**ABSTRACT**

Despite being the backbone of the global economy, farmers have always struggled with challenges like what crop to plant, how much fertilizer to use, and when to detect the disease beforehand. These problems not only affect farmers but also lead to various concerns, such as food safety, which will greatly impact sustainable agriculture practices. The old farming methods depend on emotion, intuition, feeling, and experience, reducing yield production and economic losses. This paper proposes a deep-learning-based Smart Farming and Advisory System (SFAS) to overcome the challenges faced and help our farmers. The proposed model helps farmers by providing real-time crop suggestions based on soil, climate, and market conditions. It also uses soil nutrient composition to provide custom-made fertilizer recommendations and uses Deep Learning (DL) methods to detect plant diseases by processing images. Additionally, the proposed system works on multi-language support, which enables farmers of diverse linguistic backgrounds to access agricultural insights in their native languages. This research seeks to bridge the knowledge gap in agriculture by using Machine Learning (ML), the Internet of Things (IoT), and multilingual accessibility to make farmers self-reliant through data-driven decision-making and sustainable agriculture practices. The experimental results show that the system is predicting well regarding increasing yield, reducing losses, and making precision farming decisions.

Keywords: Climate-smart agriculture, Deep learning, Crop recommendation, Plant disease detection.

**INTRODUCTION**

Crop production, a keystone of the global economy, plays a vital role by generating employment, supplying raw materials for industries, and meeting the ever-growing demand for food. However, today's farmers face many challenges while determining which crop is best for their land, how much and what kind of fertilizer is best, and what conditions are met that cause plants to suffer at the right time. The integration of advanced technologies such as DL, IoT, and Artificial Intelligence (AI) has reinvented various industries, and agriculture is one of them.

ML algorithms are used to analyze large and complicated datasets, which include measurements of plant health factors, soil composition, and weather conditions to predict the ideal crop for the farmers. By analyzing photographs of infected crops, various diseases have been discovered by DL techniques, i.e., CNNs (Convolutional Neural Networks). IoT sensors are useful for tracking temperature, humidity, and soil health in real time, which aids decision-makers in achieving agricultural output goals and minimizing losses. With these advanced technologies, experience-based methods give way to precision agriculture, where data-driven insights optimize each step of the agricultural process.

Despite all these technological advancements, several challenges hinder the broad acceptance of smart farming systems. Conventional farmers are more vulnerable to inefficiencies, decreased crop yields, and financial loss because they still rely on manual observations and experience-based decision-making. Language barriers also make it difficult for many small-scale farm owners to understand and use cutting-edge agricultural techniques, and they frequently lack access to professional advice. As a result, these issues highlight the need for clever, data-driven solutions that enable precise and understandable suggestions for farmers.

To address these challenges, this paper introduces a DL-powered SFAS that helps farmers make efficient decisions. This technology provides real-time crop planting suggestions based on weather patterns and soil conditions. Personalized fertilizer recommendations are provided by thoroughly examining the need and nutritional level of the soil. A plant disease detection unit that uses a computer vision algorithm for identifying plant illness and includes treatment for the same. Additionally, farmers can receive all the agricultural information in their native language with the help of a multilingual option. This feature has helped to build the knowledge gap in agriculture and increase production and sustainability by combining these technologies.

The main contributions of this research are as follows:

* A crop recommendation model, i.e., SFAS, that optimizes yield and sustainability.
* A DL-based disease detection system for early and precise diagnosis of plant diseases.
* A data-driven fertilizer recommendation system that suggests the right amount and best quality of fertilizer to increase crop yield.
* A multilingual support framework helps farmers access it in their native languages.

**Article Organization**

**LITERATURE WORK**

DL and ML Techniques have been used to improve decision-making in various domains. Disease identification is often accomplished via CNNs and K-means clustering, while real-time flexibility is enhanced by IoT-based monitoring. Existing research articles demonstrate high accuracy in crop selection and disease diagnosis, but issues with dataset constraints and environmental changes still exist.

