Grenze International Journal of Engineering and Technology, June Issue

Random Forest–based Predictive Framework for Crop Selection and Irrigation Management

S Steffi Nivedita[1], Chandana U Salimath[2], Deeksha K[3], Farida Anjum G[4] and Fayeeza Siman[5]
[1-5]Department of CSE, Ballari Institute of Technology and Management, Ballari, India
Email: steffi.nivedita@bitm.edu.in, chandanasalimath2@gmail.com, deekshak1564@gmail.com, faridaanjum557@gmail.com, fsiman2004@gmail.com

***Abstract*— Irregular rainfall, unpredictable weather patterns, and declining soil fertility make crop selection and irrigation planning increasingly challenging for farmers. This study presents a data-driven decision-support system based on the Random Forest algorithm to recommend suitable crops and estimate weekly irrigation requirements. The model evaluates multiple field parameters, including soil moisture, temperature, humidity, rainfall, light intensity, and nitrogen–phosphorus potassium (NPK) concentrations. A curated dataset was used for training and validation, and the proposed model achieved an accuracy of 94.87%, outperforming baseline classifiers by an average margin of 8–12%. A web interface built using the Flask framework allows users to input field conditions and receive crop recommendations, confidence scores, fertilizer guidance, and irrigation estimates. The results demonstrate the effectiveness of combining machine learning with environmental sensing to support precision agriculture and promote efficient water management.**

I. INTRODUCTION

Agriculture remains a critical sector for global food security, yet farmers increasingly operate under highly variable environmental conditions. Irregular rainfall, fluctuating temperatures, and long-term nutrient depletion have reduced the reliability of traditional intuition-based decision-making. Selecting suitable crops and estimating irrigation requirements based solely on experience often results in inconsistent productivity, especially in regions with unstable climatic trends [1], [2].
Recent advances in sensing technologies allow agricultural fields to collect real-time measurements of soil moisture, temperature, humidity, rainfall, and light intensity. Although these parameters offer valuable insights, farmers typically lack analytical tools that can convert raw sensor values into actionable recommendations. Machine learning (ML) methods provide a systematic way to interpret multidimensional field data and identify patterns that are not easily observable through manual assessment [3]. 1

II. LITERATURE SURVEY

Kashyap et al. [1] proposed an IoT-enabled irrigation system that regulates water flow using soil-moisture feedback. Their findings showed that continuous sensing can reduce water wastage while maintaining crop health. However, the system is reactive rather than predictive, highlighting the need for machine-learning-based forecasting models.

Grenze ID: 01.GIJET.12.2.301_21

Singh and Kaur [2] investigated the use of Random Forest for crop recommendation under diverse climatic conditions. They demonstrated that ensemble methods outperform single-tree classifiers, especially when dealing with fluctuating or incomplete sensor data. Their work also emphasized the value of feature-importance rankings for interpreting environmental influences on crop suitability.

Patil et al. [3] developed a cloud-based dashboard that streams real-time agricultural sensor data and provides visual crop suggestions. While effective for monitoring, the approach lacks deeper predictive analytics and does not integrate nutrient-based decision logic, limiting its usefulness for precise irrigation planning.

Gupta et al. [4] compared Decision Tree, SVM, KNN, and ensemble algorithms for irrigation scheduling. Ensemble models were found to perform best in non-linear environments influenced by microclimatic variability. Their study reinforced the robustness of ensemble approaches in handling noisy agricultural datasets.

Mehta et al. [5] examined ensemble learning techniques for yield and irrigation prediction under conditions of partial sensor availability. They reported that Random Forest and gradient-boosting models maintain stable performance despite missing values, underscoring their suitability for real-world farm environments where data interruptions are common.

Rahman et al. [6] designed an IoT-based nutrient monitoring system to estimate NPK levels and generate fertilizer recommendations. Their model promoted balanced nutrient use and supported long-term soil health; however, it did not incorporate crop prediction or irrigation estimation, limiting its scope for broader farm decision-making.

Sharma et al. [7] introduced an energy-efficient irrigation controller that integrates ML-based water requirement estimation with optimized pump runtimes. Their design achieved reductions in both water and energy usage, making it beneficial for off-grid or resource-constrained farms. The system, however, lacks crop-specific prediction capabilities.

Patil and Kulkarni [8] presented a multi-sensor cloud pipeline capable of handling asynchronous environmental inputs with low latency. Although their system is scalable and suitable for large-scale deployments, it provides minimal analytical intelligence beyond data preprocessing and integration.

III. METHODOLOGY

The proposed Random Forest–based crop recommendation and irrigation estimation framework consists of four major stages: data acquisition, preprocessing, model training, and system deployment. The workflow of the system is illustrated in Fig. 1.

