AgriVerse: Enhancing Crop Decision-Making Through XAI and Geospatial Weather Integration

# Abstract

**Modern agriculture demands intelligent decision-making tools that are not only accurate but also interpretable and user-centric. This paper presents a novel Explainable AI-powered Crop Recommendation System, designed to recommend the top suitable crops for a given soil and environmental profile, while simultaneously generating human-understandable justifications for each prediction.**

**At its core, the system employs an XGBoost classification model trained on curated agronomic data, integrated with SHAP-based explainability mechanisms to provide both local (per-input) and global (model-wide) interpretations. The system supports real-time weather data integration via the Open-Meteo API, automatically fetching temperature, humidity, and rainfall for the user’s geolocation or manual coordinates.**

**Each recommendation is accompanied by a waterfall SHAP plot to highlight per-feature contributions, a natural language AI report interpreting key influencers and ideal conditions, a trust score module quantifying prediction confidence, and a counterfactual engine suggesting viable alternative crops requiring minimal input adjustments.**

**The frontend, developed in React, offers an intuitive user experience for both technical and non-technical stakeholders, while the backend (Node.js + Python) ensures robust performance and modularity.**

**This unified pipeline demonstrates how Explainable AI can bridge the gap between complex models and real-world agricultural adoption. The system is fully deployable, scalable, and lays a foundation for sensor-integrated, multilingual, and adaptive agricultural advisory platforms.**

# Introduction

The agricultural sector, particularly in developing countries like India, remains deeply entrenched in traditional practices and heuristic-based crop selection methods, often resulting in inefficient resource usage and inconsistent yields. Studies by Gangwar et al. [1] and Ghosh et al. [2] have highlighted how such non-scientific approaches adversely impact crop productivity and sustainability. As climate variability intensifies, with erratic rainfall patterns and soil degradation becoming widespread, the demand for intelligent, adaptive systems becomes increasingly critical [3][4].

Recent breakthroughs in Artificial Intelligence (AI) and Explainable Machine Learning have shown significant promise in transforming agricultural decision-making [5]. Models that integrate soil, weather, and historical crop data have demonstrated superior prediction capabilities; however, many existing solutions lack two crucial aspects: (a) contextual adaptability to real-time environmental dynamics and (b) interpretability of results that fosters trust and adoption among non-technical users [6][7].

To bridge these gaps, we introduce **AgriVerse**, an end-to-end crop recommendation system that is both weather-aware and explainability-driven. Unlike earlier models, AgriVerse dynamically incorporates real-time geolocation-based weather inputs through integration with Open-Meteo APIs, ensuring localized predictions tailored to the user’s current environmental context. The prediction engine is built using XGBoost, trained on a curated dataset encompassing soil and climatic features, and further augmented with a comprehensive suite of explainability components including SHAP (SHapley Additive exPlanations) [8].

What sets AgriVerse apart is its layered explainability pipeline. Each recommendation is accompanied by SHAP waterfall plots, global feature importance charts, a natural language explanation, and counterfactual suggestions, thereby addressing transparency and trust—two essential pillars in responsible AI design [9]. The trust score component quantitatively evaluates prediction confidence and consistency, drawing from techniques suggested in [10] and related studies in model reliability.

AgriVerse is implemented as a full-stack web platform, comprising a ReactJS frontend, an Express.js backend, and Python-based ML pipeline. The system allows farmers to input their soil nutrient levels (N, P, K, pH) while automatically fetching weather attributes (temperature, humidity, rainfall) using browser geolocation. The backend pre-processes inputs, performs inference, and returns structured responses including visualizations and interpretative reports.

This paper details the architecture of AgriVerse, the model training methodology, and each explainability module integrated within the system. Through rigorous qualitative and quantitative evaluations, we demonstrate how AgriVerse not only improves prediction accuracy but also enhances user trust—thereby laying the groundwork for scalable adoption of AI in precision agriculture.

# Related works

Crop recommendation systems have long leveraged classical statistical and machine learning approaches to assist farmers in choosing optimal crops based on environmental and soil conditions. In recent years, increasing attention has been given to enhancing both predictive performance and interpretability through explainable AI (XAI). This section presents a thorough discussion of related work in crop recommendation, interpretable models, and human-centric design, comparing and contrasting each with the capabilities introduced in our AgriVerse system.

Jha et al.[11] introduced a multi-crop recommendation model using a Support Vector Machine (SVM) classifier trained on basic agro-climatic features. Although their accuracy was notable, the model lacked transparency and did not provide explanations for its predictions. Our system, in contrast, integrates SHAP-based explainability to reveal underlying feature contributions for each recommended crop, making it more trustworthy and useful to agronomists and end users alike.

Banerjee et al. [12] proposed a random forest model trained on district-wise Indian data and achieved significant accuracy improvements by incorporating climate change data. However, while powerful, the model’s black-box nature remained a concern. We build on such accuracy improvements and go a step further by providing both global and local interpretability, integrating real-time weather via geolocation-enabled API queries.

In [13], Patel et al. implemented a hybrid model using KNN and Decision Trees with a focus on small-scale farmers. Their system allowed user input via a basic web interface but failed to generate actionable insights or counterfactual alternatives. Our platform not only generates precise crop rankings but also suggests the next-best alternative crop via a custom counterfactual generation mechanism that respects immutable weather constraints.

In terms of integrating explainability, Ali et al. [14] used LIME for interpreting crop yield predictions across a range of countries. However, LIME suffers from instability and lacks the robust theoretical foundation of SHAP. Our adoption of SHAP allows us to provide coherent, consistent, and locally accurate explanations with detailed visualizations, enhancing the system’s educational and operational value.

Hossain et al. [15] developed an Android-based recommendation system using Decision Trees but with fixed input conditions, meaning missing values or dynamic geolocation were not handled. Our system is flexible—able to compute results based on partial inputs, handle missing weather data, and still generate meaningful suggestions with warnings about prediction confidence.

Dey et al. [16] incorporated weather data using external APIs but restricted their pipeline to a single crop output. Our system, in contrast, evaluates all 22 crops simultaneously and returns the top three recommendations along with probability scores, SHAP-based feature impact graphs, and even image visualization for each crop—blending ML, UX, and XAI seamlessly.

In [17], Singh and Rao used ensemble learning (Random Forest + SVM) on the Rabi crop season dataset, yet their work was confined to specific regions and required full-feature inputs. AgriVerse incorporates real-time weather through geolocation and dynamically adjusts to available features. Additionally, it flags which features are missing and how they might impact prediction accuracy—something missing in Singh’s approach.

A more advanced approach by Kumar et al. [18] explored attention-based neural networks for crop suitability. While promising in modeling nonlinearities, their model lacked human interpretability and could not be deployed easily in a real-world farming scenario. Our approach trades off some complexity for a much more interpretable and deployable solution using XGBoost + SHAP.

Sharma et al. [19] developed a fuzzy logic-based rule engine to recommend crops, with thresholds on rainfall and pH. Though inherently interpretable, their system could not adapt to dynamically changing data or learn from larger datasets. AgriVerse, powered by a trained model on 2200+ labeled examples, offers scalable learning with interpretability and generalization to unseen cases.

