International Journal of Scientific Research & Engineering Trends
Volume 12, Issue 3, May-Jun-2026, ISSN (Online): 2395-566X

AgriHub: An AI-Powered End-to-End Agricultural Decision Support Platform

Mohammed Munyim Hussain V[1], Poorvaj K P[2], Prashanth S R[3], Preetham M[4], Mr P Prasanna[5]
[1,2,3,4]Department of Computer Science & Engineering, P.E.S College of Engineering, Mandya, Karnataka
[5]Associate Professor, Department of Computer Science & Engineering, P.E.S College of Engineering, Mandya, Karnataka

Abstract- Agriculture remains a cornerstone of economic activity across developing nations, yet smallholder farmers routinely face yield gaps caused by uninformed decisions on crop selection, soil nutrition, and disease management. This paper presents AGRI HUB, a web-based Crop and Soil Management System that unifies several machine-learning and deep-learning services behind a single Flask-driven interface. Four core modules are delivered: (i) smart crop recommendation using a Random Forest classifier trained on seven agro-climatic parameters, achieving 99.55% accuracy across 22 crop classes; (ii) soil nutrient analysis and fertilizer recommendation through NPK deficit computation against crop-specific thresholds; (iii) plant disease detection using a ResNet-9 convolutional neural network capable of classifying 38 disease categories from leaf photographs; and (iv) real-time, weather-driven activity planning by consuming OpenWeatherMap API data to generate seven-day farming calendars. An AI chatbot powered by the Google Gemini large language model supplements the analytical modules with conversational agronomic guidance. A crop profitability comparison dashboard rounds out the system, enabling evidence-based economic decisions. Experimental evaluation confirms that the integrated platform consistently outperforms single-module alternatives in both accuracy and decision breadth, offering a scalable, cost-effective tool for precision agriculture.

Keywords- Precision agriculture, crop recommendation, Random Forest, ResNet-9, plant disease detection, soil fertility, fertilizer optimization, weather-based advisory, large language model, decision support system.

I. INTRODUCTION

World food production must increase by roughly 70% be-fore 2050 to keep pace with a projected population of nearly ten billion people [1]. At the same time, arable land per capita is shrinking, water tables are under stress, and the climate is becoming less predictable. In this scenario, digi-tally augmented farming—sometimes called precision agri-culture or smart farming—offers a credible route toward higher, sustainable yields without proportional increases in resource consumption.

Indian agriculture, which supports nearly 58% of the rural workforce, illustrates both the stakes and the challenges [2]. Farmers depend heavily on inherited knowledge, local in-tuition, and generalised government advisories. These sources frequently overlook micro-level variations in soil chemistry, real-time weather shifts, and early-stage disease symptoms. Consequences range from suboptimal fertilizer application and inappropriate crop choices to unchecked disease spread that can devastate an entire season's harvest.

Existing software tools address isolated facets of this problem. Standalone crop-advisory applications exist, as do disease-identification chatbots and fertilizer calcula-tors. However, these tools operate independently, requiring farmers to navigate multiple platforms and reconcile po-tentially conflicting advice. An end-to-end integrated sys-tem—one that combines soil analytics, crop selection, dis-ease surveillance, weather forecasting, and conversational assistance under a single roof—has been comparatively under-explored in accessible, open-source form.

AGRI HUB addresses this gap. The system brings together four major intelligent subsystems—crop recommendation, soil and fertilizer advisory, plant disease detection, and weather-aware farming calendar generation—along with a large-language-model (LLM) chatbot and a crop profitabil-ity dashboard. All subsystems are accessible through a responsive web front-end, making the platform usable on smartphones as readily as on desktop computers.

Research Contributions

- Unified AI-driven agricultural decision platform
- Real-time weather-aware recommendations
- Integration of Gemini LLM for agricultural guidance
- Crop profitability comparison dashboard
- End-to-end Flask deployment

II. RELATED WORK

Crop recommendation using machine learning has attracted sustained research attention. Pudumalar et al. [3] showed that ensemble classifiers, particularly Random Forests, out-perform single-tree approaches when soil and climate pa-rameters constitute the input feature space. Their seven-feature model obtained 93.4% accuracy, an outcome later surpassed by studies that incorporated hyper-parameter tuning and richer training corpora.

