A Project Report on

Customized Crop Solution for Farmers Considering the Soil and the Climate

Submitted in partial fulfillment for award of

Bachelor of Technology

Degree

in

Computer Science and Engineering

By

A. Sowmya (Y21ACS403) **A. Vinay Datta (Y21ACS409)**

E. Rishitha (Y21ACS444) **D. Venkata Naga Sai(Y21ACS439)**

Under the guidance of
Dr. D. Kishore Babu, M.E., Ph.D
Associate 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

<u>CERTIFICATE</u>

This is to certify that the project report entitled **Customized Crop Solution for Farmers Considering the Soil and the Climate** that is being submitted by A. Sowmya (Y21ACS403), A. Vinay Datta (Y21ACS409), E. Rishitha (Y21ACS444), D. Venkata Naga Sai (Y21ACS439) 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
Dr. D. Kishore Babu
Assoc. Prof.

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.

A. Sowmya (Y21ACS403)
A. Vinay Datta (Y21ACS409)
E. Rishitha (Y21ACS444)
D. Venkata Naga Sai (Y21ACS439)

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 **Dr. D. Kishore Babu**, Assoc. Prof, Department of CSE, for his 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.

A. Sowmya (Y21ACS403)

A. Vinay Datta (Y21ACS409)

E. Rishitha (Y21ACS444)

D. Venkata Naga Sai (Y21ACS439)

Table of Contents

List of Figures

List of Tables

Abstract

The last few years have seen considerable changes in agriculture owing to the use of data-oriented technologies. The global demand for food is on the rise due to climate changes, resource limitations, and soil degradation. Thus, there is a desperate need to make great advances in agricultural productivity in a sustainable manner. While time-tested, conventional farming practices prove inadequate to address the intricacies and vagaries of modern agriculture.

Machine learning and data analytics begin to look like potent allies for farmers, allowing them to make informed decisions concerning their environmental and soil conditions. These technologies permit precision agriculture to recommend crops, predict yields, and optimize resource use through large datasets. Supported by historical data and real-time field inputs, efficient agricultural systems can increase efficiency, minimizing waste, and improving crop management in real time subsequently.

The current project reviews the appreciable link of machine learning in agriculture, which has the potential to change the conventional decision-making processes in crop selection and yield prediction. With the digital transformation era, agriculture is more and more adopting data-driven approaches to improve productivity and sustainability. This project introduces a complete machine learning-based framework for crop recommendation and yield prediction as support for farmers and agricultural planners to make wise decisions.

Keywords: Crop recommendation, yield estimation, machine learning, agriculture analytics, precision farming, supervised learning, decision support system, crop productivity, smart agriculture, sustainable agriculture.

1 Introduction

In the face of increasing population, limited agricultural land, and climate variability, the agricultural sector is under immense pressure to boost productivity in a sustainable manner. Traditional farming practices are often guided by experience and intuition rather than data-driven decisions, which can lead to inefficient use of resources and poor yield outcomes. As a result, farmers frequently face challenges in selecting the most appropriate crops for their soil and environmental conditions and in predicting the likely yield of those crops.

The integration of modern technologies such as machine learning and data analytics into agriculture has led to the development of smart farming tools that can optimize crop production. These tools can leverage historical data and environmental parameters to recommend crops best suited for a given condition and to estimate expected yield. By adopting such intelligent systems, farmers can make informed decisions that reduce input waste, improve crop selection, and enhance yield predictability.

1.1 Background

India, being an agrarian economy, heavily relies on agriculture for food security and economic stability. However, farmers often lack access to modern tools and expertise to make accurate decisions about crop selection and yield forecasting. This gap results in poor crop choices, inefficient resource use, and unpredictable harvest outcomes.

The rapid development of machine learning techniques and the growing availability of agricultural data have created an opportunity to transform traditional farming into precision agriculture. Crop recommendation systems help determine which crop is most suitable based on soil and weather parameters, while yield prediction systems assist in estimating the quantity of produce a farmer can expect from a given area under specific conditions.

The availability of diverse environmental and agronomic data—such as soil composition, temperature, humidity, rainfall, and fertilizer usage—has opened the door to intelligent decision-support systems. By processing this data using machine learning algorithms, it is possible to build models that assist farmers in achieving higher productivity and sustainability.

This project brings together both aspects—recommendation and prediction—into a unified framework accessible via a user-friendly web interface. The recommendation model uses classification algorithms trained on soil and weather data, while the yield prediction model employs regression techniques to forecast crop output. The system aims to empower farmers with scientific guidance to improve crop planning and overall productivity.

1.2 Problem Statement

Agriculture remains a cornerstone of the global economy, particularly in countries where a large portion of the population relies on it for livelihood. Despite advancements in technology, many farmers continue to make decisions based on intuition, traditional practices, or generic recommendations that fail to account for the specific conditions of their land. This often results in suboptimal crop selection, inefficient use of resources, and poor yield outcomes. The lack of scientific tools accessible at the grassroots level contributes to significant challenges in sustainable agriculture.

One major problem is the absence of an integrated platform that offers both crop recommendation and yield prediction based on precise soil and weather data. While some systems focus on recommending crops, they do not estimate the potential yield, leaving a gap in strategic planning. Conversely, yield prediction models may fail to account for the suitability of the crop to the given soil conditions, which can lead to misleading estimations. Additionally, environmental factors like temperature, humidity, and rainfall vary widely across regions and significantly impact productivity, making it difficult to generalize solutions.

The increasing unpredictability of climate and the degradation of soil quality further exacerbate the issue, demanding the need for intelligent and adaptive systems. Farmers require decision support tools that are data-driven, localized, and capable of learning from patterns in environmental conditions and crop behavior. The challenge lies in developing a solution that not only provides accurate crop recommendations but also forecasts expected yield based on historical and real-time data.

This project addresses these challenges by proposing a dual-system solution—a Crop Recommendation System that uses soil and climate features to suggest suitable crops, and a Crop Yield Prediction System that uses those inputs along with the selected crop to predict yield. This unified, machine learning-powered system bridges the gap between recommendation and prediction, enabling farmers to make informed, data-driven decisions.

1.3 Motivation

Agriculture is not just a vital industry but the backbone of food security and economic stability in many nations. Yet, it is often subject to volatile conditions, resource mismanagement, and outdated farming techniques. The growing demand for food, combined with limited arable land and the adverse effects of climate change, calls for smarter, data-driven agricultural solutions. This project is inspired by the need to enhance agricultural productivity and sustainability by equipping farmers and agronomists with intelligent tools for decision-making. The core motivation stems from various technical, environmental, and societal challenges, as outlined below.

1.3.1 Inefficiency in Resource Utilization

Fertilizers, water, and land are finite resources that must be used judiciously. Misjudging the crop-soil compatibility not only leads to poor yields but also exhausts the soil and depletes resources. An efficient crop recommendation system can prevent such waste and promote more sustainable practices.

1.3.2 Dependence on Traditional Practices

In many rural regions, farmers still rely heavily on ancestral knowledge or guesswork to decide which crops to cultivate. These methods often overlook soil health, weather conditions, or recent advancements in agronomy. Without scientific analysis, this can lead to the cultivation of unsuitable crops, resulting in financial losses and food insecurity.

1.3.3 Unpredictability of Crop Yield

Even when the right crop is chosen, factors such as unexpected weather changes, improper fertilization, and varying soil nutrient content can drastically affect yield. Without a reliable forecast, farmers cannot plan their investments or manage supply chains effectively. Yield prediction models can serve as planning aids for harvest scheduling, market engagement, and logistics.

1.3.4 Need for Localized Agricultural Advice

Generalized farming advice often fails because agricultural conditions vary significantly by region. A data-driven system that takes into account local environmental parameters can provide personalized recommendations that are more effective and applicable on the ground.

1.3.5 Integration of Machine Learning in Agriculture

The advancement in machine learning offers a great opportunity to build predictive and adaptive systems. The ability of ML algorithms to analyze patterns and learn from data presents a promising avenue for automating agricultural recommendations and predictions with high accuracy.

1.4 Objective

The primary objective of this project is to develop a robust and intelligent agricultural decision-support system that integrates both crop recommendation and crop yield prediction based on key environmental and soil parameters. This system aims to assist farmers, agronomists, and agricultural planners in optimizing crop selection and estimating potential yield more accurately, thereby promoting sustainable farming practices and improving productivity.

1.4.1 Development of a Crop Recommendation System

One of the key objectives is to create a machine learning-based model that recommends the most suitable crop for cultivation based on parameters such as nitrogen, phosphorus, potassium levels in the soil, pH value, temperature, humidity, and rainfall. The model should be trained on diverse agricultural data to ensure generalized and reliable recommendations under varying environmental conditions.

1.4.2 Implementation of a Crop Yield Prediction System

Another critical objective is to design a yield prediction model using regression algorithms that can forecast the expected output for a given crop. This model leverages features including soil nutrients, temperature, rainfall, fertilizer usage, and other relevant inputs to estimate the yield in terms of quantity per acre or hectare. The goal is to help farmers make better investment and logistical decisions ahead of the harvest.

1.4.3 Comparative Evaluation of Machine Learning Models

A crucial part of this research is to experiment with and evaluate multiple machine learning algorithms for both classification (for crop recommendation) and regression (for yield prediction). Models like Random Forest, SVM, Logistic Regression, Decision Tree, Naïve Bayes, XGBoost, and others are assessed based on accuracy, mean squared error, and other performance metrics to select the most effective ones for deployment.

1.4.4 Integration into a Web-Based Interface

To ensure the practical utility of the system, the final objective includes integrating the models into a user-friendly web interface. This will allow users to input soil and environmental data and receive real-time crop recommendations and yield predictions without needing technical expertise.

1.5 Significance

The integration of machine learning into agriculture has opened new avenues for enhancing farming practices and decision-making processes. This project holds significant value in the context of modern agriculture, where data-driven insights are becoming increasingly vital for sustainable development, food security, and resource optimization.

One of the most important contributions of this project is its ability to help farmers choose the most suitable crop for their specific land and climatic conditions. In traditional farming, crop selection is often based on past experience or general knowledge, which may not always yield the best results. By using a crop recommendation model trained on historical data that includes soil nutrients, pH, temperature, humidity, and rainfall, this system provides personalized and data-informed recommendations. This leads to better crop-soil compatibility, improved productivity, and minimized risk of crop failure.

In addition to recommending crops, the system also addresses the uncertainty associated with estimating crop yield. Accurate yield prediction allows farmers to plan their harvests, manage storage and transportation, estimate revenue, and make better financial decisions. For policymakers and agricultural stakeholders, yield predictions can also aid in supply chain planning and food distribution strategies.