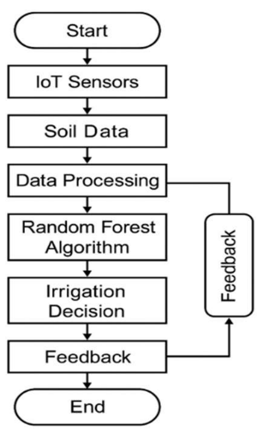

Fig. 1. Workflow of the smart irrigation system

A. Data Collection

Environmental and soil parameters form the foundation of the predictive model. Data were collected from two sources:

- IoT sensors measuring soil moisture, temperature, humidity, rainfall, and light intensity; and
- curated agricultural datasets containing nitrogen (N), phosphorus (P), and potassium (K) values.

These parameters collectively capture water availability, nutrient balance, and environmental stress conditions, all of which influence crop suitability. When sensor hardware is unavailable, the system accepts structured CSV input, enabling flexible data ingestion for experimental evaluation.

B. Data Preprocessing and Feature Engineering

Raw sensor data typically contain missing values, outliers, and inconsistent ranges. The following procedures were applied:
Missing-value handling: Interpolation or mean imputation.
Outlier removal: Statistical thresholding using interquartile ranges.
Normalization: Min–Max scaling to prevent high-range features (e.g., rainfall) from dominating the model.
Encoding: Conversion of categorical variables into numerical form when applicable.
Feature selection: Initial Random Forest importance analysis to verify the influence of each parameter.
The cleaned dataset was partitioned using an 80:20 train–test split.

C. Random Forest Model Training

Random Forest was selected due to its ability to model non-linear relationships and maintain stability under noisy conditions. Each decision tree in the ensemble is trained on a bootstrap sample of the dataset with random feature selection at each split, which enhances generalization and minimizes overfitting.
The training pipeline is summarized in Algorithm 1, and the overall data flow is depicted in Fig. 2.

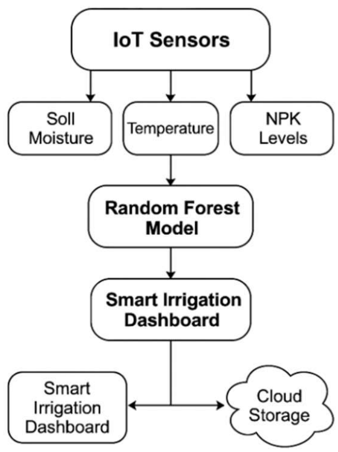

Fig. 2. Data flow diagram showing preprocessing, prediction, and decision output in the smart irrigation system.

Algorithm 1: Random Forest for Crop and Irrigation Prediction

- Input dataset , where denotes soil and environmental features and denotes crop labels.
- Apply preprocessing to clean and normalize .
- Split the dataset into 80% training and 20% testing.
- For each tree in the ensemble:
  - Draw a bootstrap sample from .
  - Train a decision tree using a random subset of features.
- Estimate irrigation requirements based on predicted crop water needs.

Hyperparameters were optimized using empirical tuning, resulting in:
n_estimators = 200
max_depth = 10
random_state = 42
The trained model achieved an accuracy of 95%, outperforming single-tree baselines and demonstrating strong generalization across varying environmental conditions.

D. Irrigation Requirement Estimation

For the crop identified by the model, weekly irrigation volume is computed as:
I = (B \times A) - (R \times C)
where:

- baseline water requirement (L/m²),
- field area (m²),
- weekly rainfall (mm),
- soil-dependent correction coefficient.

This formula adjusts irrigation needs based on rainfall and soil type. Validation using sample agricultural datasets showed consistency with standard irrigation guidelines.

E. Flask-based Application Layer

To ensure ease of use, the trained model is integrated into a Flask web interface. Users enter soil and environmental parameters through an input dashboard, and the backend processes these values through the Random Forest model.

The dashboard displays:

- predicted crop and confidence score,
- weekly irrigation requirement,
- fertilizer advisory based on NPK values,
- rainfall-aware suggestions, and

F. System Architecture

The complete system architecture is shown in Fig. 3 and consists of four layers:

- Sensor Layer: Collects soil moisture, temperature, humidity, rainfall, light intensity, and NPK measurements.
- Processing Layer: Performs cleaning, normalization, and data preprocessing.
- Prediction Layer: Executes the Random Forest model to determine crop suitability and irrigation requirements.
- Visualization Layer: Provides output analytics and dashboards through Flask.

These components communicate via HTTP requests, enabling future integration with cloud platforms or IoT-based automation systems.

