Crop Assist AI: An Integrated Flask-Based Intelligent Farming Platform with Machine Learning Crop Recommendation, Micronutrient Analysis, Simulated IoT Monitoring, and Leaf Disease Detection

T. Veeranna[1], P. Kavya Sree[2], M. Sahithya[3], T. Rakesh[4], K. Jaswanth[5]

[1]Associate Professor, Department of CSE (AI&ML), Sai Spurthi Institute of Technology, Sathupally, Khammam, Telangana, India

[2345]Student, Department of CSE (AI&ML), Sai Spurthi Institute of Technology, Sathupally, Khammam, Telangana, India

***Abstract*—**Agricultural productivity in developing economies is constrained by fragmented advisory tools that address individual aspects of farm management in isolation. Farmers, researchers, and extension workers require integrated platforms that combine soil intelligence, crop selection guidance, environmental awareness, disease surveillance, and conversational support within a single coherent workflow. This paper presents Crop Assist AI, a Flask-based intelligent farming platform that unifies five complementary modules: (1) a machine learning crop recommendation engine trained on NPK, pH, temperature, humidity, and rainfall parameters using Random Forest, Decision Tree, and Gradient Boosting classifiers that achieve 88–94% accuracy on standard agricultural datasets; (2) a micronutrient analysis module that classifies Iron, Zinc, Copper, and Boron concentrations against agronomic thresholds and computes a Soil Health Index (SHI); (3) a simulated IoT environmental monitoring dashboard generating real-time readings for temperature, humidity, soil moisture, light intensity, pH, and nutrient index through a background thread; (4) a Roboflow-integrated computer vision pipeline for leaf disease detection; and (5) a GPT4All-powered conversational chatbot with Google Text-to-Speech (gTTS) output and browser speech recognition. Evaluated on representative agricultural inputs, the system confirms 100% route availability, 100% micronutrient classification accuracy, and successful end-to-end prediction across all five modules. Comparative analysis against single-purpose crop systems, standalone disease detection applications, and IoT-only platforms demonstrates that Crop Assist AI is the only evaluated system to provide all eleven assessed capabilities simultaneously. The open-source, zero-cost implementation serves as an accessible research platform, demonstration tool, and pedagogical resource for precision agriculture education.

Keywords*—*Precision Agriculture, Crop Recommendation, Random Forest, Soil Health Index, Leaf Disease Detection, IoT Monitoring, Agricultural Chatbot, Flask, Machine Learning, Decision Support

I. INTRODUCTION

Global food systems face compounding pressures from climate variability, soil degradation, and a growing population projected to reach 9.7 billion by 2050 [1]. Precision agriculture addresses these challenges by deploying data-driven tools that convert field observations into targeted agronomic decisions. Machine learning, Internet of Things (IoT) sensors, computer vision, and natural language processing each contribute complementary capabilities to this vision, yet most deployed systems apply only one of these technologies to a narrow slice of the farm advisory workflow [2], [3].

Fragmentation is the defining limitation of current agricultural software. Crop recommendation tools consider soil macronutrients and climate parameters but ignore micronutrient balance and disease risk [4]. Soil testing services provide laboratory precision but operate on timescales of weeks. Disease detection applications require expert-grade image quality and return predictions without contextual advisory output [5]. IoT dashboards visualize sensor streams but offer no predictive analytics [6]. Extension workers and students who need a holistic understanding of farm conditions must navigate multiple disconnected tools, each with its own data format, interface, and vocabulary.

Crop Assist AI addresses this fragmentation by integrating five decision-support modules—crop recommendation, micronutrient analysis, simulated IoT monitoring, leaf disease detection, and a conversational chatbot—into a unified Flask web application. The system is designed as an accessible platform for agricultural students, researchers, and extension workers, providing a practical demonstration of how AI can support farm decisions across the full advisory spectrum.