Moreover, this dual-system project helps bridge the gap between agricultural data and actionable intelligence. By employing a user-friendly interface, even non-technical users can benefit from advanced predictive models without needing expertise in machine learning. This democratization of technology is particularly valuable in rural and under-resourced areas.

Lastly, the project promotes the use of open-source tools and reproducible methodologies, contributing to the research community and encouraging further development in smart agriculture. Its adaptability allows for future scaling, such as integration with IoT-based soil sensors, satellite imaging, or mobile applications, expanding its impact even further.

1.6 Existing System

In the current agricultural landscape, several decision support systems and tools have been developed to aid farmers in choosing suitable crops and estimating potential yield. However, most of these systems either focus on static recommendations based on general agro-climatic zones or provide manual guidelines that lack personalization and adaptability. The existing systems often fail to utilize the full potential of modern data science and machine learning techniques to provide dynamic, accurate, and context-specific recommendations.

Traditional crop selection methods rely heavily on regional crop calendars, empirical knowledge, or agronomist consultation. These methods, while useful, do not account for the specific variations in soil properties, local weather conditions, or micro-climatic changes that can significantly impact crop performance. As a result, the risk of poor yield due to inappropriate crop choice remains high.

When it comes to yield prediction, most conventional approaches are based on linear regression or statistical forecasting methods using limited variables. These models typically lack the ability to learn complex, non-linear relationships between multiple features like nutrient levels, environmental factors, and crop type. In many cases, such models do not scale well across different geographies or crop varieties.

With the raise of Machine Learning technologies, multiple systems of crop recommendation were designed for recommending a suitable crop. Most of these systems were built using Decision trees as the primary recommending model based different parameters of soil types, weather changes, location based, etc.

Furthermore, many existing tools are standalone applications or web-based dashboards with limited accessibility for rural farmers. They often require a steep learning curve or internet connectivity, which may not be consistently available in remote regions.

Additionally, there is often a lack of integration between crop recommendation and yield prediction systems. This fragmentation forces users to rely on separate platforms for making sequential agricultural decisions, leading to inefficiencies and increased uncertainty.

The need for an integrated, intelligent, and easy-to-use system that simultaneously recommends suitable crops and predicts expected yield, based on real-time inputs and machine learning techniques, forms the core motivation behind the development of the proposed system.

2 Literature Review

Much advancement has been attained in literature relating to crop recommendation based on soil attributes. In the opinion of many scholars, soil properties play a vital role in determining crop compatibility to yield crop means of sustainable farming practices. Crop selection is unfit, which is one of the major problems faced by Indian farmers. Agriculture is the major industry in India, which brings about income and jobs. Hence, their production is going to be reduced significantly. Precision agriculture has made it possible for the farmers to mitigate their problems. By using research related to soil properties, soil types, and crop production statistics, it is a method of agricultural production that advises the farmers on what best crops to use for the specific site. Therefore, productivity improves and fewer crops are wrongly selected. [7]

For the farmers to make wise decisions about crop storages, sale, minimum support price declaration, and import and export with the aid of government organizations and researchers, they need a crop yield projection system which was developed by Yogesh Gandge & Sandha [8]. There are several factors, including soil quality and pH, EC, N, P, and K to be taken into account during crop prediction, which makes this method ideal for data mining. Further, data required for this prediction method is large so it is best suited to data mining. This manuscript outlines how both data mining techniques will pave the way for harvesting huge amounts of data into knowledge. The study evaluates different data mining techniques used to estimate agricultural yields. The performance of any crop production prediction system is based on how accurately traits are retrieved and how well the classifier performs. This manuscript presents an overview of agricultural output forecast algorithms, their precision, and recommendations. [11]

Ji-chun Zhao and Jian xin Guo developed agricultural decision support systems to provide both scientific bases for agricultural research and directions for farm output. Big data analytics would pave the way for improving intelligent decision support systems. Researchers and industry developers study the development of intelligent decision-making systems in agriculture. The first entry into agricultural decision systems is presented after discussing frame designation and design process for intelligent decision systems.

India is concerned in agricultural production. For both farmers and the nation, it will prosper. What we have done is help in giving the right seed to the farmer based on the characteristics of the soil and boosting production for the entire nation. Further, we aim to improve by adding additional attributes and yield estimates in data collection. [10]. S. Babu et al. has discussed precision farming in detail, including its prerequisites and planning for its development. [1] concentrated on applying big data analysis in intelligent decision-making systems for agriculture. Their work highlighted how massive datasets collected from various sources-IoT devices, weather sensors, or soil databases, etc.-can be effectively analyzed to provide decision support for crop planning and management. Real-time data processing and visualization tools therefore hold priority in ensuring exactness and efficiency in providing agricultural recommendations. [2] set forth a review over machine learning techniques for prediction of crop yield and stressed on Random Forest, SVM & KNN being effective models suitable for agricultural datasets with prominent challenges like feature selection and data availability. Their work also serves as the background for the implementation of ML-based yield prediction systems. [3] The positive contribution made by Gowtham et al. (2022) was towards introducing a new crop recommendation system that used Decision Tree Classification implemented by Antlion Optimization method, geared towards enhancing its accuracy. This model, by optimizing for feature selection and classification rules, could readily recommend the best-suited crops based on soil as well as environmental parameters. The hybrid method was shown to be more accurate and adaptable than conventional decision tree models across complex agricultural datasets. [4] The study analyzes, in depth, how Decision Trees and SVMs, amongst others, can adequately use machine learning models to carry out crop recommendation based on soil properties, considering soil properties such as nitrogen, phosphorus, potassium, and pH. There is emphasis on the proper pre-processing of data, which will enhance the accuracy of recommendations, thereby contributing to precision agriculture and sustainable farming practices. [5]The work by Singh et al. is concerned with bringing out the machine learning models like Random Forest and SVR for very accurate crop yield predictions, using an integrated soil, climate, and crop data system for more reliable predictions, and emphasizing how data-oriented techniques can be used within sustainable agriculture. The second study entitled, "Crop Recommendation and Prediction System" introduces a new two-model architecture, which integrates a crop recommendation with a yield predictor. The use of soil nutrients and environmental parameters used in this work will optimize agricultural planning. Highly improved machine learning-integrated decision-making for most farmers is focusing on crop choices and forecasting yield.

In recent years, agricultural yield prediction has greatly improved through combining satellite observation with machine learning techniques. Li et al. [12] found that adding variety-specific information and UAV-based remote sensing data significantly improved the predictive accuracy of models for potato yield. Researchers have attempted to improve agricultural productivity using machine-learning techniques. A good example is ManendraSai et al. [13], who reviewed and presented an exhaustive analysis of various machine learning algorithms and their efficacy in predicting suitable crops under different soil and climatic conditions. According to Chen et al. [14], a hybrid model associating machine learning with expert domain knowledge could contribute to more reliable and interpretable crop recommendations. Wang et al. [15], on the other hand, studied biochar and vermicompost for improved soil health, which eventually increased cucumber's yield and quality, demonstrating how soil treatment can affect predictive yield result.

3 Proposed System

To address the challenges in crop selection and yield forecasting, this project proposes a unified system that integrates Crop Recommendation and Yield Prediction using machine learning. The solution comprises classification and regression models that take various environmental and soil parameters as input and deliver highly accurate recommendations and estimates to support informed agricultural decision-making.

3.1 Task

The preliminary task for developing this project is to build a machine learning system through a series of well-structured phases, which can recommend a best suitable crop based on the given soil and weather conditions as well as can predict the yield that can expected based on the soil and weather conditions.

Project Planning and Setup

i. Define project objectives, goals, and deliverables.

ii. Establish a timeline and allocate resources for the project.

iii. Set up development environments and tools required for the project.

3.1.1 Data Collection and Preparation

Collect the dataset for both the crop recommender and yield predictor:

i. For crop recommendation, data including nitrogen, phosphorus, potassium, temperature, humidity, pH, and rainfall was collected.

ii. For yield prediction, parameters such as rainfall, fertilizer use, temperature, and NPK levels were used alongside crop names and actual yield values (Q/acre).

Data cleaning to remove null values, and all features can be converted into suitable numeric formats. Label encoding and feature scaling can be applied for better model convergence.

3.1.2 Model Selection and Development

i. Multiple machine learning models were selected to find the most possible accurate results.

Classification models (for crop recommendation): Naïve Bayes, Random Forest, SVM, Logistic Regression, Decision Tree, and XGBoost.

Regression models (for yield prediction): Linear Regression, Random Forest Regressor, Decision Tree Regressor, SVR, KNN, and XGBoost,.

ii. Develop models using selected algorithms.

iii. Experiment with different models to optimize the performance for more accurate results.

3.1.3 Training and Evaluation

i. Split the dataset into training, and testing sets.

ii. Train the models using the training data.

iii. Evaluate model performance using metrics like accuracy, precision, recall for classification models and MSE, RMSE and R2 for regression models.

iv. Save the best-performing model of both crop recommender and yield predictor using pickle to integrate with the web interface.

3.1.4 Integration and Deployment

Integrate the trained models to application for real-time recommendations and predictions (Django - based web interface).

3.1.5 Testing and validation

Test the system against a variety of data samples , including known and unknown samples to ensure generalization and consistent predictions.

3.1.6 Documentation and Reporting

Create the detailed documentation, including system architecture diagrams, preprocessing steps, evaluation metrics, and visualization results.

i. Document the project workflow, including methodologies, algorithms, and implementation details.

ii. Prepare a comprehensive report summarizing the project objectives, methodology, results and conclusions.

iii. Create user documentation or guides for deploying and using the permission-based malware detection system.

3.2 Dataset

For the current project to datasets have been collected with relevant agricultural features to ensure high-quality predictions and recommendations. The datasets were collected from the open source – Kaggle.

The dataset for the crop recommendation system comprises soil and climatic attributes that are essential for determining the most suitable crop for cultivation in a specific area. It includes features such as Nitrogen (N), Phosphorus (P), and Potassium (K) content in the soil, along with pH level, temperature (°C), humidity (%), and rainfall (mm). The target variable is the crop label, which indicates the most suitable crop type for the given set of conditions.

This dataset contains approximately 2200 samples and covers 22 different crop types, making it well-suited for classification tasks. The dataset is clean and well-balanced, with no missing values or anomalies. This balance is essential to prevent bias in model predictions.