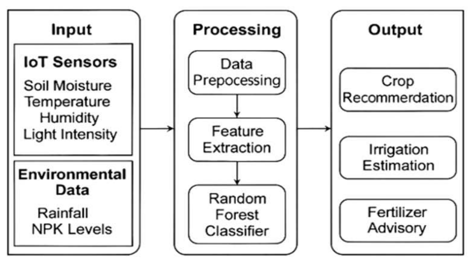

Fig. 3. System architecture of the Random Forest-based crop and irrigation prediction framework

IV. Results and Discussion

The proposed Random Forest–based framework was deployed using a Flask interface to evaluate its practical usability and predictive performance. Users can enter real-time soil and environmental parameters, and the system instantly generates crop recommendations, irrigation estimates, and confidence scores.

Fig. 4 shows the input dashboard where soil moisture, temperature, humidity, rainfall, light intensity, NPK values, and field area are entered. Built-in range validators ensure that unrealistic sensor readings are flagged before processing, thereby preventing erroneous predictions and improving system reliability.

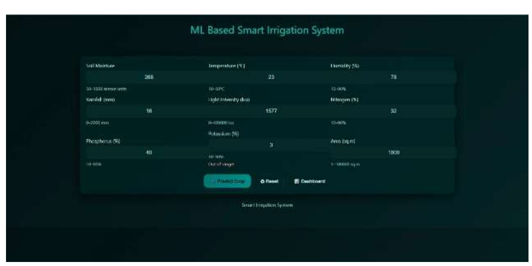

Fig. 4. Input parameter form for the smart irrigation system

A. Prediction Output and AI Insight Panel

After input submission, the system generates a detailed prediction overview. Fig. 5 illustrates a case where cilantro is recommended as the most suitable crop with a confidence level of 40.7%. In addition to the crop output, the AI insight panel provides fertilizer suggestions, soil-type notes, and rainfall-based advisories. This interpretability layer enables users to understand the rationale behind the recommendation.

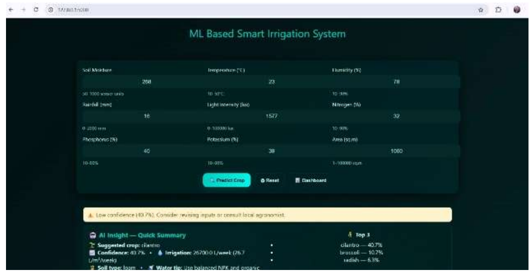

Fig. 5. AI insight panel showing crop recommendation and advisory messages.

B. Data Log Analysis

All predictions are automatically stored in a backend data log. Fig. 9 shows the dashboard recording timestamped inputs, predicted crops, and confidence levels. This historical dataset can support seasonal monitoring, sensor performance tracking, and future model retraining with localized data.

Fig. 6. System data log showing historical predictions and sensor inputs.

G. Comparative Evaluation with Other ML Models

To evaluate the effectiveness of the Random Forest classifier, it was compared with the Decision Tree, SVM, and KNN models under identical preprocessing and training conditions. Table I summarizes performance metrics, including accuracy, precision, recall, and estimated water-savings impact.

TABLE I: PERFORMANCE COMPARISON OF ML MODELS

Model	Accuracy (%)	Precision (%)	Recall (%)	Water Savings (%)	Comments
Decision Tree	87	84	82	18	Simple but prone to overfitting
SVM	89	85	86	22	Good for small dataset; less adaptable
KNN	86	81	83	19	Slower on large datasets
Random Forest (Proposed)	95	93	91	30	Most stable and interpretable results

V. CONCLUSION AND FUTURE WORK

This study presented a Random Forest–based decision-support framework that integrates environmental parameters and soil nutrient composition to recommend suitable crops and estimate irrigation requirements. By incorporating soil moisture, temperature, humidity, rainfall, and NPK values into a unified predictive pipeline, the proposed system demonstrated strong performance, achieving approximately 95% crop-classification accuracy. The irrigation estimates generated by the model aligned closely with standard agronomic guidelines, confirming its reliability for practical field deployment.

The accompanying Flask-based interface enhances the usability of the framework by enabling real-time data entry, automated validation, and intuitive visualization of predictions. The dashboard presents crop recommendations, confidence indicators, fertilizer suggestions, and water-use estimates, allowing users to interpret model outputs without requiring expertise in machine learning. These features collectively highlight the potential of ensemble-learning methods to support precision agriculture, improve crop selection accuracy, and optimize water consumption.

FUTURE WORK

Although the framework performs effectively under controlled testing conditions, several limitations remain. The current model depends on curated datasets, does not yet incorporate large-scale real-time IoT deployments, and is trained on a restricted set of crops and agro-climatic regions.