Table 1. Comparison of Existing Research Papers on Smart Agriculture

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Reference** | **Proposed Model** | **Technique** | **Compared with** | **Limitations** | **Features** |
| [1] | Plant diseases detection and fertilizer recommendation | DL and AI | Traditional farming methods using real-world datasets | Small dataset, lacks real-time adjustment | Crop health monitoring and fertilizer recommendation |
| [2] | Crop and fertilizer Recommendation based on soil data | ML | Older ML models in terms of accuracy | Needs high-quality inputs | Soil analysis, crop selection |
| [3] | Disease detection by enhancing image quality | CNN and DL | Tested on agricultural images and compared to low-resolution models | High computational power required | Disease identification |
| [4] | Plant disease detection for quick action | IoT and ML | IoT-based setups in field trials on tea plants | Needs reliable IoT setup | Early disease alerts |
| [5] | A system that gathers farming data for smarter decisions | IoT and ML | Traditional decision-making methods | High initial cost | Smart farming |
| [6] | AI-based application for farming advice | ML | Manual decision-making and tested on real-world farms | Accessibility issues for small farmers | Crop yield advisory |
| [7] | Yield estimation and disease detection | ML | Older predictive models and weather-based estimations | Accuracy depends on input quality | Yield prediction |
| [8] | A study reviewing ways to use ML for plant disease detection | ML | Summarized results from multiple ML-based agricultural studies | Needs testing with real crops | Disease classification |
| [9] | Plant health monitoring and real-time updates | IoT Sensors | Manual monitoring and tested with sensor-based data | Expensive infrastructure | Remote monitoring |
| [10] | Plant disease detection | CNN | Various DL techniques | Requires labeled datasets | Pathogen identification |
| [11] | Automatic and accurate crop disease detection | ML | Manual methods in real-world trials | High computational costs | Disease assessment and advisory |
| [12] | A blockchain-based platform for safe and transparent agricultural trading | Blockchain | Traditional trading methods | Requires industry adoption | Secure supply chain |
| [13] | Crop selection based on climate conditions | AI and climate modeling | Existing methods | Prediction can be uncertain | Climate-resistant crop selection |
| [14] | A pest control system using AI | ML | Manual pest control | Needs image datasets | Pest tracking |
| [15] | Monitor soil fertility and nutrients | AI and sensor networks | manual analysis and validated with soil sample tests | Limited to specific soil types | Nutrient and soil health analysis |
| [16] | Broad Crop health monitoring using satellite images | Satellite AI | Satellite images v/s manual farm inspections | High costs for imagery | Broad crop health analysis |
| [17] | Fast and accurate disease detection | CNN | Slower manual processes | Needs high-quality datasets | Real-time disease monitoring |
| [18] | Predict fertilizers based on soil conditions | AI and IoT | Analyzed soil data v/s predictions to actual needs | Limited scalability | Nutrient tracking |
| [19] | Prediction of crop yields using historical data | ML and Predictive Analytics | Validated predictions against weather data and historical records | Varies regionally in accuracy | Yield forecasting |
| [20] | System for pest control and pesticide use | IoT and ML | manual pest monitoring | IoT network dependency | Pesticide optimization |
| **Proposed** |  |  |  |  |  |

Table 1 highlights various technological advancements in smart agriculture, focusing on IoT-based monitoring, disease detection, predictive analytics, and crop and fertilizer recommendation. Many studies have applied ML and DL to identify crop diseases, utilizing CNNs and AI-powered advisory systems to improve the accuracy of disease detection. Moreover, IoT sensors have been widely used to track real-time environmental information such as humidity, temperature, and soil health, helping in the timely decision-making process. Some researchers have also explored image enhancement techniques, which lead to better disease detection systems, while few have emphasized yield prediction models. However, the crucial challenges include reliance on large datasets, accessibility issues, and computational cost.

