iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** VII **Month of publication:** July 2026

DOI: https://doi.org/10.22214/ijraset.2026.84227

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue VII July 2026- Available at www.ijraset.com

Crop Recommendation System Using Machine Learning

Yashashwini L[1], Supriya S[2], Anitha J[3]
[1, 2]*Students, Department of Master of Computer Applications (MCA), Ambedkar Institute of Technology, Bengaluru, Karnataka, India*
[3]*Assistant Professor, Department of Master of Computer Applications (MCA), Ambedkar Institute of Technology, Bengaluru, Karnataka, India*

Abstract: ***Choosing the right crop each season is one of the toughest calls a farmer has to make, largely because soil composition, rainfall, and local climate rarely behave predictably. Rather than leaning on field-specific, data-backed guidance, many farmers still fall back on regional tradition or broad advisory services, a habit that often leads to lower yields, wasted fertilizer, and real financial loss. This paper presents an AI-driven crop recommendation system that combines supervised machine learning with soil-climatic analytics to make crop selection more data-driven. The system takes seven measurable soil and environmental inputs, nitrogen (N), phosphorus (P), potassium (K), temperature, humidity, pH, and rainfall, and classifies the most suitable crop from a set of 22 candidate crops using a trained Random Forest classifier. Five classification algorithms, Decision Tree, Random Forest, Support Vector Machine, Naive Bayes, and Gradient Boosting, were trained and compared on a balanced dataset of 2,200 labeled samples (100 samples per crop). The Random Forest model achieved the best overall balance of accuracy and stability, reaching 99.55% test accuracy, a 99.43% out-of-bag score, and a five-fold cross-validation mean of 99.36% (± 0.39%). The system is implemented as a Flask-based web application with a trained-model backend, offering crop prediction, confidence scoring, and an integrated plant disease advisory module. Because the trained model is lightweight and does not depend on a constant internet connection once deployed, the system is well suited to rural and underserved farming communities, offering a practical, low-cost step toward precision agriculture.***

Keywords: Crop Recommendation, Machine Learning, Random Forest, Precision Agriculture, Soil Analytics, Classification, Decision Support System.

I. INTRODUCTION

Agricultural productivity sits at the center of food security and rural economic stability, yet crop selection decisions are under growing strain from unpredictable climate patterns, deteriorating soil quality, and uneven access to agronomic expertise. Whether it is a smallholder farmer working a fragmented plot or a large operation weighing seasonal risk, figuring out what to plant has become harder to judge from experience alone. Even with growing awareness of precision agriculture, optimal crop selection still eludes many farmers because of limited soil testing infrastructure, a shortage of localized advisory services, and the absence of personalized, parameter-driven recommendation tools. Traditional advisory approaches tend to rely on broad regional guidelines that overlook the diversity of soil and microclimatic conditions found across individual farm plots.

To address these gaps, artificial intelligence is increasingly being explored as a way to make agricultural decision-making both more precise and more accessible. This research focuses on building a machine learning-based system that uses measurable soil and climatic parameters, nitrogen, phosphorus, potassium, temperature, humidity, pH, and rainfall, to classify the most suitable crop for a given field. Instead of relying on generalized advisory text or image-based diagnosis, the system draws its insights from structured, quantifiable inputs, which makes it easier to implement, interpret, and deploy at scale. The proposed system was built and evaluated on the widely used Kaggle Crop Recommendation dataset, comprising 2,200 records evenly distributed across 22 crop classes, and packaged into a Flask-based web application that also incorporates a supplementary plant disease advisory module for a more complete farm decision-support experience.

II. LITERATURE REVIEW

A growing body of research has explored machine learning as a tool for crop selection and precision agriculture, motivated by the same practical gap this paper addresses: the mismatch between generalized agronomic advice and the field-specific conditions that actually determine whether a crop will thrive.

Sani et al. compared K-Nearest Neighbours, Decision Tree, Random Forest, and Support Vector Machine models for crop recommendation using nitrogen, phosphorus, potassium, and humidity as inputs, and found Random Forest gave the most consistent accuracy across the candidate algorithms, ultimately adopting it for their proposed system [1]. This mirrors the comparative approach taken in the present study.

Jha, Ranjan, and Gaur proposed a machine learning pipeline that recommends both suitable crops and complementary fertilizers, emphasizing that pairing crop advice with input-management guidance produces more actionable outcomes for farmers than crop suggestions alone [2].

A 2024 study published in Human-Centric Intelligent Systems evaluated nine different machine learning algorithms, including Logistic Regression, Support Vector Machine, K-Nearest Neighbours, Decision Tree, Random Forest, Bagging, AdaBoost, Gradient Boosting, and Extra Trees, on a Kaggle-sourced crop recommendation dataset, and reported that Random Forest consistently outperformed the other eight models across accuracy, precision, recall, and F1-score [3].