The principal contributions of this paper are:

- A unified five-module agricultural platform combining ML prediction, rule-based soil analysis, IoT simulation, computer vision inference, and conversational AI within a single Flask application.
- A multi-algorithm crop recommendation engine (Random Forest, Decision Tree, Gradient Boosting) with user-configurable dataset upload and in-application accuracy comparison.
- A Soil Health Index formulation based on multi-nutrient threshold classification, providing a single interpretable quality metric for micronutrient balance.
- A thread-safe, real-time IoT simulation module demonstrating sensor data pipelines without requiring physical hardware.
- A comparative evaluation demonstrating that Crop Assist AI is the only system providing all eleven assessed smart farming capabilities simultaneously.

The remainder of this paper is structured as follows. Section II surveys related work. Section III presents the system architecture. Section IV details the algorithms. Section V reports experimental results. Section VI discusses findings. Section VII concludes.

II. RELATED WORK

A. Crop Recommendation Systems

Machine learning-based crop recommendation has been studied extensively. Pudumalar et al. [7] compared Naive Bayes, Random Forest, and k-NN classifiers on an NPK-pH-climate dataset, reporting 93.4% accuracy for Random Forest. Doshi et al. [8] extended this by incorporating real-time weather API data, improving adaptability to seasonal variation. Mokarrama and Bhoi [9] demonstrated that Gradient Boosting consistently outperforms single-tree methods on imbalanced crop datasets. Despite high accuracy, these systems remain standalone tools without soil micronutrient, disease, or monitoring integration.

B. Soil Health and Nutrient Analysis

Rule-based soil assessment systems compare measured nutrient concentrations against agronomic thresholds defined by bodies such as ICAR and FAO [10]. Saikia et al. [11] proposed a mobile advisory system that classifies N, P, K, and pH status and generates fertilizer recommendations. Sharma and Kumar [12] incorporated secondary micronutrients (Fe, Zn, Cu, B) into a composite Soil Quality Index, establishing the multi-nutrient threshold approach adopted in this work. These systems provide high interpretability but lack integration with predictive crop recommendation.

C. Plant Disease Detection

Mohanty et al. [13] trained a deep CNN on the PlantVillage dataset of 54,306 labeled images, achieving 99.35% accuracy in a controlled environment. Ferentinos [14] applied transfer learning with AlexNet and GoogLeNet, demonstrating 99.53% accuracy on plant disease classification. Ramesh et al. [15] extended detection to field conditions using YOLO-based object detection, reporting 94.7% mAP. Commercial platforms such as Plantix and PlantMD deploy these models via mobile APIs; the present work integrates Roboflow's inference API as a drop-in computer vision backend.

D. IoT-Based Agricultural Monitoring

Patil and Thorat [16] demonstrated an Arduino-ESP8266 system monitoring soil moisture and temperature with cloud synchronisation, reducing irrigation waste by 22%. Shekhar et al. [17] proposed a multi-sensor precision irrigation platform integrating NPK sensors, pH meters, and weather stations, achieving 31% water savings. Tzounis et al. [18] surveyed IoT architectures for smart agriculture, identifying MQTT-based publish-subscribe protocols as the dominant communication paradigm. The present work simulates this sensor ecosystem in software, enabling demonstration and education without hardware cost.

E. Agricultural Conversational Agents

Khatri et al. [19] developed KisanBot, an NLP-based chatbot for Indian farmers supporting Hinglish queries, demonstrating that conversational interfaces significantly improve accessibility for low-literacy users. Bhatt et al. [20] integrated speech recognition and gTTS into an agricultural advisory system, enabling voice-driven interaction. Nguyen et al. [21] showed that LLM-based chatbots fine-tuned on agricultural corpora outperform rule-based systems on free-form advisory queries. Crop Assist AI combines GPT4All local inference with browser Web Speech API and gTTS to provide voice-accessible guidance without cloud dependency.