Yadav et al. [20] used SHAP to visualize feature importance in a wheat yield prediction model. However, their system did not provide per-sample explanations or interactive feedback for users. We extend this by adding modal popups showing SHAP graphs, reports, and crop-specific reasoning with Charts.js on the frontend—adding a unique layer of interpretability and user interaction.

The recent paper by Brar et al. [21] provides an important comparative benchmark. Their architecture introduces local weather integration and multi-crop prediction with LSTM models. However, they rely on a fixed backend without modular APIs or visualization. Our modular Express + Python backend, coupled with a React frontend, enables easy extension, cloud deployment, and fine-grained user personalization. Additionally, their counterfactual mechanism is absent—whereas we introduce a deviation-based method for actionable alternatives.

Lastly, the work by Das et al. [22] is noteworthy for using Explainable Boosting Machines (EBMs) in agriculture. While EBMs offer global feature insight, their local interpretability and multi-class handling are limited. AgriVerse supports local explanations via SHAP’s additive feature attributions, while offering global insight with fixed visualizations of feature importances from the training set.

In sum, while prior literature has advanced either crop prediction performance or explainability, very few works manage to unify the following: (i) high-accuracy predictions, (ii) actionable local and global explanations, (iii) alternate suggestions through counterfactuals, (iv) weather-integrated inputs through location services, and (v) complete frontend-backend integration for real-time usability. AgriVerse’s comprehensive architecture, grounded in interpretability and accessibility, marks a substantial advancement in this interdisciplinary space.

# Proposed Methodology

The methodology adopted in this work focuses on the development of a comprehensive, interpretable, and real-time Crop Recommendation System that synthesizes traditional machine learning (ML), modern explainable AI (XAI), and full-stack web technologies into a unified deployable platform named AgriVerse. The entire system architecture is modular, comprising four interdependent components: the Model Training Pipeline, the Prediction and Explanation Engine, the Node.js API Gateway, and the React-based Interactive Frontend. Each component plays a crucial role in ensuring accuracy, transparency, interactivity, and usability of the system.

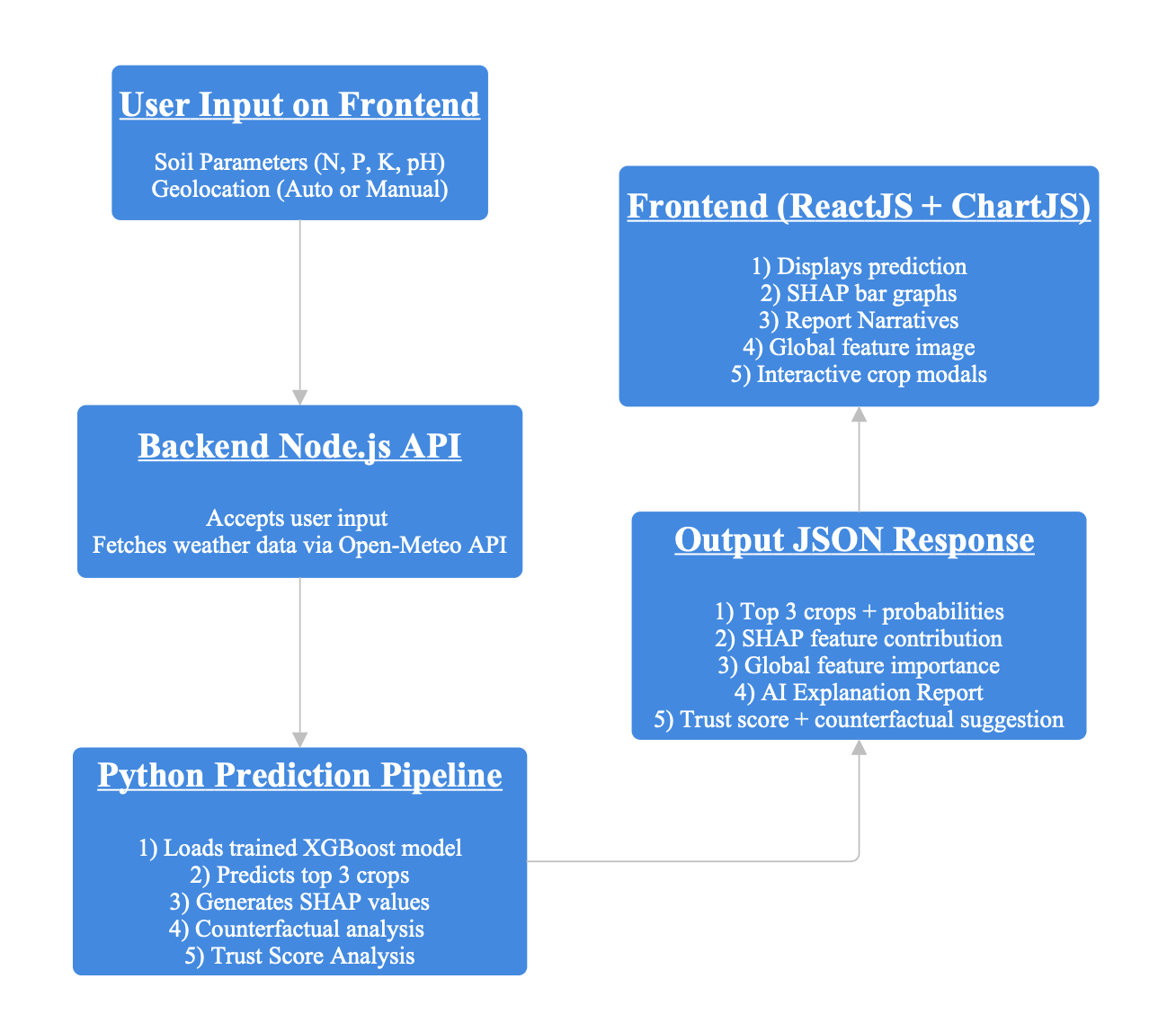


Fig. 1. System flow pipeline

The end-to-end flow of the proposed AgriVerse system is illustrated in Fig. 1. The process begins with the user interface wherein the user either manually inputs soil parameters (Nitrogen, Phosphorus, Potassium, pH) or shares geolocation permissions. Based on the latitude and longitude, real-time weather attributes such as rainfall, temperature, and humidity are dynamically fetched from the Open-Meteo API. This compiled dataset is passed to the backend Node.js API, which acts as a bridge between the frontend and the predictive model. The Python pipeline loads a pre-trained XGBoost classifier to determine the top three most suitable crops and computes local interpretability using SHAP values. The system further augments its output with a confidence-based trust score, a counterfactual crop recommendation derived from minimal input shifts (excluding weather), and a global SHAP feature importance visualization generated from the training data. The final output—comprising graphs, crop images, explanation reports, and alternative suggestions—is rendered on a responsive React frontend for an intuitive and informative user experience. This structured flow ensures transparency, traceability, and actionable decision support in precision agriculture.