Research published in the ITEGAM-JETIA journal combined XGBoost with conventional classifiers such as Decision Tree, Support Vector Machine, Logistic Regression, Random Forest, and Naive Bayes for crop recommendation, and reported ensemble boosting methods delivering competitive or superior performance to standalone tree-based models on soil-property datasets [4].

A 2023 study in Frontiers in Plant Science proposed an ensemble machine learning recommendation scheme for agricultural crop cultivation, evaluating models using error-based metrics (MAE, MSE, RMSE, R^2) in addition to classification accuracy, underlining that ensemble approaches tend to generalize better across heterogeneous soil-climate datasets than any single base learner [5].

Adebiyi, Ogundokun, and Abokhai built a machine learning-aided mobile system for farmland optimization using location, crop type, soil type, soil pH, and spacing as inputs, applying a Random Forest algorithm to classify crop subclasses and demonstrating the feasibility of deploying such models through lightweight mobile interfaces [6].

On the sensing side, a 2025 review published in the Bulletin of the National Research Centre described a low-cost, multi-sensor IoT system that measures soil pH, moisture, temperature, and NPK levels in real time and relays this data to a mobile application, illustrating how automated sensor input could eventually replace manual parameter entry in systems like the one proposed here [7].

Beyond crop selection, plant health monitoring is a closely related research thread. A 2025 review in the Journal of Imaging surveyed both classical machine learning and deep learning approaches to plant disease detection from leaf imagery, noting that classical models such as Random Forest and Support Vector Machine remain competitive on smaller, resource-constrained datasets, while deep convolutional models tend to dominate on larger benchmark datasets [8]. This distinction directly informed the design of the disease advisory module integrated into the present system.

Across this body of work, Random Forest emerges repeatedly as the strongest or most consistent performer among classical machine learning models for tabular soil-climate data, which aligns with the comparative results obtained in this study. However, most existing systems either stop at a single trained model without a transparent multi-algorithm comparison, or focus narrowly on either crop recommendation or disease detection rather than combining both into a single deployable advisory platform. The system proposed in this paper addresses that gap by (i) benchmarking five classification algorithms side by side on the same dataset and splitting methodology, (ii) reporting a full set of evaluation metrics (accuracy, precision, F1-score, out-of-bag score, and cross-validation stability) rather than accuracy alone, and (iii) packaging the resulting model into a complete, lightweight web application that also includes a plant disease advisory module, rather than a standalone notebook or research prototype.

III. PROBLEM STATEMENT

Despite the availability of soil testing services and agricultural extension programs, most smallholder and mid-sized farmers still make crop selection decisions based on regional tradition, informal peer advice, or generalized government advisory bulletins that do not account for plot-specific soil nutrient levels, pH, or microclimatic variation. This mismatch between generic advice and field-specific conditions contributes to suboptimal yields, inefficient fertilizer use, and, in some cases, complete crop failure. Formal agronomic consultation, where available, is often slow, unevenly distributed, and inaccessible to farmers in remote or under-resourced regions.

There is a clear need for a lightweight, data-driven decision-support tool that can take a small set of measurable soil and climatic parameters, values a farmer can obtain from a basic soil test kit or local weather data, and return an accurate, confidence-scored crop recommendation without requiring specialist interpretation. This paper addresses that need directly.

IV. OBJECTIVES

The key objectives of this study are as follows:

1) Design a parameter-based crop recommendation tool that classifies the most suitable crop from soil and climatic input values (N, P, K, temperature, humidity, pH, and rainfall).
2) Train and comparatively evaluate five machine learning classification algorithms, Decision Tree, Random Forest, Support Vector Machine, Naive Bayes, and Gradient Boosting, to identify the most reliable model for this task.
3) Implement a standardized, seven-feature input set for consistent, reproducible classification across all candidate crops.
4) Deliver a confidence-scored crop recommendation along with a simple, actionable output for end users.
5) Package the trained model into a lightweight, deployable Flask web application suitable for rural advisory centers, mobile-friendly browsers, or kiosk-based agricultural support systems.
6) Extend the platform with a supplementary plant disease advisory module to move toward a more complete, end-to-end farm decision-support tool.

V. METHODOLOGY

The proposed system brings together structured soil and climatic data analysis with supervised machine learning to support data-driven crop selection. It follows a systematic process built around data preprocessing, feature-based classification, comparative model evaluation, and recommendation generation. Because the approach relies on structured numeric input rather than image-based or free-text data, it stays practical for everyday use by farmers and agricultural extension workers, even in areas with limited connectivity.