Background

1. Dataset description

The primary objective of the research is to identify datasets that are not only efficient and concise as effectiveness of machine learning models depends on datasets that are lightweight and computationally efficient. The selection of appropriate datasets is crucial especially when there is limited hardware and there is need for real-time performance. The implementation of large and complex datasets would be impractical in environment where resources are limited and particularly when deployment is to be done in rural or field conditions.

The crop recommendation dataset was developed my researcher to meet the specifications of the system. The dataset comprises of about 50,000 records which makes it ample enough to support generalizing capacity and maintaining it computationally manageable. Each of the entry includes seven features which are- Nitrogen(N), Phosphorus(P), Potassium(K), Temperature, Humidity, pH level and Rainfall. These parameters were taken as they have a direct impact on crop and soil health. The target variable is the Crop Label as that is the most suitable crop based on the input conditions. To train and evaluate the machine learning model, the dataset was split into 80% training and 20% testing data. This ensured that models learned patterns effectively while being tested on unseen data.

The another dataset for fertilizer recommendation was also created by researchers to support fertilizer suggestions. It focuses on improving nutrient management based on their crop type and soil status. It includes features such as Soil type, Nitrogen, Phosphorus, Potassium and Crop Type and Moisture content. The output we get is the recommended fertilizer type.

For disease detection, we have used a publicly available dataset was used from Kaggle: New Plant Diseases Dataset. This dataset consists of over 87,000 high resolution images of both healthy and diseased plant leaves which includes crops such as apple, corn, grapes, tomato etc with multiple disease categories such as spot, rust, blight and mildew. These images were first pre-processed using standard techniques such as resizing, normalization and augmentation and then fed into convolutional neural network for classification. To train and evaluate the machine learning model, the dataset was split into 80% training and 20% testing data. The large size of dataset enables model to recognize pattens with higher accuracy.

1. Classification models

For the selection of most appropriate classification model for this project in field of agriculture involves the careful balance of accuracy, interpretability and efficiency. The models used in this project were chosen due to their strengths and performance on the data. XGBoost is known for its robust boosting mechanism and speed, CatBoost is used as it effectively handles categorial data with less preprocessing Whereas TabNet combines decision tress and attention mechanisms which helps to model complex relationships. An MLP based meta learner is used in stacking and integrate the power of all base models and help improve generalization. Resnet9 on the other hand was used for disease prediction. It allows for deeper represenatation learning ehich makes it well suited for image classification tasks.

1. Extreme Gradient Boosting(XGBoost)

It is a scalabe boosting technique that enhances the prediction accuracy through gradient boosting decision trees. It also handles overfitting by L1 and L2 regularization and offers parallel for faster training and also supports missing value handling which makes it suitable for large-scale training tasks. In our project, XGBoost is used with a learning rate of 0.01, with 300 estimators and a depth of 10. It is highly suitable for crop recommendation due to its capability to manage structured agricultural data and has worked well with numeric features such as NPK values, temperature and humidity.

1. Categorial Boosting(CatBoost)

It is used for gradient boosting which handles categorial data without requiring any extensive preprocessing which makes it powerful for agricultural datasets which doesn not include any numeric data such as soil type, crop name and region. It also avoids overfitting by implying ordered boosting and symmetric trees which then ensures high accuracy. This makes CatBoost ideal choice for recommendation tasks where efficiency and interpretability are required. In our project, it is used with a learning rate of 0.01, with 300 estimators and a depth of 10 and offers stability even with imbalanced datasets.

1. TabNet

It is a Deep Learning module which is specially designed for tabular data. It requires sequential attention to select relevant features during training which then allows for high interpretability and efficient utilization of features. In our project, TabNet is Wrapped in a scikit-learn compatible format which enables its integration into stack framework. It can capture nonlinear feature interactions and can dynamically focus on important variables which enhances model expressiveness majorly in complex soil and crop condition datasets.

