A Project Report on

Data Driven Crop – Advisory System Using Machine Learning

Submitted in partial fulfillment for award of

Bachelor of Technology

in

Computer Science and Engineering

By

L. D. R. Sathya Sree (Y21ACS493) **K. Satwika (Y21ACS465)**

K. Mounika (Y21ACS470) **M. L. Chaitanya (Y21ACS505)**

Under the guidance of

Mr. V. Naveen Kumar, M. Tech

Assistant Professor

Department of Computer Science and Engineering

Bapatla Engineering College

(Autonomous)

(Affiliated to Acharya Nagarjuna University)

BAPATLA – 522 102, Andhra Pradesh, INDIA

2024-2025

Department of
Computer Science and Engineering

CERTIFICATE

This is to certify that the project report entitled **Data Driven Crop-Advisory SystemUsing Machine Learning** that is being submitted by L.D.R.SathyaSree(Y21ACS493),K.Satwika(Y21ACS465),K.Mounika(Y21ACS470),M.L.Chaitanya(Y21ACS505) in partial fulfillment for the award of the Degree of Bachelor of Technology in Computer Science & Engineering to the Acharya Nagarjuna University is a record of bonafide work carried out by them under our guidance and supervision.

Date:

Signature of the Guide
Mr. V. Naveen Kumar
Assistant Professor

Signature of the HOD
Dr. M. Rajesh Babu
Assoc. Prof. & Head

DECLARATION

We declare that this project work is composed by ourselves, that the work contained herein is our own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

L. D. R. SathyaSree(Y21ACS493)

K. Satwika(Y21ACS465)

K. Mounika(Y21ACS470)

M. L. Chaitanya(Y21ACS505)

Acknowledgement

We sincerely thank the following distinguished personalities who have given their advice and support for successful completion of the work.

We are deeply indebted to our most respected guide **Mr. V. Naveen Kumar,** Assistant Professor, Department of CSE, for his/her valuable and inspiring guidance, comments, suggestions and encouragement.

We extend our sincere thanks to **Dr. M. Rajesh Babu**, Assoc. Prof. & Head of the Dept. for extending his cooperation and providing the required resources.

We would like to thank our beloved Principal **Dr.N.Rama Devi** for providing the online resources and other facilities to carry out this work.

We would like to express our sincere thanks to our project coordinator **Dr. P.Pardhasaradhi,** Prof. Dept. of CSE for his helpful suggestions in presenting this document.

We extend our sincere thanks to all other teaching faculty and non-teaching staff of the department, who helped directly or indirectly for their cooperation and encouragement.

L. D. R. SathyaSree(Y21ACS493)

K. Satwika(Y21ACS465)

K. Mounika(Y21ACS470)

M. L. Chaitanya(Y21ACS505)

Table of Contents

List of Figures

List of Tables

ABSTRACT

Agriculture remains the backbone of many developing and developed countries, supporting livelihood of much of the world's population and ensuring food security. The choice of the right crop to cultivate based on environmental and soil conditions is an important concern to farmers, especially those with poor access to skilled advice. For this problem to be addressed, the present work suggests a crop advisory system using machine learning that is based on historical agricultural information to make intelligent suggestions.

The model utilizes critical parameters such as Nitrogen, Phosphorous, and Potassium levels in soil, temperature, humidity, soil pH, and likely rainfall to predict the optimal crop for a given range of conditions. Data processing and analysis for the dataset are stated to be done using the Random Forest algorithm, which is among the most accurate and trustworthy algorithms used in classification issues. The final model is finally trained and tested rigorously to ensure that it generalizes on new unseen data well. Strict performance measures like accuracy, precision, recall, and 1-score are employed to ensure the efficacy of the model.

This new method varies from the old traditional ones in reducing manual decision-making and expert intervention to the minimum. In the long term, the solution presented has the potential to support precision farming, minimize wastage, and enhance national food security. By leveraging data-driven technology, the project aims to empower farmers by providing accurate, affordable, and usable information.

Keywords: Crop Advisory System, Precision Agriculture, Soil Nutrients (NPK), Sustainable Farming, Random Forest Classifier.

1 INTRODUCTION

Nowadays, agriculture remains an important sector, not only to provide food but also to sustain the livelihood of millions of people. But deciding on the appropriate crop for cultivation is now very complex because of different climatic conditions and the degradation of soil. Farmers depend on traditional knowledge, and it might not always coincide with the changing environment. As technology continues to improve, data-driven methods provide a strong means to aid farming choices. Machine learning, for example, can be used to change the way farming is done through smart prediction systems.

1.1 Background

The principal objective of this initiative is to create and implement a crop recommendation system grounded in machine learning to guide farmers in selecting the most suitable crop for their soil and environmental conditions. The system is based on a dataset containing fundamental agronomic attributes such as nitrogen (N), phosphorous (P), potassium (K), temperature, humidity, pH level, and rainfall. These soil and environmental conditions are evaluated in detail and used to train a Random Forest classifier, which is well regarded for its high accuracy and robustness in solving classification problems.

The model learns the pattern and the association of the supplied environmental factors with suitable crops for the given conditions. Upon training, the model can accept actual user input in real time and predict the best-suited crop to recommend. This diminishes the reliance on guesswork and nudges one toward more science-based decision-making, which will increase productivity.

To ensure the system's reliability, multiple performance metrics such as accuracy, precision, recall, and F1-score are employed. These metrics help in evaluating the extent to which the model generalizes to new, unseen data and the quality of its predictions.

Also, the project aims for usability. The recommendation engine is topped with a user interface that is easy to navigate and can be implemented on a personal computer or stretched out to cloud platforms in the future. This gives the solution scalability and puts it within reach of a much larger audience, especially in rural regions where agriculture support is minimal.

This tool can aid in determining the most suitable crops to plant, using actual environmental data, under the maxim of "Right Crop, Right Place, Right Time," which translates to maximizing land use, minimizing resource wastage, and generally improving the productivity of our agriculture.

1.2 Problem Statement

Agriculture is seriously challenged to maximize crop choice decisions in diverse soil and climatic conditions. Farmers tend to rely on conventional knowledge or generic suggestions that do not take into consideration the unique characteristics of their field. This misalignment between field conditions and crop needs leads to inefficient yields, wastage of resources, and financial losses.

Soil content (especially nitrogen, phosphorus, and potassium levels) and environmental conditions (temperature, humidity, pH, and rainfall) play a crucial role in crop development and productivity. Yet, identifying the intricate interplay among these factors and selecting the most appropriate crop options needs specialized knowledge that most farmers do not possess.

In addition, soil degradation and climate variability have complicated agricultural decision-making and made classical methods less certain. Farmers require convenient, data-driven tools to deal with more than one environmental parameter at once and offer tailor-made crop suggestions.

This project overcomes these challenges through the creation of a machine learning-based crop recommendation system that utilizes soil composition and environmental information to suggest the best crop options. Through the employment of sophisticated classification algorithms, the system seeks to close the gap in knowledge, minimize uncertainty during crop selection, and offer science-based advice aimed at optimizing agriculture productivity while also encouraging sustainable use of land practices.

1.3 Motivation

The origin of this crop advisory system came from the observation of my local small-scale farmers struggling with continuous crop failure despite their passion and effort. Through field trips to various farm communities, I saw how a lack of proximity to soil test facilities and agronomic knowledge made farmers rely on tradition or a neighbor's decision when planting, instead of following the specific attributes of their lands.

There is a specific interview with a third-generation farmer. Even though he had cultivated the same soil for decades, his harvests had gradually decreased as the soil conditions varied. There was no scientific advice, and he kept planting the same crops that his father used to cultivate without knowing that varying levels of nitrogen in the soil had rendered these decisions progressively inappropriate.

This experience brought to light the way that the knowledge gap in agriculture generates a cycle of inefficiency in which money is spent on crops that are not adapted to prevailing conditions. The economic impact is dire - seed, fertilizer, water, and labor wasted, and ultimately, less income for farm families who are already working thin margins.

The advances in machine learning technology created a chance to bridge this gap. We could potentially democratize agricultural knowledge by developing a system that parses advanced environmental variables to suggest the right crops and offering data-driven advice to those in need.

This project was therefore driven by the need to develop a useful tool that converts technical climate and soil data into actionable crop advice, enabling farmers to make informed decisions that optimize their land's productive capacity while encouraging sustainable agriculture.

1.4 Scope

This project involves the creation of a crop recommendation system based on machine learning with the following specified boundaries and deliverables:

The system only serves to provide crop selection advice and not the complete farming cycle. It gives priority to the first point of decision, i.e., what crop to cultivate, given prevailing soil and climatic conditions, and does not go as far as cultivation techniques, disease control, or harvesting optimization.

Data processing ability is restricted to seven major agricultural parameters: nitrogen, phosphorus, and potassium content in soil, and environmental conditions such as temperature, humidity, pH value, and rainfall readings. The system processes these parameters as static inputs instead of monitoring their temporal changes.

The advisory mechanism uses three particular machine learning algorithms—Random Forest, Support Vector Machine, XGBoost and KNN—to build classification models that map environmental situations to respective crop selections from a list of 22 pre-defined crop varieties.

