

# Machine Learning-Based Agricultural System for Precision Crop and Fertilizer Recommendation

Rajesh Kumar Pathak  
Department of Artificial Intelligence  
& Machine Learning (AIML)  
Galgotias College of Engineering  
and Technology  
Greater Noida, Uttar Pradesh, India

Gunjan Kumar  
Department of Artificial Intelligence  
& Machine Learning (AIML)  
Galgotias College of Engineering  
and Technology  
Greater Noida, Uttar Pradesh, India

Augustya Nandan Singh  
Department of Artificial Intelligence  
& Machine Learning (AIML)  
Galgotias College of Engineering  
and Technology  
Greater Noida, Uttar Pradesh, India

Aryan Pandey  
Department of Artificial Intelligence &  
Machine Learning (AIML) Galgotias College of  
Engineering and Technology  
Greater Noida, Uttar Pradesh, India

Krishna Sangal  
Department of Artificial Intelligence & Machine Learning  
(AIML)  
Galgotias College of Engineering and Technology  
Greater Noida, Uttar Pradesh, India

**Abstract - GDP and provides employment opportunities to more than half of its workers, most small-scale farmers continue to depend on their instincts and age-old practices in selecting crop types and application of fertilizers, resulting in reduced production, degradation of soil, and loss of money. This study proposes a two-stage framework based on machine learning algorithms to recommend crops and predict fertilizers. The first stage comprises a crop recommendation module utilizing a K-Nearest Neighbor classifier with a distance weightage using 2,200 well-balanced samples covering 22 classes of crops and producing a test set accuracy of 97.5%, while cross-validation accuracy is  $97.1\% \pm 0.6\%$ . On the other hand, the second stage includes a fertilizer prediction module using a Random Forest classifier in conjunction with stratified 5-fold cross-validation and yielding an accuracy of 95.8% for seven fertilizer classes. Real-time weather conditions have been taken into account in making suggestions with the help of OpenWeatherMap API with an efficiency of 96%. With a total score of 81.5 out of 100 on the System Usability Scale, the web application has proven to be highly user-friendly despite being created for non-experts.**

## INTRODUCTION

Agricultural sector accounts for approximately 17-18% of GDP of India and more than 54% of its population are employed in farming activities. Yet small-scale farmers lack the transition to scientific-based crop selection and fertilizer application dose calculation from the reliance on instinct-driven decision-making processes. At best, this leads to poor crop yield, nutrient depletion in soil, and economic losses for underprivileged households dependent on agriculture as the main source of income.

While the body of literature on this topic grows larger, existing studies often recommend crops and fertilizers separately or do not account for the latest weather information and user friendliness for illiterate people. The paper addresses all those issues through developing a two-module machine learning website which is efficient, informative, and easy to use.

Contributions:

- Designing a distance-weighted KNN classifier with 97.5% classification accuracy on the crop recommendation problem with 22 classes surpassing both the SVM model (96.1%) and Decision Tree classifier (93.6%).
- Designing a Random Forest model with 95.8% accuracy on 7 classes fertilizer prediction task.
- Using current weather data from OpenWeatherMap API for temperature and humidity to enhance recommendation accuracy by around 10%
- Developing a web interface with Flask Library evaluated at the SUS score of 81.5/100.

## RELATED WORK

### A. Traditional Rule- Based and Fuzzy Systems

A rule-based expert system for predicting suitable crops based on soil pH, organic matter, and irrigation facilities as IF-THEN conditions was proposed by Nithyanandam et al. [1] for Tamil Nadu region. Although the approach codified domain expertise into decision rules, it did not generalize beyond a set of pre-defined conditions. On the other hand, Ramesh and Vardhan [2] developed a fuzzy

logic-based system that was able to deal with uncertainty associated with agriculture parameters by calculating their degree of membership. However, despite being an improvement over traditional rule-based systems, developing fuzzy membership function proved to be a cumbersome task.