1. Multi-Layer Perceptron(MLP)

It is used as a meta-classifier in stacking the various base models. It learns from the output probabilities of the base models, in our project which are XGBoost, CatBoost, and TabNet and then combines their strengths for final predictions. The MLP has a deep architecture with layers and uses a ReLU activation function. It enables early stopping to avoid overfitting. It effectively generalizes the combined outputs of the base models and ensures stable predictions for the recommendations of most suitable crop.

1. ResNet9

It is very efficient convolutional neural network derived from deeper ResNet architectures. It utilizes residual connections which helps in mitigating vanishing gradient problem and helps in improving convergence during training. It has smaller size and faster inference time which makes it suitable for image classification tasks on moderate sized datasets. We have used it to classify various plant leaf images into healthy and diseased categories because it provides a good balance of performance and computational efficiency. It contibes to the accuracy as it has ability to extract high-level features from images data.

**Proposed Model**

Our research has included various technologies such as AI, Deep learning, and IoT to support rural planning and development. Crop recommendation, fertilizer recommendation and plan disease detection are the various modules featured in our study. Information regarding system design, data collection. model development and integration is explained further in this section.

**A. System Overview**

Our research suggests a real-time recommendation system which also has a potential to develop into a website that collects agricultural data for analysis. The platform may process user data such as soil constituents, weather, humidity, and plant photos using various deep learning models to provide useful insights. Hence our system consists of three major modules which are:

1. **Crop Recommendation Module**: This module recommends the best suitable crop for cultivation based on soil properties and weather conditions.
2. **Fertilizer Recommendation Module**: It suggests the right fertilizer to improve soil nutrients and further promote crop development by analyzing soil nutrient concentration as well.
3. **Plant disease detection module**: This module uses deep learning-based image processing technique to detect plant disease by using a leaf image and hence providing treatment recommendations for the same.

**B. The Application**

By offering suggestions for crop selection, fertilizer use, and disease detection, our application seeks to support farmers in making better decisions and improving sustainability and production. The home page of the application is where users get to select from three different functionalities. The crop recommendation tool, when entering values of different soil factors, recommends the best crop to crop, whereas the fertilizer recommendation tool examines the crop and nutrients in the soil and gives suggestions accordingly. Users can upload pictures in the disease detection section to determine the disease their plants are suffering from and know the right treatment for it. This also increases efficiency in the agricultural field.