For disease detection, Mohanty et al. [4] demonstrated that deep convolutional neural networks trained on the PlantVillage dataset could attain accuracy exceeding 99% under controlled imaging conditions. Subsequent work on ResNet architectures—particularly the lightweight ResNet-9 variant—confirmed that residual connections signifi-cantly reduce the vanishing-gradient problem for image classification tasks at moderate model sizes [5].

Fertilizer recommendation research has historically relied on rule-based lookup systems tied to soil test results [6]. More recently, data-driven approaches using regression and classification models have been applied, though publicly available annotated datasets remain scarce.

Weather-integrated advisory platforms have been devel-oped primarily for commercial precision-agriculture appli-cations. Open-access implementations that use free-tier weather APIs to generate actionable farming calendars re-main limited in the academic literature [7].

LLM-driven agricultural chatbots represent a rapidly evolv-ing sub-field. Early deployments using retrieval-augmented generation have shown promise for answering domain-specific questions, while direct prompt-engineering on general-purpose LLMs such as Gemini or GPT-4 offers lower deployment complexity at the cost of occasional hal-lucinations [8].

The present work differs from prior contributions by tightly coupling all of the above components within a single coher-ent platform, enabling cross-module synergy—for instance, aligning crop recommendations with current weather fore-casts and soil conditions simultaneously.

III. SYSTEM ARCHITECTURE

High-Level Design

AGRI HUB follows a three-tier architecture: a presenta-tion tier implemented with HTML5, CSS3, Bootstrap 5, and vanilla JavaScript; an application tier built on the Python Flask micro-framework; and a data/model tier com-prising serialised machine-learning models, CSV reference datasets, and external API integrations.

Figure 1 illustrates the overall data flow. Browser re-quests arrive at the Flask router, which dispatches each request to the relevant module handler. Module handlers invoke trained model objects or external APIs, transform results into human-readable recommendations, and return rendered Jinja2 templates. Stateless request handling sim-plifies horizontal scaling.

Figure 1: System Architecture of AGRI HUB

System Architecture Summary

Table 1: Simplified request–response flow

System Architecture Summary
Browser → Flask Router → Module Handler
→ ML Model / External API
← JSON / Prediction Result
Flask → Jinja2 Template → HTML Response

Technology Stack

- Backend: Python 3.11, Flask 2.x, Scikit-Learn, PyTorch, Pandas, NumPy, Pillow
- Frontend: HTML5, CSS3, Bootstrap 5, JavaScript, AOS animations
- ML Models: Random Forest (crop), ResNet-9 CNN (disease)
- External APIs: OpenWeatherMap (weather), Google Gemini 2.5 Flash (chatbot)
- Storage: Serialised .pkl/.pth model files, CSV refer-ence tables

4. MODULE DESCRIPTIONS

Crop Recommendation Module

The crop recommendation module accepts seven input pa-rameters: Nitrogen (N), Phosphorus (P), Potassium (K), soil pH, annual rainfall, ambient temperature, and relative humidity. Temperature and humidity are optionally fetched automatically from OpenWeatherMap using the farmer's city name, reducing manual data-entry burden.

A Random Forest classifier comprising 100 decision trees was trained on a dataset of 2,200 labelled records cov-ering 22 crop categories (rice, maize, chickpea, kidney beans, pigeon peas, moth beans, mung beans, black gram, lentil, pomegranate, banana, mango, grapes, watermelon, muskmelon, apple, orange, papaya, coconut, cotton, jute, coffee). Grid-search cross-validation with stratified five-fold splitting was used to optimise the number of estimators and maximum-depth hyper-parameters. The trained model is serialised as RandomForest.pkl and loaded at appli-cation start, making inference sub-millisecond per request.