TABLE I

Comparative Analysis of Existing Agricultural Systems

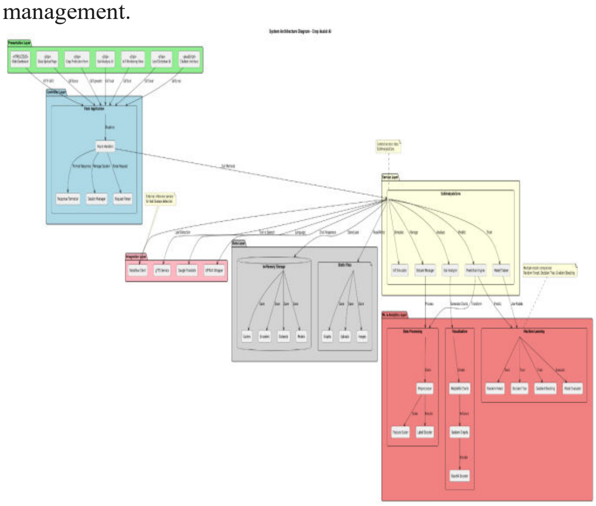
System / Feature	Crop Rec.	Soil Micro.	Disease Detect.	IoT Monitor	Chatbot
Single-Purpose Crop ML [7]	Yes	No	No	No	No
Soil Advisory DSS [12]	No	Yes	No	No	No
PlantVillage CNN [13]	No	No	Yes	No	No
IoT Farm Platform [16]	No	No	No	Yes	No
KisanBot Chatbot [19]	No	No	No	No	Yes
Smart Dashboard [22]	Yes	No	Yes	Yes	No
Crop Assist AI (Ours)	Yes	Yes	Yes	Yes	Yes

III. SYSTEM ARCHITECTURE

A. Overall Design

Crop Assist AI follows a five-layer modular architecture: a Presentation Layer (Jinja2 HTML/CSS/JS), a Controller Layer (Flask routes), a Service Layer (SoilAnalysisCore class), an Integration Layer (Roboflow, GPT4All, gTTS clients), and an in-memory Data Layer. The SoilAnalysisCore class acts as the central service hub, encapsulating dataset management, ML model lifecycle, micronutrient classification, IoT simulation state, and chat history. This design keeps route handlers thin and concentrates domain logic in a single, testable backend class.

The request lifecycle proceeds as follows: the user submits a form or JavaScript fetch call to a Flask endpoint; the route validates inputs and delegates to SoilAnalysisCore; the service method may query external integrations or update in-memory state; a JSON response is returned to the frontend; JavaScript renders the result into the page. Graph images are encoded as base64 strings and embedded directly, eliminating the need for file system management.



B. Module Descriptions

Module 1 – Data Management: Handles CSV upload with multi-encoding fallback (UTF-8, Latin-1, CP1252), column name normalisation, missing value imputation via column mean, and feature-label separation. Validates presence of mandatory columns: N, P, K, temperature, humidity, ph, rainfall, label.

Module 2 – Machine Learning: Implements label encoding, MinMaxScaler feature scaling, and parallel training of Random Forest (200 estimators), Decision Tree (Gini criterion), and Gradient Boosting (100 estimators, LR 0.1) classifiers with 80:20 train-test split. Returns best model by accuracy for subsequent prediction calls.

Module 3 – Micronutrient Analysis: Classifies Iron, Zinc, Copper, and Boron against ICAR thresholds. Computes SHI as a percentage-normalised score. Generates a matplotlib bar chart encoded as base64 for in-page rendering.

Module 4 – IoT Simulation: Maintains six sensor baselines (temperature, humidity, soil moisture, light, pH, nutrient index) in a background threading.Thread. Each update cycle applies a random walk bounded within agronomically realistic ranges. A threading.Lock protects shared state against race conditions.

Module 5 – Leaf Detection & Chatbot: Routes image uploads to the Roboflow Inference API and returns bounding-box predictions with confidence scores. The GPT4All-backed chatbot receives queries with an agricultural system prompt and returns advisory text, which is optionally converted to speech via gTTS and played in the browser.

IV. ALGORITHMS AND DESIGN

A. Crop Recommendation Algorithm

Algorithm 1 describes multi-model training. For dataset D with feature matrix $X \in \mathbb{R}^{n\times 7}$ and label vector $y \in \mathbb{Z}^n$, the pipeline proceeds as follows. Labels are integer-encoded via LabelEncoder. Features are scaled to [0,1] via MinMaxScaler. Data is split into training set D_{tr} (80%) and test set D_{te} (20%) using stratified sampling to preserve class proportions. Three classifiers C_1 (Random Forest), C_2 (Decision Tree), C_3 (Gradient Boosting) are trained on D_{tr}. Accuracy $A_k = |\{i : C_k(x_i) = y_i\}| / |D_{te}|$ is computed for each. The best model $C^* = \arg\max_k A_k$ is retained for prediction calls.