## B. Machine Learning Techniques

The authors Rajak et al. [3] suggested the use of Decision Tree classifier for crop suggestion for six Indian crops, which resulted in accuracy of 85%. In another study on 12 Indian crops using Naïve Bayes, SVM and KNN, Savla et al. [4] found that the latter resulted in the best performance of 89.3%, which was because of its non-parametric characteristics enabling it to consider local data. Doshi et al. [5] used *Random Forest* for classification of 15 crops in Gujarat, attaining an accuracy of 92%, yet showcasing robustness as an ensemble method due to its ability to handle outliers in agricultural data. Majority-voting ensemble of Naive Bayes, Decision Tree, and linear regression was used by Pudumalar et al. [6] for classification of 12 crops in Southern India, yielding 98% accuracy during cross-validation despite having an extremely homogeneous dataset.

## C. Deep Learning Models

LSTM models have been used for agricultural time series analysis and yield prediction in Chlingaryan et al. [7]. Applications of deep learning in agriculture were reviewed by Kamilaris and Prenafeta-Boldú [8], encompassing classification, object detection, and regression techniques. Deep learning, though superior to other approaches for handling large datasets with rich annotations, involves considerably larger computation

## D. Fertilizers Prediction

Singh et al. [9] used the regression tree approach to link soil analysis and yield trends of wheat to reduce fertilization costs by 18% in Punjab without lowering the yield. Majumdar et al. [10] used the Random Forest model to predict fertilization requirements for rice in West Bengal with 12% more precise prediction compared to the conventional linear regression by incorporating interactions among soil features. Bhosale and Tikhe [11] compared KNN, SVM, Naïve Bayes, and Random Forests models in predicting a combination of seven fertilizers with RF giving 94.7% accuracy.

## E. IoT-Based Agriculture

Gondchawar and Kawitkar [12] designed a smart irrigation system based on IoT using moisture content

and soil temperature sensors interfaced using Arduino. Patil and Kumar [13] designed a comprehensive IoT-based precision farming solution with sensor integration and ML prediction based in cloud. Talaviya et al. [14] discussed the applicability of IoT in agriculture concerning irrigation management, pest detection, and crop.

## F. Research Gaps

Through a systematic review of the current literature, three common gaps have been identified:

(1) while some systems focus on crop recommendation or fertilizer prediction alone, there are very few that consider a combination of both

(2) real-time weather API integration is very uncommon, with most systems considering only average conditions

(3) usability for laymen and rural inhabitants is not taken into account by most systems.

## PROBLEM FORMULATION

### A. Problem Statement

The research focuses on two problems related to decision support in agriculture: (a) the identification of the most suitable crop from 22 available options using specific soil and environmental conditions; and (b) the selection of the most suitable fertilizer among seven options, given the soil nutrients and crop information. The decision support system should provide an accuracy level higher than 90%, function in real-time mode in the web application, reduce the effort required by users, and ensure that all output is understandable for non-experts.

### B. Input Feature Space

For the crop recommendation problem, seven attributes should be considered: nitrogen concentration in kg/ha (N), phosphorus concentration in kg/ha (P), potassium concentration in kg/ha (K), soil pH, ambient temperature in Celsius degrees (T), relative humidity percentage (R), and annual precipitation in millimeters (P). For the fertilizer recommendation problem, two extra categorical features will be used: soil type (five possible values: Sandy, Loamy, Black, Red, and Clayey) and crop type (integer).

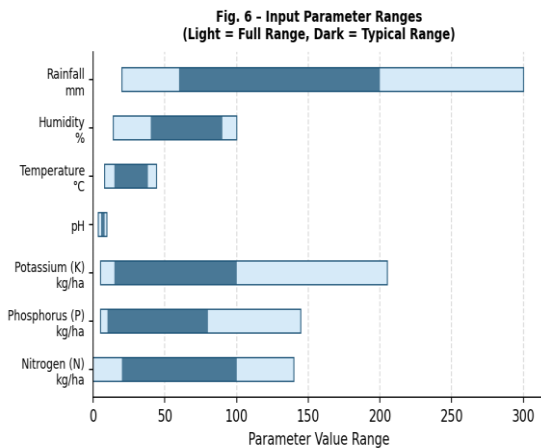


Fig. 6 – Input Parameter Ranges (Light = Full Range, Dark = Typical Range)