Model building takes a systematic path through exploratory data analysis, feature correlation check, normalization of data, training of algorithms, and performance comparison with usual metrics (accuracy, precision, recall, and F1-score).

The project intentionally does not integrate real-time environmental monitoring, multi-season crop rotation scheduling, economic consideration (input prices, market prices), and yield quantity forecasting. The project does not also consider implementation over diverse geographic locations beyond the coverage of the dataset.

This targeted strategy allows for complete investigation of the classification issue while laying groundwork for future extensions to cater for larger agricultural decision support requirements.

1.5 Existing System and Drawbacks

The existing systems for crop recommendation are largely based on traditional farming practices and, in certain cases, on simple decision-support systems. Such approaches tend not to apply the entire set of environmental and soil parameters required to make accurate predictions. While there are digital advisory platforms available, these tend to be too generic or not locally or climatically specific.

Earlier systems used basic machine learning models like Naive Bayes, Logistic Regression, and K-Nearest Neighbors (KNN). Although these are computationally fast and simple to use, they tend to find it difficult to represent intricate, non-linear relations between features such as soil nutrient, weather patterns, and crop suitability. The predictive accuracy thus tends to be low and could fall short of the practical demands of farmers relying on high-precision advisory systems.

Moreover, most available models are learnt using small or old datasets, which limits their ability to generalize to actual-world scenarios. A further significant disadvantage is the absence of an interactive interface for entering user input, making such systems less accessible to non-technical users like farmers. Additionally, some systems do not translate numerical outputs into easily interpretable crop names, lessening the simplicity and utility of recommendations.

In general, the inaccuracy, usability, and data flexibility limitations of the current systems point to the necessity of a more scalable, robust, and farmer-friendly crop recommendation solution that takes advantage of powerful algorithms such as Random Forest, and provides real-time, region-specific crop recommendations in an easy-to-understand and simple manner.

2 Literature Survey

We examine the Sangeetha et al. [1] proposed a machine learning-based crop advice system taking into account environmental factors like temperature, pH, and humidity to advice farmers with best crop selection. Their system uses different classification algorithms such as Decision Tree, Logistic Regression, Random Forest, CNN, and SVM to predict the best crops and nutrients required based on real-time farm data. While the system shows improved crop choice with the application of predictive modeling, it is very much reliant upon sensor-based weather and soil data availability and quality, which might pose limitations on small-scale farmer access. In addition, though several algorithms are tried out, not much comparison has been analyzed on performance with different regions or consistency in accuracy, indicating the need for optimization of algorithms as well as further region-specific training.

B. Swathi Sri et al. [2] suggested a better machine learning-based crop recommendation system that employs different ML methods to improve crop prediction accuracy. Their system takes into account soil characteristics and crop leaf health to predict diseases and suggest fertilizers. Based on historical data, the system helps farmers choose the best crops for a given district. The research emphasizes the application of Random Forest for classification and seeks to enhance agricultural productivity and economic results.

Balakrishnan et al. [3] created a machine learning-based crop recommendation system to help farmers select the most suitable crops depending on soil type, climate, and past yields. The system is available through web or mobile interfaces and seeks to enhance productivity and minimize farming expenses. Their methodology includes data gathering, preprocessing, feature engineering, and model assessment to provide accurate, data-based crop recommendations [3]. This framework enables farmers to make well-informed decisions, resulting in profitable and sustainable agricultural practices.

Prof. A. M. Ghime et al [4]. designed a Crop Recommendation System based on Machine Learning for recommending crops to farmers based on soil content and climatic conditions. The system classifies the crops using K-Nearest Neighbor (KNN) and suggests crops through a web or Android app. Support Vector Machine, Random Forest, Artificial Neural Networks, and Multiple Linear Regression were also implemented [4]. The system provides real-time crop advice, yield prediction, and fertilizer application timing recommendations.

Shwetha A N et al [5]. created a Smart Crop Recommendation Systemwith machine learning to assist farmers in selecting suitable crops based on influential factors such as soil characteristics, pH levels, moisture levels, water level, season, and pesticide usage. They understood that most farmers continue to use traditional methods and sought to update crop selection with data from government-sponsored soil testing in conjunction with predictive models. The web and mobile-accessible system was tested with a variety of machine learning algorithms such as Multivariate Linear Regression, ANN, Random Forest, KNN, and Naive Bayes. Of these, Naive Bayes performed the best with an accuracy of 95%, and hence it was the model of choice [5]. The research showcases how the integration of historical and real-time data can enhance yield, maintain soil health, and reduce environmental footprint, ultimately enabling more sustainable and informed farming choices.

Mayur Desai et al. [6] proposed a Machine Learning-based Crop Recommendation System to assist Indian farmers in selecting the appropriate crops based on factors like soil type, pH, rainfall, and humidity. On a Kaggle dataset, they tested seven machine learning models and concluded that the XGBoost algorithm worked best in terms of accuracy and reliability. It was deployed as a web application in order to ensure easy accessibility to farmers. Their approach brings forward the potential of machine learning in enhancing crop selection, productivity, and sustainable farming. Future study includes expanding the model to project more than one crop.

An Intelligent Crop Recommendation System based on Machine Learning was proposed by Priyadharshini A et al [7]. to address the issues Indian farmers have while selecting suitable crops due to their reliance on traditional techniques. To assist farmers in choosing the best crops, their system considers important factors including the time of year to plant, the properties of the soil, and the location of the farm. Decision trees, K-Nearest Neighbors (KNN), Naive Bayes, neural networks, and support vector machines (SVM) are among the machine learning techniques that are compared in this study [7]. The Neural Network model had the highest accuracy (89.88%), followed by Naive Bayes (88.26%) and KNN with cross-validation (88%)].

3 Proposed System

The Data Driven Crop Advisory System takes soil content (N, P, K) and environmental conditions (temperature, humidity, pH, rainfall) as input through a preprocessing pipeline using StandardScaler for normalization. The core intelligence of the system uses four machine learning algorithms (Random Forest, SVM, XGBoost, and KNN), each providing various mathematical methods for crop classification. After model training, an extensive evaluation framework measures performance using accuracy, precision, recall, and F1-score metrics. The top-performing model then provides recommendations from 22 crops of interest based on input conditions, mapping raw environmental information to useful agronomic guidance that assists farmers in optimizing crop selection decisions.

3.1 Advantages of the Proposed System

The crop advisory system has the substantial real-world advantages through its customized methodology in agricultural decision-making. Processing local-relevant parameters of climate and soil, the system presents guidance that is calibrated specifically for local field conditions in contrast to the generic regional instructions. Such customized approach presumably mitigates inefficient, resource-consuming trial-and-error forms of farming in which unnecessary expenditure occurs in producing unsuitable crops.

The multi-algorithm comparative system offers intrinsic validation by cross-validating outcomes, bringing increased confidence in advisories alongside suggesting where predictions may need further scrutiny. This system generates an inbuilt self-validation mechanism rare in standard advisory procedures.

The system's targeted input parameters (seven measurable factors only) balance comprehensiveness with real-world applicability, making investments in soil testing more worthwhile by simplifying technical data into practical crop selection advice. This reduction of technical measurements to practical advice narrows the knowledge gap between laboratory analysis of soils and decision-making at the field level.

In addition, the model's excellent demonstrated high rates of accuracy (specifically, the 99.32% reported by Random Forest) indicates good performance to depend on which might drastically diminish selection errors as opposed to experience-driven methods, particularly in areas under changing patterns of climate or outside normal historic soil conditions.

3.2 Machine Learning Algorithms

The project has a strategic multi-algorithm framework to solve the crop recommendation problem using complementary classification methods. All algorithms contribute unique mathematical strengths for detecting patterns of relationships between environmental parameters and compatible crops. Standardized input features are processed within parallel model training streams in order to facilitate comparative performance evaluation. This varied analysis framework increases reliability in recommendations due to multiple perspectives of computation regarding the same agronomic classification challenge. The implementation has uniform evaluation criteria for all the algorithms, forming an equal basis for comparison as well as offering insight into which methodology accurately represents the environmental-crop relationships.

3.2.1 Random Forest Classifier Approach

The project uses a Random Forest Classifier with 20 decision trees to identify the intricate interactions between environmental conditions and soil nutrients. Using an ensemble technique, several decision paths are used to arrive at crop suitability, reducing the chance of overfitting but keeping high prediction performance. By observing feature interactions in nitrogen, phosphorus, and potassium levels, temperature, humidity, pH, and rainfall parameters, the model registered an outstanding 99.32% accuracy on the test set. Random Forest's built-in feature importance tracking also gave clues as to which environmental variables most strongly impact crop choice decisions.

3.2.2 Support Vector Machine Methodology

The Support Vector Machine classifier was set with default hyperparameters and a fixed random state to guarantee reproducibility. The algorithm maps the seven-dimensional input space (N, P, K, temperature, humidity, pH, rainfall) to determine the best decision boundaries between the 22 crop classes. Although it achieved a good 96.82% accuracy, the SVM performed slightly worse than ensemble models, indicating that the crop classification task is aided by the more sophisticated decision boundaries of tree-based models. The SVM, however, brings useful diversification to the system's analytical strengths.