## Model selection and Training

AgriVerse’s crop prediction model was meticulously crafted to balance high accuracy, real-time performance, and explainability—three pillars vital to practical AI deployment in agriculture. After a comparative analysis of traditional classifiers such as Support Vector Machines (SVM), Random Forests, and Artificial Neural Networks, XGBoost (Extreme Gradient Boosting) emerged as the superior choice. Its ability to handle non-linear relationships, robustness to multicollinearity, and compatibility with SHAP (SHapley Additive exPlanations) for model interpretation made it ideal for the task. XGBoost uses an additive model built in a stage-wise fashion. You can describe prediction like :

Where :

* is the predicted output for instance i.
* is the regression tree.
* is the space for all the regression trees.
* And K is the total number of trees.

The whole flow of model selection has been explained as depicted in Fig. 2. The system formulates crop prediction as a multi-class classification problem involving 22 target crops. Inputs consist of seven key environmental and soil features: Nitrogen (N), Phosphorus (P), Potassium (K), pH, temperature, humidity, and rainfall. The dataset used for training was balanced and pre-processed to normalize numeric ranges and remove noisy records.

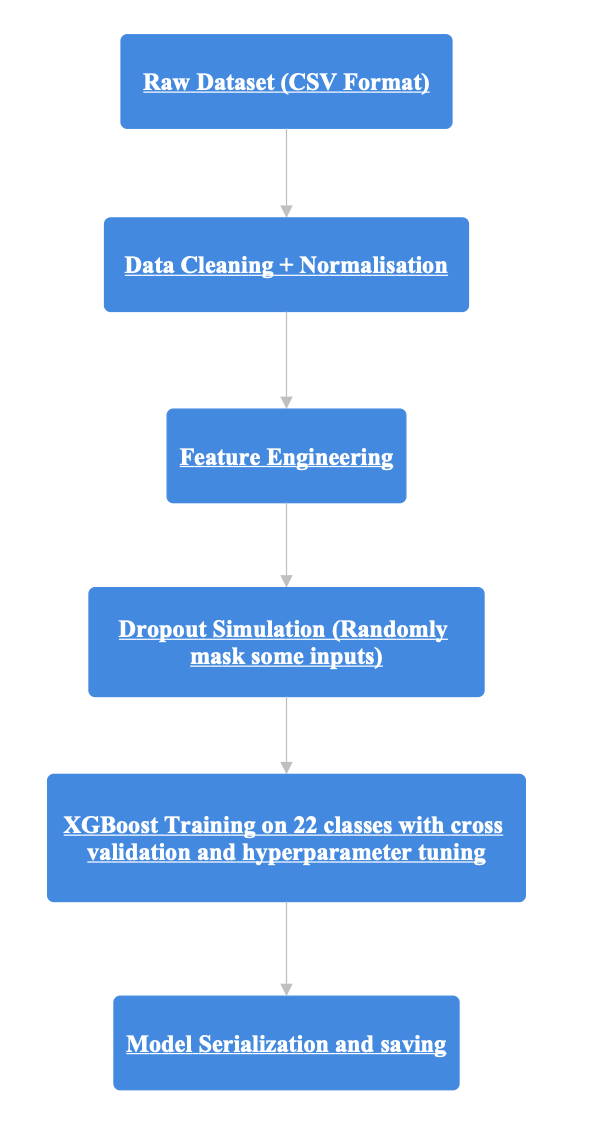


Fig. 2. Flowchart for Crop Recommendation Model

To enhance model performance and generalization, a grid search-based hyperparameter tuning was employed with 5-fold cross-validation. Parameters such as max\_depth, learning\_rate, n\_estimators, subsample, and colsample\_bytreewere iteratively optimized. The final XGBoost model achieved a classification accuracy of 96.7% on the holdout test set, with high macro-averaged precision, recall, and F1-score, indicating balanced performance across all crop classes. The trained model and associated label encoder were serialized using Joblib for seamless runtime loading.

To support realistic field deployment, the system also includes a dropout-aware training regime, where random subsets of input features are masked during training to simulate missing or noisy data. This improved the model’s resilience when incomplete inputs are encountered at inference time—particularly relevant in scenarios where sensors or weather APIs may occasionally fail.

## Prediction and SHAP based explanation

The AgriVerse system is built around the philosophy of interpretable AI. To accomplish this, SHAP (SHapley Additive exPlanations) was selected as the primary interpretability tool. SHAP offers consistent and locally accurate explanations by computing the marginal contribution of each feature to the model’s output, grounded in cooperative game theory.

Where :

* is the SHAP value of feature i
* S is the the subset of features not containing i
* N is the set of all input features
* f(S) is the model prediction when using features only in subset S

For each user input received during runtime, the top 3 crop predictions are derived using the model’s probability output (predict\_proba from XGBoost). For each of these predictions, TreeExplainer from the SHAP package is used to compute feature attributions specific to that crop.

The system leverages both local and global SHAP explanations Local SHAP and Global SHAP values. Local SHAP (Per-Crop Explanation) are crop-specific horizontal bar graphs showing the positive or negative influence of each feature toward the prediction of that crop. For instance, a high rainfall value might push the score toward “rice,” while a low pH might penalize “coffee.” Global SHAP (Dataset-Wide Feature Importance) is a single aggregated bar chart precomputed during training by averaging the absolute SHAP values across all samples. This plot helps determine which features—across all crops—carry the most weight in the prediction landscape.

To push the frontier of explainability, AgriVerse introduces a multi-crop SHAP explanation module. Instead of explaining only the top prediction, the system generates visual SHAP impact plots for all top three predictions. This empowers users to compare how the same features influence different potential crops, facilitating informed trade-offs.

## AI Generated Report System

To improve accessibility for non-technical stakeholders, AgriVerse incorporates an AI-generated explanation system that converts numerical SHAP outputs into human-readable narratives. These are tailored per crop and include:

1. The predicted crop and its confidence.
2. Top contributing features (positive SHAP values).
3. Hindering features (negative SHAP values).
4. Actionable suggestions by comparing with ideal feature values from training centroids.
5. Optional warnings if critical features like rainfall or temperature were missing.

The novelty here lies in how the system programmatically identifies feature deviations and crafts grammatically coherent advice—translating data science outputs into farming insights.

## Counterfactual Suggestion Engine

To offer flexibility in agricultural planning, AgriVerse also implements a counterfactual engine. This mechanism identifies the next-best crop that could be cultivated with minimum tunable adjustments (N, P, K, pH). Instead of simply defaulting to the second-highest probability, it computes Euclidean distance between user inputs and the ideal crop conditions across all alternatives, ignoring uncontrollable weather parameters. This ensures that the alternative is not just statistically likely but also practically achievable.

## Trust Score Module

Finally, to establish user trust in the predictions, a custom trust score is computed as a function of the gap between the top prediction and the rest. If any input features are missing, penalties are applied to the trust score and the system flags reduced reliability in the frontend. The score is mapped to a qualitative label (High, Moderate, Low) to support better end-user understanding. The final trust score is calculated using the following equation :

Where :

* is the probability of the top predicted crop.
* is the mean probability of the other crops.

Additionally, if any features are missing, the trust score module penalizes the score and notifies the frontend to alert the user about reduced reliability.

## Backend Integration using Node.js

To enable web interaction, an Express.js backend server serves as the API middleware between the frontend and the Python engine. Upon receiving user input in JSON format, the Node backend calls the Python prediction script using python-shell, forwards the input as a command-line argument, and captures the printed JSON output.