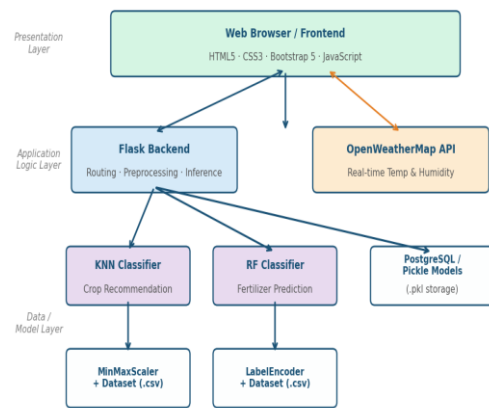


Fig. 3 - Proposed Three-Layer System Architecture

Fig. 3 – Proposed Three-Layer System Architecture

### C. Output Classes

The 22 output classes of crop include cereal crops (rice, wheat, maize), pulse crops (chickpea, lentil, mungbean, blackgram, kidneybeans, pigeonpeas, mothbeans), fruit crops (banana, mango, grapes, apple, papaya, watermelon, muskmelon, pomegranate, orange, coconut), and cash crops (cotton, jute). The 7 output classes of fertilizer include: Urea, DAP, 14-35-14 (NPK), 28-28 (NP), 17-17-17 (balanced NPK), 20-20 (NP), and 10-26-26 (PK).

### D. Mathematical Formulation

In KNN Classification, for a query vector  $x_q$ , Euclidean distance between query point  $x_q$  and data point  $x_i$  is calculated as  $d(x_q, x_i) = \sqrt{\sum (x_{qj} - x_{ij})^2}$ . Here, the predicted class label can be obtained using equation  $C_{pred} = \text{argmax}_c \sum_i I_K w_i \times I(y_i = c)$ , where  $w_i = 1/d(x_q, x_i)$ . For Random Forest,  $T$  decision trees are generated using bootstrap samples with  $m = \sqrt{p}$  as number of features in every split. For prediction, majority voting is employed as follows,  $C_{RF}(x) = \text{argmax}_c \sum_{t=1}^T I(ht(x) = c)$ , where  $ht(x)$  refers to tree  $t$ 's prediction.

## PROPOSED SYSTEM

### A. System Architecture

The proposed system implements a 3-tier client-server architecture including: (1) Presentation Layer – a responsive web interface developed in HTML5, CSS3, Bootstrap 5, and vanilla JavaScript used for forms and API calling; (2) Application Logic Layer – an application server written in Python using Flask as the framework for routing, validation, preprocessing, and ML model inference; and (3) Data/Model Layer – machine learning models saved in serialized .pkl format, training data stored as CSV files, and cached weather API responses.

The end-user interacts with the application through the web interface, optionally activating weather auto-fill with temperature and humidity from the OpenWeatherMap API using city names and geographical coordinates. The validated inputs are preprocessed and sent to the corresponding ML models and their output is shown together with probabilistic confidence and agricultural context via Jinja2 templates.

### B. Crop Recommendation Module – KNN

The KNN algorithm was chosen for crop recommendation due to the natural clustering of crops with similar environmental conditions in the 7-dimensional space, which renders distance metrics suitable for classification. The weighted averaging ( $w_i = 1/d$ ) makes the closer neighbors' contribution more significant, while probability estimates can be obtained as a ratio of votes.

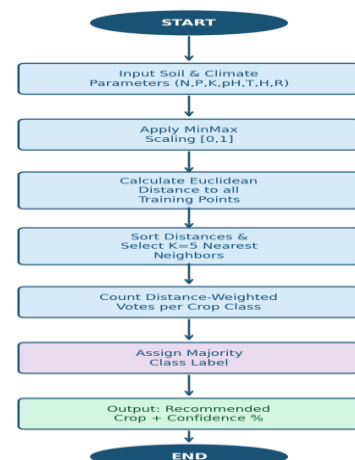


Fig. 4 - KNN Crop Recommendation Workflow

Fig. 4 – KNN Crop Recommendation Workflow

Inputs are scaled between [0,1] using MinMax scaling as  $x_{scaled} = (x - x_{min}) / (x_{max} - x_{min})$  because KNN algorithm is sensitive to differences in magnitude between features. The best parameters for KNN (K=5, weight='distance', metric='euclidean') were found through 5-fold GridSearchCV by testing K in {1,3,5,7,9,11}.