3.2.3 XGBoost Algorithm Integration

The solution integrates XGBoost with 10 estimators to perform multiclass classification using the mlogloss evaluation metric. In order to handle XGBoost's zero-based indexing, the code has label adjustment prior to training and prediction steps. This gradient boosting model builds trees sequentially that make up for mistakes made by previous iterations, providing an accuracy of 98.64% that is close to Random Forest performance. The XGBoost solution provides an optimal trade-off between computational speed and prediction accuracy, which may prove useful for system deployment on resource-poor systems.

3.2.4 K-Nearest Neighbors Implementation

The K-Nearest Neighbors approach offers an entirely different solution in the form of classifying crops by similarity to familiar cases instead of inferring decision rules. The distance-based solution scored 95.91%, clearly showing that even minimalistic mathematical solutions can solve the problem of recommending crops very well when they are given well-scaled input features. The KNN version has computational ease and explainability, mapping new field conditions to the most similar conditions experienced in the past and their successful crops.

3.3 Dataset Information

This project applies to a dedicated agricultural data set consisting of 2200 records of climate and soil parameters with their corresponding optimal crop recommendations. Each record combines seven important agricultural parameters: three measurements of soil macronutrients (nitrogen, phosphorus, and potassium content in ratio values) together with four environmental factors (temperature in degrees Celsius, relative humidity as a percentage, pH acidity/alkalinity scale, and rainfall in millimeters). The data set includes 22 different crop types such as staple cereals (rice, maize), pulses (chickpea, lentil, pigeonpeas), fruit (apple, banana, grapes, mango, orange, papaya, pomegranate, watermelon), commercial crops (coffee, cotton, jute), and others - forming a well-rounded data distribution across various categories of agriculture. Such diversity allows the system to make recommendations in a wide range of farming environments and local conditions. Exploratory initial analysis validated data quality with no missing values or duplicates found. Distribution analysis showed different ranges for different parameters, indicating the need for standardization in preprocessing. Correlation heat mapping indicated strong correlations between some environmental factors and certain crop suitability patterns, which were useful for the next modeling stage. The structure of the dataset defines a well-defined supervised learning problem, correlating input environmental variables with target crop categories, and is therefore perfectly appropriate for training classification models that can apply these relationships to novel field conditions.Table 1Description of Instances in Dataset

Table 3.1 Description of Instances in Dataset

S. No	Instance	Description
1.	Nitrogen (N)	Soil macronutrients are essential for leaf growth and plant proteins, measured in ratio values. Range spans from 0 to 140.
2.	Phosphorus (P)	Soil macronutrient critical for root development and flowering, measured in ratio values. Range spans from 5 to 145.
3.	Potassium (K)	Soil macronutrient is important for overall plant health and disease resistance, measured in ratio values. Range spans from 5 to 205.
4.	Temperature	Environmental factors affecting plant metabolism and growth rate, measured in degrees Celsius. Range spans from 8.8 to 43.7.
5.	Humidity	Environmental factors influencing transpiration and water requirements, measured as percentage. Range spans from 14.3 to 99.9.
6.	pH	Soil acidity/alkalinity level affecting nutrient availability, measured on standard pH scale. Range spans from 3.5 to 9.9.

7.	Rainfall	Environmental factors determining water availability, measured in millimeters. Range spans from 20.2 to 298.6.
8.	Label	Target variable indicating optimal crop for given conditions. Contains 22 distinct crop varieties encoded numerically from 1 to 22.

4 System Analysis

The system analysis for this project involves understanding the agricultural needs for crop recommendations based on soil and environmental parameters, gathering requirements for an ML-based advisory system, designing a predictive model architecture using classification algorithms, implementing multiple models with feature preprocessing techniques, and thoroughly evaluating performance to select the optimal Random Forest algorithm for deployment. The system transforms raw agricultural input data (N, P, K, pH, temperature, humidity, rainfall) into actionable crop recommendations for farmers.

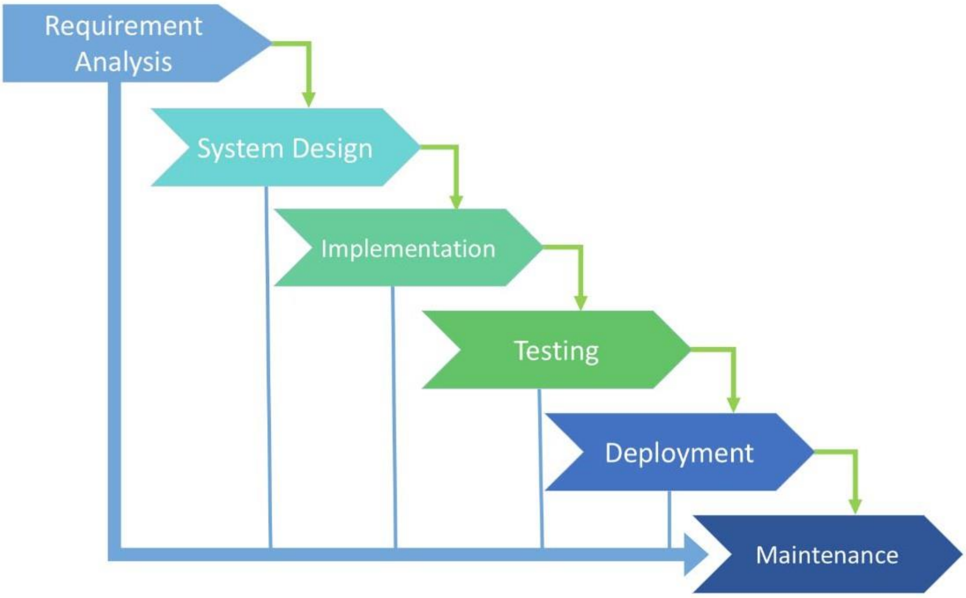

Figure 4.1 Software Development Life Cycle

The system analysis for this project involves understanding the problem statement, gathering requirements, designing the system, implementing the system, testing the system, and deploying the system.

4.1 Requisites Accumulating and Analysis

During this stage, the requirements of the project are obtained via communication with the stakeholders, such as farmers and agricultural specialists. The structure and nature of the crop recommendation dataset are investigated. Requirements for preprocessing data, such as the management of missing values, duplicate checking, and numerical encoding of crop labels, are determined. The analysis entails gaining a deep knowledge of agriculture domain knowledge, recognizing the significant soil parameters (N, P, K, pH) and environmental conditions (temperature, humidity, rainfall) affecting crop suitability. Exploratory Data Analysis (EDA) is conducted to understand the feature distributions, relationship between the soil nutrients and the environmental conditions, and possible issues in predicting best crop recommendations.

4.2 System Design

The framework is broken down into multiple phases, beginning with data collection and preprocessing, which cleans and normalizes raw crop data in preparation for analysis. Important parameters influencing crop productivity are then revealed using feature selection and visualization techniques. To choose the best machine learning model, a number of them are then trained and verified. Finally, based on predetermined input parameters, the best-performing model is implemented to recommend suitable crops.

4.3 Implementation

Implementation was initiated with structuring the dataset and performing preprocessing operations such as missing value handling and feature scaling. Several models like Random Forest, SVM, and Decision Tree were trained over the preprocessed data. The performance of models was evaluated in terms of parameters such as accuracy, precision, and F1-score to measure reliability. The most effective model was then utilized in a well-structured prediction system that advises crops on providing environmental inputs.

4.4 Testing

To estimate the expected accuracy of the trained models, the testing phase was conducted to evaluate them against an alternate test dataset. The strengths and weaknesses of each algorithm were measured based on performance metrics such as accuracy, precision, recall, F1-score, and confusion matrix. Graphs such as bar charts and heatmaps helped interpret the data better. To ensure its reliability and deployment readiness, the top-performing model was validated further.

4.5 Deployment

During deployment, the machine learning model, once trained, is added to a user-friendly interface for easy access by farmers as well as agricultural specialists. The system is then hosted on a suitable platform to enable it to process real-time inquiries and make crop recommendations according to the inputs of users. Deployment involves the deployment of backend services, maintaining the accuracy of the model response, and offering secure data management. Correct documentation is prepared to help users navigate through the features of the system. The solution is implemented in a live environment to test how it will run in real conditions.

4.6 Maintenance

The maintenance cycle of the Crop Advisory System involves frequently updating soil, weather, and crop data to provide precise recommendations. With changing agricultural practices and seasonal trends, the system is updated accordingly. Any problem faced by the users is resolved quickly to provide reliability. The machine learning model is regularly retrained with new data to improve performance. Ongoing feedback from farmers and experts makes the system more functional and easier to use over time.

4.7 Methodology

The approach adopted in this project is a defined pipeline to form an efficient crop advisory system. First, relevant agricultural data covering soil nutrients, weather, and crop types is gathered from trustable sources. The data then undergoes preprocessing methods like managing missing values, normalization, and label encoding by cleaning and converting it. The cleaned dataset is employed for training machine learning algorithms such as Random Forest, SVM, KNN, and XGBoost, with each being trained on patterns from the features to make precise crop predictions. The models are tested using unseen data after being trained in order to confirm their performance. The models are then compared using different evaluation metrics in order to determine which of them performs best and can be deployed in real-world scenarios.