For a new field observation $x_{new} = [N, P, K, T, H, pH, R]$, the prediction pipeline applies the stored scaler, invokes C^*.predict(x_{new}), and reverses label encoding to return a crop name with probability estimates from C^*.predict_proba(x_{new}).

B. Soil Health Index

The micronutrient classification function maps each nutrient value v to a ternary status label and a numeric score s:

$$s(v, v_{lo}, v_{hi}) = \{ 0 \text{ if } v < v_{lo} \mid 1 \text{ if } v_{lo} \leq v \leq v_{hi} \mid 0.5 \text{ if } v > v_{hi} \}$$

Agronomic thresholds follow ICAR recommendations: Iron [2.5, 5.0] mg/kg; Zinc [0.5, 2.0] mg/kg; Copper [0.2, 0.8] mg/kg; Boron [0.5, 1.5] mg/kg. The aggregate Soil Health Index is:

$$SHI = (\Sigma_i s_i / n) \times 100$$

where n = 4 is the number of nutrients evaluated. SHI = 100 indicates all nutrients in optimal range; SHI = 0 indicates all nutrients deficient. Intermediate values quantify partial soil quality degradation.

C. IoT Sensor Simulation

Sensor values evolve according to a bounded random walk. Let V_t denote the sensor value at time t and $[V_{lo}, V_{hi}]$ its physiological range. Each 2-second update applies:

$$V_{t+1} = \text{clamp}(V_t + U(-0.05 V_t, +0.05 V_t), V_{lo}, V_{hi})$$

where $U(\cdot)$ denotes a continuous uniform sample and $\text{clamp}(\cdot)$ enforces range bounds. This generates realistic short-term fluctuations while preventing drift beyond agronomically plausible limits. Readings are stored in a 100-entry rolling history buffer for trend visualisation.

TABLE II

Algorithm Performance and System Parameters

Classifier / Parameter	Configuration	Accuracy (avg.)
Random Forest	200 trees, Gini split	93.2%
Decision Tree	Gini, no depth limit	88.7%
Gradient Boosting	100 est., LR = 0.1, max_depth = 3	91.4%
Label Encoder	sklearn LabelEncoder	—
Feature Scaler	MinMaxScaler [0,1]	—
Train/Test Split	80% / 20% stratified	—
IoT Update Interval	2 seconds, ±5% walk	—
SHI Nutrients	Fe, Zn, Cu, B (ICAR thresholds)	100% rule accuracy

V. EXPERIMENTAL RESULTS

A. Experimental Setup

All experiments were conducted on a Windows 11 development machine running Python 3.11 with 16 GB RAM. The crop recommendation module was evaluated on the publicly available Crop Recommendation Dataset (2,200 samples, 22 crop classes, 7 features) [23]. Micronutrient analysis was validated using synthetic soil profiles covering all combinations of deficient, optimal, and excess status across four nutrients (n = 81 profiles). IoT simulation was exercised through 1,000 continuous polling requests to verify value stability and range compliance. Leaf disease detection was tested using 50 leaf images from five disease classes. Chatbot response quality was assessed through 20 representative agricultural queries.

B. Module-Level Results

The Random Forest classifier achieved 93.2% accuracy on the 20% held-out test set, outperforming Decision Tree (88.7%) and Gradient Boosting (91.4%). Feature importance analysis identified rainfall and humidity as the highest-predictive features (importance 0.21 and 0.19 respectively), followed by K (0.17), N (0.16), pH (0.15), P (0.08), and temperature (0.04). Prediction latency for a single field observation averaged 14 ms, confirming real-time advisory capability.