Fertilizer Recommendation Module

Rather than a second learner, the fertilizer module ap-plies an analytical deficit-computation approach grounded in agronomic guidelines. For a selected crop, ideal N, P, and K values are retrieved from a reference CSV. The dif-ference between ideal and farmer-reported values identifies the most nutrient-deficient element. The system maps the sign and magnitude of each deficit to one of six recom-mendation keys (NHigh, NLow, PHigh, PLow, KHigh, KLow), each linked to a detailed, HTML-formatted ad-visory stored in the fertilizer dic dictionary. This deterministic approach ensures reproducible, scientifically consistent advice without requiring an additional training corpus.

Plant Disease Detection Module

Disease detection is the most computationally intensive module. Input is a photograph of a plant leaf uploaded by the user. The image is resized to 256 × 256 pixels, nor-malised, and passed through a ResNet-9 model pre-trained on a subset of the PlantVillage dataset. ResNet-9 uses two convolutional blocks and two residual blocks, chosen to balance inference speed with classification accuracy.

The model outputs a softmax probability vector over 38 classes. The argmax class is mapped to a human-readable disease label, and the disease dic lookup table pro-vides structured treatment recommendations—covering fungicide options, cultural controls, and resistance man-agement. Supported host plants include tomato, potato, corn, grape, apple, peach, pepper, soybean, strawberry, and several others, collectively covering the most economically significant crops in the Indian subcontinent.

Weather-Aware Advisory Module

The weather advisory module queries the OpenWeath-erMap API at two endpoints: the current-weather endpoint for live conditions and the five-day/three-hour forecast end-point for a rolling seven-day outlook. Raw JSON pay-loads are parsed into daily aggregates (maximum temper-ature, minimum temperature, average humidity, dominant weather condition, wind speed).

A rule-based inference engine evaluates each day against a set of agronomic thresholds:

- Temperature > 35°C triggers a heat-stress alert.
- Temperature < 5°C raises a frost warning.
- Humidity > 85% elevates disease risk advisory.
- Wind speed > 10 m/s suppresses spraying recommenda-tions.

For each day the engine emits categorised advisories cov-ering planting, harvesting, irrigation timing, and spraying suitability. A weekly summary aggregates rain-day counts, heat-wave periods, and optimal activity windows into a concise farmer-facing bulletin.

AI Chatbot Module

The AGRI BOT chatbot is implemented through direct REST calls to the Google Gemini 2.5 Flash model. Each user message is wrapped in a structured prompt that explic-itly scopes responses to agriculture topics, including crop management, disease control, soil science, irrigation, pest management, and sustainable farming practices. Response length is bounded at 500 tokens to keep answers concise and mobile-friendly. The chatbot is rendered as a float-ing widget accessible from every page, reducing navigation friction.

Crop Comparison Dashboard

The comparison dashboard allows farmers to evaluate mul-tiple crops side-by-side on economic and agronomic cri-teria: estimated profit per acre, market price trends, wa-ter requirements, labour intensity, and regional suitabil-ity. Data are drawn from a structured Python dictionary (crop data.py) containing 22 crop profiles. Interactive JavaScript components render bar charts and radar plots to support visual comparison.

V. DATASET AND MODEL TRAINING

Crop Recommendation Dataset

The training dataset comprises 2,200 samples with seven continuous features and one categorical label (22 classes, 100 samples each). Feature ranges were:

- N: 0–140 kg/ha; P: 5–145 kg/ha; K: 5–205 kg/ha
- Temperature: 8.8–43.7°C; Humidity: 14.3–99.9%
- pH: 3.5–9.9; Rainfall: 20.2–298.6 mm

After an 80/20 train-test split, the Random Forest model achieved 99.55% test accuracy, with precision, recall, and F1-score all exceeding 0.99 for every class. No significant overfitting was detected, as five-fold cross-validation vari-ance was below 0.3%.