For the yield prediction functionality, a separate dataset was used, tailored toward quantifying crop output based on environmental and cultivation factors. The features in this dataset include Rainfall (mm), Fertilizer usage, Temperature (°C), and primary soil nutrients such as Nitrogen (N), Phosphorus (P), and Potassium (K). The target output is Yield, measured in quintals per acre (Q/acre), representing the productivity of the crop under the given conditions. This dataset contains approximately 110 samples**.**

3.3 Input

The system is designed to accept a comprehensive set of user-provided inputs that represent soil composition and environmental factors critical for agriculture. The inputs are categorized based on the module they serve:

For Crop Recommendation, the following parameters are required:

i. Nitrogen (N): Amount of nitrogen in the soil (mg/kg)

ii. Phosphorus (P): Amount of phosphorus in the soil (mg/kg)

iii. Potassium (K): Amount of potassium in the soil (mg/kg)

iv. pH Level: Acidity or alkalinity of the soil

v. Temperature (°C): Average ambient temperature

vi. Humidity (%): Atmospheric moisture level

vii. Rainfall (mm): Amount of precipitation in the region

For Yield Prediction, the required inputs include:

i. Crop Name: Selected from the output of the recommender or chosen manually

ii. Rainfall (mm): Precipitation levels for the crop season

iii. Fertilizer (kg/acre): Quantity of fertilizer applied

iv. Temperature (°C): Climate temperature during cultivation

v. Nitrogen (N), Phosphorus (P), Potassium (K): Nutrient levels in the soil

These inputs are collected through a user-friendly web interface developed using Django, where values are entered via form fields.

3.4 Output

The system produces two types of outputs corresponding to its two primary modules:

i. Crop Recommendation Output:

Based on the provided soil and weather data, the system returns the most suitable crop to cultivate under those conditions. The result is displayed on the web interface as:

"Recommended Crop: Rice" (example)

ii. Yield Prediction Output:

Once the user submits both environmental and crop-specific details, the system estimates the expected crop yield, expressed in quintals per acre. This output is also displayed clearly to the user as:

"Predicted Yield: 11.50 Q/acre" (example)

The system is designed to be intuitive and informative, allowing farmers or stakeholders to easily access actionable insights that can guide cultivation decisions and resource management.

The application allows the user to choose one of the two modules, either the crop recommender or the yield predictor. The modules provides the corresponding output in the above mentioned format based on the user inputs of soil and the weather conditions.

4 Algorithms

4.1 Introduction to Machine Learning

Machine Learning (ML) is a rapidly growing field that empowers systems to automatically learn and improve from experience without being explicitly programmed. At its core, ML focuses on developing algorithms that can process large volumes of data, identify patterns, and make intelligent decisions. As the agriculture sector faces mounting challenges such as climate change, unpredictable weather conditions, and fluctuating soil fertility, machine learning offers innovative solutions to optimize productivity and sustainability.

In this project, ML techniques are used to analyse complex relationships between soil and environmental parameters and crop performance. By utilizing historical agricultural data, the system can provide accurate crop recommendations and yield predictions, enabling farmers to make data-driven decisions for better outcomes.

Machine Learning models can be classified into several types based on the learning paradigm. Two of the most prominent and foundational types are Supervised Learning and Unsupervised Learning.

4.1.1 Supervised Learning

Supervised learning is the most commonly used ML approach, particularly in real-world applications where historical data is available with clearly labelled outcomes. In supervised learning, the algorithm is trained using a dataset that contains both input features (e.g., soil nutrients, temperature, rainfall) and target labels (e.g., crop name or crop yield). The goal is for the model to learn the mapping function that can predict the target label for unseen input data.

There are two primary types of supervised learning tasks:

i. Classification: This involves predicting a discrete label or category. In the context of our project, classification is used in the Crop Recommendation System, where the input parameters (such as N, P, K, temperature, etc.) are analysed to classify which crop would be best suited for the conditions.

ii. Regression: This involves predicting a continuous numerical value. For our Crop Yield Prediction System, regression models are trained to estimate the expected yield (in tons/hectare or quintals/acre) based on soil properties and environmental factors.

Some common supervised algorithms used in this project include: Random Forest Classifier and Regressor, Support Vector Machine (SVM), Decision Trees, Logistic Regression, Naïve Bayes, XGBoost, Linear Regression.

Each algorithm has its own strengths depending on the complexity, nature of the data, and interpretability requirements.

4.1.2 Unsupervised Learning

Unsupervised learning, on the other hand, is used when the data does **not** have labelled outcomes. Instead of learning from labelled examples, the model tries to explore the underlying structure and patterns of the data. This is particularly useful in exploratory data analysis, anomaly detection, and clustering similar data points.

In agriculture, unsupervised learning can be applied for:

i. Clustering fields based on similar soil profiles to recommend regional crop strategies.

ii. Identifying patterns in farmer practices or crop behaviour under different climate conditions.

iii. Reducing dimensionality in datasets with many input features using techniques like PCA (Principal Component Analysis), which helps visualize and understand complex datasets better.

While unsupervised learning is not a core component in this version of the system, it holds promising potential for future extensions, such as regional crop zoning or detecting outlier weather patterns.

4.1.3 Role of Machine Learning in Agriculture

Machine Learning bridges the gap between raw data and actionable insights. By leveraging ML in this project:

i. Farmers can predict the most suitable crop for their land based on scientific analysis.

ii. They can also estimate the expected yield, allowing better planning and risk mitigation.

iii. Agricultural experts can analyse the performance of different models to continuously refine the recommendations.

Ultimately, integrating ML with agriculture transforms traditional farming into smart farming, which is data-driven, sustainable, and efficient.

4.2 Algorithms for Crop Recommendation (Classification Models)

The crop recommendation module of this system is essentially a classification problem, where the goal is to assign a specific crop label based on given soil and weather parameters. Several supervised machine learning algorithms were evaluated for this task to identify the most accurate and efficient one.

The key input features used for crop classification include: Nitrogen (N), Phosphorus (P), Potassium (K), Temperature (°C), Humidity (%), pH Value of soil, Rainfall (mm).

The classification algorithms explored in the project are detailed below:

4.2.1 Naïve Bayes Classifier

The Naïve Bayes classifier is a probabilistic model based on Bayes' Theorem, assuming that features are conditionally independent given the class label. Despite its simplicity, it performs surprisingly well in many real-world applications, especially when dealing with high-dimensional data.

In the field of agricultural analytics, classification models play a critical role in recommending crops based on various environmental and soil conditions. One of the fundamental algorithms used for classification is Naïve Bayes. This probabilistic model applies Bayes' Theorem to compute the likelihood of a given crop being suitable based on the input features such as nitrogen, phosphorus, potassium levels, temperature, humidity, rainfall, and soil pH. It operates under the assumption that all input features are conditionally independent, which, despite being a simplification, allows the model to perform remarkably well in many real-world applications due to its computational efficiency and robustness in high-dimensional data.

4.2.2 Random Forest Classifier

Another powerful method for classification is the Random Forest algorithm, which belongs to the family of ensemble learning techniques. It constructs a multitude of decision trees during training and aggregates their predictions through a majority voting mechanism. The use of multiple trees and random feature selection at each split helps in reducing variance and improving the model's generalization ability. This technique is particularly effective when the dataset includes noisy, complex, and non-linear patterns, making it a reliable choice for crop recommendation systems.

4.2.3 Support Vector Machine (SVM)

Support Vector Machines (SVM) are also widely employed due to their ability to construct optimal decision boundaries between different crop classes. SVM identifies the hyperplane that maximally separates the classes in the feature space and can be extended to handle non-linear data through the use of kernel functions. This capability makes SVM particularly useful when the crop classification task involves subtle distinctions between data points in high-dimensional space.

4.2.4 Logistic Regression

Though primarily used for binary classification, Logistic Regression can be extended to multiclass problems using one-vs-rest or multinomial logistic strategies. It models the probability that a given input belongs to a particular class.

Logistic Regression is another frequently used technique, especially for binary and multi-class classification problems. It models the probability that a given input belongs to a particular class using the logistic function. Although relatively simple in nature, logistic regression can yield highly interpretable results and performs efficiently when the relationship between the independent variables and the target class is approximately linear.

4.2.5 Decision Tree Classifier

A Decision Tree classifier splits the data based on feature values to build a tree-like structure of decisions. It is intuitive, easy to visualize, and can handle both numerical and categorical data.

Decision Trees provide a straightforward yet effective approach to classification. These models recursively split the data based on feature thresholds to create a tree-like structure where each path from the root to a leaf represents a rule for classification. Decision trees are intuitive and easy to visualize, making them an appealing choice for agricultural experts who value model transparency alongside predictive power.

4.2.6 XGBoost Classifier

XGBoost (Extreme Gradient Boosting) is a highly efficient implementation of gradient boosting algorithms. It works by combining multiple weak learners (typically decision trees) into a strong one, focusing on minimizing errors from previous iterations.

XGBoost has gained popularity due to its high accuracy and computational speed. It is a gradient boosting technique that builds models sequentially, where each new model attempts to correct the errors of its predecessor. XGBoost incorporates regularization to prevent overfitting and supports advanced features like parallelization and missing value handling. These properties make it a strong candidate for complex classification tasks such as crop recommendation, where precision and performance are critical.

4.2.7 Model Selection Strategy

Each of the above models was trained and evaluated using performance metrics such as:

i. Accuracy: it is defined as the ratio of the correctly predicted crop labels to the total number of predictions made. This measure essentially gives an overall view of how good a model is performing descriptive-wise.

ii. Precision and Recall: Precision measures the exactness of the classifier predicting an instance of a specific crop while recall measures its ability to identify all relevant instances of the given crop. This is viable while analysing the performance of individual classes in a multi-class classifier.

iii. F1-Score: F1 is a measure for the harmonic mean of precision and recall, thus giving one measure by making a balance between both of them. This is especially useful in the case of a highly imbalanced dataset.

iv. Confusion Matrix: Regarding visualization of the model performance, a confusion matrix was generated for appreciation of understandings of misclassification tendencies across different crop categories.

4.3 Algorithms for Yield Predictor (Regression Models)

Regression algorithms are fundamental in predictive modelling tasks where the objective is to estimate a continuous outcome variable based on a set of input features. In this project, the focus was on predicting crop yield using various environmental and agricultural inputs such as rainfall, fertilizer quantity, temperature, and soil nutrient levels.

4.3.1 Linear Regression

Among the regression models explored, Linear Regression served as a foundational approach. It operates under the assumption of a linear relationship between the dependent variable and the independent features. By fitting a straight line that minimizes the sum of squared differences between the predicted and actual values, linear regression offers both simplicity and interpretability. Despite its limitations in capturing non-linear relationships, it often serves as a benchmark for evaluating more complex models.