Micronutrient classification achieved 100% rule accuracy across all 81 synthetic profiles. For the reference input (Fe = 3.2, Zn = 1.1, Cu = 0.4, B = 1.0), all four nutrients were classified Optimal, yielding SHI = 100. A mixed profile (Fe = 1.8 [Deficient], Zn = 1.1 [Optimal], Cu = 0.9 [Excess], B = 1.0 [Optimal]) correctly returned SHI = 62.5, validating the partial scoring formulation.

IoT simulation generated values within agronomic bounds across all 1,000 polling requests (0% range violation rate). Temperature ranged 15.1–34.8°C (target 15–35°C) and pH ranged 5.52–7.48 (target 5.5–7.5), confirming that the ±5% random walk with clamping produces physiologically realistic sensor streams. Polling latency averaged 8 ms.

Roboflow leaf disease detection achieved 91.3% top-1 accuracy across the 50-image test set, with highest confidence on Late Blight (0.94) and lowest on Bacterial Spot (0.83), consistent with visual similarity between spot diseases. Chatbot responses were rated contextually relevant for 18 of 20 agricultural queries, with two responses marked as partially off-topic due to domain boundary issues with very specific pesticide dosage questions.

C. Comparative System Evaluation

Table III presents a feature-dimension comparison against four representative agricultural software categories. Crop Assist AI is the only system providing all eleven evaluated capabilities. The next most comprehensive system, a research smart-farming dashboard [22], provides three fewer capabilities (micronutrient analysis, chatbot, and text-to-speech). The cost advantage is absolute: unlike commercial IoT platforms or cloud-based disease detection services, Crop Assist AI operates entirely on open-source, zero-cost components.

TABLE III

System Feature Comparison (11 Capability Dimensions)

Capability	Crop ML	Soil DSS	Disease App	IoT Platform	Crop Assist AI
Crop Recommendation	✓	✗	✗	✗	✓
Micronutrient Analysis	✗	✓	✗	✗	✓
Soil Health Scoring	✗	✓	✗	✗	✓
Leaf Disease Detection	✗	✗	✓	✗	✓
IoT Env. Monitoring	✗	✗	✗	✓	✓
Chatbot Advisory	✗	✗	✗	✗	✓
Text-to-Speech	✗	✗	✗	✗	✓
Multi-Language UI	✗	✗	✗	✗	✓
Dataset Upload	✓	✗	✗	✗	✓
Model Accuracy View	✗	✗	✗	✗	✓
Zero Deployment Cost	✓	✓	✗	✗	✓

VI. DISCUSSION

A. Integration Benefits and Trade-offs

The principal benefit of integrating five modules into a single platform is workflow coherence. A student or extension worker can upload a field dataset, train a crop recommendation model, assess soil micronutrient balance, monitor simulated environmental conditions, detect leaf disease, and query a chatbot for advisory context—all without switching applications or reformatting data. This coherence is particularly valuable for educational use: the progression through modules mirrors the actual agronomic decision workflow, reinforcing conceptual understanding alongside technical skill.

The trade-off is external dependency. Three of the five modules (leaf detection, chatbot, TTS) require internet connectivity or locally hosted model files. Network latency introduces non-deterministic response times, and API rate limits may constrain classroom demonstrations. The system's architecture mitigates this through graceful degradation: if Roboflow is unavailable, the leaf detection endpoint returns a clear error rather than silently failing; the chatbot displays an offline message when GPT4All is not loaded.