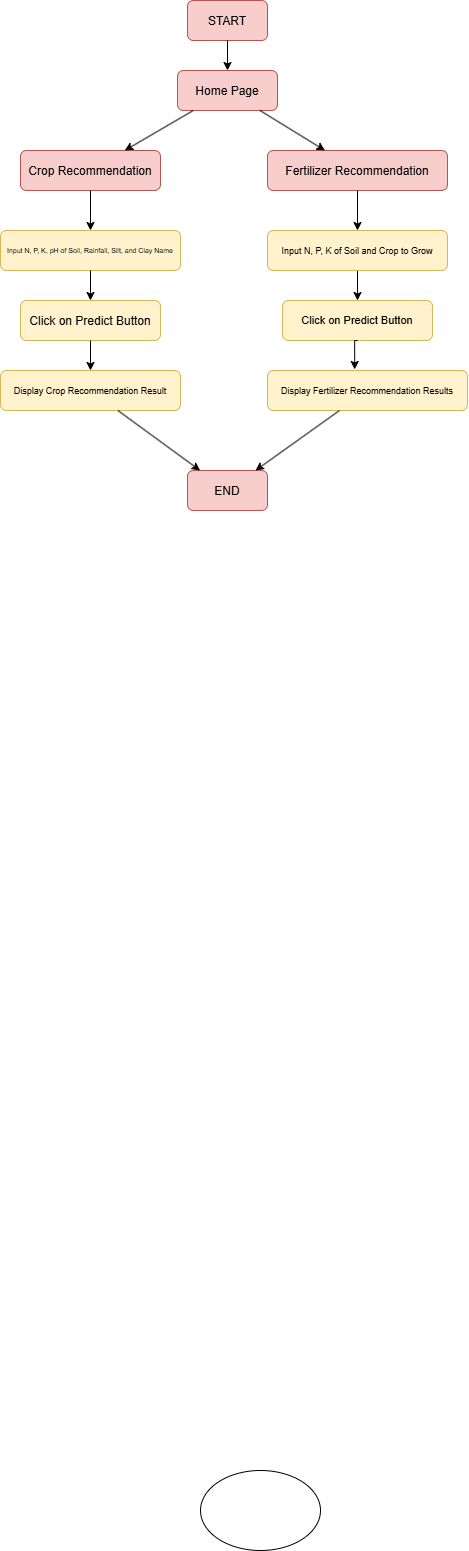


Figure 1

1) Crop Recommendation

The user will input values for all the soil characteristics that are nitrogen (N), phosphorus (P), potassium (K), temperature, humidity, rainfall, and soil PH. The system will then send a POST request to Flask API where the train model will further process the request. The model then returns an HTTP request to the frontend, displaying the most suitable crop to grow. The crop recommendation flow diagram in shown in Figure 2.

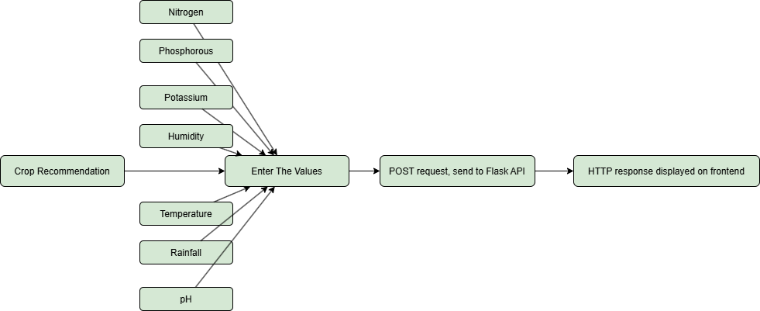


Figure 2

Machine Learning Approach

This module is trained on a dataset of soil nutrients, weather data, and crop information. It has attributes like nitrogen (N), phosphorus (P), potassium (K), temperature, humidity, rainfall, and soil pH in the dataset. This requires normalizing these values and dealing with missing/inconsistent entries through data preprocessing.

The most appropriate classifier for crop recommendation is implemented and compared using multiple classification algorithms. The models like Decision Tree, Naïve Bayes, Support Vector Machine (SVM), Logistic Regression, Random Forest, XGBoost and stacked models. The best model based on cross-validation accuracy (after hyperparameter tuning) is selected.

|  |
| --- |
| Algorithm 1 Proposed Algorithm for Crop Recommendation |
| 1. Results: Performance is calculated using the Confusion Matrix and ROC Curve 2. Input: Dataset for Crop levels of Nitrogen, Phosphorous, Potassium, Temperature, Humidity, pH and Rainfall 3. Output: Crop recommendation based on input conditions 4. Procedure:   Step 1: Preprocess data-encoding the crop labels, splitting of dataset and balancing classes  Step 2: Select XGBoost, CatBoost and TabNet as base classifiers.  Step 3: Using the selected models for implementing a stacking ensemble and an MLP as the meta-classifier.  Step 4: Optimizing hyperparameters  Step 5: Evaluate model using metrics(Accuracy, Precision, Recall, F1 score, Specificity, NPV) and visualize with ROC  Step 6: Save trained models and encoders for deployment and repeat  Steps 3 to 5 for alternate ensemble strategies for comparison. |

2) Fertilizer Recommendation

The user has to provide the nutrients level of soil which are nitrogen (N), phosphorus (P), potassium (K) and the crop name. The a POST request is sent to the Flask API which is hosting the fertilizer classifier. After processing this request, API returns a response and frontend displays the best fertilizer and the quantity to be used. The fertilizer recommendation flow diagram in shown in Figure 3.