4.3.2 Decision Tree Classifier

Decision Tree Regressor is another widely used model for regression tasks. It splits the input space into distinct regions based on feature values, recursively partitioning the data to minimize variance within each leaf node. Unlike linear models, decision trees are capable of capturing non-linear interactions between variables and are not sensitive to outliers. However, they tend to overfit the data if not properly pruned or regularized. Their interpretability and adaptability to non-parametric data make them an attractive choice for modelling yield outcomes influenced by multiple factors.

4.3.3 Random Forest Regressor

The Random Forest Regressor, an ensemble-based technique, extends the idea of decision trees by constructing a large number of trees and averaging their predictions to enhance stability and reduce variance. Each tree is trained on a random subset of data and features, which leads to better generalization performance. This model is particularly effective for high-dimensional data with complex feature interactions, making it highly suitable for yield prediction where multiple environmental and soil conditions interplay.

4.3.4 XGBoost

To further improve performance, XGBoost (Extreme Gradient Boosting) was also utilized. It builds upon the concept of gradient boosting by sequentially adding decision trees that correct the errors of the previous ones. XGBoost incorporates regularization terms to penalize model complexity and includes efficient algorithms for handling missing data and parallel computation. These features enable it to achieve superior accuracy and robustness in various machine learning tasks, including agricultural yield forecasting.

4.3.5 Support Vector Machine (SVR)

Another model explored was Support Vector Regression (SVR), which is a regression counterpart of the SVM classification algorithm. SVR attempts to find a function that deviates from the actual target values by a value no greater than a specified margin while being as flat as possible. It uses kernel functions to transform data into higher dimensions where linear relationships can be found. Although computationally intensive, SVR performs well when the number of features is high relative to the number of observations.

4.3.6 K-Nearest Neighbors

K-Nearest Neighbors (KNN) Regressor is a non-parametric method that predicts the output based on the average of the closest training examples in the feature space. It is simple to implement and works well when the dataset has local smoothness, but it may be sensitive to the choice of k and the presence of noise. Due to its instance-based nature, it can capture local trends effectively, which is beneficial when yield varies significantly with specific conditions.

4.3.7 Model Selection Strategy

i. Mean Squared Error (MSE)

MSE measures the average of the squares of the errors between the predicted values and the actual target values.

Each prediction error (difference between actual and predicted) is squared to avoid negative values cancelling out positive ones. It is a fundamental metric used for optimization during training (especially in linear regression models).

ii. Root Mean Squared Error (RMSE)

RMSE is simply the square root of MSE and brings the error back to the original scale of the target variable.

RMSE is one of the most popular metrics for regression problems. Since it's in the same unit as the predicted variable, it's easier to understand and interpret than MSE.

iii. R – Squared (R^2 score)

R^2 measures the proportion of variance in the dependent variable that is predictable from the independent variables.

R^2 gives an idea of how well your independent variables explain the variability in your dependent variable. It compares your model to a baseline model that always predicts the mean.

5 System Design

The system design encapsulates the structural and behavioural architecture of the proposed crop recommendation and yield prediction framework. This design ensures the system is modular, scalable, and user-friendly. The system is divided into several logical layers: the input interface layer, the model processing layer, and the result visualization layer. These layers work cohesively to deliver accurate recommendations and yield predictions based on user-provided agricultural data.

5.1 System Architecture

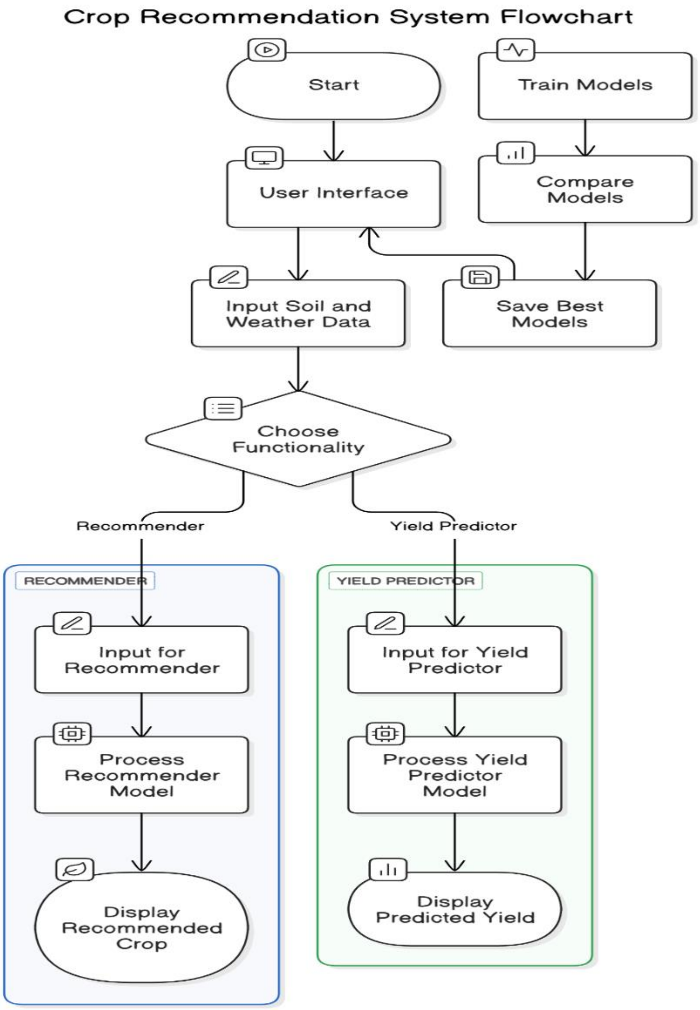


Figure 5.1 System Architecture

The Crop Recommendation System Flowchart visually represents the architecture and workflow of the agricultural decision support system. The system begins with the initial phase, where machine learning models are trained using historical agricultural datasets that include various soil properties (Nitrogen, Phosphorus, Potassium, pH), weather conditions (Temperature, Humidity, Rainfall), and crop yield data. Multiple models are evaluated, and their performance is compared using standard metrics like Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and R-squared (R^2). The best-performing models are saved and later used during prediction to ensure optimal results.

Following the model training phase, the system transitions to the user interaction phase. The user interface, developed using Django, enables users to input real-time soil and weather parameters. Once the input is received, users are prompted to choose the functionality they desire—either Crop Recommender or Yield Predictor.

If the Recommender module is selected, the system accepts input parameters such as soil nutrients and weather attributes. This input is then processed by the crop recommendation model, which predicts the most suitable crop to grow under those specific conditions. The recommended crop is then displayed on the user interface.

Alternatively, if the Yield Predictor functionality is chosen, the user is required to enter not only the environmental parameters but also the specific crop they intend to cultivate. The system then processes this data through the pre-trained yield prediction model, which outputs the expected crop yield (typically in tons per hectare). This prediction is then presented to the user through the interface.

This architecture ensures a modular and streamlined workflow where data flows logically from input to prediction. It combines backend model processing with frontend user interaction, making it practical and efficient for real-world agricultural use. The separation of modules also allows easy scalability and maintenance of each component independently.

5.2 Use Case Diagram

Figure 5.2 illustrates the use case diagram for the project. It is the similar process for both the models of crop recommender and the yield predictor.

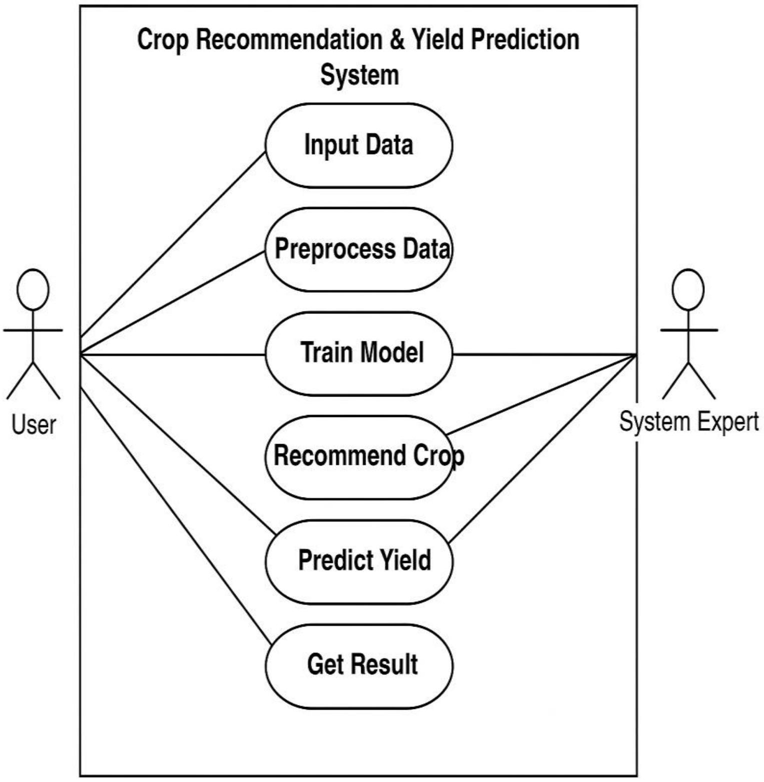


Figure 5.2 Use Case Diagram

The use case diagram for the Crop Recommendation and Yield Prediction System outlines the interactions between two main actors—User and System Expert—and the various functionalities offered by the system. The system is designed to help users determine the most suitable crop for cultivation and to predict the expected yield based on environmental and soil condition.

5.3 Class Diagram

Figure 5.3 illustrates the use case diagram for the project.

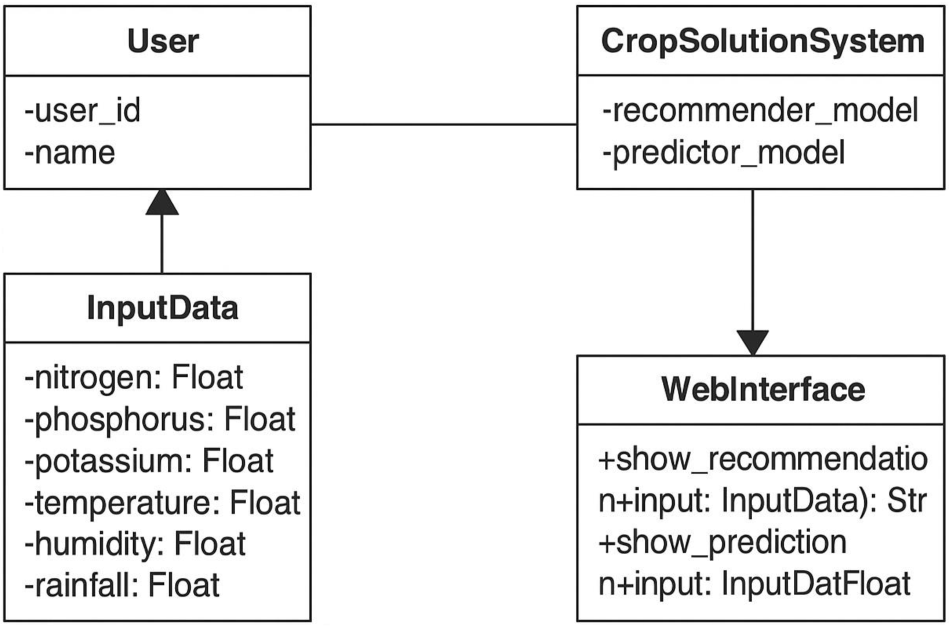


Figure 5.3 Class Diagram

The class diagram provides a structural overview of the major components in the Crop Recommendation and Yield Prediction System. It illustrates the relationships between classes such as User, InputData, CropSolutionSystem, and WebInterface, highlighting their attributes and methods.

The CropSolutionSystem contains the two main attributes of the project , the recommender model and the predictor model.

6 Implementation

The implementation phase transforms the design and planned methodology into a functional solution. This system integrates both the Crop Recommendation System and the Crop Yield Prediction System, developed using machine learning algorithms and deployed via a web-based interface for user interaction.

6.1 Requirements

Requirements are critical to the success of a project. By meeting these requirements, you can ensure that the project's software runs smoothly, reduces the risk of compatibility issues, and ensures that all stakeholders can access and use the project's software. By ignoring these requirements, you risk encountering compatibility issues, errors, and crashes, which can negatively impact the project's success.

6.1.1 Hardware Requirements

Hardware requirements are essential to ensure that the project's software can run efficiently and effectively. These requirements include the minimum specifications for the CPU, RAM, storage, and other hardware components. By meeting these requirements, the project's software can run smoothly, reducing the risk of crashes, errors, and other issues.

i. RAM (min 16GB)

ii. Hard Disk (min 128GB)

iii. CPU.

iv. X64 based Processor.

v. 64-bit operating system

6.1.2 Software Requirements

Software requirements are also critical to the success of a project. These requirements include the specific versions of software and operating systems that are compatible with the project's software. By ensuring that the project's software is compatible with the required software and operating systems, you can reduce the risk of compatibility issues and ensure that the project's software functions as intended.

i. Operating System: Windows 10/Linux

ii. Programming Language: Python 3.9+

iii. Web Framework: Django

iv. IDE: Visual Studio Code / Jupyter Notebook / Anaconda

v. Frontend: HTML, CSS

vi. Backend: Python (Django Framework)

6.1.3 Libraries

Libraries play a crucial role in software projects by providing pre-written code and functionality that developers can leverage to expedite development, improve code quality, and enhance the capabilities of their applications.

i. Pandas is the core library used for data manipulation and analysis. It allows for efficient handling of structured data and provides data structures like

DataFrames which are ideal for processing tabular datasets. It was primarily used for loading the datasets, cleaning, and preprocessing the data.

ii. NumPy, short for Numerical Python, complements pandas by offering support for multi-dimensional arrays and a range of mathematical functions. It was used for transforming data into the required formats, such as arrays for feeding into machine learning models, and for performing numerical operations like reshaping and scaling.

iii. Matplotlib and Seaborn are visualization libraries. Matplotlib enables the creation of static, interactive, and animated plots. It was used to plot histograms and scatter plots for analysing feature distributions and prediction accuracy. Seaborn, built on top of matplotlib, offers a more aesthetic and high-level interface for drawing informative statistical graphics such as heatmaps and correlation matrices to explore relationships among features.

iv. Scikit-learn (sklearn) is the backbone of the machine learning part of the project. It provides a suite of efficient tools for classification, regression, model selection, and evaluation. It includes algorithms such as Random Forest, Support Vector Machines, Decision Trees, Logistic Regression, and tools like train-test splitting, data scaling (via MinMaxScaler), encoding, and evaluation metrics like mean squared error (MSE), R-squared (R^2), precision, recall, and F1-score.

v. XGBoost is an advanced library optimized for gradient boosting. It was used both in the classification and regression phases due to its high accuracy and ability to handle complex patterns in data. XGBoost’s regularization capabilities make it particularly robust against overfitting.

vi. Pickle was used for model persistence. After training and selecting the best models, they were saved to disk using pickle so that they can be loaded later during prediction in the web interface, without the need to retrain.

vii. Django, a high-level Python web framework, was used to develop the web interface. It enabled integration of the trained machine learning models with a user-friendly frontend. Django handles the routing, data input forms, and presentation of results.

Together, these libraries formed the technological foundation of the system, providing the capability to ingest raw data, process and visualize it, build accurate predictive models, and deliver results via an interactive web platform.

6.2 Code

The implementation phase of the Crop Recommendation and Yield Prediction System involves meticulous development of Python-based machine learning models and a web interface using Django. This section elaborates on each aspect of the codebase to provide a comprehensive understanding of how the system operates—from data preprocessing to deployment.

6.2.1 GitHub Link

https://github.com/AmbatiSowmya1x/Crop_Project.git

The above GitHub repository contains code implementation, and the installation requirements and steps to run the project.

6.2.2 Importing Libraries

To begin with the implementation, a wide array of Python libraries was imported to support data analysis, model development, evaluation, and web integration:

i. Pandas and NumPy: Used for data manipulation and numerical computations.

ii. Matplotlib and Seaborn: Employed for data visualization and plotting histograms, heatmaps, and scatter plots.

iii. Scikit-learn (sklearn): Provided utilities for machine learning algorithms, preprocessing, metrics, and model selection.

iv. XGBoost: An optimized gradient boosting framework used for both classification and regression tasks.

v. Pickle: Used for serializing trained models, encoders, and scalers for reuse during inference.

vi. Django Framework: Used to create the frontend and backend interface of the application, integrating the trained models with user input and output.

This modular import structure ensures a clean and efficient development pipeline that is easy to debug and scale.

6.2.3 Dataset Loading and Preprocessing

Two primary datasets were utilized for this project:

i. Crop Recommendation Dataset: Contained soil and environmental features—such as Nitrogen (N), Phosphorus (P), Potassium (K), pH, temperature, humidity, and rainfall—along with a target variable representing the best-suited crop for the given conditions.

ii. Crop Yield Prediction Dataset: Included features like rainfall (mm), fertilizer amount, temperature, and NPK values, along with the crop yield expressed in quintals per acre.

The datasets were loaded using pandas.read_csv() and pandas.read_excel(). Missing values were handled using .dropna() or imputation techniques as needed. Columns were standardized for uniformity, and data types were cast explicitly (e.g., temperature as integers).

The LabelEncoder from sklearn.preprocessing was used to convert crop names (categorical data) into numerical labels for model compatibility. The mappings were stored using pickle for consistent decoding during prediction.

6.2.4 Data Analysis and visualization

Crop Recommendation Dataset

A heatmap based on Pearson correlation was constructed to analyze the relations among the features is shown in the Figure 6.1 Phosphorus and Potassium showed a moderate positive correlation. However, overall low interaction among variables suggests that there is little multicollinearity, making these variables good candidates for use in supervised learning models without excessive feature elimination.

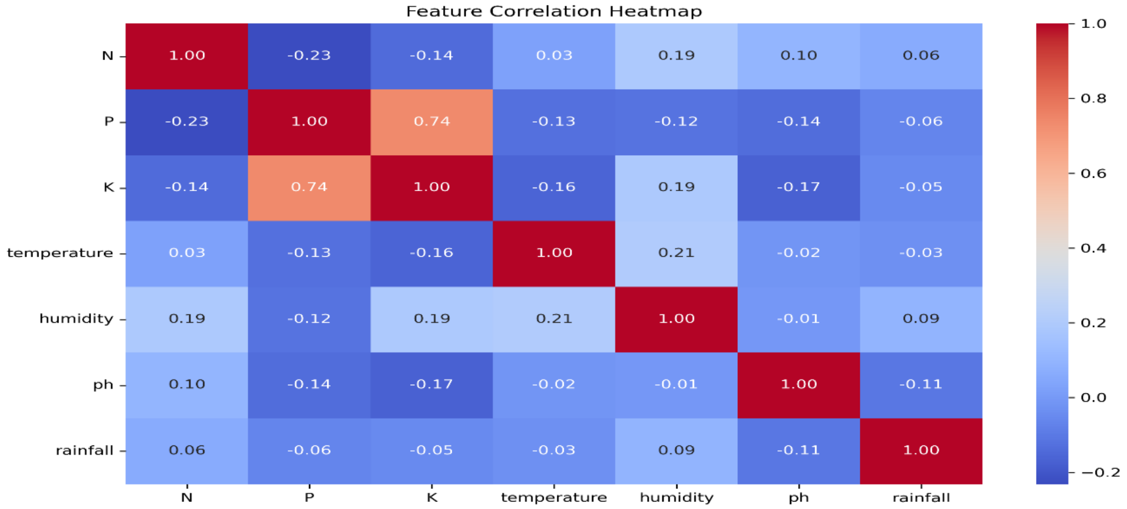


Figure 6.1 Feature Correlation Heatmap of

Crop Recommendation dataset

Figure 6.2 illustrates the distributions of the input features used for crop recommendation, including soil nutrients (Nitrogen, Phosphorus, Potassium), environmental parameters (temperature, humidity, rainfall), and pH value. Most features exhibit a bell-shaped or moderately skewed distribution. These histograms help assess the necessity for scaling or normalization prior to model training.

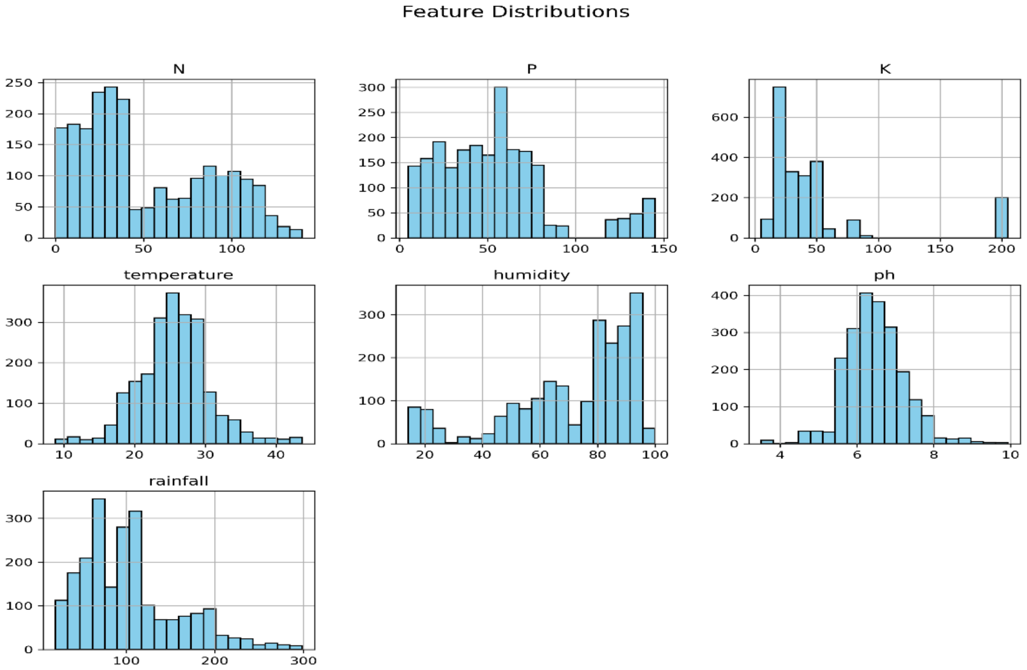


Figure 6.2 Distribution of input Features in the Crop Recommendation Dataset

Yield Prediction Dataset:

The histograms available in Figure 6.3can be used as good references toward understanding the impact of each feature. The values for different features such as Rainfall and Temperature, among others, do not follow a uniform distribution, and there is skewness, which means that their values exist in a specified range. Rainfall appears to have a bimodal distribution, suggesting that there are two separate clusters of data related to it. The values of fertilizers also show some skews, suggesting that the amount of nutrients applied varied among the samples.

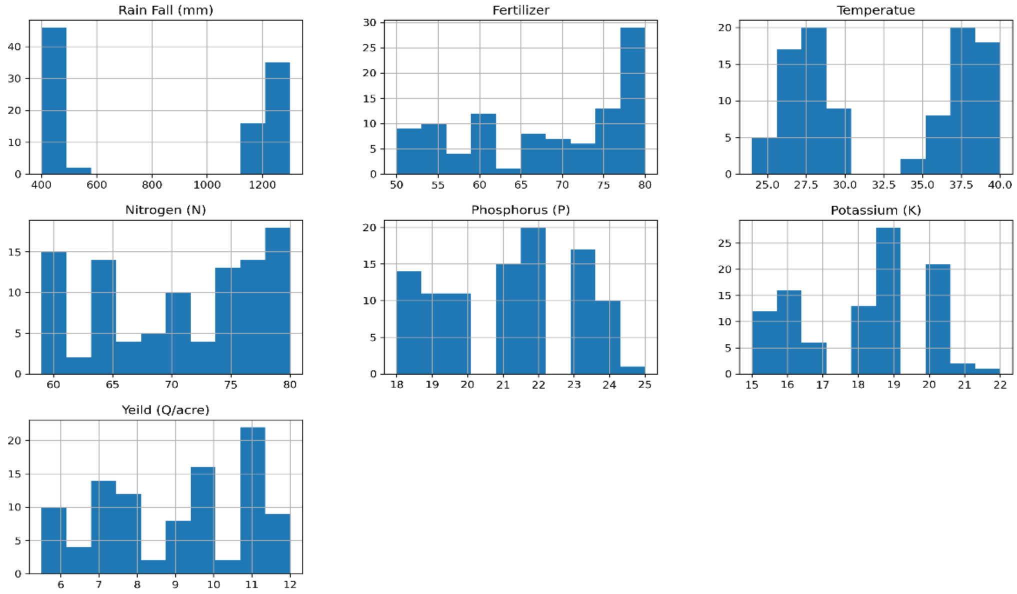


Figure 6.3 Distribution of input features in Yield Prediction Dataset

The correlation matrix is depicted in Figure 6.4, which measures the linear relationship among the features and the dependent variable, that is, Yield. By contrast with the previous dataset, the new dataset indicates higher positive correlations of some input features with yield:

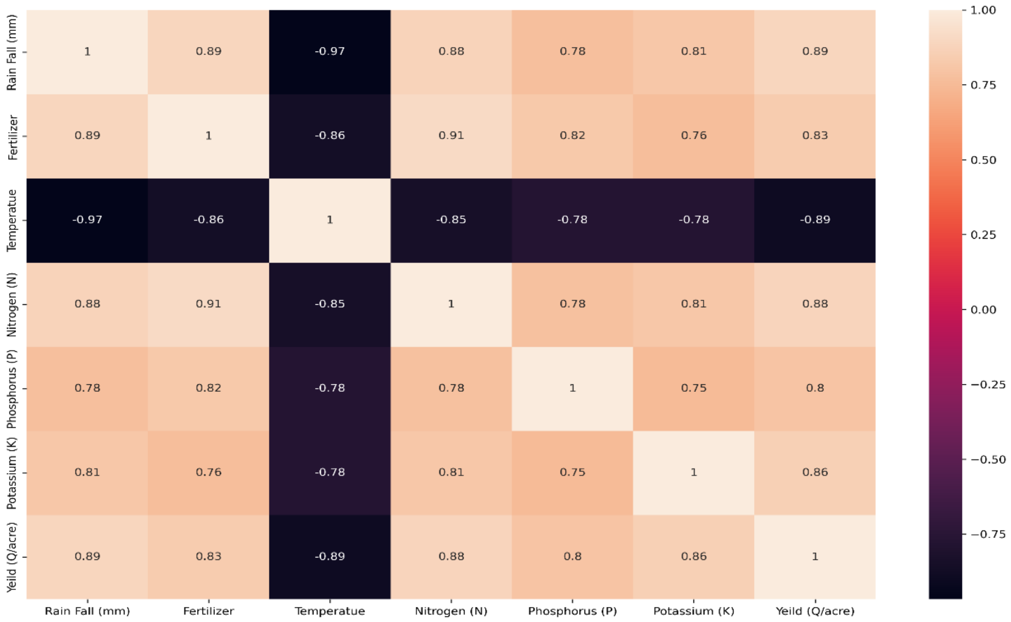


Figure 6.4 Feature Correlation Heatmap of Yield Prediction Dataset

Rainfall and Nitrogen (N) are both highly positively correlated to yield (about 0.88 to 0.89), thus showing their vital importance concerning crop production. Fertilizer and Potassium (K) are also positively highly correlated from 0.83 to 0.86, affirming their contributions to yield enhancement. Interestingly enough, Temperature boasts a strong negative correlation with Yield (check value ≈ -0.89), which indicates that too high a temperature could be harmful to crop output.

6.2.5 Data Scaling and Transformation

To normalize the input features and reduce the impact of skewness in the yield prediction dataset:

MinMaxScaler was used to scale the features within a 0–1 range, improving model convergence and consistency.

Both the input scaler (x_scaler) and output scaler (y_scaler) were stored using pickle so they could be reused during inference in the deployed application.

6.2.6 Splitting Data for Training and Testing

Using train_test_split() from sklearn.model_selection, the datasets were split into 80% training and 20% testing data. To ensure a balanced distribution of yield values across the test set, stratified sampling was used based on yield quantile bins (pd.qcut).

This ensured a more representative split, which is especially important in datasets with skewed distributions or varying output ranges.

6.2.7 Model Training

Classification Models (Crop Recommendation): The following classification models were trained:

i. Naïve Bayes

ii. Random Forest

iii. Support Vector Machine (SVM)

iv. Logistic Regression

v. Decision Tree Classifier

vi. XGBoost Classifier

Each model was trained on the soil and weather data, and performance was evaluated using metrics like accuracy, precision, recall, and F1-score. Overfitting and underfitting were detected by comparing training and testing accuracy.

Regression Models (Yield Prediction): The following regression models were implemented and trained:

i. Linear Regression

ii. Decision Tree Regressor

iii. Random Forest Regressor

iv. SVR (Support Vector Regressor)

v. KNN Regressor

vi. XGBoost Regressor

Evaluation metrics included Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and R^2 Score. Hyperparameter tuning was done using GridSearchCV to optimize the performance of Random Forest and other ensemble models.

6.2.8 Model Saving and Reuse

Once the best-performing model was identified (based on test metrics), it was serialized using pickle.dump() and saved into the /models directory. This included:

i. The best classification model (best_crop_model.pkl)

ii. The best regression model (yield_prediction_model.pkl)

iii. Encoders and scalers (crop_label_encoder.pkl, yield_predictor_encoder.pkl, yield_predictor_scaler.pkl, yield_predictor_y_scaler.pkl)

These saved objects were later loaded during deployment to perform inference without retraining the models, ensuring faster and consistent predictions.

6.2.9 Django Web Interface Development

The application interface was developed using Django’s template system. It included:

i. An index page with navigation links to the Crop Recommender and Yield Predictor.

ii. Separate forms for each module where users could input data.

iii. Backend logic to preprocess the inputs using the saved scalers/encoders, make predictions using the loaded models, and render results on a result page.

6.2.10 Integration, Output Display, and UI Styling

Upon form submission, Django’s views.py invoked the model pipeline, executed the prediction, and passed the result back to the HTML template using context variables. These results were styled and presented to the user in a clean and interactive manner.

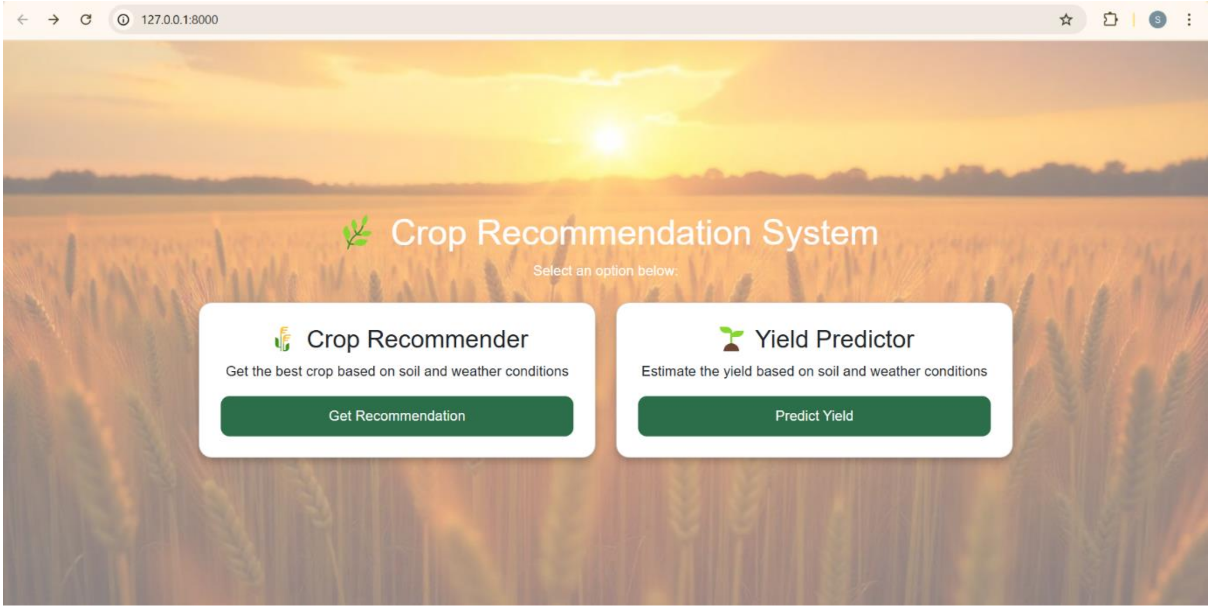
CSS and background images were incorporated into the templates to enhance visual appeal, keeping the interface responsive and engaging while maintaining simplicity.

7 Results

7.1 Interface

Interface is designed using python django frame work and html. This is the starting page of the application when the application is executed on Editor Terminal, the application is hosted on a web and the below page is opened on the browser.

The index page of the user interface of the application is shown below. Figure 7.1 shows how the index is designed with the two options, One to the Crop recommender and the other to the Yield Predictor.



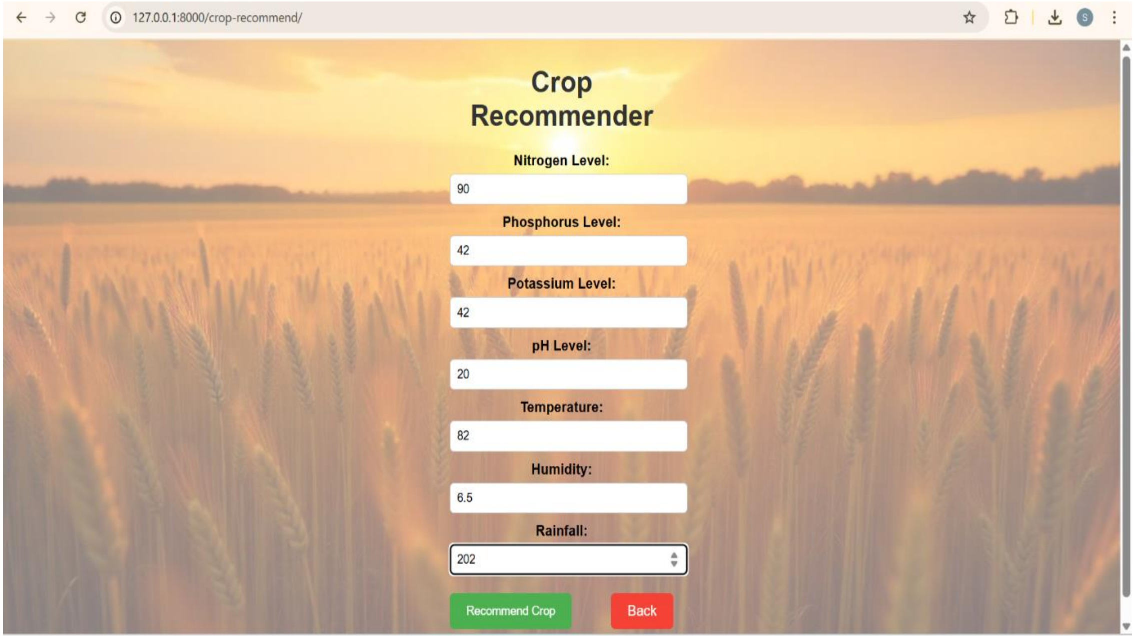
127.0.0.1:8000
Crop Recommendation System
Select an option below:
Crop Recommender
Get the best crop based on soil and weather conditions
Get Recommendation
Yield Predictor
Estimate the yield based on soil and weather conditions
Predict Yield

Figure 7.1 User Interface - Index Page

The two options takes the user to the new corresponding pages as shown.

7.1.1 Crop Recommender Interface and Result

The interface for the Crop Recommender is designed with a form which accepts the input form the user to provide a suitable crop recommendation.



Crop Recommender

Nitrogen Level: 90

Phosphorus Level: 42

Potassium Level: 42

pH Level: 20

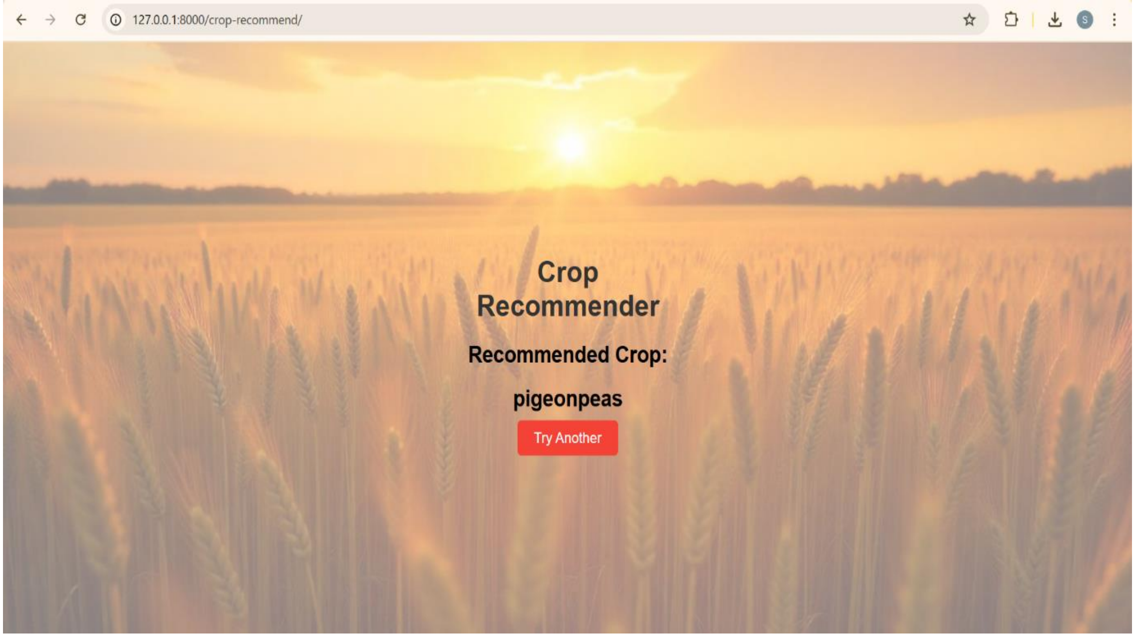
Temperature: 82

Humidity: 6.5

Rainfall: 202

Figure 7.2 Crop Recommender Interface

Figure 7.2 shows the crop recommender interface with the sample input.



Crop Recommender

Recommended Crop:

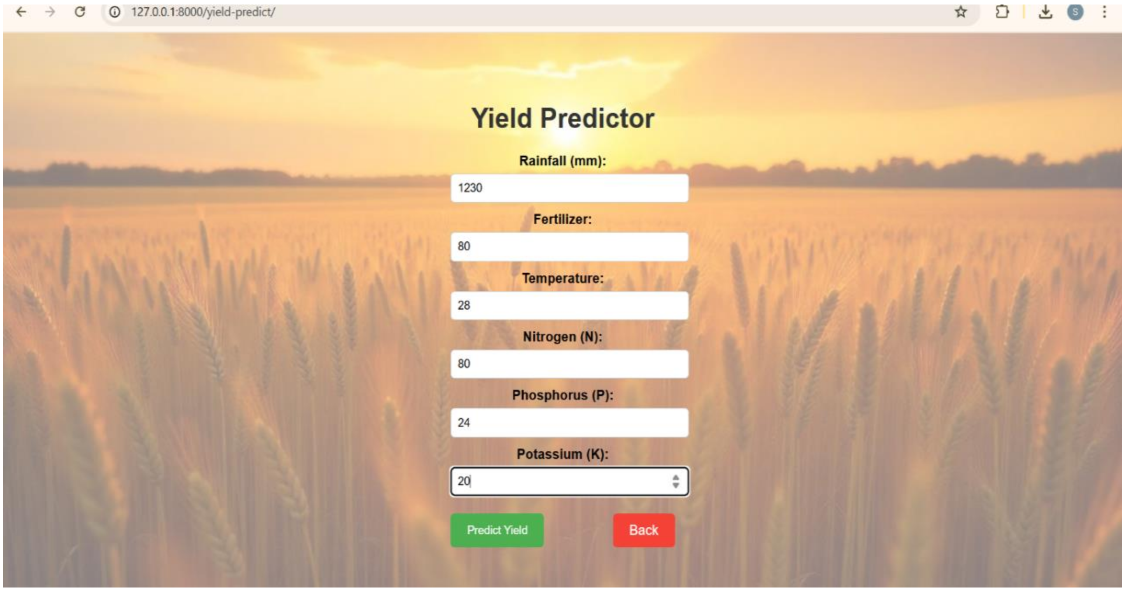
pigeonpeas

Figure 7.3 Crop Recommender Result

Figure 7.3 presents how the recommended crop is displayed after being processed by the machine learning model in the backend.

7.1.2 Yield Predictor Interface and Result

The interface for the Yield Predictor is designed with a form which accepts the input form the user to predict the yield (Q/hectare).



127.0.0.1:8000/yield-predict/

Yield Predictor

Rainfall (mm): 1230

Fertilizer: 80

Temperature: 28

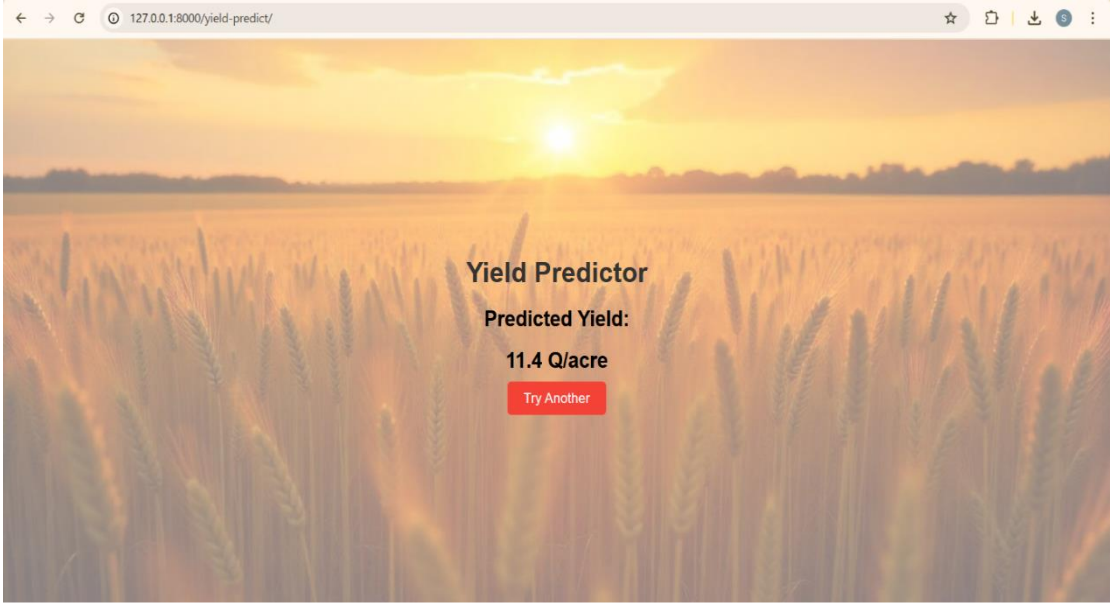
Nitrogen (N): 80

Phosphorus (P): 24

Potassium (K): 20

Figure 7.4 Yield Predictor Interface

Figure 7.4 shows the crop recommender interface with the sample input.



127.0.0.1:8000/yield-predict/

Yield Predictor

Predicted Yield:

11.4 Q/acre

Figure 7.5 Yield Predictor Result

Figure 7.5 presents how the predicted yield is displayed after being processed by the machine learning model in the backend.

7.2 Comparison

The proposed system was developed with two core modules: a Crop Recommendation System and a Crop Yield Prediction System, both of which were evaluated separately to validate the effectiveness and accuracy of the machine learning models used. This section highlights the performance of various machine learning models applied in both modules and provides a comparative analysis to identify the best-performing techniques.

7.2.1 Crop Recommendation Model Comparison

To evaluate the classification performance of the crop recommendation model, experimentation is done with multiple supervised learning algorithms including Random Forest, Decision Tree, Naïve Bayes, Support Vector Machine (SVM), Logistic Regression, and XGBoost.

Table 7.1 Classification Models Comparison (Crop Recommender)

Model	Accuracy	Precision	Recall	F1-Score
Naïve Bayes	99.545	0.995	0.995	0.995
Random Forest	99.318	0.993	0.993	0.993
SVM	97.954	0.979	0.979	0.979
Logistic Regressor	94.091	0.942	0.940	0.940
Decision Tree	98.409	0.984	0.984	0.984
XGBoost	98.636	0.986	0.986	0.986

The dataset used for training and testing contained features like nitrogen, phosphorus, potassium, temperature, humidity, pH, and rainfall, with the target variable being the type of crop. All the trained models were then compared based on various evaluation metrics such as Accuracy, Precision, Recall, and F1-Score.

Table 7.1 shows the evaluation metrics results for various models. Based on the presented classification results, it is evident that the models used for crop recommendation have yielded significantly high performance across all evaluation metrics. Among all the models, Naïve Bayes achieved the highest accuracy of 99.54%, along with precision, recall, and F1-score values of 0.995, indicating exceptional consistency in both predicting the correct class and maintaining balance between precision and recall.

Following closely, the Random Forest classifier exhibited a robust performance with 99.31% accuracy, and precision, recall, and F1-score all at 0.993, showcasing its effectiveness in handling complex, nonlinear relationships in the data. Similarly, XGBoost, a gradient boosting ensemble model, also performed competitively with 98.64% accuracy and high evaluation scores of 0.986, validating its strong generalization capability.

7.2.2 Yield Predictor Model Comparison

The regression-based yield prediction module was evaluated using models like Random Forest Regressor, Linear Regression, Decision Tree Regressor, Support Vector Regressor (SVR), and K-Nearest Neighbors Regressor. The dataset included nine features such as soil nutrients (N, P, K), temperature, humidity, pH, rainfall, crop type (encoded), and yield (target).

The regression models evaluated for crop yield prediction demonstrate varying levels of performance across the Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and R² Score metrics.

Table 7.2 Regression Models Comparison (Yield Predictor)

Model	MSE	RMSE	R2
Linear Regressor	0.439	0.663	0.906
Decision Tree	0.962	0.981	0.794
Random Forest	0.447	0.668	0.904
XGBoost	0.666	0.816	0.857
SVR	0.540	0.735	0.884
KNN Regressor	0.336	0.580	0.928

Table 7.2 shows the evaluation metrics results for various models. Among them, the K-Nearest Neighbors (KNN) Regressor emerged as the best-performing model with the lowest MSE of 0.336, an RMSE of 0.580, and the highest R² score of 0.928. This indicates its superior ability to closely approximate actual yield values and effectively capture the local structure of the data.

The Linear Regression model also showed strong predictive capability, achieving an R² score of 0.906 with a low MSE of 0.439 and RMSE of 0.663, suggesting a good linear fit between the input features and target variable. Similarly, Random Forest Regressor recorded an R² score of 0.904, which is nearly identical to that of Linear Regression, along with a slightly higher MSE and RMSE. This suggests that Random Forest is capable of capturing non-linear relationships with decent accuracy.

8 Conclusion

The Crop Recommendation and Yield Prediction System developed in this project addresses critical challenges faced in modern agriculture by providing intelligent decision support through machine learning techniques. The system is designed to recommend the most suitable crop based on soil nutrients and environmental conditions such as temperature, rainfall, humidity, and pH, while also predicting the potential yield of the selected crop. This dual functionality, implemented using a range of regression and classification algorithms, empowers farmers with data-driven insights to make informed choices, thereby enhancing productivity and sustainability.

The performance evaluation of various machine learning models indicates that the KNN Regressor achieved the highest accuracy for yield prediction, while the crop recommendation models demonstrated consistent performance across different classifiers. The use of a Django-based web interface ensures ease of access and usability for end users, bridging the gap between advanced analytics and practical farming applications.

Overall, this project contributes to the field of precision agriculture by facilitating smarter crop planning and yield estimation.

9 Future Scope

While the current system effectively delivers crop recommendations and predicts yield with commendable accuracy, there remain several avenues for future enhancement and expansion.

The system can be enhanced by integrating real-time weather data using APIs, enabling more dynamic and location-specific recommendations. Incorporating soil image analysis with deep learning can help evaluate soil health visually, enhancing prediction accuracy. Expanding the dataset using satellite and IoT-based sensor data will allow for more regionally adaptable models.

A fertilizer recommendation module can be introduced to guide optimal nutrient management. Mobile app integration with multilingual and voice input support can increase accessibility, especially in rural areas. These advancements will make the system more intelligent, scalable, and useful for a broader farming community.

Another promising direction is the **development of a mobile application** to improve accessibility for farmers, especially those in rural and remote areas. A user-friendly mobile interface, possibly with voice input and regional language support, can bridge the technological divide and bring the system closer to its target audience.

10 References

[1] P. Patil and R. Sherekar, “Performance analysis of Naive Bayes and J48 classification algorithm for data classification,” *International Journal of Computer Science and Applications*, vol. 6, no. 2, pp. 256–261, 2013.

[2] S. H. Patil and R. K. Srivastava, “Crop recommendation system using machine learning: A review,” *International Journal of Computer Applications*, vol. 180, no. 38, pp. 1–4, 2018.

[3] R. Sharma, D. Singh, and A. Kumar, “Machine learning techniques for crop yield prediction: A review,” *International Journal of Advanced Science and Technology*, vol. 29, no. 8, pp. 9582–9589, 2020.

[4] M. Kamilaris and F. X. Prenafeta-Boldú, “Deep learning in agriculture: A survey,” *Computers and Electronics in Agriculture*, vol. 147, pp. 70–90, Apr. 2018.

[5] B. Sonwane and V. D. Shewale, “Soil fertility prediction and recommendation system using machine learning,” in *Proc. Int. Conf. Inventive Computation Technologies (ICICT)*, Coimbatore, India, 2018, pp. 1–4.

[6] H. Kaur, N. Kumari, and D. Gupta, “Soil fertility and crop yield prediction using machine learning techniques,” in *Proc. 2019 Int. Conf. on Machine Learning, Big Data, Cloud and Parallel Computing (COMITCon)*, Faridabad, India, 2019, pp. 289–293.

[7] S. Pudumalar, R. Elangovan, R. Rajashree, C. Kavya, T. Kiruthika, and J. Nisha, “Crop recommendation system for precision agriculture,” in *2017 International Conference on Advanced Computing (ICoAC)*, Chennai, India, 2017, pp. 32–36. doi: 10.1109/ICoAC.2017.7951740.

[8] Y. Gandge and Sandhya, “A study on various data mining techniques for crop yield prediction,” in *2017 IEEE International Conference on Electrical, Electronics, Communication, Computer, and Optimization Techniques (ICEECCOT)*, Mysuru, India, 2017, pp. 420–423. doi: 10.1109/ICEECCOT.2017.8284541.

[9] A. R. Dasari, “Crop recommendation system to maximize crop yield in Ramtek region using machine learning,” *International Journal of Scientific Research in Science and Technology*, vol. 6, pp. 485–489, 2019.

[10] S. Babu, “A software model for precision agriculture for small and marginal farmers,” in *2013 IEEE Global Humanitarian Technology Conference: South Asia Satellite (GHTC-SAS)*, Trivandrum, India, 2013, pp. 352–355. doi: 10.1109/GHTC-SAS.2013.6629937.

[11] R. Gandhi, “Support Vector Machine — Introduction to Machine Learning Algorithms,” *Towards Data Science*, 2018.

[12] D. Li, Y. Miao, S. K. Gupta, C. J. Rosen, F. Yuan, C. Wang, L. Wang, and Y. Huang, “Improving potato yield prediction by combining cultivar information and UAV remote sensing data using machine learning,” *Remote Sensing*, vol. 13, no. 16, 2021.

[13] D. ManendraSai, M. S. Dekka, M. M. Rafi, M. M. R. D. Apparao, M. T. Suryam, and M. G. Ravindranath, “Machine learning techniques based prediction for crops in agriculture,” *Journal of Survey in Fisheries Sciences*, vol. 10, no. 1S, pp. 3710–3717, 2023.

[14] H. Chen, Z. Yang, L. Jiang, X. Zhang, and X. Zhao, “A hybrid model for crop recommendation based on machine learning and expert knowledge,” *Information Processing in Agriculture*, vol. 9, no. 1, pp. 153–168, 2022.

[15] F. Wang, X. Wang, and N. Song, “Biochar and vermicompost improve the soil properties and the yield and quality of cucumber (*Cucumis sativus* L.) grown in plastic shed soil continuously cropped for different years,” *Agriculture, Ecosystems & Environment*, vol. 315, p. 107425, 2021.