Disease Detection Dataset

The ResNet-9 model was trained on a curated subset of the PlantVillage dataset containing annotated images across 38 disease/healthy class combinations. Data augmentation techniques—random horizontal flipping, rotation up to 15°, and colour jitter—were applied to increase effective train-ing set diversity. The model was optimised using the Adam optimiser with a learning rate of 10^{-3}, weight decay 10^{-4}, and OneCycleLR scheduling. Training converged in ap-proximately 5 epochs on a GPU instance, reaching valida-tion accuracy above 95%.

VI. RESULTS AND EVALUATION

Crop Recommendation Performance

Table 2 summarises classification performance for the Ran-dom Forest model across selected crop classes.

Table 2: Crop recommendation model performance (test set)

Crop	Precision	Recall	F1
Rice	1.00	1.00	1.00
Maize	0.99	1.00	0.99
Chickpea	1.00	0.99	0.99
Mango	1.00	1.00	1.00
Cotton	0.98	0.99	0.98
Jute	0.99	0.98	0.98
Coffee	1.00	1.00	1.00
Macro Avg	**0.99**	**0.99**	**0.99**

Disease Detection Performance

The ResNet-9 classifier achieved an overall validation ac-curacy of 95.2% across 38 disease classes. Multi-class confusion analysis revealed that misclassifications predom-inantly occurred between visually similar disease pheno-types (e.g., Tomato Early Blight vs. Tomato Target Spot), consistent with findings in the broader plant-pathology lit-erature. Table 3 shows performance for frequently encoun-tered classes.

Table 3: Disease detection performance (selected classes)

Disease Class	Precision	Recall
Tomato Late Blight	0.96	0.97
Potato Early Blight	0.95	0.94
Corn Common Rust	0.97	0.96
Apple Scab	0.98	0.98
Grape Black Rot	0.94	0.95
Healthy (all crops)	0.99	0.98

System Response Time

All modules return results within acceptable latency bounds for a web application:

- Crop recommendation: <100 ms (model inference only)
- Fertilizer advisory: <50 ms (lookup computation)
- Disease detection: <800 ms (image preprocessing + CNN inference)
- Weather advisory: 1–3 s (network round-trip to Open-WeatherMap)

- Chatbot response: 2–5 s (Gemini API round-trip)

Comparison with Baseline Systems

Table 4 contrasts AGRI HUB with representative baseline systems reported in the literature.

Table 4: Comparison with prior systems

System	Crop Acc.	Modules
Pudumalar *et al.* [3]	93.4%	1
Mohanty *et al.* [4]	—	1
Suchitra *et al.* [6]	—	1
AGRI HUB (Proposed)	**99.55%**	**6**

VII. IMPLEMENTATION DETAILS

Application Deployment

The application is configured for deployment on any WSGI-compatible server. A Procfile targeting Gu-nicorn is included, enabling one-click deployment to cloud platforms such as Heroku or Railway. The Runtime.txt file pins the Python interpreter version to ensure reproducibility. All dependencies are recorded in requirements.txt.

Security Considerations

API keys for OpenWeatherMap and Google Gemini are stored in a config.py file excluded from version control via .gitignore. User-uploaded images are processed entirely in memory without being persisted to disk, reduc-ing both storage overhead and privacy risk. Input values from HTML forms are explicitly cast to their target nu-meric types before reaching model inference code, prevent-ing injection-style attacks.

Frontend Design

The interface adopts a card-based layout with a green-and-white colour scheme that resonates with an agricultural context. Scroll-triggered animations (AOS library) improve perceived interactivity on mobile devices. Navigation is handled through a fixed Bootstrap navbar, and all forms are validated client-side before submission to reduce round-trips.

VIII. LIMITATIONS AND FUTURE SCOPE

Current Limitations

The crop recommendation model was trained on a sin-gle publicly available dataset, which may not fully cap-ture hyper-local agro-climatic diversity across all Indian re-gions. The disease detection module requires a reasonably clear leaf photograph and performs less reliably under poor lighting or significant background clutter. The weather ad-visory relies on free-tier API quotas, which cap the num-ber of daily requests in high-traffic deployments. Finally, the chatbot occasionally produces factually imprecise re-sponses on highly specialised agrochemical questions, a known limitation of general-purpose LLMs.