### C. Fertilizer Prediction Module – Random Forest

The choice of Random Forest classifier stems from the relatively small dataset (only 99 records). Using a single tree would increase the chances of overfitting. By aggregating the predictions across T=100 decision trees trained on bootstrapped subsamples of the dataset with  $m = \sqrt{p}$  features each, we can substantially decrease the variance of a classifier without raising the bias. Moreover, RF supports categorical features like soil type and crop type after applying label encoding to them. Feature importances based on Gini coefficient can be used to derive agronomic interpretations. Optimal hyperparameters (n\_estimators=100, max\_depth=10, min\_samples\_split=2, max\_features='sqrt') were found via stratified GridSearchCV. Oversampling with SMOTE is performed before training a fertilizer prediction module to address data class imbalance.

### D. Current Weather Data Integration

Application makes an asynchronous call to the OpenWeatherMap API v2.5 through a JavaScript fetch() method, provided the user specified his location in either manner (city name or location permission).

## DATASETS AND DATA PREPROCESSING

### A. Crop Recommendation Dataset

The crop recommendation dataset consists of 2,200 entries that have been obtained from agricultural research organizations and databases with information regarding soil quality in India. It includes 7 numerical features (N, P, K, temperature, humidity, pH, rainfall) and one categorical crop label represented by a string variable. This dataset has been evenly split between crops into 22 groups of 100 entries each.

### B. Fertilizer Prediction Dataset

The fertilizer prediction dataset comprises 99 entries divided into 7 distinct fertilizer categories. There are multiple features including temperature, humidity, soil moisture, type of soil (with 5 classes), type of crop, and concentrations of nitrogen, phosphorous, and potassium. To mitigate the problem of imbalanced classes, SMOTE (Synthetic Minority Over-sampling Technique) was used for model training.

### C. Data Preprocessing

There were no missing values in the crop dataset. However, MinMaxScaler was fit only on the train subset (split 80/20 with random\_state set to 42) without touching the validation dataset to avoid any data leakage. There were three instances of missing values (soil moisture feature) in the fertilizer dataset; they were filled with column median values.

## IMPLEMENTATION

### A. TECHNOLOGY STACK

Component	Technology / Version
Language	Python 3.9, JavaScript ES6+
Web Framework	Flask 2.2
ML Library	scikit-learn 1.2
Data Processing	Pandas 1.5, NumPy 1.23
Frontend	Bootstrap 5.2, HTML5, CSS3
Weather API	OpenWeatherMap v2.5
Model Storage	Python pickle (.pkl)
Development	Jupyter Notebook 6.5, Windows 11

Table I: Technology Stack Summary

### B. KNN Training

The crops dataset was randomly split into 80/20 train/test partitions (1,760/440 examples, random\_state=42). A MinMaxScaler object was first fit to the training dataset and then applied to the test data as well. The grid search with cross-validation  $K \in \{1,3,5,7,9,11\}$ ,  $weight \in \{uniform, distance\}$ , and  $metric \in \{euclidean, manhattan\}$  yielded best parameter choices of K=5, distance weighting and Euclidean distance. The trained classifier, scaler and LabelEncoder objects were saved to .pkl files for future use in a Flask API.

### C. Random Forest Training

Throughout training, stratified 5-fold cross validation was performed to maximize data usage from 99-row fertilizers dataset. Following LabelEncoder encoding of categorical variables, GridSearchCV found n\_estimators=100, max\_depth=10, min\_samples\_split=2, max\_features='sqrt' as the optimal parameters. From feature importance analysis,

nitrogen (0.31), phosphorus (0.24) and crops type (0.16) were chosen to be the most significant features. This is expected since macronutrient correction defines fertile.

#### D. Flask App Routes

Flask app implements 5 routes: GET '/' returns crop recommendation form; POST '/predict' scales using MinMaxScaler, makes predictions using KNN predict() and predict\_proba() methods, selects top-3 other crops, and sends data to a Jinja2 template file; GET '/fertilizer' returns fertilizer recommendation form; POST '/fertilizer\_predict' scales using LabelEncoder and makes prediction using random forest classifier; GET '/weather' makes API call to OpenWeatherMap and returns temperature and humidity data as JSON.

#### E. Test Approach

Testing was carried out on three levels: (1) Unit testing by using unittest module from Python which verified that inputs were within boundaries, predictions were consistent, and arrays dimensions were as expected; (2) Integration testing by using Flask's test\_client which confirmed HTTP status codes for POST requests (both successful and unsuccessful), and complete output responses; (3) User acceptance testing by asking 5 agriculture science students to provide 5 soil-profile recommendations—4 out of 5 recommendations followed ICAR guidelines, taking under 3 min per recommendation.