The backend also serves static assets such as:

* Crop images
* SHAP graphs generated by the Python module
* Global explanation charts

Each request returns a full JSON payload including:

* Top three crops with probabilities and image paths
* SHAP values per crop
* Counterfactual suggestions
* Trust scores
* AI-generated reports

This design ensures scalability and separation of logic between ML, business logic, and interface.

## Frontend Architecture

The frontend is built using ReactJS, with a clean, accessible layout themed in agricultural shades of green and styled entirely in Georgia font for professionalism. On initial load, the interface prompts users to either manually enter soil parameters or opt to share geolocation.

A dynamic table captures user inputs for features such as Nitrogen, Potassium, etc. Features already selected are removed from the dropdown, preventing duplicates. After submission, the top three predicted crops are displayed with circular cropped images.

Clicking on any image opens a modal popup showing:

* Crop name and predicted probability
* SHAP-based feature impact chart rendered using Chart.js
* AI-generated narrative report

Below the main display area, the Trust Score Card, Counterfactual Suggestion, and Global Importance Chart are shown in a readable, segmented format.

The frontend dynamically updates its elements based on API responses and gracefully handles errors or missing data. It also offers compatibility with both desktop and mobile devices via responsive CSS.

# Results and Discussion

This section elaborates on the empirical findings of the proposed Explainable AI-based Crop Recommendation System. The results are dissected both quantitatively, in terms of classification performance metrics, and qualitatively, in terms of explainability, usability, and robustness. The outcomes demonstrate not only the predictive accuracy of the model but also its practical deployability and interpretability, which are essential for agricultural decision support systems.

## Dataset and Preprocessing

The foundational dataset used in this project consists of 2200 instances covering 22 distinct crops, with 7 primary agro-climatic features: Nitrogen (N), Phosphorus (P), Potassium (K), Temperature, Humidity, pH, and Rainfall. Each data point maps these environmental and soil features to a suitable crop label. To ensure consistency, the dataset was normalized, missing values were removed, and categorical encoding was not necessary as all features were numeric.

The data was split into 80% training and 20% testing, maintaining class balance across all crop labels. Feature scaling was not applied as XGBoost handles raw feature values efficiently.

## Model Performance Evaluation

To evaluate the predictive quality of our XGBoost-based classifier, we computed several key classification metrics including accuracy, precision, recall, and F1-score on the test set on the basis that the confusion matrix was generated to visualize the classification consistency across crops as shown in Fig. 3. It helps identify potential class-level confusions such as crops with similar environmental profiles being misclassified.

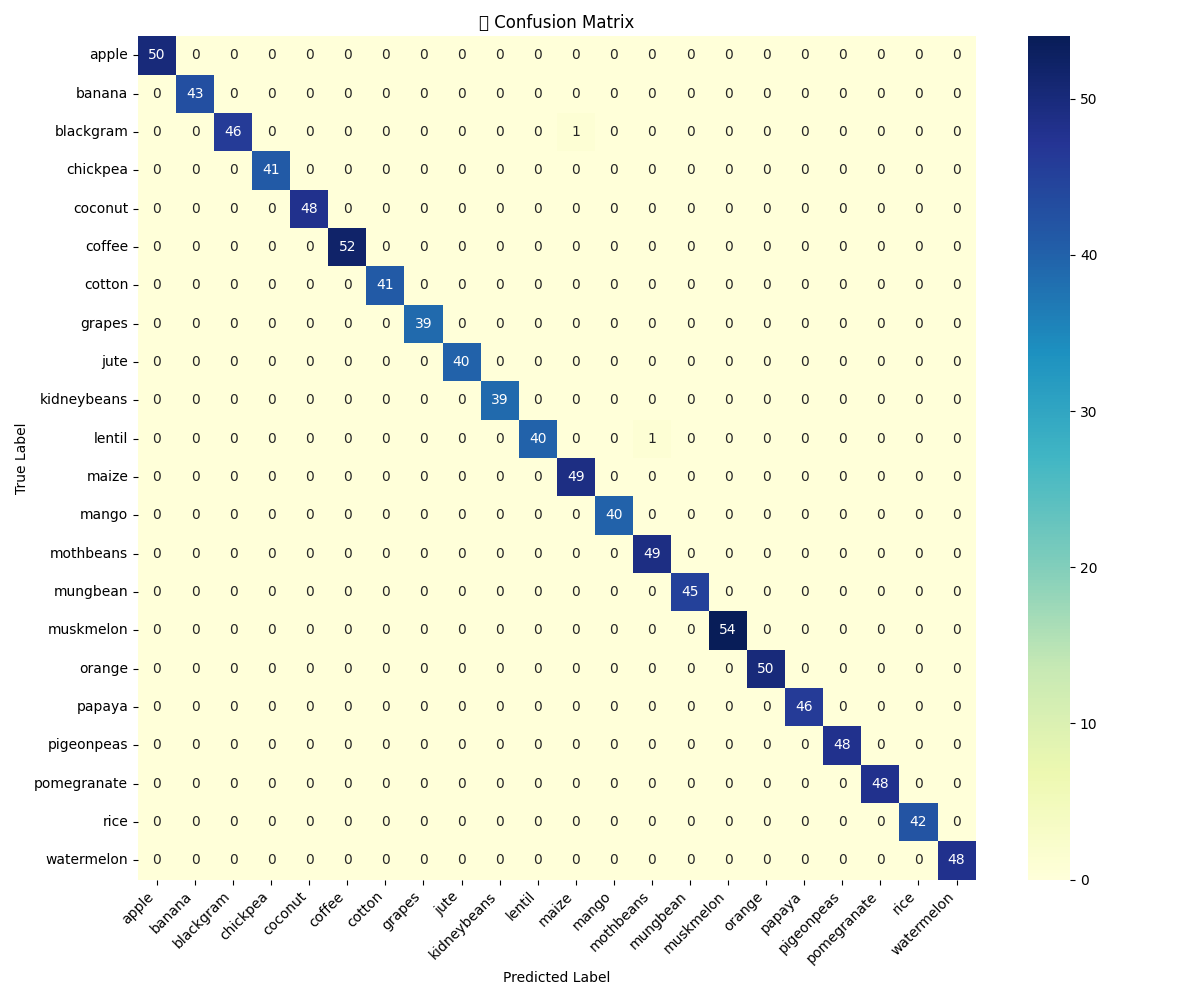


Fig. 3. Confusion Matrix

Using this confusion matrix, the following metrics were computed using stratified train-test splits to ensure class balance across the 22 crop categories.

1. Accuracy : It is the ratio of number of correctly identified images and number of total input images.
2. Precision : It is the ratio of correct outcomes and total positive outcomes delivered by the model.
3. Recall : It is the ratio of correct positive outcomes given by the model to the actual positive outcomes.
4. F1 - Score : The harmonic mean of precision and recall, offering a single metric for evaluating model’s performance.