4.7.1 Gathering Data

The first stage of the project is to collect a balanced dataset containing key agricultural parameters like Nitrogen (N), Phosphorus (P), Potassium (K), pH value, temperature, humidity, and rainfall. These are key factors that determine the appropriate crops for an environment. The data is obtained from authentic sources such as agricultural research institutions, open government databases, and academic datasets. Efforts are made to ensure the data is accurate, current, and regionally applicable. A balanced and varied dataset enables the model to learn better and enhances its capacity to generalize across conditions. This groundwork is crucial in building a credible and data-driven crop advisory system.

4.7.2 Data Preprocessing

Following the gathering of raw data, data preparation is an essential step that prepares the dataset for analysis by cleaning it. This includes identifying and managing null or missing data that may otherwise skew the learning process. Duplicate records are eliminated during model training in order to prevent bias and redundancy.

In order to enable more effective algorithm convergence, numerical features are scaled using the proper scaling techniques such that all of the values fall within a comparable range. Categorical data is converted into numerical values that machine learning algorithms can easily understand through the use of label encoding. The general consistency and quality of the dataset are enhanced by this meticulous preparation, giving the model a solid base.

4.7.3 Training the Model

Once preprocessed, the dataset is divided into training and testing subsets. The training data are utilised to train various machine learning algorithms such as Random Forest, SVM, KNN, and XGBoost. Each model is trained to learn patterns and relationships among input features and the suitability of crops.

4.7.4 Testing the Model

A different testing dataset that was not utilized for training is used to assess the models after they have been trained. Understanding the models' ability to process novel, unknown input and generate precise predictions depends on this stage. We can identify problems like overfitting, in which the model learns training data by heart but fails on fresh inputs, or underfitting, in which the model is unable to recognize patterns in the data, by examining how well they perform on the test set. Through testing, the model's accuracy on paper and dependability in real-world applications are confirmed.

4.7.5 Evaluating the Model

Model performance is evaluated using key metrics such as accuracy, precision, recall, and F1-score. A comparative analysis of the results from each algorithm helps determine the most effective model for crop recommendation. Visualizations such as confusion matrices and performance graphs are also used for deeper insights.

4.7.5.1 Evaluation Metrics

Evaluation measures are required to understand how well a machine learning model can make predictions. In this project, measures such as Accuracy, Precision, Recall, and F1-Score are utilized to gain a holistic understanding of the strengths and weaknesses of the model. Accuracy indicates the overall accuracy of the model, whereas precision indicates the number of predicted positives that are actually correct. Recall measures the model's capacity to mark all instances of relevance, and the F1-Score weighs precision and recall, which is particularly helpful when used with imbalanced datasets. These are utilized in comparing more than one algorithm and choosing the algorithm with the highest performance for crop recommendation.

4.7.5.1.1 Accuracy

Accuracy is the percentage of total accurate predictions by the model out of all predictions. In the context of the crop advisory system, it informs us about how many times the model accurately predicts the most appropriate crop given the available soil and climatic conditions. Accuracy is a good place to start, but accuracy alone won't be good enough if the dataset is not balanced.

4.7.5.1.2 Precision

Precision is concerned with the accuracy of positive predictions. It estimates how many crops it predicted as good actually were good. A high precision implies the model is cautious and makes recommendations for crops only when it is quite sure, which is very important when incorrect crop recommendations result in losses or poor harvests for farmers.

4.7.5.1.3 Recall

Recall, or sensitivity, is a measure of how well the model can identify all the crops that actually qualify. Here, it serves to prevent the system from missing any good crops. A high recall value indicates that the model is good at capturing all correct recommendations, albeit with some false ones.

4.7.5.1.4 F1- Score

The F1-Score is the harmonic mean between precision and recall and provides a balanced representation whenever there is a compromise between the two. For the crop advisory system, F1-Score ensures that both the accuracy of predictions (precision) and completeness of predictions (recall) are considered. This is especially critical whenever the dataset contains several crop classes or slight imbalances.

5 Design

The crop advisory system employs a modular and layered strategy for maintenance, scalability, and efficient implementation. It begins with an easy-to-use interface for entering agricultural parameters, which are processed through a data preprocessing module. The data is then passed to machine learning models such as Random Forest, SVM, KNN, and XGBoost, which are trained on historical data. The top-performing model is used for crop recommendation, which is then notified to the user.

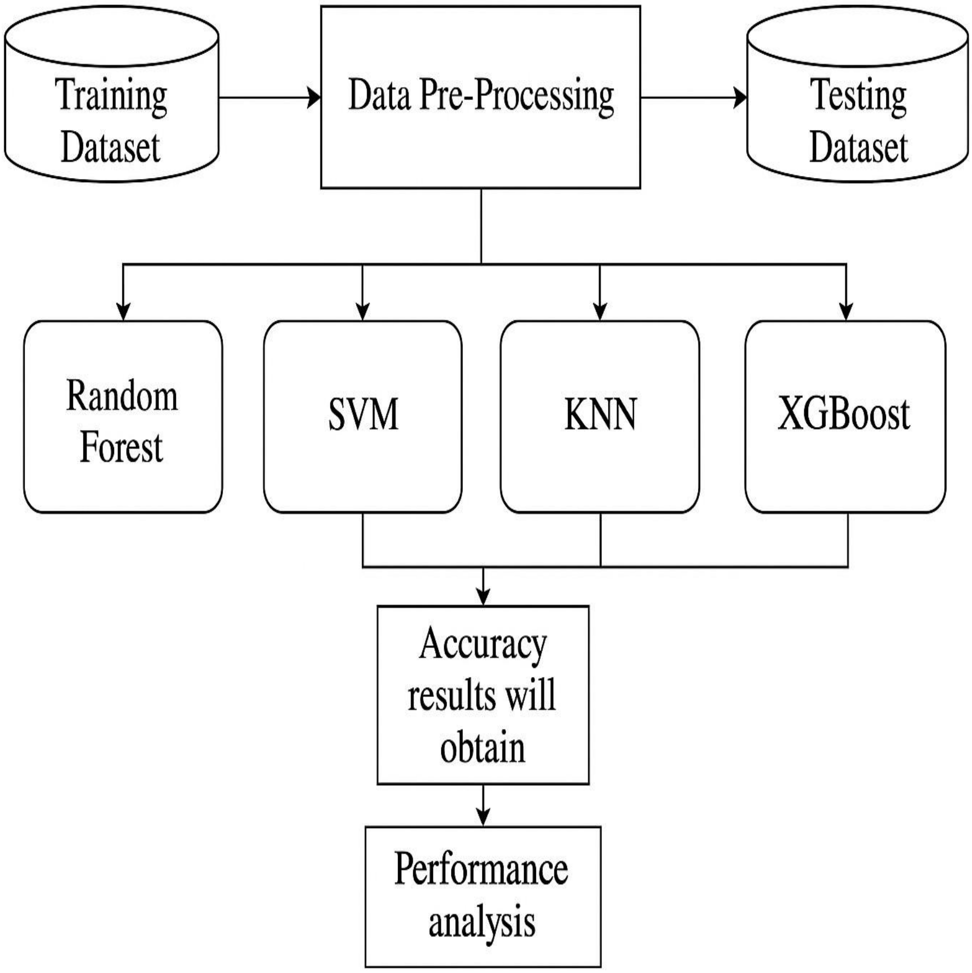

Figure 5.1 System Architecture

5.1 Use Case Diagram

Use-case diagrams in Crop Advisory System emphasize the major functionalities and interactions among users (e.g., farmers or agricultural officers) and the system. These diagrams explain how users provide critical agricultural parameters such as soil nutrients (N, P, K), pH level, temperature, humidity, and rainfall to the system. The system processes data from these inputs and applies machine learning models (Random Forest, SVM, KNN, XGBoost) to produce precise crop recommendations. The figure also exhibits other possible application scenarios such as reviewing previous suggestions, getting informed on best agricultural practices, and getting access to forecast outcomes. Overall, it establishes the scopes and functions of every component to provide a valid advisory solution.