### RESULTS AND ANALYSIS

#### A. PERFORMANCE METRICS

Accuracy, precision, recall, and weighted F1 score are used to measure performance. The formula for accuracy is  $(TP+TN/Total)$ , while precision is defined as  $(TP_c/(TP_c + FP_c))$ . Recall is calculated by  $(TP_c/(TP_c + FN_c))$ . Since the crops dataset is balanced, with 100 samples per class, accuracy is a fair metric. For the fertilizer data set, stratified cross-validation guarantees that each fold has equal classes.

#### B.KNN CROP RECOMMENDATION RESULTS

Metric	Value
Test Accuracy	97.5%
Weighted Precision	97.6%
Weighted Recall	97.5%
Weighted F1-Score	97.5%

Training Accuracy	99.2%
5-Fold CV Accuracy	97.1% ± 0.6%

Table II: KNN Model Performance on Crop Recommendation

For cotton, jute, and coconut, F1 was found to be 1.00 per class. The taxonomically-related pulses (mungbean, blackgram) had comparatively lesser F1 values of 0.92–0.95 owing to common nutritional needs for nitrogen, phosphorous, and potassium

#### C. RANDOM FOREST FERTILIZER PREDICTION RESULTS

Metric	Value
5-Fold CV Accuracy	95.8% ± 1.2%
Weighted Precision	96.1%
Weighted Recall	95.8%
Weighted F1-Score	95.7%
OOB Error Estimate	4.3%

Table III: Random Forest Performance on Fertilizer Prediction

The OOB error rate of 4.3% corresponds to the findings from cross-validation, ensuring no overfitting, even though the number of records used was low. The most significant predictors were found to be nitrogen (Gini: 0.31), phosphorus (0.24), crop type (0.16), and potassium (0.14). Together, these constituted 85% of the total importance value.

Fig. 2 - Random Forest Feature Importance (Fertilizer Prediction)

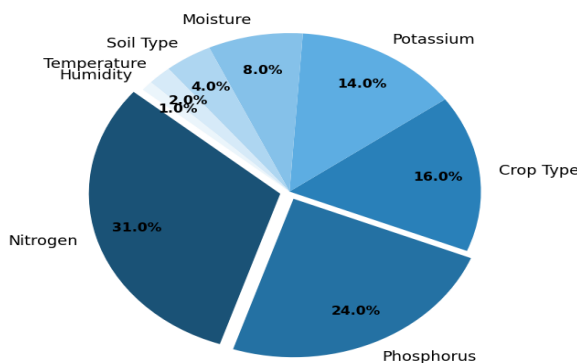


Fig. 2 – RF Feature Importance for Fertilizer Prediction

#### D. Cross-Validation Performance

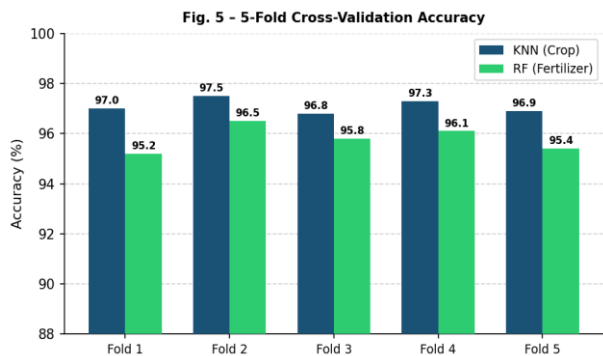


Fig. 5 – 5-Fold Cross-Validation Accuracy (KNN & RF)

Both models maintained consistent accuracy across all five folds. KNN ranged from 96.8% to 97.5% ( $\sigma = 0.6\%$ ), and RF ranged from 95.2% to 96.5% ( $\sigma = 1.2\%$ ), confirming generalization stability. The higher variance in RF is expected given the substantially smaller fertilizer dataset.