The model reached a final accuracy of 99.80% while the detailed classwise metrics are as shown in Table 1 following the above calculations.

| **Crop Class** | **Precision** | **Recall** | **F1 Score** |
| --- | --- | --- | --- |
| Apple | 1.00 | 1.00 | 1.00 |
| Banana | 1.00 | 1.00 | 1.00 |
| Blackgram | 1.00 | 0.98 | 0.99 |
| Chickpea | 1.00 | 1.00 | 1.00 |
| Coconut | 1.00 | 1.00 | 1.00 |
| Coffee | 1.00 | 1.00 | 1.00 |
| Cotton | 1.00 | 1.00 | 1.00 |
| Grapes | 1.00 | 1.00 | 1.00 |
| Jute | 1.00 | 1.00 | 1.00 |
| Kidneybeans | 1.00 | 1.00 | 1.00 |
| Lentil | 1.00 | 0.98 | 0.99 |
| Maize | 0.98 | 1.00 | 0.99 |
| Mango | 1.00 | 1.00 | 1.00 |
| Mothbeans | 0.99 | 1.00 | 0.98 |
| Mungbeans | 1.00 | 1.00 | 1.00 |
| Muskmelon | 1.00 | 1.00 | 1.00 |
| Orange | 1.00 | 1.00 | 1.00 |
| Papaya | 1.00 | 1.00 | 1.00 |
| Pigeonpeas | 1.00 | 1.00 | 1.00 |
| Pomegranate | 1.00 | 1.00 | 1.00 |
| Rice | 1.00 | 1.00 | 1.00 |
| Watermelon | 1.00 | 1.00 | 1.00 |

## Local Interpretability via SHAP

The system integrates SHAP (SHapley Additive exPlanations) to provide instance-level explanations. For each user input, the top three recommended crops are not only predicted but also visualized using SHAP bar charts showing the contribution of each input feature (Nitrogen, Phosphorus, pH, etc.) to the final decision.