A. Data Collection and Feature Set

The system relies on seven standardized input parameters that agronomic literature consistently identifies as primary drivers of crop suitability: nitrogen (N), phosphorus (P), and potassium (K) soil content, ambient temperature, relative humidity, soil pH, and seasonal rainfall. Depending on the deployment context, these values can come from standard soil testing kits, regional meteorological data, or IoT-based field sensors. The 22 candidate crops the system classifies among include:

- Cereal crops: rice, maize
- Pulses and legumes: chickpea, kidneybeans, pigeonpeas, mothbeans, mungbean, blackgram, lentil
- Fiber crops: cotton, jute
- Fruit crops: pomegranate, banana, mango, grapes, watermelon, muskmelon, apple, orange, papaya, coconut
- Plantation/beverage crop: coffee

This standardized feature set ensures that every candidate crop is judged against the same consistent, measurable input profile across all classification models.

B. Data Preprocessing

Before training, the collected data was checked for missing or inconsistent values (none were found in the dataset) and feature ranges were examined for outliers. A StandardScaler transformation was then applied so that features measured on very different numeric scales, such as rainfall in millimeters versus pH on a roughly 3.5-10 scale, do not unfairly dominate distance-sensitive classifiers such as Support Vector Machine. Categorical crop labels were encoded numerically using a label encoder. The dataset was subsequently split into training and testing subsets (80% train, 20% test) using stratified sampling, so that every crop class was proportionally represented in both subsets and model generalization could be evaluated fairly.

C. Rule-Assisted Suitability Scoring

Alongside the primary classification output, the system also computes a suitability confidence score for the recommended crop, drawn from the classifier's prediction probability. This score is then mapped to a defined confidence band, shown in Table 1, so end users can quickly gauge how much weight to place on a given recommendation.

Suitability Score Range	Recommendation Confidence
0.90 - 1.00	High Confidence
0.70 - 0.89	Moderate Confidence
0.50 - 0.69	Low Confidence
< 0.50	Not Recommended

Table 1: Suitability Score Range and Recommendation Confidence

D. Machine Learning-Based Classification

To go beyond simple feature-matching logic, the system applies trained classification models to the structured input parameters. Machine learning earns its place here in a few important ways:

- Multi-Class Crop Classification: Maps a combined soil-climate feature vector to the most probable crop class among the 22 candidates.
- Feature Importance Analysis: Identifies which input parameters most strongly influence crop suitability for a given prediction (see Section 8, Table 5).
- Confidence Estimation: Derives a prediction probability score used to compute the suitability confidence band shown in Table 1.
- Comparative Model Validation: Evaluates five algorithms against one another to select the most reliable classifier for deployment (see Section 12, Table 4).

Rather than generating interpretations from images or text, this machine learning step works from structured numeric input, giving a data-driven read on measurable field conditions.

E. Recommendation Generation

Once classification is complete, the system presents the user with a simple, actionable result. The recommendation includes:

- The recommended crop (e.g., "Recommended Crop: Rice")
- A visual suitability score summary
- A confidence-based suggestion:
    - – "Highly suitable for current field conditions"
    - – "Moderately suitable — consider supplementary soil treatment"
    - – "Low suitability — consider alternative crop options"

This makes the platform well suited to advisory support, planning assistance, and field-level decision-making, though it is not meant to replace professional agronomic consultation for complex or high-value cultivation decisions.

F. Lightweight and Offline Capability

The trained classification model is compact enough to run on lightweight hardware, keeping the system both accessible and cost-effective.

Since it does not need a constant internet connection once deployed, it can operate comfortably in remote or resource-limited settings, including rural agricultural extension centers and community farming cooperatives.

VI. SYSTEM ARCHITECTURE

The architecture behind the proposed AI-driven crop recommendation system is built for modularity, scalability, and ease of deployment.

It follows a layered structure that processes soil and climatic input efficiently and delivers meaningful crop suitability insights without demanding complex agronomic infrastructure. The system is implemented as a Flask web application, with the following interconnected components:

A. User Input Layer

This front-end module, implemented through the application's template pages (including index, login, register, dashboard, and predict pages), is where users enter soil and climatic parameter values, either by hand or through integration with IoT soil sensors and weather APIs. The interface is kept deliberately minimal and farmer-friendly, relying on numeric input fields, which makes it easy to deploy on mobile devices, tablets, or kiosks at agricultural extension centers.

B. Preprocessing Engine

Once a user submits input values, the system validates and normalizes them using the saved StandardScaler object (persisted as scaler.pkl) into the format the trained classification model expects, keeping things consistent no matter what the original measurement scale or data source was.

C. Classification Module

This backend module, implemented in crop_intelligence.py and app.py, runs the trained Random Forest classifier (persisted as crop_model.pkl, alongside its label encoder in label_encoder.pkl) on the preprocessed feature vector, producing a predicted crop class together with a prediction probability. Five candidate algorithms were evaluated during development to settle on Random Forest as the most reliable classifier, as detailed in Section 12.

D. Suitability Scoring Engine

This module converts the classifier's prediction probability into a suitability confidence band (Table 1), keeping the output consistent and explainable regardless of which classification algorithm sits underneath.