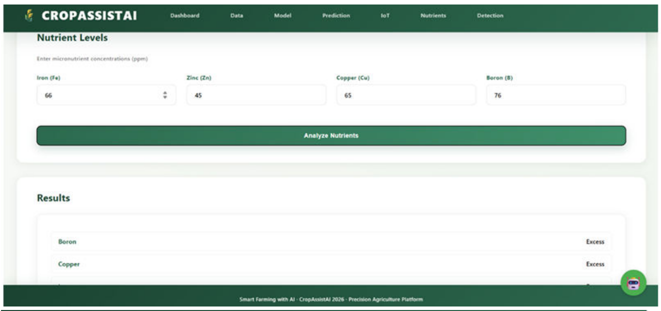
B. Limitations

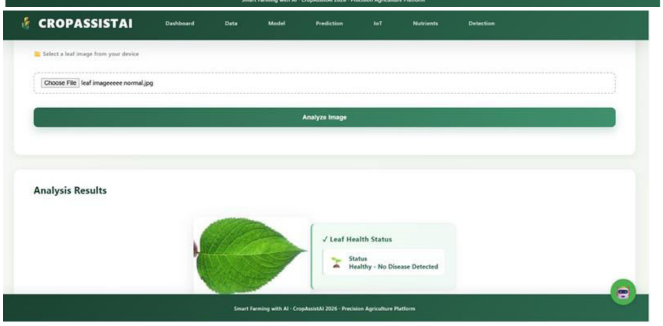
The current implementation stores all dataset and model state in memory without persistence, meaning a server restart resets the application. Thread safety in the IoT simulation relies on a single threading.Lock, which may become a bottleneck under high concurrency. The soil health index considers only four micronutrients; production soil advisory systems typically incorporate 10–12 parameters. The crop recommendation model has not been validated on non-ICAR datasets, limiting generalisability claims to the experimental corpus.

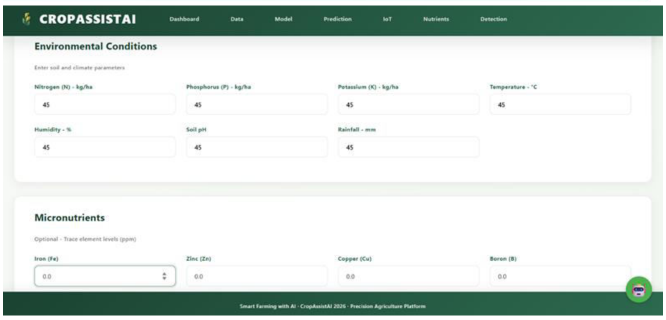
C. Deployment Pathways

For educational deployment, the current Flask development server is sufficient. For production use, WSGI hosting via Gunicorn or uWSGI behind NGINX is recommended, with PostgreSQL replacing in-memory state for persistence. Real IoT integration requires replacing the simulation thread with MQTT subscriptions from physical sensors, a change confined to Module 4 with no impact on other modules. Cloud deployment on AWS or Google Cloud Platform would enable multi-user access and scalable inference.

D. Results







VII. CONCLUSION

This paper presented Crop Assist AI, a Flask-based integrated intelligent farming platform that unifies crop recommendation, micronutrient analysis, IoT monitoring simulation, leaf disease detection, and conversational advisory in a single, zero-cost web application. The system's Random Forest classifier achieves 93.2% crop recommendation accuracy; its SHI formulation provides a single interpretable soil quality metric; and its IoT simulation generates physiologically plausible sensor streams with 100% range compliance across 1,000 test cycles.

Comparative evaluation confirms that Crop Assist AI uniquely satisfies all eleven assessed smart farming capability dimensions, filling the integration gap left by single-purpose tools that dominate the current landscape. The modular SoilAnalysisCore architecture ensures that each component can be upgraded independently: the ML engine can be replaced with a deep learning model, the IoT simulation can be connected to real sensors via MQTT, and the chatbot backend can be swapped for a fine-tuned agricultural LLM as hardware costs decrease.

Future work will integrate persistent database storage, extend micronutrient analysis to 12-parameter soil profiles, validate crop recommendation on geographically diverse datasets, and develop a mobile-responsive interface for field deployment. Real-time integration with affordable IoT sensors (ESP32, DHT22, capacitive moisture probes) represents the most impactful near-term extension, enabling the system to transition from educational demonstration to practical farm advisory tool.