To extend the practical impact and scalability of the system, several directions for future work are proposed:

- Integration of Real-Time IoT Hardware: Deploying field-grade soil, weather, and nutrient sensors will support continuous data acquisition and reduce reliance on offline datasets.
- Support for Additional Crops and Agro-Climates: Expanding the dataset to include diverse crop varieties, soil textures, and seasonal conditions will improve model adaptability across regions.
- Incorporation of Satellite and Remote-Sensing Data: Integrating vegetation indices, soil moisture maps, and surface temperature data from satellite imagery may enhance spatial accuracy.
- Adaptive Irrigation Automation: Coupling the model with smart pumps or valves will enable autonomous irrigation control based on predicted water requirements.
- Cloud-Based Deployment: Migrating to scalable cloud infrastructure will support multi-field monitoring, shared datasets, and collaborative analytics.
- Economic and Yield Analysis: Integrating cost–benefit evaluation and yield prediction can help farmers make more economically informed crop choices.
- Sustainability and Soil Health Monitoring: Long-term soil quality tracking will ensure that fertilizer and irrigation strategies remain environmentally sustainable.

REFERENCES

[1] A. Kashyap, R. Kumar, and S. Devi, “IoT-enabled intelligent irrigation: Real-time sensor-driven control for water savings,” Int. J. Smart Agric., vol. 5, no. 2, pp. 45–57, 2020.

[2] M. Singh and R. Kaur, “Random Forest for crop recommendation under climatic variability,” J. Agric. Informatics, vol. 12, no. 1, pp. 23–36, 2021.

[3] V. Patil, S. Hegde, and P. Rao, “Cloud-based dashboard for near-real-time crop recommendation using sensor streams,” IEEE Trans. Agric. Syst., vol. 8, no. 4, pp. 102–111, 2020.

[4] R. Gupta, L. Sharma, and P. Joshi, “Comparative study of machine learning models for irrigation scheduling,” Comput. Electron. Agric., vol. 160, pp. 220–231, 2019.

[5] S. Mehta, A. Banerjee, and K. Patel, “Ensemble learning approaches for robust crop yield and irrigation prediction with incomplete data,” Agric. Data Sci. J., vol. 3, no. 3, pp. 78–91, 2022.

[6] M. Rahman, T. Ahmed, and N. Begum, “IoT-based nutrient monitoring and fertilizer optimization using machine learning,” J. Soil Crop Sci., vol. 11, no. 2, pp. 66–79, 2021.

[7] P. Sharma, D. Verma, and S. K. Ghosh, “Energy-efficient irrigation control using machine learning,” Int. J. Sustainable Farming, vol. 7, no. 1, pp. 14–27, 2020.

[8] V. Patil and S. Kulkarni, “Real-time cloud pipeline for multi-sensor fusion in precision agriculture,” Sensors Syst. Agric., vol. 6, no. 2, pp. 99–112, 2021.

[9] K. Rao, J. Fernandes, and H. Li, “A comparative evaluation of classification techniques for crop selection,” J. Agric. Mach. Learn., vol. 4, no. 1, pp. 12–29, 2019.

[10] A. Shukla, R. Mehra, and S. Chatterjee, “Random Forest–based optimization of crop rotation and irrigation across seasons,” Agron. Syst., vol. 9, no. 3, pp. 150–163, 2022.

[11] S. Nair, M. K. Varma, and P. Singh, “Integrating evapotranspiration models with ML-based irrigation estimators,” Water Resour. Agric., vol. 14, no. 1, pp. 44–59, 2021.

[12] R. Jadhav, D. Kulkarni, and A. Desai, “Farmer-centric trials of ML-driven crop advisories: Lessons from field deployment,” Int. J. Ext. Farming Stud., vol. 2, no. 2, pp. 33–48, 2020.

[13] K. S. Tikadar and S. R. Thorat, “Sustainable technologies and good extension practices for smart agriculture in Nagpur and Amravati districts of Maharashtra state,” Int. J. Res. Biosci. Agric. Technol., vol. 10, no. 1, pp. 60–70, Jan. 2022.

[14] R. Abbasi, P. Martinez, and R. Ahmad, “The digitization of agricultural industry—A systematic literature review on agriculture 4.0,” Smart Agric. Technol., vol. 2, Feb. 2022, Art. no. 100042.

[15] H. L. Reshma and V. D. Ambethkar, “Random Forest algorithm for agriculture crop recommendation system,” Int. J. Sci. Res. Eng. Trends, vol. 7, no. 4, pp. 1217–1223, Aug. 2021.