E. Recommendation and Report Generator

By combining the classification output with the suitability score and crop-specific profile data (persisted as crop_profiles.pkl), this module puts together a human-readable summary through the application's results and crop_detail pages: the recommended crop, its confidence level, an explanation of the key contributing parameters, and a suggested course of action.

F. Data Handling and Extended Advisory Layer

The system is built around lightweight, efficient data handling, with minimal storage of personally identifying information and support for both online and offline deployment. Beyond core crop recommendation, the application also integrates a supplementary plant disease advisory module (disease_predict, disease_info, and disease_guide pages, backed by a separate lightweight disease detection model), a fertilizer guidance page, and a field analysis page, extending the platform from a single-purpose classifier into a broader farm decision-support tool.

VII. MACHINE LEARNING MODEL

At the core of the recommendation engine is a supervised multi-class classification model trained to map a seven-dimensional soil-climate feature vector to one of 22 crop labels. This section describes the model family selected for deployment and the rationale behind that choice, ahead of the full comparative results presented in Section 12.

The deployed model is a Random Forest Classifier, an ensemble learning method that constructs a large number of decision trees during training and outputs the class that receives the majority vote across all trees. Random Forest was selected as the production model for three main reasons: it handles the non-linear relationships between soil nutrient levels and crop suitability more robustly than a single decision tree, it is far less prone to overfitting due to its use of bootstrap aggregation (bagging) and random feature subsampling, and it naturally produces feature importance scores that make the model's reasoning interpretable to non-technical stakeholders. The production configuration uses 500 estimators, balanced class weighting to account for any class imbalance, "sqrt" feature subsampling at each split, and out-of-bag scoring enabled for an additional, built-in validation signal. Model artifacts, the trained classifier, label encoder, feature scaler, and crop profile data, are persisted using joblib for fast reloading inside the Flask application without retraining.

VIII. DATASET DESCRIPTION

The system was trained and evaluated on the Crop Recommendation dataset (Crop_recommendation.csv), a widely used, publicly available agronomic dataset originally compiled from Indian soil, climate, and fertilizer data sources and distributed via Kaggle [10].

Feature	Min	Max	Mean	Std. Dev.
Nitrogen, N (kg/ha)	0.00	140.00	50.55	36.92
Phosphorus, P (kg/ha)	5.00	145.00	53.36	32.99
Potassium, K (kg/ha)	5.00	205.00	48.15	50.65
Temperature (°C)	8.83	43.68	25.62	5.06
Relative Humidity (%)	14.26	99.98	71.48	22.26
Soil pH	3.50	9.94	6.47	0.77
Rainfall (mm)	20.21	298.56	103.46	54.96

Table 2: Statistical Summary of Dataset Features

The dataset contains 2,200 records with no missing values and no duplicate rows, evenly distributed across 22 crop classes with exactly 100 samples per class, which avoids the class-imbalance problems that often complicate multi-class agricultural datasets. Table 2 summarizes the statistical range of each of the seven numeric features. Nitrogen, phosphorus, and potassium values span a wide range (0-205 kg/ha depending on nutrient), reflecting the very different nutrient demands of crops such as pulses versus fruit trees, while temperature (8.8-43.7 °C), humidity (14.3-100%), pH (3.5-9.9), and rainfall (20.2-298.6 mm) together capture a broad diversity of agro-climatic zones. The 22 target crop labels are: rice, maize, jute, cotton, coconut, papaya, orange, apple, muskmelon, watermelon, grapes, mango, banana, pomegranate, lentil, blackgram, mungbean, mothbeans, pigeonpeas, kidneybeans, chickpea, and coffee.

IX. ALGORITHMS USED

Five supervised classification algorithms, spanning simple, ensemble, kernel-based, and probabilistic approaches, were trained and evaluated on identical training and test splits to identify the most reliable classifier for deployment.

1) Decision Tree: A single tree-structured classifier that recursively splits the feature space based on information gain. Included as a fast, interpretable baseline against which ensemble improvements can be measured.
2) Random Forest: An ensemble of decision trees trained on bootstrapped samples with random feature subsampling, combined by majority vote. Chosen as the production model for its robustness to overfitting and built-in feature importance ranking.
3) Support Vector Machine (SVM): A kernel-based classifier (radial basis function kernel used here) that finds the optimal separating boundary between crop classes in a transformed feature space. Included to test whether a margin-maximizing approach could outperform tree-based methods on this dataset.
4) Naive Bayes: A probabilistic classifier (Gaussian variant used here) based on Bayes' theorem with a conditional-independence assumption between features. Included as a lightweight, low-compute baseline given its very fast training and inference time.
5) Gradient Boosting: An ensemble method that builds trees sequentially, with each new tree correcting the errors of the previous ensemble. Included as a boosting-family comparison point alongside the bagging-based Random Forest model.