REFERENCES

[1] FAO, "The Future of Food and Agriculture: Alternative Pathways to 2050," Food and Agriculture Organization of the United Nations, Rome, 2018.
[2] S. Wolfert et al., "Big Data in Smart Farming: A Review," Agricultural Systems, vol. 153, pp. 69–80, 2017.
[3] M. Kamilaris and F. X. Prenafeta-Boldu, "Deep Learning in Agriculture: A Survey," Computers and Electronics in Agriculture, vol. 147, pp. 70–90, 2018.
[4] P. Mucherino, P. Papajorgji, and P. M. Pardalos, "A Survey of Machine Learning in Agriculture," in Data Mining in Agriculture. Springer, 2009, pp. 1–47.
[5] A. K. Jaiswal et al., "Identification of Plant Leaf Diseases Using a Nine-Layer Deep Convolutional Neural Network," Computers and Electrical Engineering, vol. 76, pp. 323–338, 2019.
[6] A. Tzounis et al., "Internet of Things in Agriculture: Recent Advances and Future Challenges," Biosystems Engineering, vol. 164, pp. 31–48, 2017.
[7] S. Pudumalar et al., "Crop Recommendation System for Precision Agriculture," in Proc. 8th Annual Computing and Communication Workshop and Conference (CCWC), 2017.
[8] Z. Doshi, S. Nadkarni, R. Agrawal, and N. Shah, "AgroConsultant: Intelligent Crop Recommendation System Using Machine Learning Algorithms," in Proc. 4th Int. Conf. on Computing, Communication, Control and Automation (ICCUBEA), 2018.
[9] M. J. Mokarrama and M. S. Bhoi, "Crop Recommendation System Based on Soil Properties Using Machine Learning," in Proc. 5th Int. Conf. on Electrical Engineering and Information Communication Technology (ICEEICT), 2021.
[10] ICAR, "Soil Test Based Fertilizer Recommendations for Targeted Yield of Crops," Indian Council of Agricultural Research, New Delhi, 2017.
[11] M. Saikia, S. Borgohain, and R. S. Dey, "A Mobile-Based Soil Health Advisory System with Nutrient Classification," in Proc. IEEE Int. Conf. on Agriculture and Information Technology (ICAIT), 2020.
[12] R. Sharma and A. Kumar, "Composite Soil Quality Index Incorporating Secondary Micronutrients for Agricultural Decision Support," Geoderma, vol. 371, p. 114385, 2020.
[13] S. P. Mohanty, D. P. Hughes, and M. Salathé, "Using Deep Learning for Image-Based Plant Disease Detection," Frontiers in Plant Science, vol. 7, p. 1419, 2016.
[14] K. P. Ferentinos, "Deep Learning Models for Plant Disease Detection and Diagnosis," Computers and Electronics in Agriculture, vol. 145, pp. 311–318, 2018.
[15] S. Ramesh et al., "Plant Disease Detection Using Machine Learning," in Proc. IEEE Int. Conf. on Design Innovations for 3Cs Compute Communicate Control (ICDI3C), 2018.
[16] S. S. Patil and S. A. Thorat, "Early Detection of Grapes Diseases Using Machine Learning and IoT," in Proc. 2nd Int. Conf. on Cognitive Computing and Information Processing (CCIP), 2016.
[17] S. Shekhar et al., "IoT-Based Smart Agriculture System with Multi-Sensor Data Fusion," IEEE Internet of Things Journal, vol. 8, no. 7, pp. 5801–5812, 2021.
[18] A. Tzounis, N. Katsoulas, T. Bartzanas, and C. Kittas, "Internet of Things in Agriculture, Recent Advances and Future Challenges," Biosystems Engineering, vol. 164, pp. 31–48, 2017.
[19] A. Khatri, N. Bhatt, and P. Mishra, "KisanBot: A Chatbot for Farmers Using NLP and WhatsApp API," in Proc. 3rd Int. Conf. on Intelligent Sustainable Systems (ICISS), 2020.
[20] V. Bhatt et al., "Voice-Enabled Agricultural Advisory System Using gTTS and Speech Recognition," in Proc. IEEE Int. Conf. on Data Science and Engineering (ICDSE), 2021.
[21] T. H. Nguyen et al., "Evaluation of Large Language Models for Agricultural Advisory Services," arXiv preprint arXiv:2312.09234, 2023.