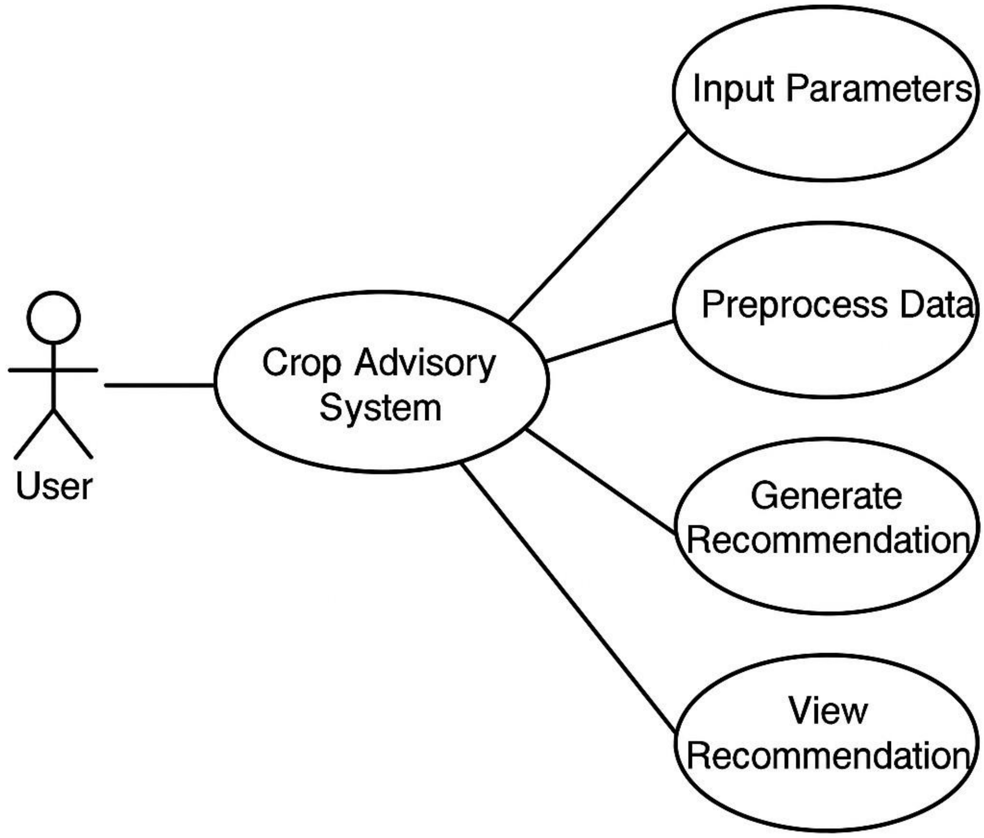

Figure 5.2 Use Case Diagram

5.2 Class Diagram

The structural organization of the Crop Advisory System is illustrated in the class diagram, which enumerates the major components and their relationships. The CropAdvisor class, the main controller for handling user input and model execution, interacts with the User class. Supporting classes like DataPreprocessor and MLModel manage data cleansing and machine learning logic, respectively. To promote modularity, every model from Random Forest, SVM, KNN, to XGBoost is defined as a subclass of MLModel. Due to the separation of concerns facilitated by such organization, the system is scalable and easy to manage.

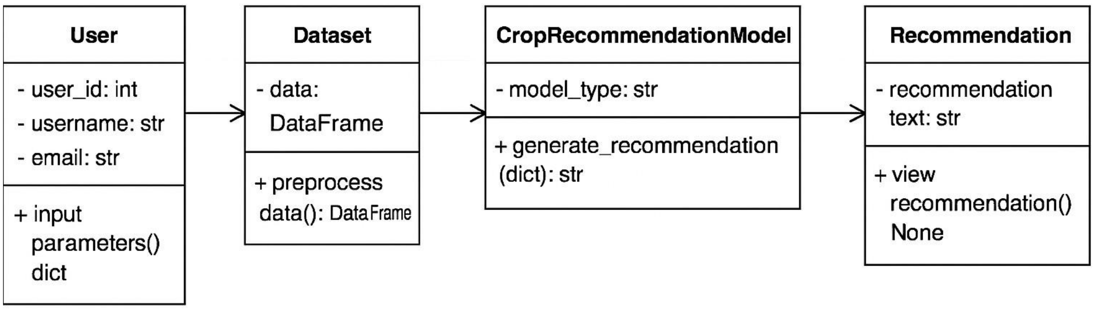

Figure 5.3 Class Diagram

5.3 Activity Diagram

The crop advisory system activity diagram outlines the exact actions taken from data entry to the final crop recommendation. It begins when a user enters agricultural factors, such as weather and nutrient levels. Prior to being sent to a preprocessing module for data scaling and cleaning, the inputs are verified. Machine learning algorithms trained on the qualities are used in the model selection process to assess the processed data. The system determines the optimal crop based on the evaluation and gives recommendations in line with that determination.

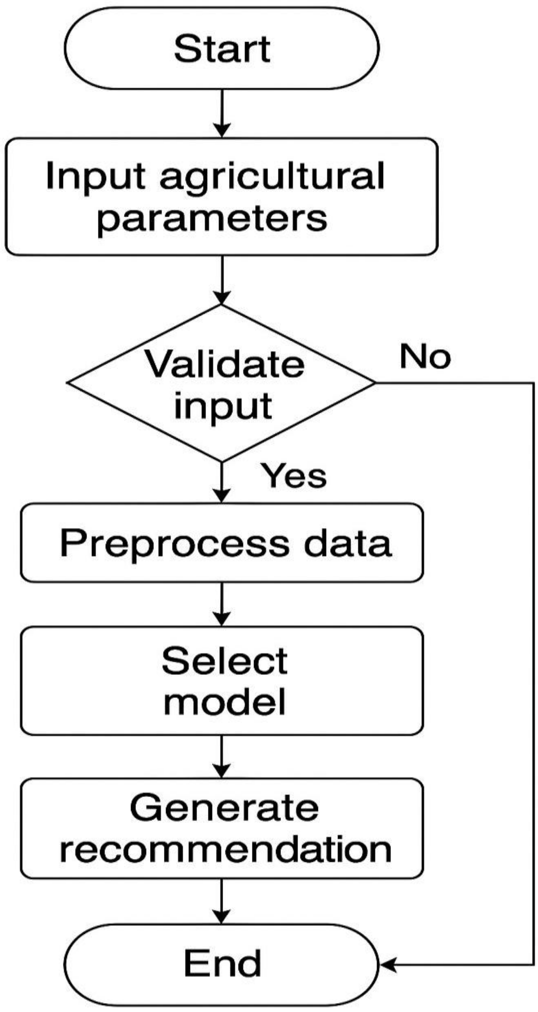

Figure 5.4 Activity Diagram

5.4 Sequence Diagram

The sequence diagram defines the order in which the user interacts with the system in a crop recommendation process. The process starts when the user provides agricultural parameters, which initiate the system to preprocess and validate the data. Once preprocessing is successful, the system chooses an optimal model and produces an optimal crop recommendation. The entire step is performed in a predefined sequence, indicating how the data moves and actions are initiated in terms of time between the user and system entities.

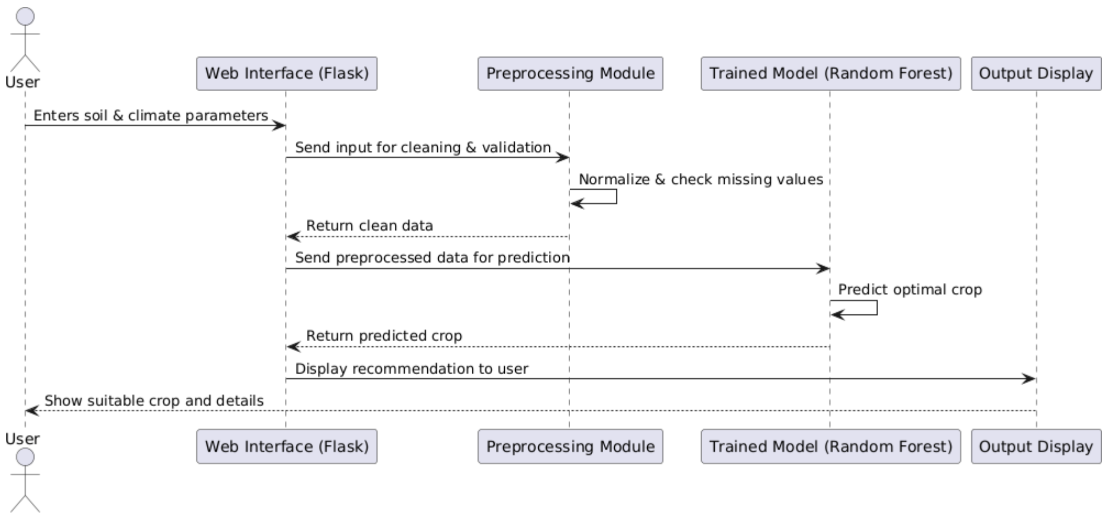

Figure 5.5 Sequence Diagram

5.5 Collaboration Diagram

The collaboration diagram illustrates the way various parts of the crop recommendation system interact to serve user requests. It highlights the structural organization of objects and message exchange sequence. Each player, e.g., the user interface, validation module, data processor, and model selector, is positioned to represent direct lines of communication. The diagram efficiently conveys how duties are divided and each segment functions together in coordination to produce the final recommendation.