X. PROPOSED SYSTEM

The proposed system reframes crop selection as a structured, parameter-based multi-class classification problem, replacing generalized regional advisory text with field-specific, data-driven recommendations. Unlike traditional advisory approaches, which rely heavily on farmer experience and broad regional guidance, the proposed system automates multi-feature analysis of measurable soil and climatic conditions and returns a recommendation in real time.

Parameter	Traditional Methods	Proposed System
Decision Basis	Farmer experience / regional tradition	Data-driven ML classification (Random Forest)
Soil Testing Use	Manual interpretation by agronomist	Automated seven-feature analysis
Scalability	Limited by expert availability	Scalable to a large user base at negligible marginal cost
Personalization	Generic regional advice	Field-specific, confidence-scored recommendation
Turnaround Time	Days (lab testing + consultation)	Seconds (real-time prediction)

Table 3: Comparison with Traditional Crop Selection Methods

By replacing subjective, experience-based judgment with a trained classifier evaluated against held-out test data, the proposed system offers a transparent, reproducible, and scalable alternative to conventional crop advisory practice, while still leaving room for a human agronomist to review borderline, low-confidence recommendations.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue VII July 2026- Available at www.ijraset.com

XI. IMPLEMENTATION DETAILS

The proposed system was implemented as a Flask-based web application backed by a persisted scikit-learn model pipeline. The codebase is organized around a small number of clearly separated responsibilities, which keeps the system maintainable and easy to extend.

1) Application Entry Point (app.py): Defines the Flask routes that connect the front-end template pages to the underlying prediction logic, handling user requests for crop prediction, disease prediction, dashboard views, and account management (login and registration).

2) Core Prediction Logic (crop_intelligence.py): Encapsulates the feature ordering, crop profile construction, and prediction helper functions used by the trained model, keeping the machine learning logic decoupled from the web-serving layer.

3) Model Training Pipeline (model_training.py): Loads Crop_recommendation.csv, encodes crop labels, scales the seven input features, splits the data into stratified training and test sets, trains the production Random Forest classifier (500 estimators, balanced class weighting, out-of-bag scoring enabled), evaluates it via hold-out testing and five-fold cross-validation, and persists the resulting model, label encoder, scaler, and crop profile artifacts using joblib for reuse by the web application.

4) Disease Detection Module (disease_training.py, lightweight_disease_detector.py): A supplementary module that trains a lightweight image-based classifier on a labeled plant disease image dataset, saving the resulting model for use by the disease_predict and disease_predict_results template pages.

5) Front-End Templates (templates/): A set of HTML pages, including index, login, register, dashboard, predict, results, crop_detail, field_analysis, fertilizer_guide, disease_guide, disease_info, disease_predict, and disease_predict_results, that together provide the full user-facing workflow from account access through crop prediction to supplementary advisory content.

6) Persisted Model Artifacts (models/): Stores the trained crop_model.pkl, label_encoder.pkl, scaler.pkl, and crop_profiles.pkl files, along with the disease detection model, so that the web application can load a ready-to-use pipeline at startup rather than retraining on every request.

XII. SCREENSHOTS

Figures 1 and 2 show the organization of the project's codebase and trained model artifacts, illustrating how the machine learning pipeline, web application, and front-end templates are structured within the repository.

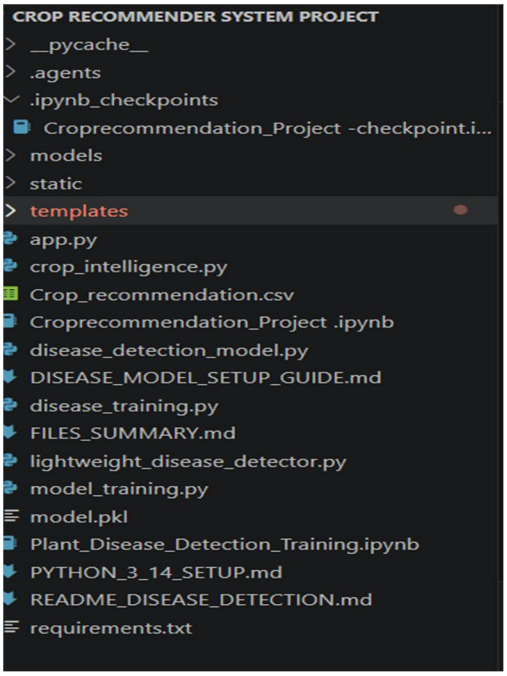


Fig. 1. Project directory structure showing the templates folder (front-end pages including dashboard, predict, results, crop_detail, and the disease advisory pages) alongside the core application files app.py, crop_intelligence.py, and the trained-data source Crop_recommendation.csv.

