prediction Model 2, based on the [] dataset, predicts both Fertilizer and Remark labels using two independent Random Forest classifiers. The model achieved 99 percent accuracy for fertilizer prediction and 99 percent for remark prediction.

For usage, the user's Rainfall and Temperature data are dynamically obtained from a weather API [5] based on the user's geographic location. Soil NPK values (Nitrogen, Phosphorus, Potassium) are captured using an NPK sensor as part of a precision farming approach. These integrations allow the models to deliver personalized, location-aware recommendations for optimal crop nutrition.

5) Query LLM Node: This final node takes the predictions and recommendations generated by the previous ML nodes and converts them into natural language outputs. This ensures that the information is user-friendly and easily understandable for farmers, regardless of their technological expertise. Upon receive the "predicted disease" prediction from the disease-detection node, the query LLM communicates with the general-question answering node to retrieve information about the predicted disease, prevention methods and other such useful insights from it and presents it to the user along with the disease prediction.

E. Translation layer

Uses google translation python library to provide a translation abstraction layer allowing the user to input their query in any language. The translation layer auto-detects the language of the user query and translates it into english before passing it to the intent classifier node. the models response is then translated back to the user's language thus removing the language barrier to access this technology.

F. Speech inputs

We integrated a speech-to-text module using OpenAI's multi-lingual Whisper model "turbo" to enable voice-based queries. The system records audio via the sounddevice library, saves it and transcribes it using Whisper. This is then integrated to the chatbot interface by providing speech input button and a sidebar for selecting the desired language of speech, enabling the user to interact with the assistant at ease through speech in their preferred language. The Whisper module supports over 90+ languages including regional Indian languages.

G. Fine-tuning using LoRA on distilbert

To enable accurate classification of user queries in our agricultural assistant, we fine-tuned a lightweight transformer-based intent classifier using DistilBERT enhanced with Low-Rank Adaptation (LoRA). The classifier maps natural language queries to six predefined intents: General Farming Question, Fertilizer Classification, Crop Recommendation, Yield Prediction, Image Plant Disease Detection, and Unclear. We chose the distilbert-base-uncased model for its efficiency, and applied LoRA by injecting low-rank trainable matrices into the attention layers (q-lin and v-lin) with a rank of 8 and an alpha scaling of 32. This significantly reduced the number of trainable parameters while maintaining high performance, making it well-suited for fine-tuning on resource-constrained hardware.

The dataset was provided in JSON format and stratified by intent labels into training (80%), validation (10%), and test (10%) sets to ensure balanced class representation across all splits. We used Hugging Face's ClassLabel for intent encoding, and applied preprocessing with the DistilBERT tokenizer (max length 128, truncation, dynamic padding). Training was conducted using Hugging Face's Trainer API for 25 epochs with a batch size of 16, learning rate of 2e-5, weight decay of 0.01, and FP16 mixed precision enabled for GPU acceleration. Regular evaluation and checkpointing were performed every 100 steps, using accuracy and weighted F1-score as evaluation metrics. The best-performing checkpoint—achieving a validation accuracy of 98.8%—was selected and subsequently evaluated on the held-out test set to assess generalization.

For deployment, we saved the fine-tuned model and tokenizer, and wrapped inference into a lightweight function for real-time intent prediction. This enables seamless integration into interactive applications, such as voice/chat interfaces for farmers. The use of LoRA allowed the model to remain compact and efficient, providing a scalable, low-latency solution for intent recognition in agriculture-focused conversational systems.

ACKNOWLEDGMENT

We would like to express our sincere gratitude to Dr. Uma D, for her invaluable guidance, unwavering support, and insightful feedback throughout the development of this project. Her expertise and mentorship have been fundamental to the successful execution of our work. We also wish to acknowledge the authors and contributors of the research literature, datasets, and open-source frameworks that have been referenced and utilized in this project. Their foundational work has significantly contributed to the design and implementation of our proposed system.We extend our appreciation to our team members — Sai Tarun Arveity, Sambhav Birudavolu, Sanjana Boggaram J, and Shraddha—for their dedicated efforts, collaborative spirit, and technical contributions across all phases of the project.This endeavor has been both intellectually enriching and technically rewarding, and we are grateful for the opportunity to work on a project with meaningful real-world impact

REFERENCES

[1] Paddy Doctor, Pandarasamy Arjunan (Samy), and Petchiammal. Paddy Doctor: Paddy Disease Classification.

https://kaggle.com/competitions/paddy-disease-classification, 2022. Kaggle.
[2] https://www.kaggle.com/datasets/kushagra3204/wheat-plant-diseases/data
[3] J, ARUN PANDIAN; GOPAL, GEETHARAMANI (2019), "Data for Identification of Plant Leaf Diseases Using a 9-layer Deep Convolutional Neural Network", Mendeley Data, V1, doi: 10.17632/tywbtsjrjv.1.
[4] Noon, Serosh Karim et al. 'Computationally Light Deep Learning Framework to Recognize Cotton Leaf Diseases'. 1 Jan. 2021 : 1 – 16.
[5] R. Jadhav and P. Bhaladhare, 'Farmer's Assistant in Agricultural Sector by using Machine Learning and Deep Learning'
[6] J. Kanimozhi, 'A SMART FARMING ASSISTANT – COLLABORATIVE HELP FROM INTERNET AND AGRICULTURAL EXPERTS,' ACTA TECHNICA CORVINIENSIS – Bulletin of Engineering, vol. XIV, no. 1, pp. 111, 2021.
[7] Syed Hassan Ali, Muhammad Farrukh Shahid, M. Hassan Tanveer, and Abdul Rauf, 'Integrating LLM for Cotton Soil Analysis in Smart Agriculture System,' International Journal of Innovations in Science Technology, Special Issue pp 283-294, Oct 2024.
[8] R. Sapkota et al., 'Multi-Modal LLMs in Agriculture: A Comprehensive Review,' IEEE Transactions on Automation Science and Engineering, pp. 1–13, 2024.
[9] S. Sagar, M. Javed, and D. S. Doermann, 'Leaf-Based Plant Disease Detection and Explainable AI,' arXiv preprint arXiv:2404.16833, 2024.