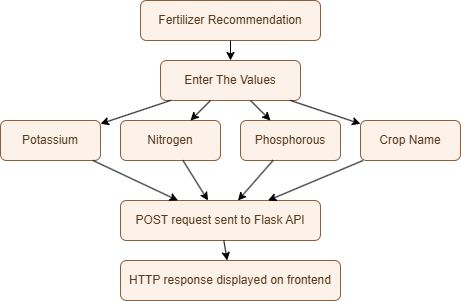


Figure 3

Machine Leaning Approach

This module uses a dataset that correlates the soil nutrient level and crop type to the corresponding fertilizer mixing ratio. It contains data like soil pH, nitrogen, phosphorus, potassium levels, and the recommended combinations of fertilizers.

Fertilizer recommendations are based on a rule-based classifier. The model evaluates if there is a deficiency or an excess of nitrogen, phosphorus, or potassium in the soil and recommends the right amount of fertilizers to maintain nutrient levels.

|  |
| --- |
| Algorithm 2 Proposed Algorithm for Fertilizer Recommendation |
| 1:Results: For validating our Fertilizer recommendation system’s accuracy we used real-world datasets.  2: Input: Attributes that are required are Crop name and soil nutrients level (Nitrogen, Phosphorous, Potassium)  3: Output: Recommendation of fertilizer is done based on the nutrients imbalance.  4: Procedure:  Step 1: Preprocessing the fertilizer dataset i.e to clean and normalize the data.  Step 2: Calculate the difference between the soil values and the ideal crop requirement values.  Step 3: Using the above step find the most imbalanced nutrients and recommend fertilizer accordingly using predefined dictionary.  Step 4: Optimizing the recommendation thresholds using validation data.  Step 5: Store the final logic and predefined dictionary for system deployment and repeat steps 3-4 as new data comes. |

3) Disease Detection

The user will upload a plant leaf image. The image is then processed by backend model which is using the Resnet9 pretrained model for the disease classification process. Then an HTTP response is send to frontend which then provides the diagnosis of the disease and the treatment to proceed with. The disease detection flow diagram in shown in Figure 4.



Figure 4

Machine Learning Approach

This module is trained with a large dataset of plant leaf images of many healthy and diseased samples. The images are pre-processed by resizing to a consistent size, normalizing pixel values, and augmenting the dataset to increase model generalization.

The plant disease is classified using a Convolutional Neural Network (CNN) based deep learning model. A transfer learning approach trains the model utilizing a pre-trained ResNet9 architecture for improved accuracy. The final layer is then fine-tuned to classify plant diseases and suggest appropriate treatments.

|  |
| --- |
| Algorithm 3 Proposed Algorithm for Disease Detection |
| 1: Results: Evaluation of disease detection system is done and model is used for its training and deployment.  2: Input: Leaf images of various plant is given as the input for disease detection.  3: Output: Prediction of the disease labelled for the given image using our dataset.  4: Procedure:  Step 1: Techniques like image augmentation, normalization of pixels etc are applied for preprocessing and loading of the dataset.  Step 2: ResNet9 was chosen as the base model and transferred learning was used if our dataset was small.  Step 3: Training and fine- tunning using cross-entropy and Adam optimizer was used. Other techniques like learning rate scheduling, dropout, and grid search were also used.  Step 4: Confusion matrix, accuracy, precision, recall, F1-score, specificity, FNR, FPR; plot ROC and convergence curves were used to evaluate our model.  Step 5: Trained model and class mapping was stored |

Performance Evaluation

The experimental setup was conduscted on HP Pavilion Aero Lapop equipped with an AMD Ryzen 7 5800U with Radeon Graphics, 16GB RAM and a 512 GB SSD. The operating system used was Windows 11 and all the implementations were done using Python 3.10 with the libraries such as scikit-learn, pickle, pandas, PyTorch etc. During the experiment both training and testing computation times were recordes for all the different models. The training time for Decision tree, Naïve Bayes, SVM, Logistic Regression, Random forest, XGBoost, Stacked model using MLP meta-learner were 0.37,0.01, 2.70, 4.14, 21.52, 9.21respectively and the corresponding testing times were 0.01, 0.01, 7.36, 0.02, 0.43, 0.10, 0.67 . These performance times indicate that the proposed system is capable of giving efficient and accurate predictions within a reasonable computation window which makes it more feasible.