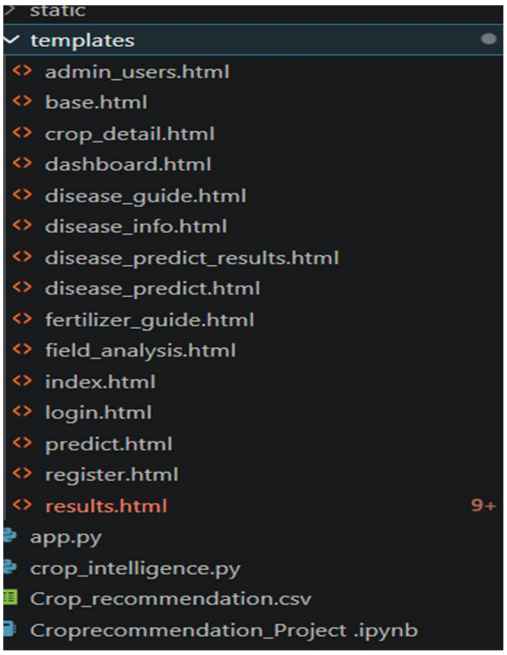


Fig. 2. Full project structure showing the models directory (trained classifier and encoder artifacts), the training scripts (model_training.py, disease_training.py), supporting documentation files, and the requirements.txt dependency manifest.

XIII. RESULTS AND DISCUSSION

The proposed AI-driven crop recommendation framework was evaluated by training and testing all five candidate algorithms on the same stratified 80/20 split of the 2,200-record dataset, yielding 1,760 training samples and 440 held-out test samples across the 22 crop classes.

Algorithm	Accuracy (%)	Precision (%)	F1-Score (%)
Decision Tree	97.95	98.06	97.94
Random Forest	99.55	99.57	99.55
Support Vector Machine	98.41	98.56	98.40
Naive Bayes	99.55	99.59	99.54
Gradient Boosting	98.86	98.97	98.87

Table 4: Comparative Performance of Evaluated Classification Algorithms (Test Set, n = 440)

Random Forest and Naive Bayes achieved the joint-highest classification accuracy on the held-out test set (99.55%), with Random Forest edging ahead marginally on F1-score (99.55% versus 99.54%) and Naive Bayes edging ahead marginally on precision (99.59% versus 99.57%). Decision Tree, the simplest model tested, trailed the ensemble methods at 97.95% accuracy, consistent with its greater susceptibility to overfitting on a dataset with 22 fine-grained classes. Support Vector Machine (98.41%) and Gradient Boosting (98.86%) both performed respectably but did not surpass Random Forest. Random Forest was selected as the production model both for this narrow performance edge on F1-score and because bagging-based ensembles of this kind tend to be more robust to noisy or borderline feature values at inference time than a single probabilistic model, while also providing interpretable feature importance scores (Table 5) that a single Naive Bayes classifier cannot.

Feature	Relative Importance
Rainfall	22.3%
Humidity	21.4%
Potassium (K)	17.9%
Phosphorus (P)	15.2%
Nitrogen (N)	10.5%
Temperature	7.5%
Soil pH	5.3%

Table 5: Feature Importance Ranking (Production Random Forest Model)

Where misclassifications did occur across the five models, they were concentrated among crop pairs with agronomically overlapping requirements, for example between crops that share similar rainfall and humidity bands, which is consistent with the feature importance ranking in Table 5: crops that are close together in rainfall-humidity space are naturally harder for any classifier to separate than crops from very different climatic zones.

XIV. PERFORMANCE EVALUATION

Beyond the single train/test split reported in Table 4, the production Random Forest model was further validated using out-of-bag (OOB) estimation and five-fold cross-validation to check that its strong test-set performance was not an artifact of a favorable split.

- Held-out test accuracy: 99.55% (n = 440 test samples)
- Out-of-bag (OOB) score: 99.43%, computed internally during training from the bootstrap samples left out of each tree
- Five-fold cross-validation mean accuracy: 99.36% (standard deviation ± 0.39%), confirming stable performance across different data partitions rather than a single lucky split

The close agreement between the held-out test accuracy (99.55%), the OOB score (99.43%), and the cross-validation mean (99.36%) indicates that the model generalizes consistently rather than overfitting to a single split of the data. Table 5 shows that rainfall and humidity are the two most influential predictors in the trained model, together accounting for roughly 44% of total feature importance, followed by the three soil-nutrient features (potassium, phosphorus, and nitrogen) at a combined 43.6%, with temperature and pH contributing smaller but non-trivial shares. This ranking is agronomically sensible: water availability (rainfall, humidity) and soil nutrient composition are well-established primary determinants of crop suitability, while temperature and pH act more as secondary constraints within a broadly suitable range.