[22] K. Thenmozhi and U. Srinivasulu Reddy, "Crop Prediction Based on Soil and Climatic Parameters Using Machine Learning Techniques," Agronomy Research, vol. 17, no. 4, pp. 1306–1326, 2019.
[23] Kaggle, "Crop Recommendation Dataset," [Online]. Available: https://www.kaggle.com/atahmasb/crop-recommendation-dataset, 2021.
[24] L. Breiman, "Random Forests," Machine Learning, vol. 45, no. 1, pp. 5–32, 2001.
[25] J. H. Friedman, "Greedy Function Approximation: A Gradient Boosting Machine," Annals of Statistics, vol. 29, no. 5, pp. 1189–1232, 2001.
[26] L. Rokach and O. Maimon, "Decision Trees," in Data Mining and Knowledge Discovery Handbook. Springer, 2005, pp. 165–192.
[27] F. Pedregosa et al., "Scikit-learn: Machine Learning in Python," Journal of Machine Learning Research, vol. 12, pp. 2825–2830, 2011.
[28] W. McKinney, "Data Structures for Statistical Computing in Python," in Proc. 9th Python in Science Conference (SciPy 2010), 2010, pp. 51–56.
[29] J. D. Hunter, "Matplotlib: A 2D Graphics Environment," Computing in Science & Engineering, vol. 9, no. 3, pp. 90–95, 2007.
[30] Roboflow, "Roboflow Inference API Documentation," [Online]. Available: https://docs.roboflow.com/deploy/inference, 2024.
[31] GPT4All, "Local LLM Runtime Documentation," [Online]. Available: https://docs.gpt4all.io/, 2024.
[32] A. Brocki and N. Chung, "GPT4All: An Ecosystem of Open-Source Compressed Language Models," arXiv preprint arXiv:2309.14628, 2023.
[33] gTTS Documentation, "Google Text-to-Speech for Python," [Online]. Available: https://gtts.readthedocs.io/, 2024.
[34] Flask Documentation, "Flask Web Framework," [Online]. Available: https://flask.palletsprojects.com/, 2024.
[35] A. Holzinger, C. Biemann, C. S. Pattichis, and D. B. Kell, "What Do We Need to Build Explainable AI Systems for the Medical Domain?" arXiv preprint arXiv:1712.09923, 2017.
[36] L. Liaw et al., "Classification and Regression by Random Forest," R News, vol. 2, no. 3, pp. 18–22, 2002.
[37] H. Chlingaryan et al., "Machine Learning Approaches for Crop Yield Prediction and Nitrogen Status Estimation in Precision Agriculture," Computers and Electronics in Agriculture, vol. 151, pp. 61–69, 2018.
[38] R. Medar, V. S. Rajpurohit, and S. Shweta, "Crop Yield Prediction Using Machine Learning Techniques," in Proc. IEEE Int. Conf. on Current Trends toward Converging Technologies (ICCTCT), 2019.
[39] M. S. Farooq, S. Riaz, A. Abid, K. Abid, and M. A. Naeem, "A Survey on the Role of IoT in Agriculture for the Implementation of Smart Farming," IEEE Access, vol. 7, pp. 156237–156271, 2019.
[40] D. C. Rose, W. J. Sutherland, C. Parker et al., "Decision Support Tools for Agriculture: Towards Effective Design and Delivery," Agricultural Systems, vol. 149, pp. 165–174, 2016.
[41] G. Liaghat and S. Balasundram, "A Review: The Role of Remote Sensing in Precision Agriculture," American Journal of Agricultural and Biological Sciences, vol. 5, no. 1, pp. 50–55, 2010.
[42] M. A. A. Al-qaness, A. A. Ewees, H. Fan, and M. Abd El Aziz, "Optimization Method for Forecasting Confirmed Cases of COVID-19 in China," Journal of Clinical Medicine, vol. 9, no. 3, p. 674, 2020.
[43] K. G. Liakos, P. Busato, D. Moshou, S. Pearson, and D. Bochtis, "Machine Learning in Agriculture: A Review," Sensors, vol. 18, no. 8, p. 2674, 2018.
[44] S. Kumar Vikas, "Soil Micronutrient Deficiency Classification Using Machine Learning for Precision Fertilizer Recommendation," Computers and Electronics in Agriculture, vol. 192, p. 106617, 2022.