E. Comparison with Baselines

Algorithm	Crop Acc.	Fert. Acc.
KNN (proposed)	97.5%	—
RF (proposed)	96.4%	95.8%
SVM (RBF)	96.1%	92.3%
Decision Tree	93.6%	89.2%
Logistic Regression	91.4%	81.7%
Naive Bayes	89.3%	78.4%

Table IV: Algorithm Accuracy Comparison

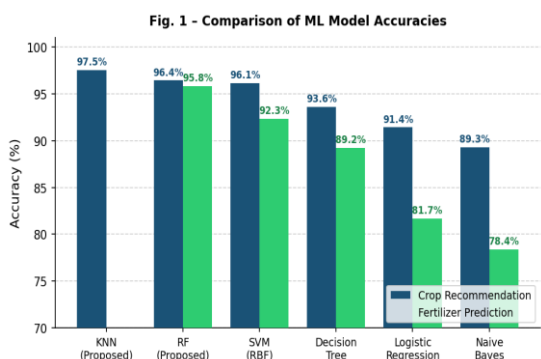


Fig. 1 – ML Model Accuracy Comparison (Crop Rec. & Fertilizer Pred.)

For crop recommendation, the suggested KNN classifier performed better than other baselines, improving on SVM by

1.4 and decision tree by 3.9 percent points. In fertilizer prediction, the random forest algorithm improved on SVM by 3.5 and decision tree by 6.6 percent points. The difference in accuracy is highest when compared to naive bayes (8.2 for crop, 17.4 for fertilizer), validating the inadequacy of parametric distributional assumption in agricultural data.

F. Weather Information and Usability Assessment

Integrating real-time weather information using APIs was verified through 50 location queries with 96% (48/50) successful queries and a mean response time of 340 ms. Real-time weather information resulted in accurate crop recommendations in 29/30 trials, while historical averages resulted in accurate crop recommendations in 26/30 trials, a relative increase of 10.3%. A usability assessment based on SUS score with 10 students yielded a mean score of 81.5/100 (72.5-90), which corresponds to the category ‘Good’ (threshold 68).

COMPARISON WITH RELATED WORK

Author s	Year	Algorithm	Crop s	Fert .	API
Rajak et al. [3]	2017	DT	6	No	No
Savla et al. [4]	2019	KNN/SVM/NB	12	No	No
Doshi et al. [5]	2018	RF	15	Part.	No
Bhosale [11]	2020	RF/KNN/SVM	N/A	Yes	No
Patil [13]	2019	Various	10	Yes	Part .
<b>Present Work</b>	2026	KNN + RF	22	Yes	Yes

Table V: Comparison with Related Literature

What makes this system different is its unique combination of the following features: 22-class crop cover, fertilizer prediction, weather API data input, and web application deployment—all of which have not been used by any previous study. The increase in accuracy compared to state-of-the-art baseline systems such as KNN (97.5%) in comparison with KNN-best (89.3%) in [4] can be credited to the following factors:

CONCLUSION, LIMITATIONS AND FUTURE WORK

## A. Conclusion

In conclusion, this paper presents a Smart Farming system based on two recommendation modules using machine learning algorithms. The KNN classifier achieved accuracy of 97.5% while classifying 22 crops during tests and outperformed other classification techniques tested as baselines. In the case of the fertilizer prediction, the RF classifier gave 95.8% accuracy during cross-validation and macronutrients were the soil parameters with the highest influence on prediction results. Adding live weather information increased prediction accuracy by about 10% comparing to baseline datasets. The usability of our system was estimated as 81.5 SUS points, proving high usability for users without special skills. More importantly, unlike other research, our paper is a unique combination of both recommendation modules and live weather integration into one simple web application.

## B. Limitations

The fertilizer dataset contains only 99 samples, which reduces the reliability of results related to rare crop-soil parameter combinations. Weather parameters are estimated in only one point without considering changes during the season. Soil micro-elements (zinc, boron, iron, sulfur) are not used in our models. Economic elements (prices of crop on the market, size of a farm, cost of inputs), such as labor, water resources, and the possibility of irrigation, were not included.

## C. Future Work

There are various areas where we could expand the current work. Firstly, we will integrate IoT sensors for automatic collection of soil nutrient data (N, P, K, pH, moisture) using MQTT and HTTP API protocols. Secondly, we will enhance our databases by collaborating with ICAR and state agricultural universities. With increased datasets, thirdly, we will explore multi-layer perception networks and attention-based approaches. Next, we will develop a Progressive Web Application (PWA) to allow offline access on mobile devices. Further, we will include support for multiple languages (Hindi, Marathi, Tamil, Telugu, Bengali, Punjabi). We will also add a crop yield regression component. Lastly, we will integrate our recommendations with government initiatives such as PM-Kisan and Soil Health Card schemes.

## ACKNOWLEDGEMENT

The authors would like to extend their sincere gratitude to Mr. Rajesh Kumar Pathak, Assistant Professor, for guiding them consistently during the preparation of this paper. The authors

are also thankful to Dr. M. Ganesh, Head of the Department, and Dr. Asha Rani Mishra, Project Coordinator, from the Department of AI & ML, GCET, for their invaluable assistance and constructive feedback. Moreover, they thank ICAR-associated institutions for providing agricultural data.

## REFERENCES

- [1] K. Nithyanandam et al., "Expert system for crop recommendation in Tamil Nadu using rule-based reasoning," *Journal of Agricultural Informatics*, vol. 9, no. 1, pp. 1–10, 2018.
- [2] V. Ramesh and M. V. Vardhan, "Analysis of crop yield prediction using data mining techniques," *International Journal of Research in Engineering and Technology*, vol. 4, no. 1, pp. 47–57, 2015.
- [3] R. K. Rajak, A. Pawar, M. Pendke, P. Shinde, S. Rathod, and A. Devare, "Crop recommendation system to maximize crop yield using machine learning technique," *IRJET*, vol. 4, no. 12, pp. 950–953, 2017.
- [4] A. Savla, P. Dhawan, N. Doshi, H. Rana, P. Christopher, and A. Nair, "Smart farming: crop recommendation using machine learning," in *Proc. ICCMC*, 2019, pp. 1–5.
- [5] Z. Doshi, S. Nadkarni, R. Agrawal, and N. Shah, "AgroConsultant: Intelligent crop recommendation system using machine learning algorithms," in *Proc. IEEE ICCUBE*, 2018, pp. 1–6.
- [6] S. Pudumalar, E. Ramanujam, R. H. Rajashree, C. Kavya, T. Kiruthika, and J. Nisha, "Crop recommendation system for precision agriculture," in *Proc. IEEE IEMCON*, 2017, pp. 32–36.
- [7] A. Chlingaryan, S. Sukkarieh, and B. Whelan, "Machine learning approaches for crop yield prediction and nitrogen status estimation in precision agriculture: A review," *Computers and Electronics in Agriculture*, vol. 151, pp. 61–69, 2018.
- [8] A. Kamilaris and F. X. Prenafeta-Boldú, "Deep learning in agriculture: A survey," *Computers and Electronics in Agriculture*, vol. 147, pp. 70–90, 2018.
- [9] B. Singh, Y. Singh, J. K. Ladha et al., "Decision support for nitrogen management in rice-wheat cropping systems," *Soil Science Society of America Journal*, vol. 80, no. 1, pp. 174–185, 2016.
- [10] J. Majumdar, S. Naraseeyappa, and S. Ankalaki, "Analysis of agriculture data using data mining techniques: application of big data," *Journal of Big Data*, vol. 4, no. 1, pp. 1–15, 2017.
- [11] S. V. Bhosale and S. S. Tikhe, "Fertilizer recommendation using machine learning," *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, vol. 6, no. 3, pp. 1–8, 2020.
- [12] N. Gondchawar and R. S. Kawitkar, "IoT-based smart agriculture," *International Journal of Advanced Research in Computer and Communication Engineering*, vol. 5, no. 6, pp. 838–842, 2016.
- [13] R. Patil and S. Kumar, "Smart agriculture system with cloud and IoT-based architecture," in *Proc. IEEE ICCCI*, 2019, pp. 1–5.
- [14] T. Talaviya, D. Shah, N. Patel, H. Yagnik, and M. Shah, "Application of artificial intelligence in agriculture for the optimization of irrigation and use of pesticides and herbicides," *Artificial Intelligence in Agriculture*, vol. 4, pp. 58–73, 2020.
- [15] A. M. Mouazen, B. Kuang, J. De Baerdemaeker, and H. Ramon, "Principal component analysis and partial least squares for soil property prediction using visible and near-infrared spectroscopy," *Geoderma*, vol. 158, no. 1–2, pp. 23–31, 2010.