XV. ADVANTAGES

The proposed system offers several practical benefits over both traditional advisory practices and many existing agricultural mobile applications:

- Accessibility: Designed for broad use, from agricultural extension centers to individual farmer self-assessment via a mobile-friendly web interface.
- Data-Driven Precision: Replaces generalized regional advisory with field-specific, parameter-based recommendations, backed by a model validated to 99.55% test accuracy.
- Transparency: Feature importance scores (Table 5) let users and extension officers see which inputs are driving a given recommendation, rather than treating the model as an opaque black box.
- Scalability: A single trained model can serve a large number of users simultaneously with negligible marginal computational cost.
- Offline-Capable Deployment: The persisted model artifacts (crop_model.pkl, scaler.pkl, label_encoder.pkl) allow the system to run in low-connectivity rural settings once deployed locally.
- Extensibility: The same platform already integrates a supplementary plant disease advisory module, fertilizer guidance, and field analysis pages, positioning it as a broader farm decision-support tool rather than a single-purpose classifier.

Parameter	Typical Existing Agri-Apps	Proposed System
Primary Function	Disease detection via image upload, or market price lookup	Soil-climate crop suitability classification, with an integrated disease module
Input Method	Image upload or manual lookup	Seven-parameter soil & climate entry
Model Transparency	Often opaque to the end user	Feature importance reported (Table 5)
Offline Capability	Mostly online-dependent	Lightweight, offline-deployable trained model
Validation Reporting	Accuracy figures rarely disclosed	Test accuracy, OOB score, and 5-fold CV all reported

Table 6: Comparison with Existing Agricultural Advisory Applications

XVI. FUTURE SCOPE

Several avenues exist for future enhancement:

1) Real-World Field Validation: Deploying the system with real farmer cohorts and validating recommendations against actual harvest outcomes rather than held-out test data alone.

2) IoT Sensor Integration: Automating input collection through real-time soil moisture, NPK, and weather sensor feeds to eliminate manual data entry, building on IoT-based precision agriculture approaches already demonstrated in the literature [7].

3) Regional and Crop Variety Expansion: Expanding the candidate crop set beyond the current 22 classes to include region-specific varieties and local cultivar preferences.

4) Fertilizer and Irrigation Recommendations: Extending the fertilizer_guide module to suggest complementary, quantitative fertilizer and irrigation strategies alongside crop selection.

5) Yield Prediction Integration: Adding a secondary regression model to estimate expected yield for the recommended crop under given conditions, rather than suitability classification alone.

6) Disease Detection Enhancement: Expanding the current lightweight disease detector with a larger labeled image dataset and, where computational resources allow, a convolutional neural network backbone, informed by the deep-learning approaches surveyed in recent literature [8].

7) Explainability Enhancements: Extending the existing feature-importance reporting into per-prediction explanations (e.g., SHAP values) so farmers can see exactly why a specific crop was recommended for their entered values.