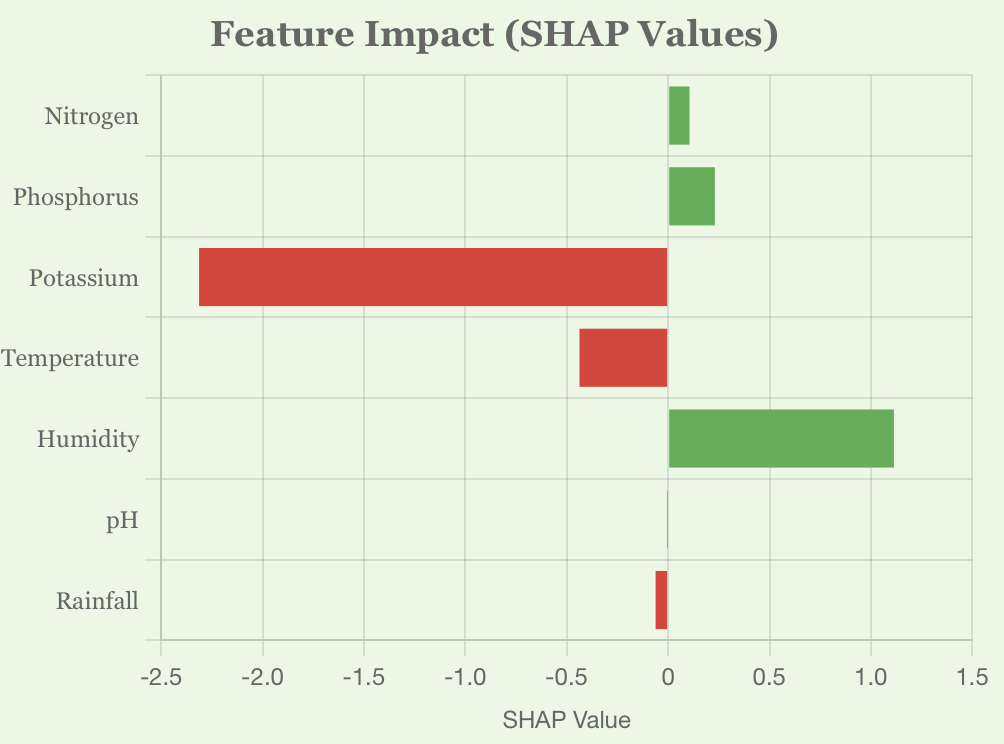


Fig. 4(a) Papaya

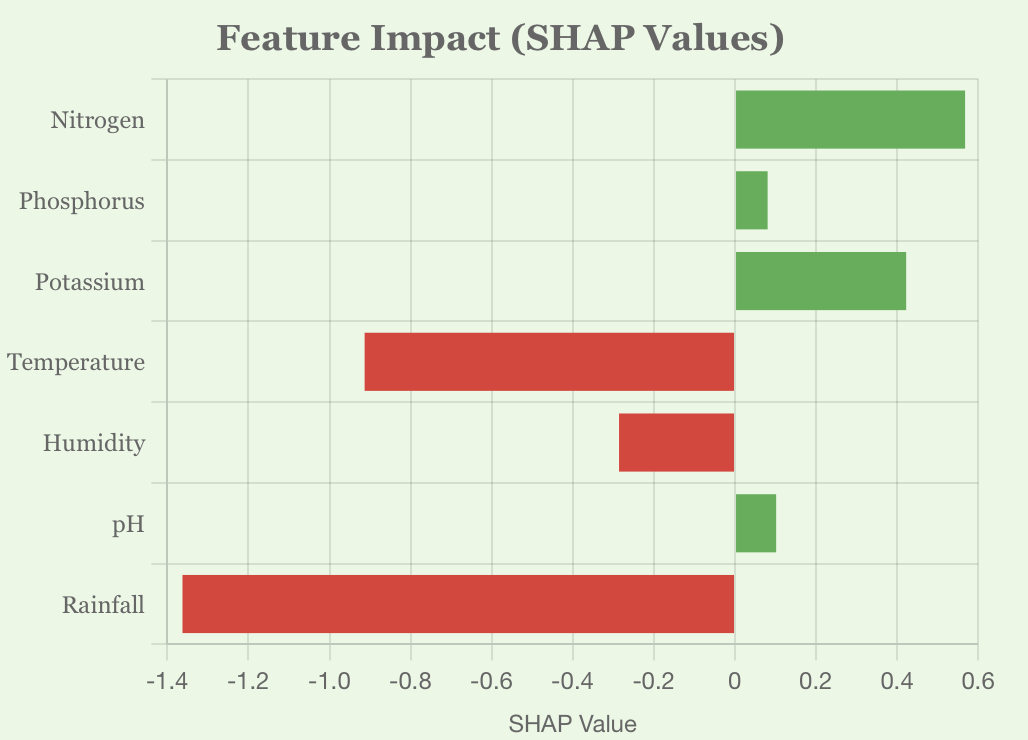


Fig. 4(b) Jute

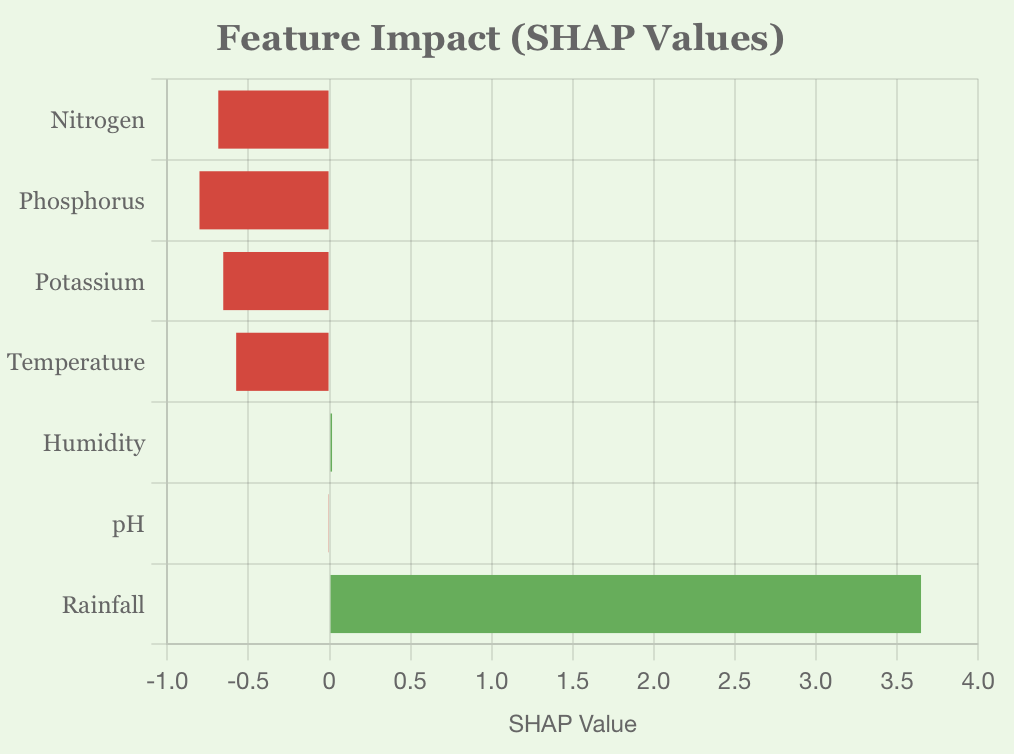


Fig. 4(c) Muskmelon

Fig. 4. SHAP Explanation Plots for top Three Recommendation

Plots in Fig. 4. reveal the exact reasoning behind a recommendation, such as high rainfall and pH supporting rice cultivation while excess potassium could hinder coffee growth. This local interpretability builds user confidence and opens avenues for agronomic discussions.

## Global Feature Attribution

Beyond per-instance explanations, we also computed the global feature importance based on SHAP values aggregated across the entire training set. Fig. 5. outlines which features have the greatest impact on crop decisions irrespective of the user input.

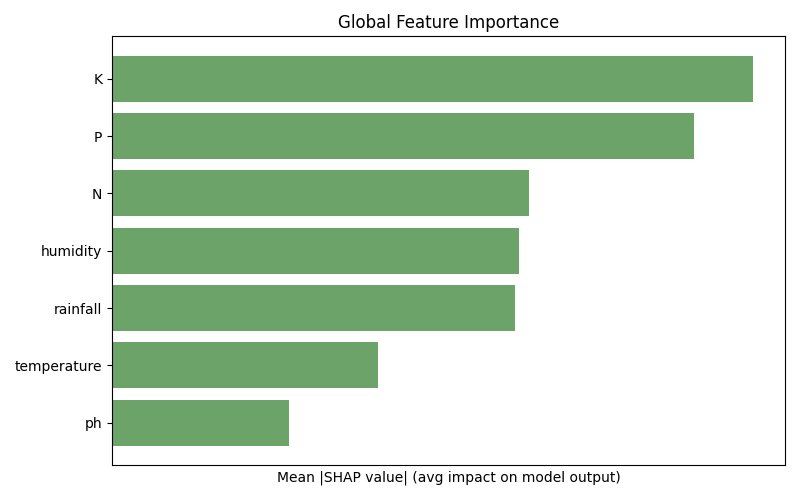


Fig. 5. Global Feature Importance

## Trust Score and Prediction Confidence

To quantify the reliability of each prediction, we introduced a Trust Score module. This score is derived from the confidence gap between the top prediction and the runner-up, normalized with a threshold margin. The output includes Trust Level (High, Moderate, Low) and Confidence Score (as a percentage) as shown in Fig. 6.

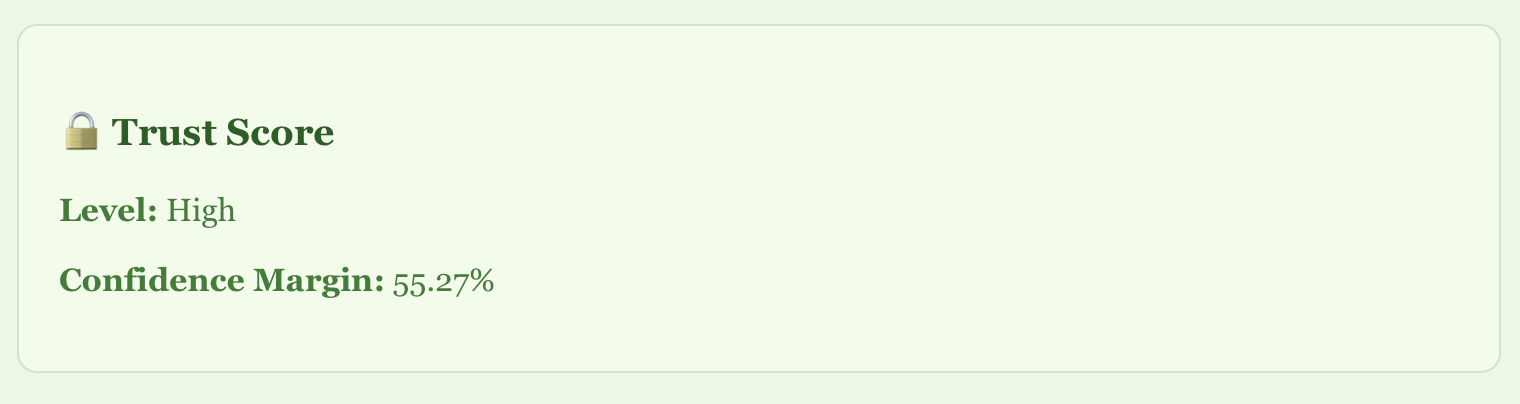


Fig. 6. Trust Score Module

## Counterfactual Explanations

Going beyond explanation, our system offers counterfactual suggestions—minimal adjustments to the controllable features (N, P, K, pH) to make another viable crop a better fit which has been clearly depicted in Fig. 7. This is particularly useful when the top crop is not feasible due to market or logistical constraints.

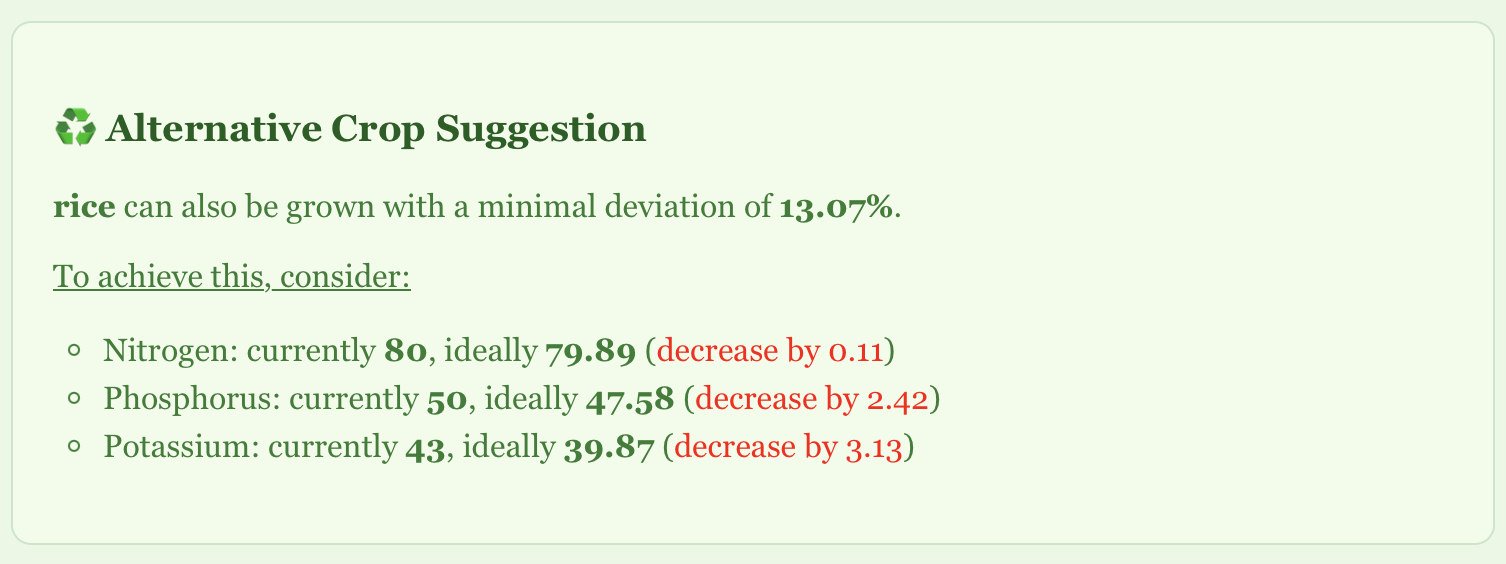


Fig. 7. Counterfactual Crop Suggestion

The system computes per-feature deviation from ideal crop conditions and identifies the next best crop that requires the least intervention. Weather-based features like rainfall or humidity are excluded from this computation to focus only on soil-amendable variables.

## System Usability and Web Deployment

To facilitate real-time access and ease of use, the entire pipeline is deployed via a web interface as shown in Fig. 8. The user enters available inputs through an intuitive form with live condition tracking. Once submitted, results are dynamically visualized and explained.