1. RESULTS AND DISCUSSIONS

**Results based on Confusion Metrics**

**Results based on ROC Curve**

**Results based on Cross-Validation**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Models | Accuracy  (%) | Precision  (%) | Recall  (%) | F-1 Score  (%) | Specificity  (%) | FNR  (%) | FPR  (%) |
| Decision Tree | 99.26 | 99.26 | 99.26 | 99.26 | 99.96 | 0.74 | 0.04 |
| Naïve Bayes | 98.50 | 98.52 | 98.50 | 98.50 | 99.93 | 1.52 | 0.07 |
| SVM | 96.47 | 96.42 | 96.42 | 96.42 | 99.83 | 3.63 | 0.17 |
| Random Forest | 99.79 | 99.79 | 99.79 | 99.79 | 99.99 | 0.21 | 0.01 |
| XGBoost | 99.44 | 99.44 | 99.43 | 99.43 | 99.97 | 0.58 | 0.03 |
| Stacked Model | 99.85 | 99.85 | 99.85 | 99.85 | 99.99 | 0.15 | 0.01 |

A comparison Table for different algorithms used in the Crop Recommendation model

Evaluation Metrics Formulae

|  |  |
| --- | --- |
| Parameter | Formula |
| Precision(PPV) | (TP / (TP + FP)) \* 100 |
| Recall or Sensitivity | (TP / (TP + FN)) \* 100 |
| Specificity | (TN / (FP + TN)) \* 100 |
| F1-Score | (2 \* (PPV \* Sensitivity)) / (PPV + Sensitivity) \* 100 |
| Accuracy | (TP + TN) / (TP + FN + FP + TN) \* 100 |
| FPR | FP / N |
| FNR | FN / P |

|  |  |  |  |
| --- | --- | --- | --- |
|  | Models | Description | Accuracy (%) |
| [1] | Decision Tree | A simple rule-based model that creates hierarchical decisions based on soil and climate features. | 99.26 |
| [2] | Naïve Bayes | A probabilistic classifier assuming feature independence; fast but less accurate for complex decision tasks. | 98.50 |
| [3] | SVM (Support Vector Machine) | Uses kernel tricks to classify crops based on boundary margins; effective for small to medium-sized datasets. | 96.47 |
| [4] | Random Forest | An ensemble of decision trees that reduces overfitting and increases robustness in prediction. | 99.79 |
| [5] | XGBoost | Boosted trees algorithm with regularization; known for speed and accuracy on structured crop data. | 99.44 |
| Proposed | Stacked Ensemble Model | Combines XGBoost, CatBoost, and TabNet with an MLP meta-classifier for generalized, high-performance crop recommendation. | 99.85 |

1. FUTURE WORK and CONCLUSION

Our approach has great potential in agricultural decision-making with AI-based suggestions for crops, fertilizers, and plant disease detection. Future improvements may include a multi-class classification system to detect multiple diseases in a single plant, and an offline mode will ensure accessibility for farmers in remote areas with poor connectivity. Additionally for yield testing and pest control, we can implement predictive analytics.  
While Our system has already validated system’s effectiveness but further refinements can enhance it’s accessibility and reliability and will make it more robust and farmer-centric precision agriculture solution.

1. REFRENCES
2. Kumar, R., Gupta, S., Srivastava, Y., & Srivastava, H. (2024). AI Assisted Plant Disease Detection, Crop and Fertilizer Recommendation System. *2024 5th International Conference on Communication, Computing and Industry 4.0 (C2I4)*, 1-6.
3. Iniyan, S., Raja, M. S., & Rajendran, S. (2024). Crop and Fertilizer Recommendation System Applying Machine Learning Classifiers. *PeerJ Computer Science, 10*, e1955.
4. Ram, M. S., & Kumar, N. (2024). A Super Resolution CNN based Model for Crop Disease Detection. *International Journal of Advanced Computer Science and Applications, 15*(1), 100-107.
5. Bashir, R. N., & Hassan, S. A. (2021). Internet of Things (IoT) and Machine Learning Model of Plant Disease Prediction–Blister Blight for Tea Plant. *International Journal of Advanced Computer Science and Applications, 12*(5), 200-206.
6. Garg, S., Pundir, P., Jindal, H., Saini, H., & Garg, S. (2021). Towards a Multimodal System for Precision Agriculture using IoT and Machine Learning. *arXiv preprint arXiv:2107.04895*.
7. Gupta, S., Chopade, A., Jain, N., & Bhonde, A. (2022). Farmer's Assistant: A Machine Learning Based Application for Agricultural Solutions. *arXiv preprint arXiv:2204.11340*.
8. Ahmed, F. U., Das, A., & Zubair, M. (2024). A Machine Learning Approach for Crop Yield and Disease Prediction Integrating Soil Nutrition and Weather Factors. *arXiv preprint arXiv:2403.19273*.
9. Kumar, A., & Singh, R. (2022). Machine Learning Techniques for Crop Disease Detection: A Review. *International Journal of Engineering Research & Technology, 11*(2), 1-5.
10. Patel, H., & Patel, V. (2023). IoT Based Smart Farming for Crop Monitoring and Disease Detection. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology, 9*(3), 234-240.
11. Sharma, P., & Gupta, R. (2021). Deep Learning Approaches for Crop Disease Detection: A Review. *Journal of Artificial Intelligence and Systems, 3*(1), 1-10.
12. Verma, S., & Singh, A. (2023). Smart Agriculture: Crop Disease Detection Using Machine Learning. *International Journal of Computer Applications, 182*(20), 25-30.
13. Kapoor, R., & Malhotra, P. (2024). Blockchain Technology for Secure Agricultural Transactions. *Journal of Agricultural Informatics, 15*(1), 45-52.
14. Iyer, S., & Reddy, P. (2023). Climate-Resilient Crop Selection Using Artificial Intelligence. *International Journal of Agricultural Science and Research, 13*(2), 67-74.
15. Ramesh, K., & Sinha, P. (2024). Smart Pest Control Using Machine Learning. *Journal of Computational Agriculture, 10*(3), 150-160.
16. Desai, V., & Kulkarni, P. (2024). AI-Powered Soil Nutrient Monitoring for Precision Agriculture. *International Journal of Agricultural Technology, 20*(4), 300-310.
17. Joshi, A., & Shah, K. (2023). Crop Health Monitoring via Satellite Imagery Using Deep Learning. *Remote Sensing Applications: Society and Environment, 22*, 100-110.
18. Zhou, J., Li, X., & Wang, H. (2023). A Deep Learning-Based Approach for Real-Time Crop Disease Detection in Smart Agriculture. *International Journal of Agricultural Informatics, 18*(2), 45-60.
19. Kumar, V., Singh, D., & Sharma, A. (2024). Predictive Analysis of Soil Nutrients for Fertilizer Recommendation using IoT and AI. *Journal of Smart Agriculture & Automation, 12*(3), 78-92.
20. Patel, R., & Joshi, N. (2022). AI-Powered Crop Yield Prediction Model Based on Climate and Soil Conditions. *Computational Agriculture and Precision Farming, 9*(4), 125-140.
21. Das, P., & Roy, S. (2023). IoT-Driven Smart Pest Control System Using Machine Learning Algorithms. *International Journal of Agricultural Robotics, 7*(1), 200-215.