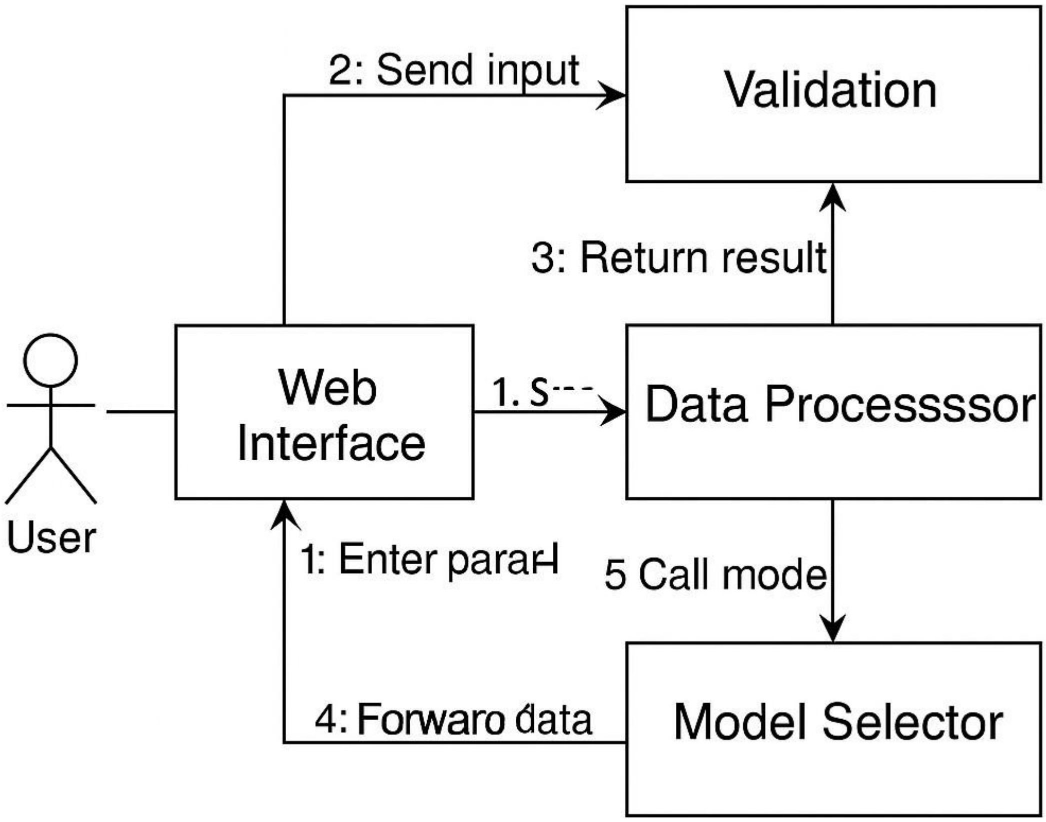

Figure 5.6 Collaboration Diagram

5.6 State Chart Diagram

By defining its phases and transitions, the state chart graphic illustrates the crop recommendation system's dynamic behavior. Depending on user engagement, it proceeds through preprocessing and model execution after starting in an idle state and proceeding to data input and validation. Every transition shows how the system reacts to both internal and external events and is event- or condition-driven. An overview of how the system changes as a result of operations during its life cycle is given by this graph.

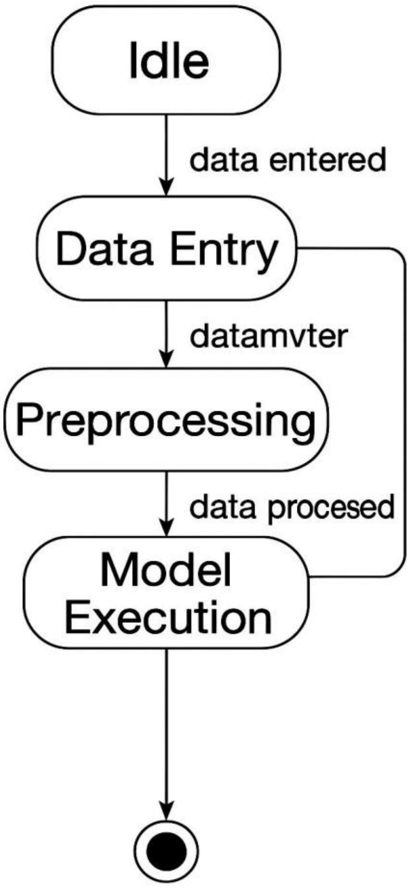

Figure 5.7 State Chart Diagram

6 Requirements

We developed an intelligent crop advisory system that leverages agricultural parameters to recommend suitable crops. By analyzing key features such as soil nutrients, weather conditions, and environmental factors, the system can predict the most appropriate crop for a given region. This approach supports informed decision-making for farmers, ultimately aiming to enhance agricultural productivity and resource management.

6.1 Software's and Libraries Used in the Project

To construct and deploy this system successfully, a range of software tools were applied, each performing a precise function during various phases of the development life cycle. They include programming environments, machine learning libraries, and visualization tools. They all work together towards smooth data management, model training and assessment, and result display. Each software was chosen based on its effectiveness, simplicity of use, and how well it supported the needs of the project. Their joint application made the system accurate and convenient to use.

6.1.1 Jupyter Notebook

Jupyter Notebook is an active open-source instrument, which promotes interactive computing and data analysis in an easily sharable and friendly document format. A significant version of Jupyter is Google Colaboratory (Colab), a cloud-based free environment kindly provided by Google. This innovative platform extends the functionality of Jupyter by utilizing the power of cloud computing. With its intuitive interface and collaborative nature, Jupyter Notebook enables With its intuitive interface and collaborative nature, Jupyter Notebook enables users to easily interact in data discovery and computational operations. It accommodates several programming languages, thus being a convenient option for users who prefer different coding options.

6.1.2 Python

Python is a general-purpose, interpreted programming language that happens to be highly programmed at a high level and is renowned for having a philosophy of design that focuses heavily on the readability of code. The language accomplishes this through a special feature called significant indentation that determines the shape of the code in terms of line indentation. This process, sometimes referred to as the "off-side rule," is critical to crafting code that can be read and seen aesthetically. The intention of Python's object-oriented programming (OOP) syntax and philosophy is to assist programmers in writing logical, consistent code. Python is a developer favorite for small and large projects due to its flexibility.

6.1.3 Pandas

It's a library of data manipulation that is applicable for data cleaning and data preprocessing. Pandas offers functionality for data writing and reading, data handling with missing values, and data transformation in other formats. It is mostly utilized in data science and machine learning for data preprocessing and feature engineering.

6.1.4 NumPy

It is a collection of numerical operations that may be used when working with matrices and arrays. High-performance implementations of numerical operations, including Fourier transforms, matrix multiplication, and linear algebra, are provided by NumPy. It is used to handle and alter data in machine learning and data science.

6.1.5 Scikit-learn (sklearn)

scikit-learn is a free software machine learning library for Python. It offers simple and efficient data mining and data analysis tools based on other popular scientific computing libraries such as NumPy, SciPy, and matplotlib. scikit-learn is intended to work well with these libraries in order to support the development of high-performance machine learning applications.

6.1.6 Matplotlib

Matplotlib is a commonly used data plotting library in Python that enables programmers and data analysts to produce numerous static, animated, and interactive plots. For this project, it is key to visually representing the data as well as the model performance. It assists in graphing bars such as confusion matrices, bar charts, and line graphs so that intricate patterns in the data are easier to comprehend. Its adaptability and ability to be easily customized support decision-making and model assessment through well-formatted, clear visuals.

6.1.7 Seaborn

Seaborn is a high-level visualization library on top of Matplotlib that streamlines the task of producing well-informed and good-looking statistical graphics. Seaborn also includes built-in themes and palettes of colors, which add visual appeal to plots. For the purpose of this project, Seaborn is utilized for creating heatmaps, correlation matrices, and distribution plots, thereby revealing patterns and relationships in the data. Its direct handling of DataFrame structures makes it effective in presenting insights during exploratory analysis as well as model evaluation.

6.2 Hardware Requirements

Hardware requirements are necessary to enable the software in the project to function efficiently and effectively. Requirements include the minimal CPU, RAM, storage, and other hardware requirements. The project's software can function easily by satisfying such requirements, preventing crashes, errors, and other problems.

I. 64-bit operating system (Windows, Linux, MacOS)

II. RAM (Minimum 8 GB)

III. CPU

IV. X64 Based Processor

V. Hard Disk (Minimum 128 GB)

VI. Input Devices (Keyboard, Mouse)

6.3 Software Requirements

Successful implementation of such a crop recommending system using machine learning is completely reliant upon well-configured software infrastructure. Each layer of the stack contributes essentially to ensuring system functionality, compatibility, and operation.

I. Software: Python 3.10 or High Version

II. Integrated Development Environment (IDE): Jupyter Notebook

7 Code & Implementation

Python is used to write the project code, with a focus on modular coding and a clean structure for easy maintenance. Using libraries like pandas and numpy, the data is initially preprocessed to ensure consistency and prepare it for model training. Scikit-learn makes use of many machine learning techniques, including Random Forest, SVM, KNN and XGBoost. To determine which model is the most accurate, the models are trained and evaluated using performance metrics. After the model is finished, it is saved to generate crop projections using fresh input data.

7.1 Code

Code for implementation of Data Driven Crop Advisory System.