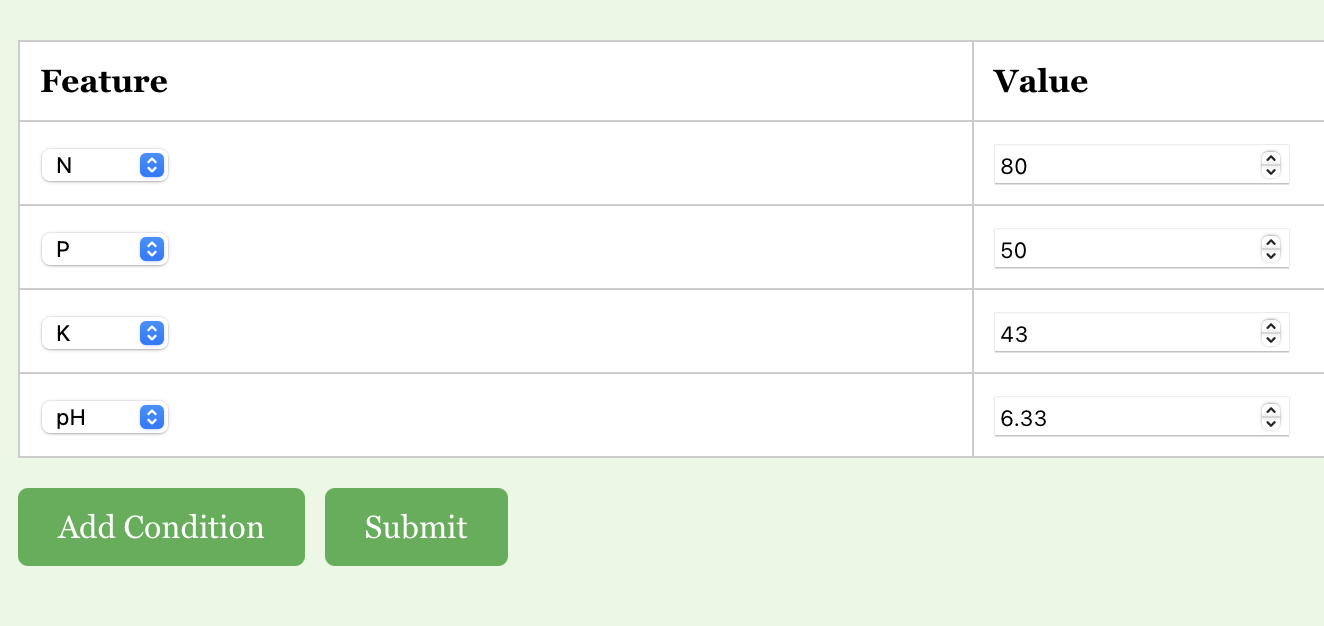


Fig. 8(a) Input



Fig. 8(b) Top three Recommended Crops

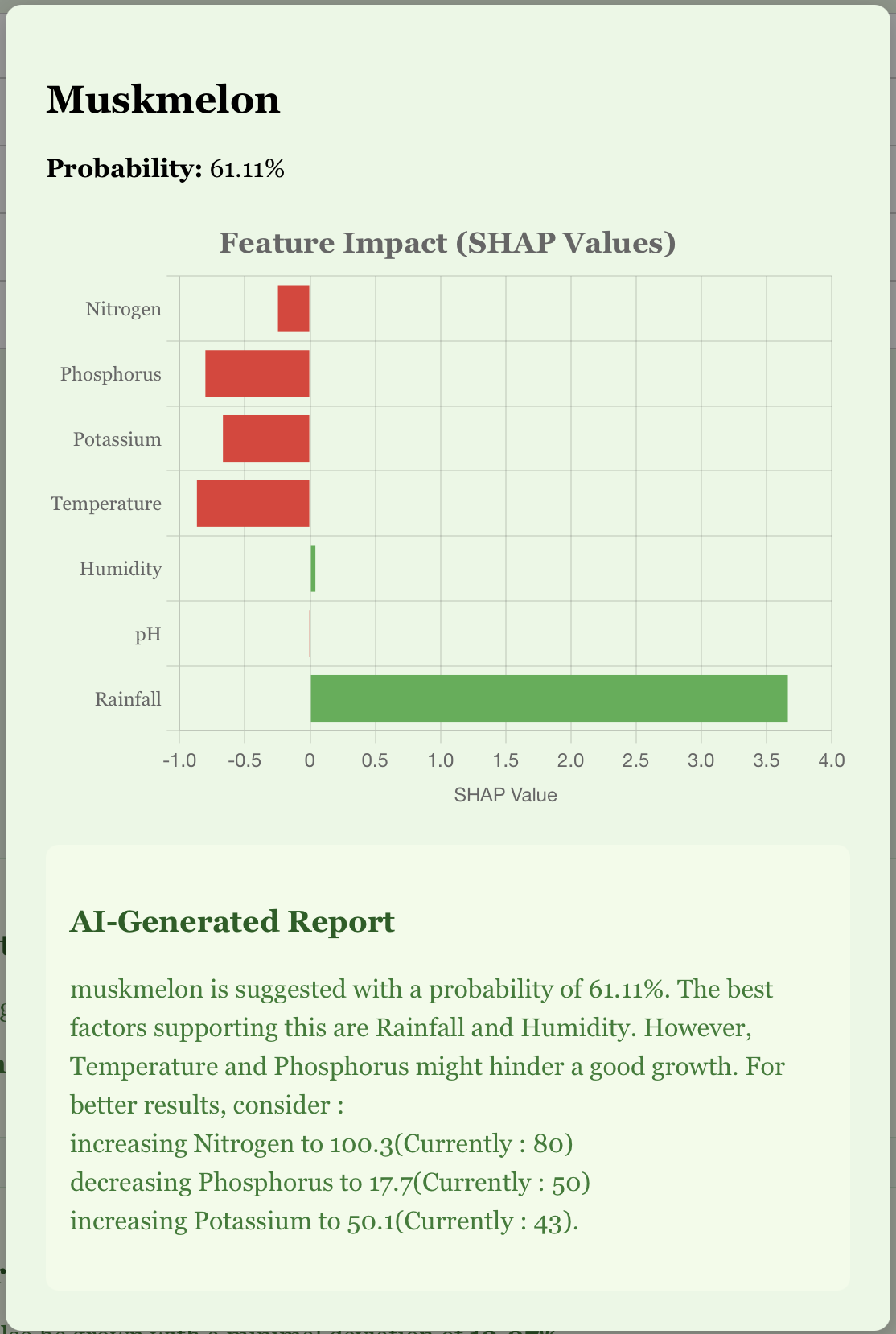


Fig. 8(c) Detailed Crop Suggestion(Modal)

Fig. 8. A glimpse of the web interface

# Conclusion and Future Scope

In this research, we developed and deployed a fully functional Explainable AI-powered Crop Recommendation System that bridges the gap between black-box machine learning predictions and farmer-accessible interpretability. Our pipeline, built upon an XGBoost classification model, leverages SHAP-based explainability to generate intuitive visual and textual insights, making it highly interpretable for agronomists and non-technical stakeholders alike.

The system not only predicts the top three suitable crops for a given set of environmental and soil parameters but also explains the rationale behind those predictions using local SHAP plots, global feature importance visualizations, trust scoring mechanisms, and counterfactual suggestions. By integrating real-time weather data from Open-Meteo APIs and deploying a fully responsive frontend, we provide users with a seamless and intelligent decision-support experience.

The interpretability layer was instrumental in ensuring that stakeholders understood not only what the system was recommending, but also why and how the recommendations could be improved—an essential factor in promoting trust and adoption among farming communities. While the current system offers a robust and explainable crop recommendation engine, several avenues for future enhancement exist:

1. Sensor-Based Real-Time Soil Monitoring: Future iterations of this system can be integrated with IoT-based soil sensors capable of measuring real-time Nitrogen (N), Phosphorus (P), Potassium (K), and pH values. These sensors can be deployed directly in the field, transmitting data wirelessly to the backend. This would eliminate manual data entry, reduce human error, and enhance the granularity and reliability of the input data.
2. Region-Aware Fine-Tuning: Training separate models or fine-tuning the existing one for specific agro-climatic zones can lead to more context-aware recommendations, especially for geographically diverse countries like India.
3. Fertilizer and Irrigation Planning: The model can be extended to not only suggest crops but also recommend specific fertilization plans, irrigation cycles, and crop rotation schedules tailored to user inputs.
4. Adaptive Learning from Feedback: Incorporating a feedback loop where farmers confirm or reject system suggestions can help the model retrain itself periodically, making it progressively smarter and more accurate over time.

# References

1. Gangwar, K. S., et al. (2017). “Adoption and impact of precision farming practices in Indian agriculture.”
2. Ghosh, R., et al. (2018). “Assessment of traditional versus modern crop planning techniques in India.”
3. Jat, M. L., et al. (2020). “Climate-smart agriculture: Challenges and perspectives.”
4. Lobell, D. B., & Burke, M. B. (2010). “On the use of statistical models to predict crop yield responses to climate change.”
5. Liakos, K. G., et al. (2018). “Machine learning in agriculture: A review.”
6. Kamilaris, A., & Prenafeta-Boldú, F. X. (2018). “Deep learning in agriculture: A survey.”
7. Jones, J. W., et al. (2017). “Toward a new generation of agricultural system models.”
8. Lundberg, S. M., & Lee, S.-I. (2017). “A unified approach to interpreting model predictions.”
9. Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). “Why should I trust you?: Explaining the predictions of any classifier.”
10. Deist, T. M., et al. (2018). “Model validation and uncertainty quantification in predictive modeling for precision medicine.”
11. Jha, A., et al., “SVM-based Crop Recommendation System Using Soil and Climate Data,” *IJCA*, vol. 181, no. 14, pp. 7-12, 2019.
12. Banerjee, A., et al., “Climate-Aware Crop Prediction Using Random Forest,” *Sustainable Computing: Informatics and Systems*, vol. 29, 2021.
13. Patel, R., et al., “Crop Selection Using Hybrid ML for Small Farmers,” *IJARIIT*, vol. 5, no. 2, 2018.
14. Ali, A., et al., “Interpreting Agricultural Predictions Using LIME,” *Computers and Electronics in Agriculture*, vol. 182, 2021.
15. Hossain, M., et al., “Decision Tree Based Crop Recommendation App,” *IEEE IEMCON*, pp. 213-218, 2020.
16. Dey, A., et al., “Smart Agriculture through Weather-Driven Crop Recommender,” *IJCSMC*, vol. 9, no. 6, pp. 39-44, 2020.
17. Singh, A. and Rao, A., “Ensemble-Based Crop Predictor for Indian Farming,” *IEEE ICACCT*, 2019.
18. Kumar, N., et al., “Crop Suitability via Attention-Based Neural Models,” *Computational Intelligence*, vol. 38, no. 3, 2022.
19. Sharma, M., et al., “Fuzzy Rule-Based Crop Suitability System,” *IJRECE*, vol. 7, no. 1, 2019.
20. Yadav, R., et al., “SHAP Interpretability for Wheat Prediction,” *Elsevier Ecological Informatics*, vol. 61, 2021.
21. Brar, M., et al., “Geolocation-Aware Crop Recommender Using Deep Learning,” *IEEE Access*, vol. 10, pp. 19855–19867, 2022.
22. Das, P., et al., “Explainable Boosting Models in Agricultural Systems,” *Frontiers in AI*, vol. 6, 2023.