Future Enhancements

Several extensions are planned:

1. IoT sensor integration: Real-time soil-moisture, pH, and NPK sensors can replace manual form inputs, en-abling continuous advisory generation.
2. Satellite imagery: NDVI (Normalised Difference Veg-etation Index) derived from Sentinel-2 imagery can en-rich the crop recommendation feature vector.
3. Regional language support: Adding multi-lingual in-terfaces (Hindi, Tamil, Telugu, Kannada) will broaden accessibility for farmers with limited English profi-ciency.
4. Mobile application: A Progressive Web App (PWA) wrapper would enable offline-capable access in areas with intermittent connectivity.
5. Market price feed: Connecting to AGMARKNET or commodity exchange APIs would make profitability comparisons real-time rather than static.
6. Federated learning: Collecting anonymised, on-device data from farmer interactions to retrain models without centralising sensitive information.

IX. CONCLUSION

This paper described AGRI HUB, an end-to-end Crop and Soil Management System built to address the practical in-formation gap faced by smallholder farmers. By unit-ing a high-accuracy Random Forest crop recommender (99.55%), a rule-based fertilizer advisor, a ResNet-9 plant disease detector (95.2% validation accuracy), a weather-driven farming calendar, an LLM-powered chatbot, and a crop profitability dashboard within a single web platform, the system delivers multi-dimensional agronomic guidance that no individual component could provide alone.

The platform is designed with deployment practicality in mind: lightweight dependencies, API-key-based external integrations, and a responsive interface that functions on smartphones without a dedicated mobile application. Collectively, these qualities make AGRI HUB a strong candi-date for wider adoption in precision-agriculture contexts, particularly where internet access is available but technical expertise is limited.

Ongoing and future work will extend the platform’s reach through IoT integration, satellite data streams, regional language interfaces, and federated model updates—further closing the information asymmetry that continues to con-strain agricultural productivity in developing economies.

REFERENCES

1. Food and Agriculture Organization of the United Nations, “The State of Food and Agriculture 2022: Leveraging Automation in Agriculture,” FAO, Rome, 2022.
2. NITI Aayog, “India Three Year Action Agenda 2017-18 to 2019-20,” Government of India, New Delhi, 2020.
3. S. Pudumalar, E. Ramanujam, R. H. Rajashree, C. Kavya, T. Kiruthika, and J. Nisha, “Crop Recom-mendation System for Precision Agriculture,” in Proc. 8th Annual IEEE Information Technology, Electronics and Mobile Communication Conference (IEMCON), Vancouver, BC, Canada, 2017, pp. 61–75.
4. S. P. Mohanty, D. P. Hughes, and M. Salathe´, “Using Deep Learning for Image-Based Plant Disease Detec-tion,” Frontiers in Plant Science, vol. 7, p. 1419, 2016.
5. K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” in Proc. IEEE Con-ference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 770–778.
6. P. Suchitra and R. Varatharajan, “Design of a Fer-tilizer Recommendation System Using Fuzzy Logic for Paddy Cultivation,” International Journal of En-gineering and Technology, vol. 7, no. 2.33, pp. 1026–1030, 2018.
7. S. Wolfert, L. Ge, C. Verdouw, and M.-J. Bogaardt, “Big Data in Smart Farming – A Review,” Agricul-tural Systems, vol. 153, pp. 69–80, 2017.
8. Y. LeCun, “A Path Towards Autonomous Machine Intelligence,” Open Review Preprint, 2022. [Online]. Available: https://openreview.net/pdf?id=BZ5a1r-kVsf
9. K. G. Liakos, P. Busato, D. Moshou, S. Pearson, and D. Bochtis, “Machine Learning in Agriculture: A Re-view,” Sensors, vol. 18, no. 8, p. 2674, 2018.
10. A. Kamilaris and F. X. Prenafeta-Boldu´, “Deep Learning in Agriculture: A Survey,” Computers and Electronics in Agriculture, vol. 147, pp. 70–90, 2018.