7.1.1 Importing all necessary Libraries

We begin by importing all the necessary tools and packages. These libraries provide support for data manipulation, visualization, and machine learning..

```
import os

import numpy as np

import pandas as pd

from sklearn import metrics

from sklearn.model_selection import train_test_split

from sklearn import preprocessing

import random
```

```
import seaborn as sns

import matplotlib.pyplot as plt
```

7.1.2 Dataset Loading and Initial Exploration

We load the dataset from a CSV file and explore its shape, size, and sample rows to understand its structure and contents.

```
crop_data = pd.read_csv("Dataset/Crop_recommendation.csv")

# View the first and last few rows to get a glimpse of the data

print(crop_data.head())

print(crop_data.tail())

# Display dataset dimensions

print(f"Total Elements in Dataset: {crop_data.size}")

print(f"Dataset Shape (Rows, Columns): {crop_data.shape}")
```

7.1.3 Data Quality Check

To maintain the quality of the dataset, we check for any missing or repeated values. Removing duplicates and handling nulls (if any) ensures better model reliability.

```
missing_values = crop_data.isnull().sum()

print("Missing Values per Column:\n", missing_values)

duplicates = crop_data.duplicated().sum()

print(f"\nTotal Duplicate Rows: {duplicates}")
```

Get column-wise data type and memory usage information

print("\nDataset Overview:")

crop_data.info()

7.1.4 Statistical Summary and Correlation

We generate statistical insights and a correlation matrix to examine the linear relationships between numerical features.

correlation_matrix = crop_data.drop(columns=['label']).corr()

print("\nFeature Correlation Matrix:\n", correlation_matrix)

Visualize correlation matrix using heatmap

plt.figure(figsize=(10, 8))

sns.heatmap(correlation_matrix, annot=True, cbar=True, cmap='coolwarm')

plt.title("Feature Correlation Heatmap")

plt.show()

7.1.5 Label Encoding

Since crop names are text-based, we convert them into numeric codes. This step is required as machine learning algorithms work with numerical inputs.

crop_mapping = {

'rice': 0, 'maize': 1, 'jute': 2, 'cotton': 3, 'coconut': 4, 'papaya': 5, 'orange': 6,

'apple': 7, 'muskmelon': 8, 'watermelon': 9, 'grapes': 10, 'mango': 11, 'banana': 12,

```
    'pomegranate': 13, 'lentil': 14, 'blackgram': 15, 'mungbean': 16, 'mothbeans': 17,
    'pigeonpeas': 18, 'kidneybeans': 19, 'chickpea': 20, 'coffee': 21
}
crop_data['label'] = crop_data['label'].map(crop_mapping)
```

7.1.6 Splitting the Dataset

The data is divided into training and testing sets. This allows us to train the model on one portion and validate its performance on unseen data.

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

7.1.7 Training the Random Forest Model

```
rf_model = RandomForestClassifier(n_estimators=10, random_state=10)
rf_model.fit(X_train_std, Y_train)
```

7.1.8 Model Evaluation

```
print(f"Accuracy:  {accuracy_rf:.4f}")
print(f"Precision: {precision_rf:.4f}")
print(f"Recall:    {recall_rf:.4f}")
print(f"F1 Score:  {f1_rf:.4f}")
```

7.1.9 Loading Pickle File

Random Forest model was learned and tested and saved for subsequent reuse. The Pickle package in Python serialized and saved the model in.pkl format so it could be loaded easily for making predictions. Such an approach works best for live applications such as web or mobile interfaces, in which re-learning on the fly is time-prohibitive and not feasible. Pickle keeps the model consistent and available at all times, which makes the project scalable and user-friendly.

7.1.10 Building a User Interface with Flask

To provide the users with the simple and interactive way to use our crop suggestion model, we built a web interface with Flask, which is an open-source lightweight web framework written in Python.

Flask makes it possible for us to take our Python program and turn it into a live web application with very little configuration. In this case, it allows the user to interact with the trained Random Forest model. As a user provides soil and climate details through the web form, Flask picks up the details, processes them, passes them to the model, and then presents the predicted crop in real time.

The good thing about Flask is that it's flexible and fast. It helped us to keep our app simple, efficient, and deployable. Because we had limited routes and HTML templates, we could integrate our machine learning code into a straightforward front end so that the tool could be accessed by non-tech people or individuals who don't have any knowledge of technicality.

8 Results

Through detailed assessment and comparison of various machine learning models, the Random Forest classifier was determined to be the best for crop prediction. The model repeatedly returned to high accuracy, demonstrating its ability to detect subtle patterns in the data between soil nutrients, temperature, humidity, and other environmental conditions. While testing, it worked strongly on unseen samples, indicating good generalization and minimal overfitting. In addition, vital performance indicators such as precision, recall, and F1-score justified its reliability.

8.1 Parameter Entry Portal

The Parameter Entry Interface enables farmers to enter key soil composition information (N, P, K, pH) and environmental factors (temperature, humidity, rainfall) within ranges derived from the dataset. This simple form captures all agricultural parameters needed by the machine learning model to predict the best crop for the conditions, with efficient user experience for agricultural decision support.

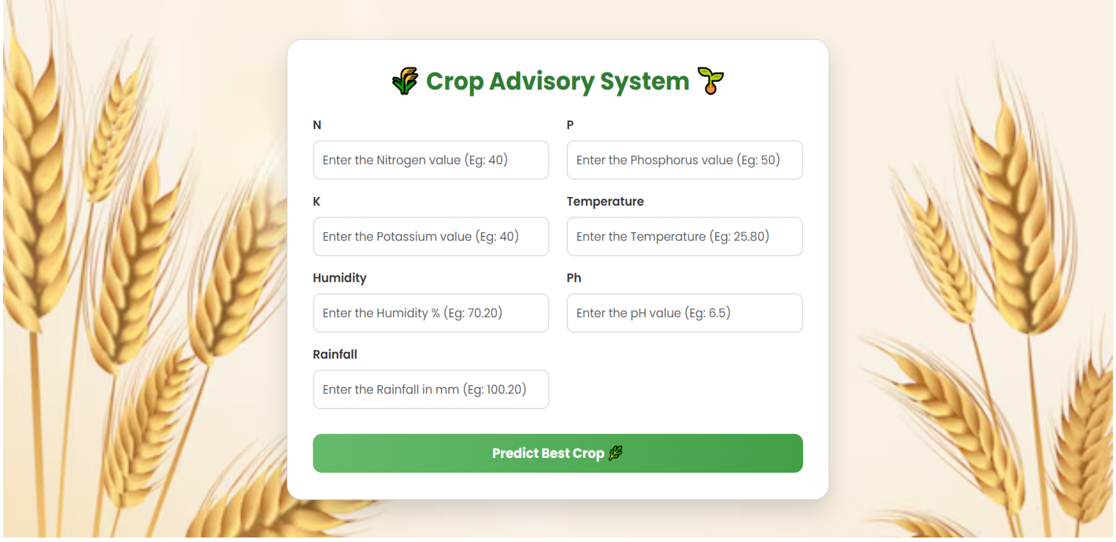

Figure 8.1 Parameter Entry Portal

8.2 Data Input Dashboard

This interface records important agricultural parameters for the machine learning model to process. Users input specific values for soil nutrients, environmental conditions and soil characteristics. These exact measurements allow the Random Forest algorithm to examine multiple agricultural variables at once, providing scientifically supported crop recommendations optimized for the particular growing conditions.

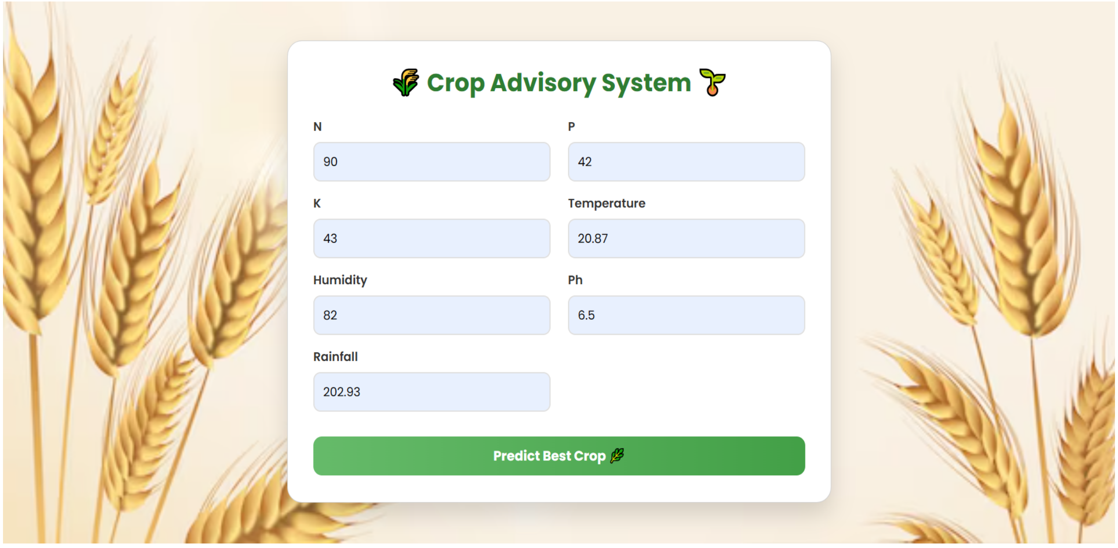

Figure 8.2 Input Dashboard

8.3 Crop Recommendation Result

The outcome of the machine learning model is shown in the prediction interface according to the user-inputted agricultural parameters.

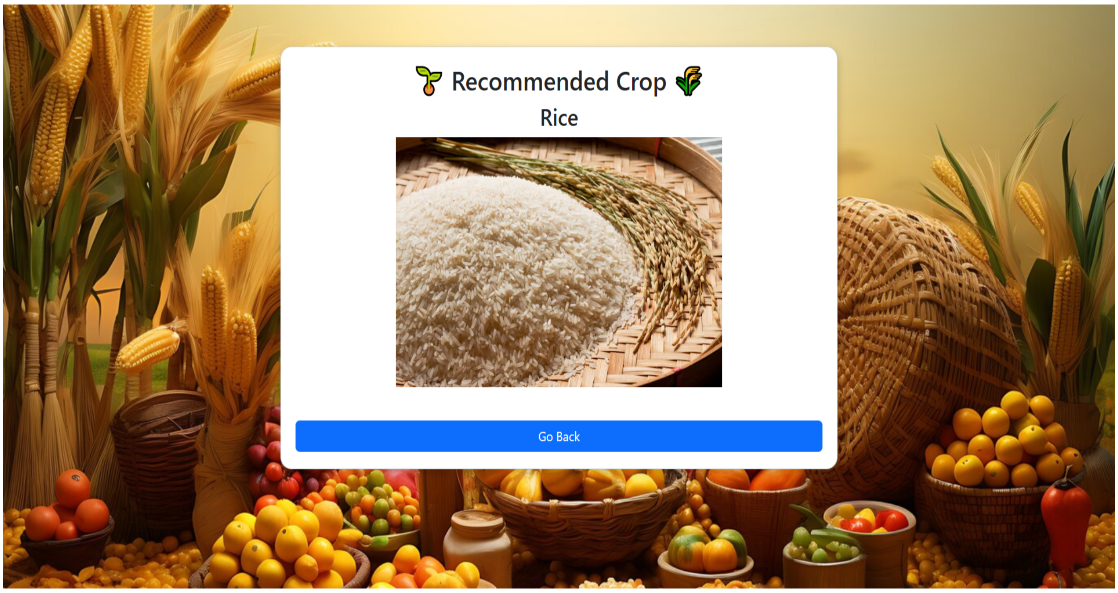

Figure 8.3 Crop Recommendation Result

8.4 Performance Analysis of Models

A variety of machine learning techniques were applied and tested on the dataset to determine the optimal model for crop recommendation. Metrics such as accuracy, precision, recall, and F1-score were utilized to compare techniques such as Support Vector Machine (SVM), K-Nearest Neighbors (KNN), XGBoost, and Random Forest. Random Forest consistently yielded the best results overall, even though each model had strengths. It could enhance generalization and reduce overfitting by combining multiple decision trees in an ensemble method. Besides showing how different models handled the data, this comparison justified Random Forest's choice based on its superior performance and stability balance on new, unseen inputs.

Table 8.1 Performance Analysis of Models

S. No	Algorithm	Accuracy	Precision	Recall	F1-Score
1.	Random Forest	98.87%	99.02%	99.87%	98.65%
2.	SVM	98.02%	97.19%	98.02%	97.52%
3.	KNN	95.76%	95.15%	95.76%	95.24%
4.	XGBoost	98.31%	98.48%	98.31%	98.20%

8.5 Algorithm Performance Comparision

The Crop Advisory System uses four machine learning models, with Random Forest showing the best performance in predicting crops. Its accuracy and precision are 99.55%, 99.58%, recall, and F1-score, making it the best choice for crop prediction. XGBoost follows with 98.64% accuracy and 98.76% precision. SVM and KNN perform competitively but with lower measurement values.

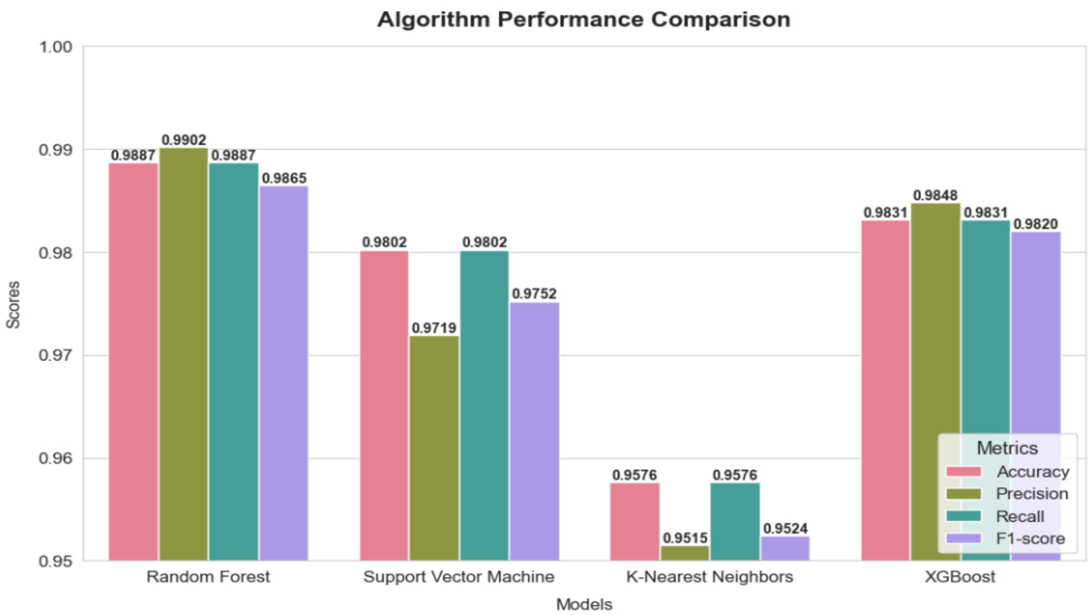

Figure 8.4 Algorithm Performance Comparison

9 Conclusion

The application of machine learning in the project provided the solution for crop recommendation based on the most important agricultural features, such as soil nutrition, temperature, humidity, and rainfall. Experimenting different classification methods like KNN, SVM, XGBoost, and Random Forest was our effort to see which algorithm could produce the most stable and accurate outcome. The best performer, following several rounds of testing and verification, was Random Forest, delivering the best balance of accuracy, generalization ability, and resistance to overfitting. Its ensemble method, which aggregates the output of multiple decision trees, enabled it to extract fine-grained relationships in data that simpler models may overlook.

In addition to achieving excellent predictive precision, the model was also shown to perform well when tested on unseen data, indicating good generalizability to real-world agricultural settings. The visualizations and performance metrics also supported its stability and reliability, thus being a good candidate for practical implementation. In general, this project demonstrates how machine learning and data science can be used to enable better-informed decision-making in agriculture, enabling farmers to choose crops suitable for their conditions and ultimately contributing to more productive and sustainable farming practices.

10 References

[1] Allam, S., Manjusha, B., Vuyyuru, A., Veeranjaneyulu, P., Rani, K. P., & Mishra, S. (2024, January). Agriculture Crop Recommendation System Using Machine learning. In *2024 International Conference on Advancements in Smart, Secure and Intelligent Computing (ASSIC)* (pp. 1-5). IEEE.

[2] Sri, B. S., Pavani, G., Sindhuja, B. Y. S., Swapna, V., & Priyanka, P. L. (2023, March). An Improved Machine Learning based Crop Recommendation System. In 2023 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS) (pp. 64-68). IEEE.

[3] Balakrishnan, D., Kumar, A. P., Reddy, K. S. K., Kumar, R. R., Aadith, K., & Madhan, S. (2023, June). Agricultural crop recommendation system. In *2023 3rd International Conference on Intelligent Technologies (CONIT)* (pp. 1-5). IEEE.

[4] Ghime, Prof & Kshirsagar, Akshay & Lohakare, Shahuraj & Quadri, Fardeen & Kare, Vikas. (2024). Crop Recommendation System Using Machine Learning. IJARCCE. 13. 10.17148/IJARCCE.2024.13476.

[5] Shwetha, A. N., & PrabodhC, P. (2024, June). Machine Learning based Smart Crop Recommendation System. In *2023 4th International Conference on Intelligent Technologies (CONIT)* (pp. 1-6). IEEE.

[6] Desai, M., & Ansari, N. (2023, August). An Innovative Method to Increase Agricultural Productivity using Machine Learning-based Crop Recommendation Systems. In *2023 Second International Conference on Augmented Intelligence and Sustainable Systems (ICAISS)* (pp. 645-651). IEEE.

[7] Priyadharshini, A., Chakraborty, S., Kumar, A., & Pooniwala, O. R. (2021, April). Intelligent crop recommendation system using machine learning. In *2021 5th international conference on computing methodologies and communication (ICCMC)* (pp. 843-848). IEEE.

[8] Das, P., Pal, A., Mullick, R., Singha, S., & Nath, S. (2024, November). Enhanced Crop Yield Prediction by Machine Learning Techniques. In *2024 IEEE International Conference of Electron Devices Society Kolkata Chapter (EDKCON)* (pp. 1-6). IEEE.

[9] Lokhande, A., & Dixit, M. (2022). Crop Recommendation System Using Machine Learning. *International Research Journal of Engineering and Technology (IRJET)*, *9*(5), 3584-3590.