XVII. CONCLUSION

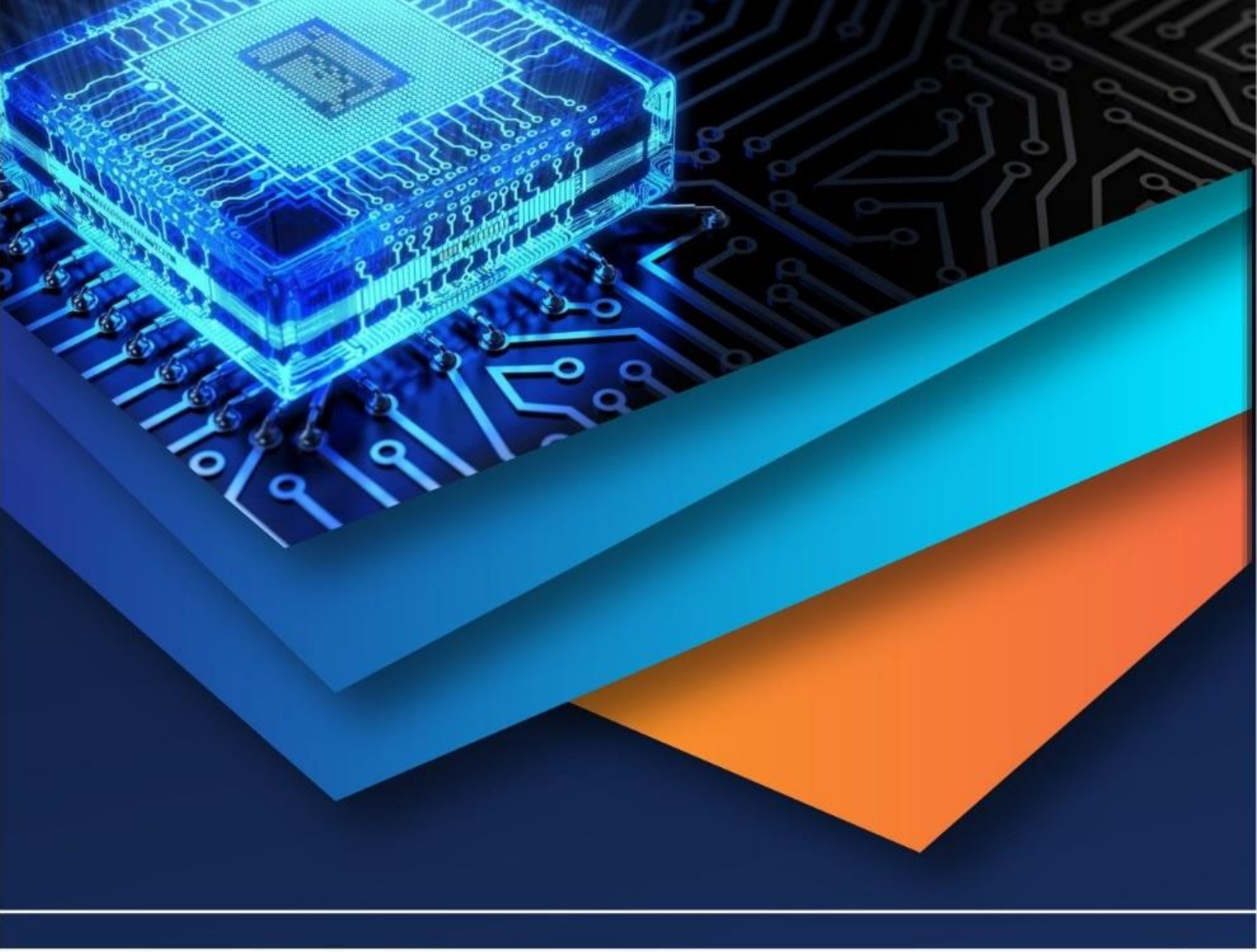
This paper presented a data-driven approach to crop recommendation built around machine learning classification of seven soil and climatic parameters. Five candidate algorithms, Decision Tree, Random Forest, Support Vector Machine, Naive Bayes, and Gradient Boosting, were trained and compared on a balanced, 2,200-record dataset spanning 22 crop classes, with Random Forest selected as the production model after reaching 99.55% test accuracy, a 99.43% out-of-bag score, and a stable 99.36% (± 0.39%) five-fold cross-validation mean. Rather than depending on generalized regional advisory or image-based diagnosis alone, the system offers field-specific, parameter-driven crop suitability classification, packaged into a lightweight Flask web application that also integrates a supplementary plant disease advisory module, fertilizer guidance, and field analysis features. Its modular architecture makes it practical across several domains, including agricultural extension centers, smallholder farming, and government outreach programs, and its lightweight, offline-capable deployment keeps it accessible even in resource-constrained rural settings.

XVIII. ACKNOWLEDGEMENT

The authors gratefully acknowledge the creators and maintainers of the publicly available Crop Recommendation dataset on Kaggle, whose work made the training and evaluation described in this paper possible.

REFERENCES

[1] S. R. Sani, S. V. S. Ummadi, et al., "Crop Recommendation System using Random Forest Algorithm in Machine Learning," in Proc. 2023 2nd Int. Conf. Applied Artificial Intelligence and Computing (ICAAIC), 2023.
[2] G. K. Jha, P. Ranjan, and M. Gaur, "A Machine Learning Approach to Recommend Suitable Crops and Fertilizers for Agriculture," in Recommender System with Machine Learning and Artificial Intelligence: Practical Tools and Applications in Medical, Agricultural and Other Industries, Springer, 2020, pp. 89-99.
[3] "Enhancing Agricultural Productivity: A Machine Learning Approach to Crop Recommendations," Human-Centric Intelligent Systems, Springer Nature, 2024.
[4] "Random Forest Algorithm Use for Crop Recommendation," ITEGAM Journal of Engineering and Technology for Industrial Applications (ITEGAM-JETIA), 2023.
[5] "Ensemble Machine Learning-Based Recommendation System for Effective Prediction of Suitable Agricultural Crop Cultivation," Frontiers in Plant Science, vol. 14, 2023.
[6] M. O. Adebiyi, R. O. Ogundokun, and A. A. Abokhai, "Machine Learning-Based Predictive Farmland Optimization and Crop Monitoring System," Scientific Programming, vol. 2020, Art. no. 9428281, 2020.
[7] "Smart Farming for a Sustainable Future: Implementing IoT-Based Systems in Precision Agriculture," Bulletin of the National Research Centre, Springer Nature, 2025.
[8] T. Nyawose, R. C. Maswanganyi, and P. Khumalo, "A Review on the Detection of Plant Disease Using Machine Learning and Deep Learning Approaches," Journal of Imaging, vol. 11, no. 10, Art. no. 326, 2025.
[9] F. Pedregosa et al., "Scikit-learn: Machine Learning in Python," Journal of Machine Learning Research, vol. 12, pp. 2825-2830, 2011.
[10] A. Ingle, "Crop Recommendation Dataset," Kaggle, 2020. [Online]. Available: https://www.kaggle.com/datasets/atharvaingle/crop-recommendation-dataset



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY