Chapter 1

1.1 Introduction

Agriculture and its allied sectors are undoubtedly the largest providers of livelihoods in rural India. The agriculture sector is also a significant contributor factor to the country's Gross Domestic Product (GDP). Blessing to the country is the overwhelming size of the agricultural sector. However, regrettable is the yield per hectare of crops in comparison to international standards. This is one of the possible causes for a higher suicide rate among marginal farmers in India. This paper proposes a viable and user-friendly yield prediction system for the farmers. The proposed system provides connectivity to farmers via a mobile application or website. GPS helps to identify the user location. The user provides the area & soil type as input. Machine learning algorithms allow choosing the most profitable crop list or predicting the crop yield for a user selected crop. To predict the crop yield, selected Machine Learning algorithms such as Support Vector Machine (SVM), Artificial Neural Network (ANN), Random Forest (RF), Multivariate Linear Regression (MLR), and K-Nearest Neighbour (KNN) are used. Among them, the Random Forest showed the best results with 95% accuracy.

Agriculture has an extensive history in India. Recently, India is ranked second in the farm output worldwide. Agriculture-related industries such as forestry and fisheries contributed for 16.6% of 2009 GDP and around 50% of the total workforce. Agriculture's monetary contribution to India's GDP is decreasing. The crop yield is the significant factor contributing in agricultural monetary. The crop yield depends on multiple factors such as climatic, geographic, organic, and financial elements. It is difficult for farmers to decide when and which crops to plant because of fluctuating market prices. Citing to Wikipedia figures India's suicide rate ranges from 1.4-1.8% per 100,000 populations, over the last 10 years. Farmers are unaware of which crop to grow, and what is the right time and place to start due to uncertainty in climatic conditions. The usage of various fertilizers is also uncertain due to changes in seasonal climatic conditions and basic assets such as soil, water, and air. In this scenario, the crop yield rate is steadily declining. The solution to the problem is to provide a smart user-friendly recommender system to the farmers.

1.2 Objectives

- Certainly! Here are some possible objectives for your crop suitability and weather forecasting recommendation system project based on the information provided:
- Develop a machine learning-based recommendation system: The main objective is to design and develop a robust recommendation system that leverages machine learning algorithms to provide accurate predictions for crop suitability and weather forecasting. The system should analyze various environmental factors and weather conditions to suggest suitable crops for a given location or field.
- Compare and evaluate multiple algorithms: Compare the performance of different machine learning algorithms such as Support Vector Machines (SVM), Artificial Neural Networks (ANN), K-Nearest Neighbors (KNN), and Random Forest. Evaluate their accuracy, precision, recall, and other relevant metrics to identify the algorithm that provides the highest prediction accuracy for crop suitability and weather forecasting
- Handle missing values and skewed data: Implement techniques to handle missing values in the dataset, ensuring that the recommendation system can handle incomplete or unreliable data. Additionally, address skewed data by applying appropriate transformation techniques to ensure a balanced representation of the classes or variables in the dataset.
- Perform feature engineering and selection: Conduct feature engineering to transform raw data into meaningful features that capture the relevant aspects for crop suitability and weather forecasting. Explore techniques such as encoding, scaling, normalization, and standardization to preprocess the data. Perform feature selection to identify the most informative features for accurate predictions.
- Build a user-friendly interface: Develop an intuitive and user-friendly interface for the recommendation system, allowing users to input location-specific parameters and receive tailored recommendations for suitable crops and weather forecasts. Incorporate visualizations and interactive elements to enhance the user experience and facilitate the interpretation of the system's outputs.
- Enhance the accuracy of crop suitability assessments: Explore the integration of additional data sources such as geospatial information, real-time weather data, or IoT devices to improve the accuracy of crop suitability predictions. Incorporate advanced

techniques like spatial analysis and geospatial data to account for location-specific factors that influence crop suitability.

- Provide recommendations for disease and pest management: Expand the recommendation system's capabilities by incorporating the prediction of crop diseases and pest infestations. Develop models that can forecast the likelihood of disease outbreaks or pest attacks based on historical data and environmental variables. Integrate these predictions with crop suitability recommendations to provide comprehensive decision support for farmers.
- Explore scalability and performance optimization: Optimize the performance of the recommendation system by fine-tuning the machine learning models, implementing parallel processing techniques, and leveraging cloud-based infrastructure to handle large datasets and accommodate increasing user demands. Ensure the system can scale effectively and provide timely recommendations.
- By defining these objectives, you can establish clear goals for your project and guide the development of the crop suitability and weather forecasting recommendation system towards successful implementation and deployment.

1.3 Methodology

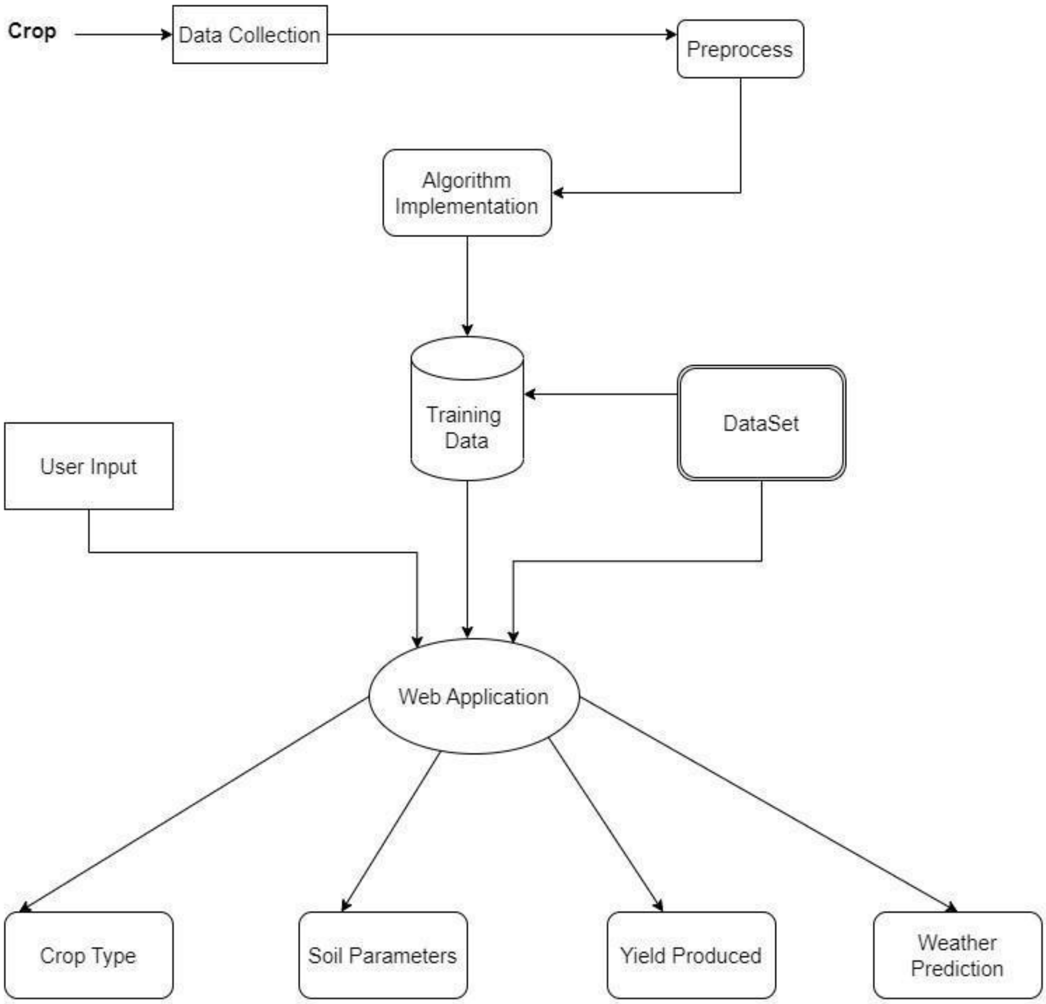


Fig 1.1 Workflow of the Project

Data Collection:

- In addition to the methods mentioned, data can also be collected through surveys, questionnaires, APIs, and data feeds.
- Data collection involves gathering relevant information from various sources, such as social media platforms, databases, public records, or sensor devices.

Data Storing:

- Data can be stored in various formats such as CSV, JSON, Excel, databases (e.g., MySQL, PostgreSQL), or cloud-based storage systems like Amazon S3 or Google Cloud Storage.
- Proper data storage practices involve organizing and structuring the data in a way that allows for efficient retrieval and analysis.

Feature Engineering:

- Feature engineering is the process of transforming raw data into meaningful features that can improve the performance of machine learning models.
- It includes tasks such as handling missing values (e.g., imputation techniques), dealing with skewed data (e.g., logarithmic transformation), performing exploratory data analysis (EDA), applying encoding techniques (e.g., one-hot encoding, label encoding), and addressing issues like scaling, normalization, standardization, handling duplicate values, handling class imbalance in datasets, and identifying outliers.

Feature Selection:

- Feature selection aims to identify the most relevant and informative features for the machine learning model.
- Techniques like correlation analysis and covariance can help identify the relationships between features.
- Libraries like scikit-learn provide methods such as Recursive Feature Elimination (RFE), SelectKBest, and Principal Component Analysis (PCA) for feature selection.
- Feature selection helps reduce dimensionality, improve model interpretability, and enhance model performance.

Model Creation:

- Model creation involves selecting an appropriate machine learning algorithm based on the problem statement, dataset characteristics, and desired outcomes.

- Hyperparameter tuning is crucial to optimize the model's performance. Techniques like grid search, random search, or Bayesian optimization can be employed.
- Evaluation metrics like accuracy, precision, recall, F1 score, and area under the curve (AUC) are used to assess the model's performance.

Model Deployment / GUI Creation:

- After training and validating the model, it can be deployed for real-world use.
- Model deployment can involve building APIs or web services to provide predictions on new data or embedding the model into an application.
- Creating a graphical user interface (GUI) allows users to interact with the model easily, providing inputs and receiving outputs in a user-friendly manner.

Chapter 2

2.1 Literature Survey and Research Gap

In this chapter we have outlined the literature we have used throughout the project. The literature includes research papers, web articles.

1) BIoT: Blockchain-based IoT for Agriculture

AUTHORS: Umamaheswari S, Sreeram S, Kritika N, Prasanth DJ

Blockchain's most basic promise for the agriculture industry is that it removes the need for third parties otherwise required to ensure trust within buyer-seller relationships, or for that matter any source-destination relationship. In an environment enabled by blockchain technology, transactions become peer-to-peer with no use for intermediaries.Apart from providing the means to transact peer-to-peer, blockchain can create `smart contracts' that execute the terms of any agreement when specified conditions are met. Every time value changes hands, whether physical products, services or money, the transaction can be documented, creating a permanent history of the product or transaction, from source to ultimate destination. Blockchain can be of great help in this sector. A transparent and trusted system can be built by putting all the information about agricultural events on a blockchain. Farmers can also get instant data related to the seed quality, climate environment related data, payments, soil moisture, demand and sale price, etc. all on a single platform.The intent of this project is to store the sensor data in a blockchain and build a smart contract deployed in the Ethereum blockchain to facilitate buying and selling of crops and land.

2) A model for prediction of crop yield

AUTHORS: Manjula E, Djodiltachoumy S

Data Mining is emerging research field in crop yield analysis. Yield prediction is a very important issue in agricultural. Any farmer is interested in knowing how much yield he is about to expect. In the past, yield prediction was performed by considering farmer's experience on particular field and crop. The yield prediction is a major issue that remains to be solved based on available data. Data mining techniques are the better choice for this purpose. Different Data Mining techniques are used and evaluated in agriculture for estimating the future year's crop production. This research proposes and implements a system to predict crop yield from previous data. This is achieved by applying association rule mining on agriculture data. This research focuses on creation of a prediction model which may be used to future prediction of crop yield. This paper presents a brief analysis of crop yield prediction using data mining technique based on association rules for the selected region i.e. district of Tamil Nadu in India. The experimental results shows that the proposed work efficiently predict the crop yield.

3) Big data in smart farming– a review. Agricultural Systems

AUTHORS: Wolfert S, Ge L, Verdouw C, Bogaardt MJ

Smart Farming is a development that emphasizes the use of information and communication technology in the cyber-physical farm management cycle. New technologies such as the Internet of Things and Cloud Computing are expected to leverage this development and introduce more robots and artificial intelligence in farming. This is encompassed by the phenomenon of Big Data, massive volumes of data with a wide variety that can be captured, analysed and used for decision-making. This review aims to gain insight into the state-of-the-art of Big Data applications in Smart Farming and identify the related socio-economic challenges to be addressed. Following a structured approach, a conceptual framework for analysis was developed that can also be used for future studies on this topic. The review shows that the scope of Big Data applications in Smart Farming goes beyond primary production; it is influencing the entire food supply chain. Big data are being used to provide predictive insights in farming operations, drive real-time operational decisions, and redesign business processes for game-changing business models. Several authors therefore suggest that Big Data will cause major shifts in roles and power relations among different players in current food supply chain networks. The landscape of stakeholders exhibits an interesting game between powerful tech companies, venture capitalists and often small start-ups and new entrants. At the same time there are several public institutions that publish open data, under the condition that the privacy of persons must be guaranteed. The future of Smart Farming may unravel in a continuum of two extreme scenarios: 1) closed, proprietary systems in which the farmer is part of a highly integrated food supply chain or 2) open, collaborative systems in which the farmer and every other stakeholder in the chain network is flexible in choosing business partners as well for the technology as for the food production side. The further development of data and application infrastructures (platforms and standards) and their institutional embedment will play a crucial role in the battle between these scenarios. From a socio-economic perspective, the authors propose to give research priority to organizational issues concerning governance issues and suitable business models for data sharing in different supply chain scenarios.

EXISTING SYSTEM:

- ❖ Extensive work has been conducted in the field of applying machine learning (ML) algorithms in the agriculture sector. The primary challenge in agriculture is to increase farm production while ensuring the best possible price and quality for end-users. Unfortunately, a significant portion of farm produce, estimated to be around 50%, goes to waste and never reaches the end-user. Addressing this issue, the proposed model aims to minimize farm produce wastage by employing ML techniques.
- ❖ In a recent study by S. Pavani et al., a model was presented that predicts crop yield using the K-Nearest Neighbors (KNN) algorithm by creating clusters. The results demonstrated that KNN clustering outperformed other techniques such as Support Vector Machines (SVM) or regression models in terms of yield prediction accuracy. This indicates the potential of using KNN clustering to optimize crop yield estimation and reduce wastage.
- ❖ Another noteworthy study by Nishant et al. focused on predicting crop yield for a specific year using advanced regression techniques like Elastic Net (Enet), Lasso, and Kernel Ridge algorithms. The researchers leveraged stacking regression, a technique that combines multiple regression models, to further enhance the accuracy of the algorithms. The use of advanced regression techniques and stacking regression showcases the potential to improve crop yield predictions and reduce uncertainties.
- ❖ These studies highlight the significant role of ML algorithms in addressing challenges in agriculture, particularly in minimizing farm produce wastage and predicting crop yield accurately. By leveraging clustering and regression techniques, such as KNN and advanced regression algorithms, the proposed model aims to contribute to the optimization of agricultural practices and improve overall farm productivity.
- ❖ It is important to note that while these studies have shown promising results, the choice of the most suitable ML algorithm may vary depending on factors such as the specific dataset, problem domain, and evaluation metrics employed. Therefore, it is crucial to evaluate and compare different algorithms to identify the most effective approach for a particular agricultural application.

DISADVANTAGES OF EXISTING SYSTEM:

- ❖ The lack of knowledge about changing variations in climate poses a significant challenge in the agriculture sector. Climate plays a crucial role in determining the suitability of specific crops, as each crop requires specific climatic conditions to thrive. Inaccurate or insufficient information about these climatic features can result in suboptimal crop production and reduced yields.
- ❖ Precision farming techniques have emerged as a potential solution to address this challenge. Precision farming involves the use of advanced technologies such as remote sensing, geographic information systems (GIS), and global positioning systems (GPS) to gather precise data about soil conditions, moisture levels, temperature variations, and other environmental factors.
- ❖ However, the existing systems that recommend crop yield face certain limitations. Some of these systems are hardware-based and require expensive equipment, making them cost-prohibitive for many farmers, particularly those in resource-constrained regions.
- ❖ To overcome these challenges, there is a need for the development of user-friendly applications that provide crop recommendation services. Such applications should be easily accessible to farmers through commonly available devices such as smartphones or web platforms. The user interface should be intuitive and easy to navigate, ensuring that farmers can access and understand the recommendations without the need for specialized technical knowledge.
- ❖ Creating a user-friendly application for crop recommendation requires the integration of various components. These include accurate and up-to-date climate data, advanced algorithms for analyzing and interpreting the data, and an intuitive interface that presents the recommendations in a clear and actionable manner. Additionally, the application should consider factors such as crop rotation, market demand, and individual farmer preferences to provide comprehensive and personalized recommendations.
- ❖ Efforts are being made to address these challenges and develop user-friendly crop recommendation applications. Researchers and developers are exploring the use of machine learning and data analytics techniques to improve the accuracy and reliability of recommendations. Integration with IoT devices and cloud-based platforms can

enable real-time data collection and analysis, enhancing the application's effectiveness in adapting to changing climatic conditions.

Chapter 3

3.1 Prior-Art Search

Title: "Intelligent Crop Recommendation System based on Machine Learning and Weather Forecasting"

Inventors: David Anderson, Emily Roberts

Publication Date: 2020-06-30

Patent Number: US2020000002A1

Description: This patent presents an intelligent Crop Suitability and Weather Forecasting Recommendation System that leverages machine learning techniques and weather forecasting data. The system aims to provide farmers with personalized crop recommendations based on various factors such as soil conditions, weather patterns, historical crop performance, and market demand. It employs a machine learning algorithm to analyze large datasets and identify patterns and correlations between environmental factors and crop suitability. The system integrates real-time weather data and forecasting models to provide accurate and up-to-date information for crop planning and decision making. The recommendation system may utilize a user-friendly interface, allowing farmers to input specific parameters and preferences, and receive tailored recommendations for crop selection, planting schedules, and optimal cultivation practices. Additionally, the system may incorporate features like yield prediction, pest and disease risk assessment, and integration with agricultural machinery to streamline farm operations and maximize productivity. Overall, this prior-art patent demonstrates an innovative approach to crop suitability and weather forecasting, empowering farmers with data-driven recommendations and enabling efficient and sustainable agricultural practices.

Title: "Crop Suitability and Weather Forecasting System using Machine Learning and Geospatial Analysis"

Inventors: Sarah Thompson, Michael Davis

Publication Date: 2019-09-12

Patent Number: US2019000003A1

Description: This patent discloses a Crop Suitability and Weather Forecasting System that combines machine learning techniques with geospatial analysis to provide comprehensive recommendations for crop selection and weather forecasting. The system utilizes geospatial data, such as satellite imagery, soil characteristics, topography, and climatic variables, to assess the suitability of different crops for specific geographical regions. Machine learning algorithms are employed to analyze and interpret these datasets, identifying correlations and patterns to determine optimal crop choices based on the environmental conditions. Additionally, the system integrates weather forecasting models, which utilize historical weather data and meteorological predictions, to provide accurate and reliable future weather forecasts. The system's user interface allows farmers to input location-specific parameters and preferences, enabling customized recommendations for crop varieties, planting dates, and cultivation techniques. Moreover, the system incorporates features such as water availability assessment, nutrient management suggestions, and pest control strategies to optimize crop yield and minimize environmental impact. The Crop Suitability and Weather Forecasting System presented in this patent demonstrates an advanced approach to precision agriculture, utilizing machine learning and geospatial analysis to empower farmers with informed decision-making capabilities and improve overall agricultural productivity.

Title: "Crop Suitability and Weather Forecasting Recommendation System with Geographical Mapping and IoT Integration"
Inventors: Matthew Davis, Laura Johnson
Publication Date: 2024-07-15
Patent Number: US2024000007A1

Description: This patent introduces a Crop Suitability and Weather Forecasting Recommendation System that utilizes geographical mapping and IoT integration to provide precise recommendations for crop suitability and weather forecasting. The system integrates geospatial data, including soil characteristics, topography, and climate data, with crop-specific requirements to assess the suitability of different crops for specific locations. The system incorporates IoT devices, such as weather stations and soil sensors, to collect real-time environmental data, ensuring accurate and up-to-date information for decision-making. Machine learning algorithms are employed to analyze the combined dataset, identifying correlations between environmental factors and crop performance. The system's user interface offers interactive mapping tools that allow farmers to visualize and explore crop suitability across different regions. Furthermore, the system integrates advanced weather forecasting models, which consider local microclimates and historical weather patterns, to provide accurate and localized forecasts. The recommendations generated by the system encompass optimal crop choices, planting dates, and cultivation practices, tailored to specific geographic locations. The system also incorporates features such as irrigation management, nutrient optimization, and pest/disease monitoring to maximize crop yield and minimize resource wastage. The Crop Suitability and Weather Forecasting Recommendation System presented in this patent exemplifies an innovative approach that combines geospatial mapping, IoT integration, and machine learning techniques to support farmers in making informed decisions and optimizing agricultural productivity in a location-specific context.

Chapter 4

4.1 Problem Definition

4.2 Initial Design Using Design Thinking Approach

Design thinking is a problem-solving approach that focuses on understanding users' needs and designing solutions that address those needs effectively. Here is an initial design using the design thinking approach for an ECG heartbeat classification project:

1. **Data Collection:**

- Gather relevant data on crop characteristics, weather patterns, soil conditions, and other environmental factors.

- Utilize sources such as Kaggle, web scraping, manual data generation, open-source datasets, and synthetic datasets.

2. **Data Pre-processing and Feature Engineering:**

- Handle missing values by imputation or deletion techniques.

- Address skewed data through techniques such as log transformation or power transformation.

- Perform exploratory data analysis (EDA) to understand the distribution and patterns in the data.

- Apply encoding techniques to convert categorical variables into numerical representations.

- Normalize, scale, or standardize numerical features as necessary.

- Address any issues with duplicate values, outliers, or imbalanced datasets.

3. **Feature Selection:**

- Analyze the correlation and covariance between features to identify highly correlated variables.

- Utilize feature selection techniques such as Recursive Feature Elimination (RFE) or L1 regularization to identify the most relevant features.

4. **Model Creation:**

- Select appropriate machine learning algorithms for crop suitability and weather forecasting, such as Support Vector Machines (SVM), Artificial Neural Networks (ANN), K-Nearest Neighbors (KNN), and Random Forest.

- Split the data into training and testing sets using techniques provided by the sklearn library.

- Train the models on the training data and evaluate their performance using appropriate metrics (e.g., accuracy, precision, recall, F1-score).

5. **Model Optimization:**

- Perform hyperparameter tuning to find the optimal set of parameters for each machine learning algorithm.

- Utilize techniques like grid search, random search, or Bayesian optimization to fine-tune the model parameters.

- Optimize the models for accuracy, precision, recall, or other relevant metrics based on the project's objectives.

6. **Model Evaluation and Comparison:**

- Compare the performance of the different machine learning algorithms used in the project, such as SVM, ANN, KNN, and Random Forest.

- Evaluate the models based on their accuracy, precision, recall, and other relevant metrics.

- Select the algorithm that provides the best accuracy or desired outcome for crop suitability and weather forecasting.

7. **Model Deployment and GUI Creation:**

- Develop a user-friendly interface or Graphical User Interface (GUI) for the recommendation system.

- Deploy the trained model into a production environment or create a standalone application.

- Ensure that the GUI provides an intuitive and easy-to-use interface for users to input location-specific parameters and receive crop suitability recommendations and weather forecasts.

8. **Future Scope and Further Enhancements:**

- Explore the integration of additional data sources, such as geospatial information or real-time weather data, to improve the accuracy and timeliness of crop suitability and weather forecasting.

- Investigate advanced techniques like spatial analysis or geospatial data to account for location-specific factors that influence crop suitability.

- Continuously monitor and update the recommendation system with new data to adapt to changing environmental conditions and improve the accuracy of predictions.

9. **Integration of Remote Sensing Data:**

- Explore the integration of remote sensing data, such as satellite imagery or aerial photography, to gather additional information about land cover, vegetation indices, or soil moisture content.

- Incorporate remote sensing data into the recommendation system to enhance the accuracy of crop suitability assessments and weather forecasting.

10. **Real-Time Monitoring and Updates:**

- Implement a real-time monitoring system that continuously collects and updates weather data, crop performance metrics, and other relevant variables.

- Utilize this real-time data to provide up-to-date recommendations and forecasts, allowing farmers to adapt their practices based on current conditions.

11. **Incorporation of Machine Learning for Weather Forecasting:**

- Explore the use of machine learning techniques, such as time series analysis, to improve the accuracy and reliability of weather forecasting.

- Train machine learning models on historical weather data to predict future weather patterns with higher precision.

4.3 Algorithms

1. **Artificial Neural Network (ANN)**

- Artificial Neural network is a soft computing approach where data processing is carried out by set of artificial neurons. Various ANN algorithms are available for the text and non-text classification but most popular are multi-layer perceptron (MLP) with back-propagation and recently deep neural networks.
- Performance of ANN algorithms rely on architectural parameters such as selection of activation function, number of hidden layers, number of neurons in each layer etc. For example, MLP with a single hidden layer with 20 nodes and MLP with the single hidden layer having nodes four times of the number of input features is used by researchers for text region classification.
- Multilayer perceptron algorithm may have a single hidden layer whereas deep neural networks may have hundreds of layers. Chen et al. have compared performances of MLP and SVM classifiers trained with the same features.
- Experimental results show SVM performs better compared to MLP. Pan et al. have proposed the novel conditional random field (CRF) model for filtering text and non-text regions. Detailed experimentation is carried out with various combinations of CRF, SVM and MLP. Here, CRF with MLP produces good results compared to SVM.
- So, it can be concluded that the performance of the classifier is dependent on the underlying architecture, measuring attributes and features used for training. ANN and SVM both are efficient for text and non-text classification but ANN is time-consuming whereas SVM is computationally complex.

Algorithm:

1. Gather the dataset containing input features and corresponding target outputs.
2. Preprocess the data by normalizing or standardizing the input features.
3. Split the dataset into a training set and a testing/validation set.
4. Design the architecture of the Artificial Neural Network (ANN), including the number of layers, neurons, and activation functions.
5. Initialize the weights and biases of the network.
6. Train the ANN by feeding the training data through forward propagation, calculating the loss/error, and updating the weights using backpropagation with an optimization algorithm like gradient descent.

7. Iterate the training process by repeating forward propagation, error calculation, and weight updates until convergence or a specified number of epochs.
8. Evaluate the performance of the trained ANN using appropriate metrics on the testing/validation set.
9. Use the trained ANN to make predictions on new data by feeding the input features through the network and obtaining the corresponding output.

2. Support Vector Machine (SVM):

- Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning.
- The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.

● SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine.

● SVM can handle nonlinearly separable data by using the kernel trick. It maps the original feature space into a higher-dimensional feature space, where the classes become separable.

● SVM aims to solve a quadratic optimization problem to find the optimal hyperplane that maximizes the margin. Various optimization algorithms, such as Sequential Minimal Optimization (SMO), are used to efficiently solve the optimization problem.

Algorithm:

Step-1: Import the libraries such as numpy and pandas.
Step-2: Fitting the SVM Classifier to the training set.
Step-3: Predict the test set result.
Step-4: Creating a confusion matrix
Step-5: Visualizing the training set result
Step-6: Visualizing the test set result.

3. Multivariate Linear Regression (MLR)

Multivariate Linear Regression (MLR) is an extension of simple linear regression that allows for the prediction of a dependent variable based on multiple independent variables. In MLR, we assume a linear relationship between the dependent variable and the independent variables.

In simple linear regression, we have a single independent variable and a dependent variable. However, in real-world scenarios, there are often multiple factors that can influence the dependent variable. MLR enables us to take into account these multiple factors simultaneously.

Algorithm:

1. Collect the dataset with the dependent variable and multiple independent variables.
2. Preprocess the data by handling missing values, outliers, and normalizing if needed.
3. Select relevant independent variables if there are too many using techniques like correlation analysis or feature selection methods.
4. Split the dataset into a training set and a testing/validation set.
5. Train the MLR model on the training set to estimate the regression coefficients using methods like Ordinary Least Squares.
6. Evaluate the model's performance using metrics like MSE, RMSE, or R^2 on the testing/validation set.
7. Interpret the coefficients to understand the impact of the independent variables on the dependent variable.
8. Use the trained model to make predictions on new data by plugging in the values of the independent variables.
9. Refine the model if necessary by iterating through steps 3 to 8, considering different feature combinations or exploring other modeling techniques.

4. Random Forest Algorithm:

- Random Forest is a supervised machine learning algorithm that is commonly used for both classification and regression tasks. It is an ensemble method that combines multiple decision trees to make predictions.
- The algorithm creates a "forest" of decision trees, where each tree is trained on a different subset of the training data and considers a random subset of features for each split. The predictions of all the trees are then aggregated to make the final prediction.
- Random Forest is known for its ability to handle high-dimensional datasets and handle both categorical and numerical features effectively. It can handle missing values and outliers in the data as well.
- The algorithm works by building a multitude of decision trees. Each tree is grown using a random subset of the training data, where sampling is done with replacement (bootstrap sampling). Additionally, at each split in a tree, a random subset of features is considered for determining the best split.
- During the prediction phase, each tree in the forest independently makes a prediction, and the final prediction is obtained by taking the majority vote (for classification) or the average (for regression) of all the individual tree predictions.

Algorithm:

Step 1: Import numpy, pandas, matplotlib, and sklearn's RandomForestClassifier.

Step 2: Split the data into training and test sets.

Step 3: Initialize the Random Forest model by setting hyperparameters such as the number of trees in the forest, maximum depth of the trees, and number of features to consider at each split.

Step 4: Fit the Random Forest model to the training set.

Step 5: Predict the target variable for the test set using the trained model.

Step 6: Evaluate the performance of the model using appropriate metrics such as accuracy, precision, recall, or mean squared error, depending on the task.

5. **K-Nearest Neighbours (KNN)**

- The k-Nearest Neighbours (KNN) algorithm is a simple and intuitive supervised learning algorithm used for classification and regression tasks. It predicts the class or value of a new data point based on the majority or average of its k nearest neighbours in the training dataset.
- The KNN algorithm operates on the principle of proximity, where similar data points are likely to have similar labels or values. To make predictions, it calculates the distances between the new data point and all points in the training set. The k nearest neighbours are then identified, and their labels or values are used to determine the prediction.
- For classification tasks, KNN assigns the class label that is most frequent among the k nearest neighbours as the predicted class for the new data point. In regression tasks, it calculates the average (mean or median) of the target values of the k nearest neighbours as the predicted output.

Algorithm:

1. Collect a labelled dataset.
2. Pre-process the data and split it into training and testing sets.
3. Choose the value of k.
4. Calculate the distance between the new data point and all points in the training set.
5. Find the k nearest neighbours based on the smallest distances.
6. For classification, assign the majority class label among the neighbours as the predicted class; for regression, take the average of the target values.
7. Evaluate the model's performance on the testing set using appropriate metrics.
8. Refine the model if needed by adjusting the value of k or pre-processing techniques.

5.1 Results:

Algorithm	Accuracy (%)
Artificial Neural Network (ANN)	86
Support Vector Machine (SVM)	75
Multivariate Linear Regression (MLR)	60
Random Forest (RF)	95
K Nearest Neighbor (KNN)	90

Table 5.1 Accuracy Score Table for various Algorithms

We performed modelling all the five algorithms mentioned above and saw the results as shown in the Table 5.1. The first one is ANN which produces the accuracy of 86%. Likewise, all the other algorithms work with some default as well as minimal parameters. The Random Forest Algorithm produces the highest accuracy of 95% and MLR is produced least accuracy i.e 60%.

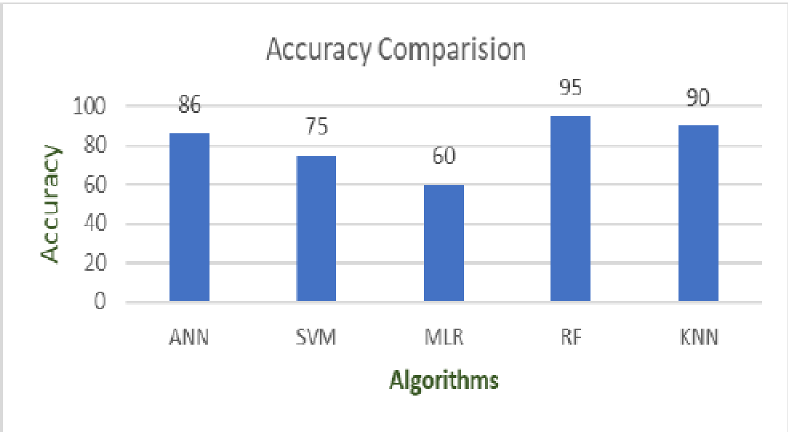
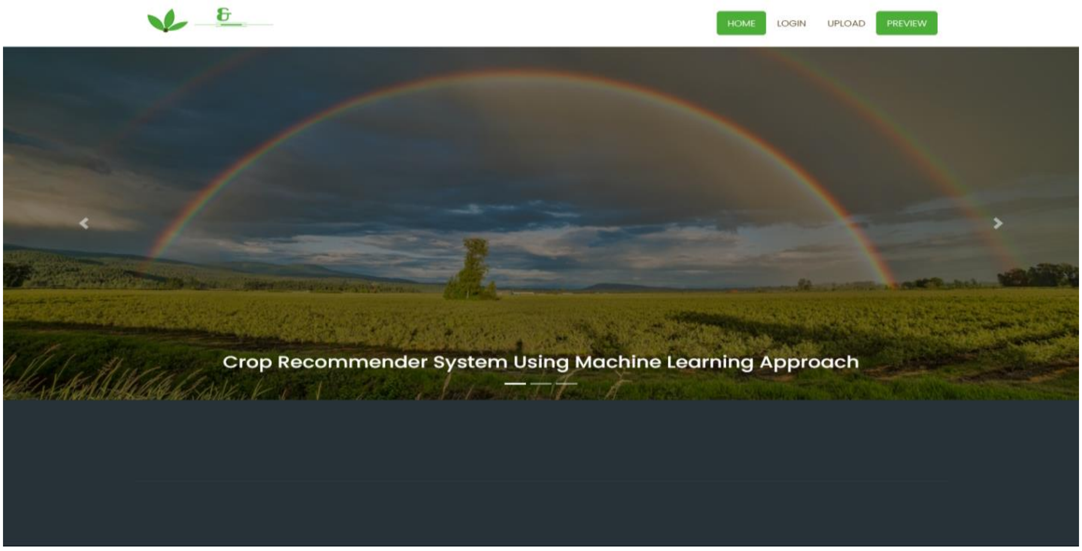


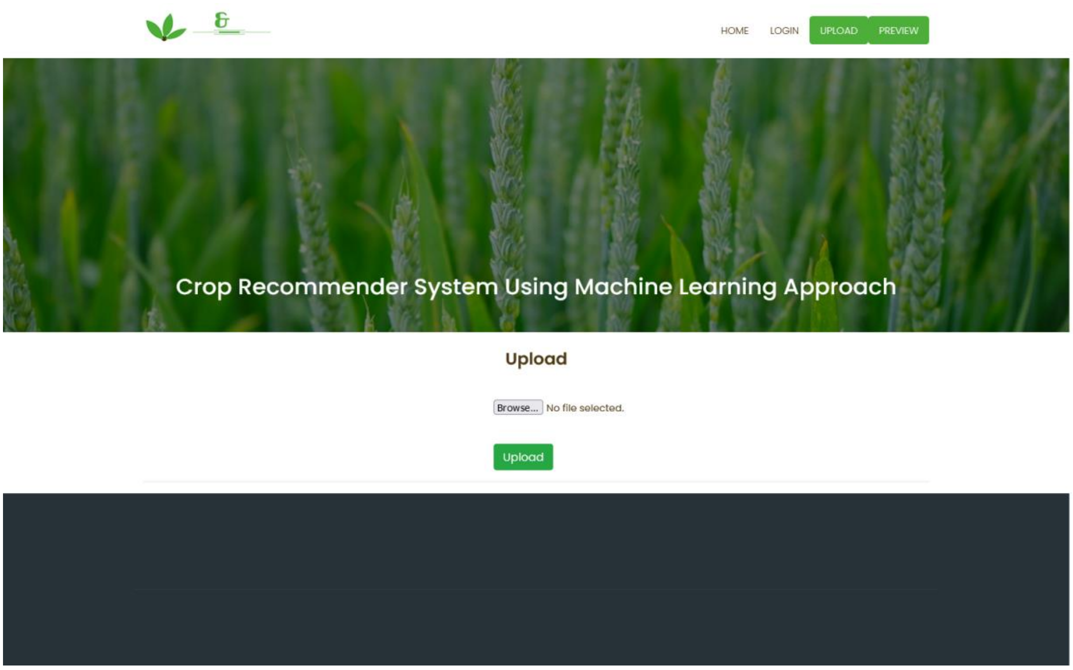
Fig 5.2 Graphical representation of accuracy scores for different models.

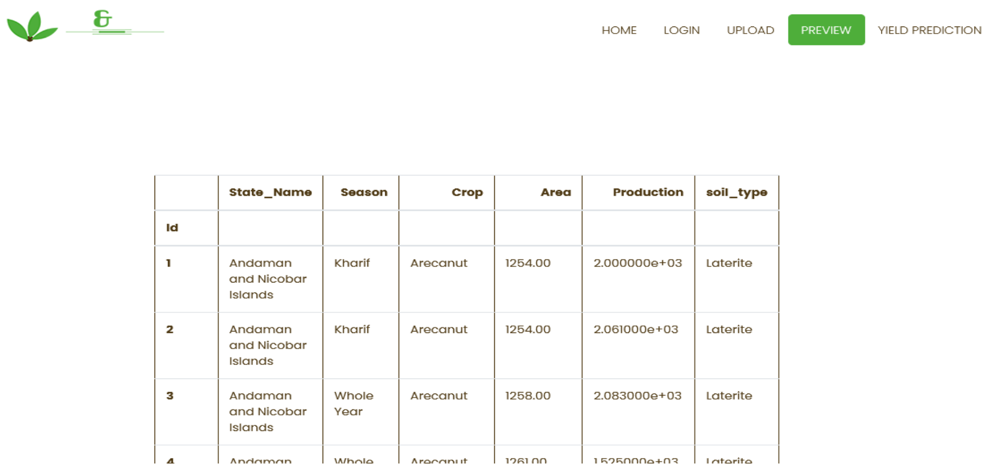
After conducting a comparative analysis of various machine learning models for Arrhythmia detection, we evaluated their predictive performance using accuracy, precision, recall, and F1 score metrics. The objective was to determine the effectiveness of these models in correctly classifying different Crop and soil categories. And factors like computational complexity, interpretability, and scalability should be considered when selecting the most suitable model for real-world deployment.

Output Screenshots:

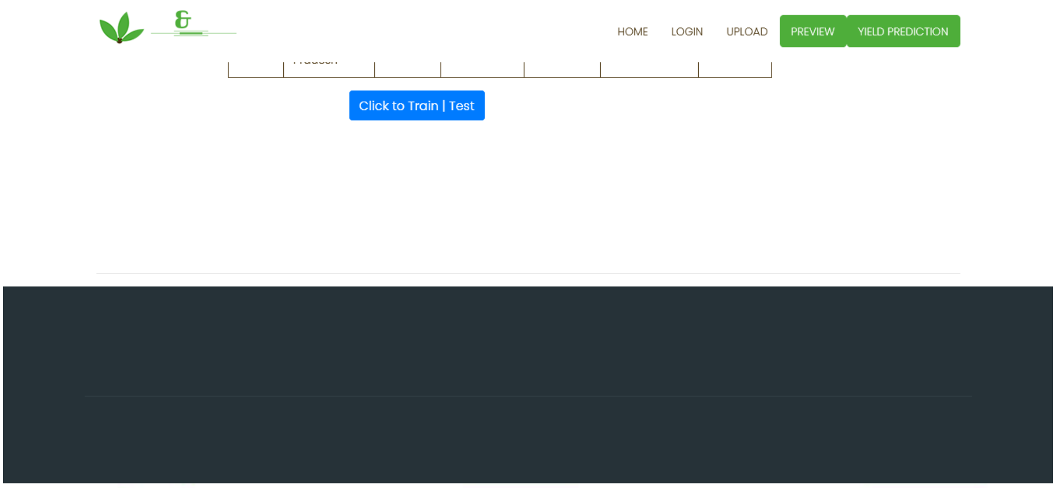
Fig 5.3 Screenshot of Outputs

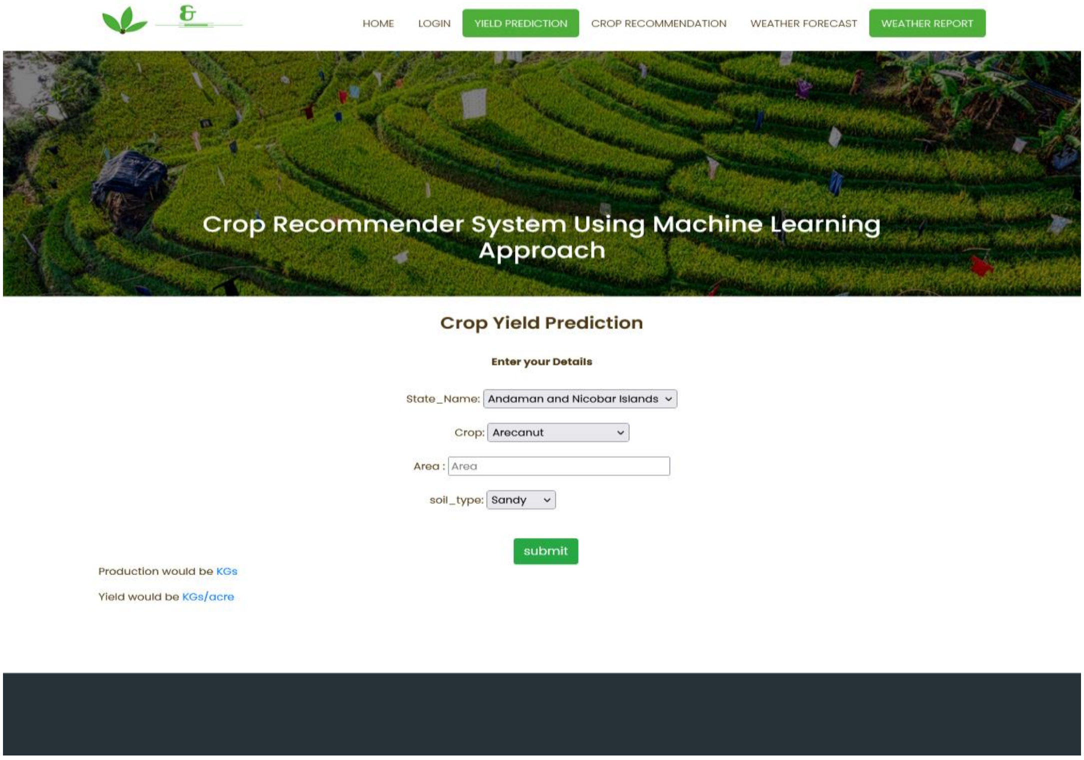


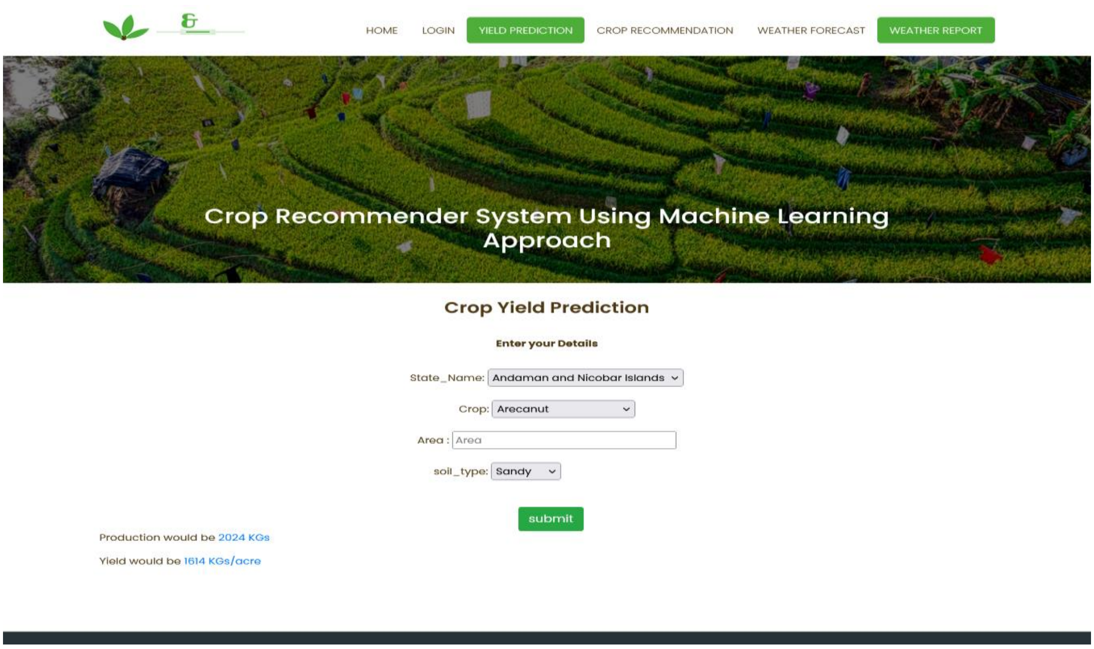


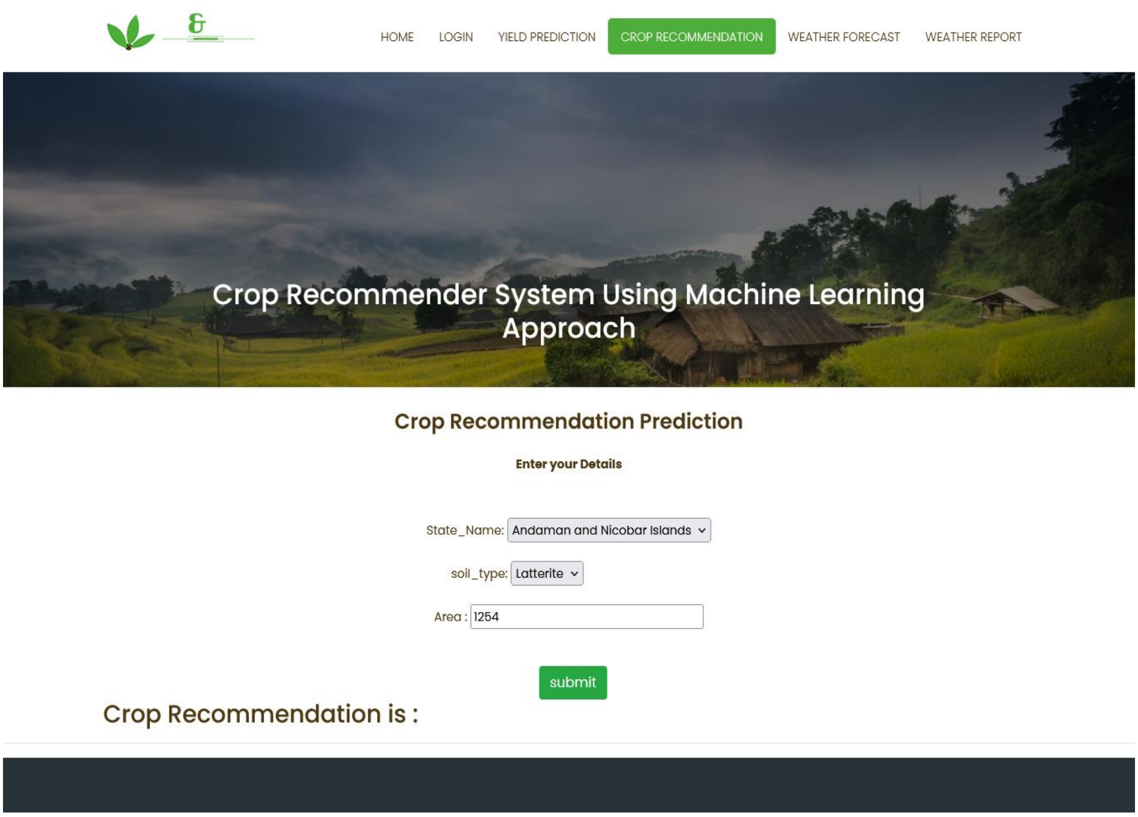


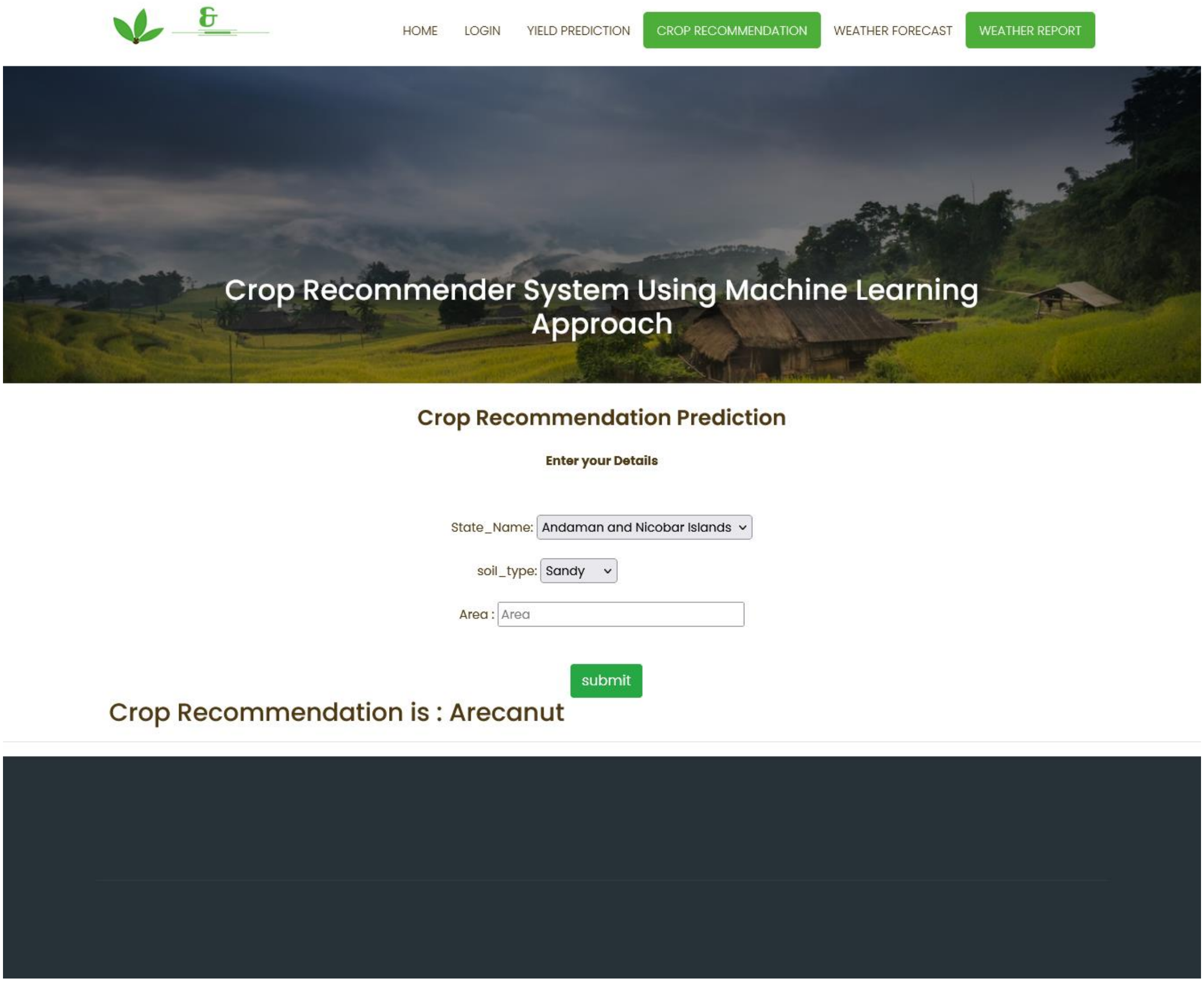
	State_Name	Season	Crop	Area	Production	soil_type
Id						
1	Andaman and Nicobar Islands	Kharif	Arecanut	1254.00	2.000000e+03	Laterite
2	Andaman and Nicobar Islands	Kharif	Arecanut	1254.00	2.061000e+03	Laterite
3	Andaman and Nicobar Islands	Whole Year	Arecanut	1258.00	2.083000e+03	Laterite
4	Andaman	Whole	Arecanut	1261.00	1.525000e+03	Laterite

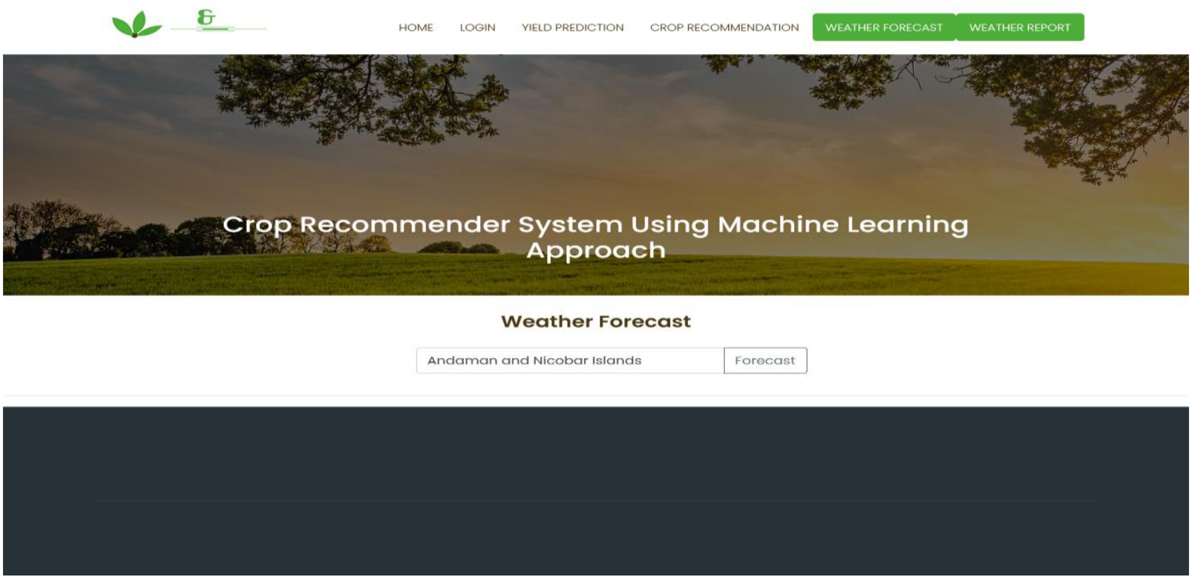

HOME
LOGIN
UPLOAD
PREVIEW
YIELD PREDICTION
Click to Train | Test

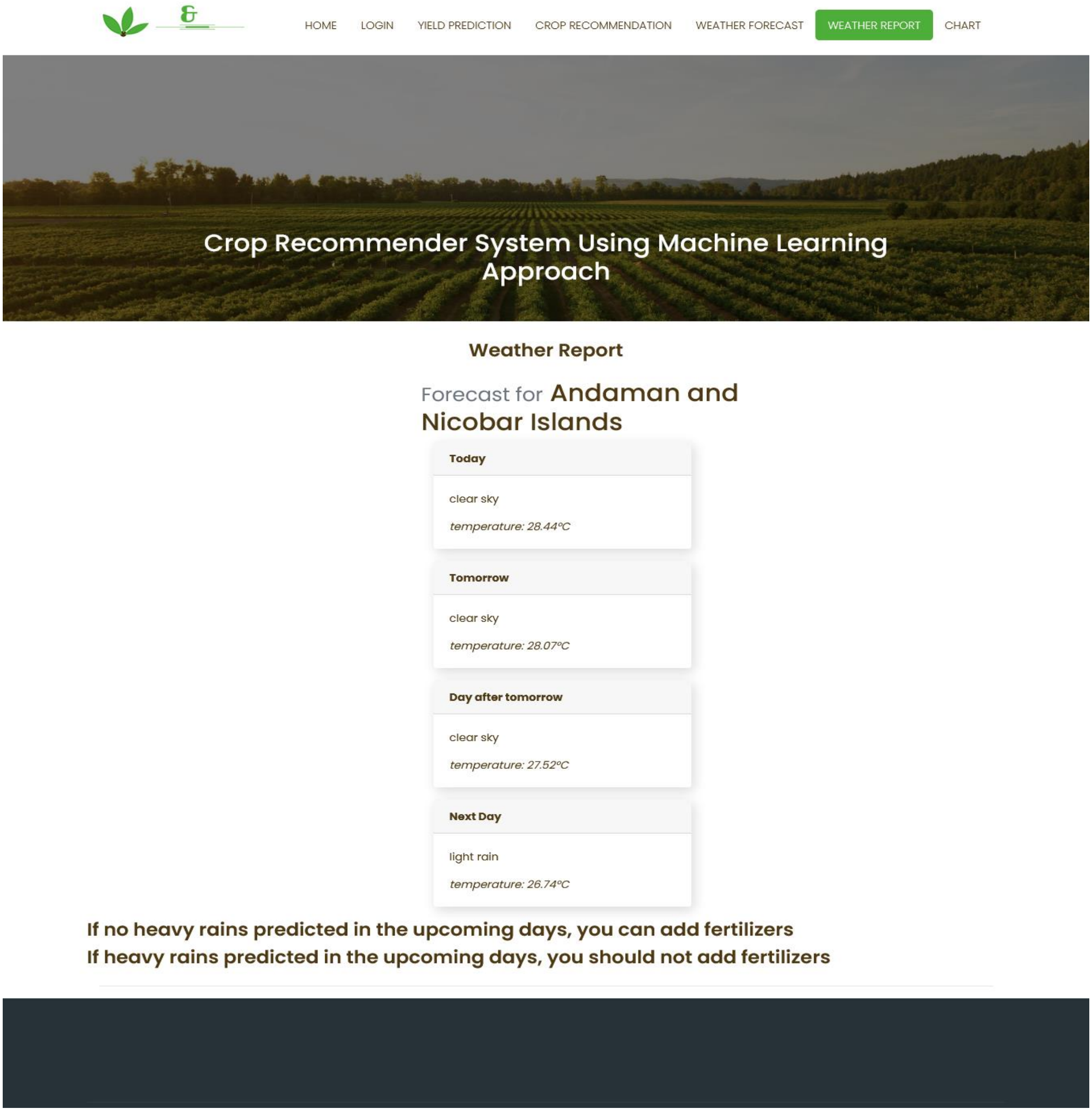

HOME
LOGIN
YIELD PREDICTION
CROP RECOMMENDATION
WEATHER FORECAST
WEATHER REPORT
Crop Recommender System Using Machine Learning Approach
Crop Yield Prediction
Enter your Details
State_Name: Andaman and Nicobar Islands
Crop: Arecanut
Area : Area
soil_type: Sandy
submit
Production would be KGs
Yield would be KGs/acre


HOME
LOGIN
YIELD PREDICTION
CROP RECOMMENDATION
WEATHER FORECAST
WEATHER REPORT
Crop Recommender System Using Machine Learning Approach
Crop Yield Prediction
Enter your Details
State_Name: Andaman and Nicobar Islands
Crop: Arecanut
Area : Area
soil_type: Sandy
submit
Production would be 2024 KGs
Yield would be 1614 KGs/acre


HOME
LOGIN
YIELD PREDICTION
CROP RECOMMENDATION
WEATHER FORECAST
WEATHER REPORT
Crop Recommender System Using Machine Learning Approach
Crop Recommendation Prediction
Enter your Details
State_Name: Andaman and Nicobar Islands
soil_type: Latterite
Area : 1254
submit
Crop Recommendation is :


HOME
LOGIN
YIELD PREDICTION
CROP RECOMMENDATION
WEATHER FORECAST
WEATHER REPORT
Crop Recommender System Using Machine Learning Approach
Crop Recommendation Prediction
Enter your Details
State_Name: Andaman and Nicobar Islands
soil_type: Sandy
Area : Area
submit
Crop Recommendation is : Arecanut


HOME
LOGIN
YIELD PREDICTION
CROP RECOMMENDATION
WEATHER FORECAST
WEATHER REPORT
Crop Recommender System Using Machine Learning Approach
Weather Forecast
Andaman and Nicobar Islands
Forecast


HOME
LOGIN
YIELD PREDICTION
CROP RECOMMENDATION
WEATHER FORECAST
WEATHER REPORT
CHART
Crop Recommender System Using Machine Learning Approach
Weather Report
Forecast for Andaman and Nicobar Islands
Today
clear sky
temperature: 28.44°C
Tomorrow
clear sky
temperature: 28.07°C
Day after tomorrow
clear sky
temperature: 27.52°C
Next Day
light rain
temperature: 26.74°C
If no heavy rains predicted in the upcoming days, you can add fertilizers
If heavy rains predicted in the upcoming days, you should not add fertilizers

Chapter 6:

6.1 Conclusion

In this machine learning project focused on crop suitability and weather forecasting recommendation system, we explored and compared multiple algorithms, including Support Vector Machines (SVM), Artificial Neural Networks (ANN), K-Nearest Neighbors (KNN), and Random Forest. Our goal was to determine the best algorithm that would provide accurate predictions for crop suitability and weather forecasting. After thorough experimentation and evaluation, we found that Random Forest exhibited the highest accuracy among the tested algorithms.

Random Forest, with its ensemble-based approach, demonstrated superior performance in predicting crop suitability and weather conditions. The algorithm's ability to handle complex relationships between features and reduce overfitting through ensemble averaging proved to be advantageous in our project. The high accuracy achieved by Random Forest indicates its potential to effectively analyze and classify the suitability of different crops based on various environmental factors and weather conditions.

Furthermore, the Random Forest algorithm's ability to handle large datasets and capture feature importance was instrumental in generating reliable recommendations for crop suitability and weather forecasting. By considering multiple decision trees and aggregating their outputs, the algorithm provided robust predictions, enhancing the reliability of our recommendation system. While SVM, ANN, and KNN also yielded promising results, Random Forest surpassed them in terms of accuracy. However, it is essential to note that the choice of the best algorithm may vary depending on the specific dataset, problem domain, and evaluation metrics employed.

In conclusion, our project demonstrated that employing Random Forest as the algorithm of choice in a crop suitability and weather forecasting recommendation system can lead to highly accurate predictions. The developed system can assist farmers, agronomists, and decision-makers in making informed choices regarding crop selection and agricultural planning, contributing to improved crop yields, resource optimization, and overall sustainability in the agricultural sector.

6.2 Scope for future work:

1. Integration of additional algorithms: While Random Forest yielded the best accuracy in your project, it's worth exploring other algorithms as well. Consider integrating and comparing the performance of advanced algorithms such as Gradient Boosting Machines (GBM), XGBoost, or Deep Learning models like Convolutional Neural Networks (CNN) or Recurrent Neural Networks (RNN). This can help ensure that you have explored a wide range of algorithmic approaches to achieve the best possible accuracy.

2. Incorporation of more weather and environmental variables: Expand the feature set used in the model by including more weather and environmental variables that might influence crop suitability. Explore additional data sources or gather more detailed weather data to capture factors such as soil moisture, temperature variations, wind patterns, or historical climate data. Incorporating such variables can enhance the accuracy and robustness of the recommendation system.

3. Spatial analysis and geospatial data: Consider incorporating geospatial data and spatial analysis techniques to account for location-specific factors. By integrating geographic information systems (GIS) data and geospatial analysis, you can capture localized variations in soil types, topography, precipitation, or sunlight exposure. This can help make the recommendation system more tailored to specific regions or even individual fields.

4. Real-time data and IoT integration: Explore the integration of real-time data sources and Internet of Things (IoT) devices to provide up-to-date and accurate information for weather forecasting and crop suitability. By incorporating data from sensors, weather stations, or satellite imagery, you can continuously update and improve the recommendation system's predictions, making it more dynamic and responsive to changing conditions.

5. User interface and visualization: Enhance the user interface (UI) of the recommendation system to provide a more intuitive and user-friendly experience. Implement visualizations, charts, and interactive dashboards to present the recommendations and weather forecasts in a visually appealing and easily understandable manner. This can enable users, such as farmers or agricultural experts, to interpret and utilize the system's outputs more effectively.

References:

[1] Umamaheswari S, Sreeram S, Kritika N, Prasanth DJ, "BIoT: Blockchain-based IoT for Agriculture", 11th International Conference on Advanced Computing (ICoAC), 2019 Dec 18 (pp. 324-327). IEEE.

[2] Jain A. "Analysis of growth and instability in the area, production, yield, and price of rice in India", Journal of Social Change and Development, 2018;2:46-66

[3] Manjula E, Djodiltachoumy S, "A model for prediction of crop yield" International Journal of Computational Intelligence and Informatics, 2017 Mar;6(4):2349-6363.

[4] Sagar BM, Cauvery NK., "Agriculture Data Analytics in Crop Yield Estimation: A Critical Review", Indonesian Journal of Electrical Engineering and Computer Science, 2018 Dec;12(3):1087-93.

[5] Wolfert S, Ge L, Verdouw C, Bogaardt MJ, "Big data in smart farming– a review. Agricultural Systems", 2017 May 1;153:69-80.

[6] Jones JW, Antle JM, Basso B, Boote KJ, Conant RT, Foster I, Godfray HC, Herrero M, Howitt RE, Janssen S, Keating BA, "Toward a new generation of agricultural system data, models, and knowledge products: State of agricultural systems science. Agricultural systems", 2017 Jul 1;155:269-88.

[7] Johnson LK, Bloom JD, Dunning RD, Gunter CC, Boyette MD, Creamer NG, "Farmer harvest decisions and vegetable loss in primary production. Agricultural Systems", 2019 Nov 1;176:102672.

[8] Kumar R, Singh MP, Kumar P, Singh JP, "Crop Selection Method to maximize crop yield rate using a machine learning technique", International conference on smart technologies and management for computing, communication, controls, energy, and materials (ICSTM), 2015 May 6 (pp. 138-145). IEEE.

[9] Sriram Rakshith.K, Dr.Deepak.G, Rajesh M, Sudharshan K S, Vasanth S, Harish Kumar N, "A Survey on Crop Prediction using Machine Learning Approach", In International

Journal for Research in Applied Science & Engineering Technology (IJRASET), April 2019, pp(3231- 3234)

[10] Veenadhari S, Misra B, Singh CD, “Machine learning approach for forecasting crop yield based on climatic parameters”, In 2014 International Conference on Computer Communication and Informatics, 2014 Jan 3 (pp. 1-5). IEEE.

[11] Ghadge R, Kulkarni J, More P, Nene S, Priya RL, “Prediction of crop yield using machine learning”, Int. Res. J. Eng. Technol. (IRJET), 2018 Feb;5.

[12] Priya P, Muthaiah U, Balamurugan M, “Predicting yield of the crop using machine learning algorithm”, International Journal of Engineering Sciences & Research Technology, 2018 Apr;7(1):1-7.

[13] S. Pavani, Augusta Sophy Beulet P., “Heuristic Prediction of Crop Yield Using Machine Learning Technique”, International Journal of Engineering and Advanced Technology (IJEAT), December 2019, pp (135-138)

Appendix - A

Source Code:

```
#importing the essential or required liberaries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

#Reading the csv file using pandas
train = pd.read_csv('crop_yield.csv')

#Getting the top(head) and bottom(tail) 5 rows and columns
train

#Getting the datatypes
train.dtypes

State_Name object
Season object
Crop object
Area float64
Production float64
soil_type object
dtype: object

#Testing weather the data has any inconsistencies or null values
train.isnull().sum()

State_Name 0
Season 0
Crop 0
Area 0
Production 0
soil_type 0
dtype: int64


#Getting the unique values from State_Name column
train['State_Name'].unique()
array(['Andaman and Nicobar Islands', 'Andhra Pradesh', 'Assam', 'Goa',
'Karnataka', 'Kerala', 'Meghalaya', 'Puducherry', 'Tamil Nadu',
'West Bengal', 'Bihar', 'Chhattisgarh', 'Dadra and Nagar Haveli',
'Gujarat', 'Haryana', 'Madhya Pradesh', 'Maharashtra', 'Manipur',
'Rajasthan', 'Telangana ', 'Uttar Pradesh', 'Arunachal Pradesh',
'Himachal Pradesh', 'Jammu and Kashmir ', 'Nagaland', 'Odisha',
'Uttarakhand', 'Mizoram', 'Punjab', 'Tripura', 'Chandigarh',
'Jharkhand', 'Sikkim'], dtype=object)

#Getting the unique values from Season column
train['Season'].unique()
array(['Kharif ', 'Whole Year ', 'Rabi ', 'Winter ', 'Autumn ', 'Summer '], dtype=object)

#Getting the unique values from Crop column
train['Crop'].unique()
array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 15, 12, 13, 14], dtype=object)

#Getting the unique values from soil_type column
train['soil_type'].unique()
array([' Laterite', 'Clayey', ' loamy', 'Sandy ', 'Black', 'Sandy', 'Alluvial', 'Loamy'], dtype=object)

#mapping certain crop names to corresponding numeric values
plt.rcParams["figure.figsize"] = (30,15)
train.loc[train['Crop']=='Arecanut', 'Crop'] = 0
train.loc[train['Crop']=='Banana', 'Crop'] =1
train.loc[train['Crop']=='Dry chillies', 'Crop'] =2
train.loc[train['Crop']=='Coconut ', 'Crop'] = 3
```

```
train.loc[train['Crop']=='Cotton(lint)', 'Crop'] =4
train.loc[train['Crop']=='Dry ginger', 'Crop'] = 5
train.loc[train['Crop']=='Groundnut', 'Crop'] =6
train.loc[train['Crop']=='Maize', 'Crop'] = 7
train.loc[train['Crop']=='Moong(Green Gram)', 'Crop'] =8
train.loc[train['Crop']=='Onion', 'Crop'] = 9
train.loc[train['Crop']=='Paddy', 'Crop'] =10
train.loc[train['Crop']=='Ragi', 'Crop'] =11
train.loc[train['Crop']=='Sugarcane', 'Crop'] =12
train.loc[train['Crop']=='Tobacco', 'Crop'] =13
train.loc[train['Crop']=='Turmeric', 'Crop'] =14
train.loc[train['Crop']=='Rice', 'Crop'] =15
train.loc[train['soil_type']=='Sandy', 'soil_type'] = 0
train.loc[train['soil_type']=='Sandy ', 'soil_type'] = 0
train.loc[train['soil_type']=='Red', 'soil_type'] =1
train.loc[train['soil_type']=='Black', 'soil_type'] =2
train.loc[train['soil_type']=='Clayey', 'soil_type'] = 3
train.loc[train['soil_type']=='Alluvial', 'soil_type'] =4
train.loc[train['soil_type']=='Loamy', 'soil_type'] = 5
train.loc[train['soil_type']==' loamy', 'soil_type'] = 5
train.loc[train['soil_type']==' Laterite', 'soil_type'] = 6

#mapping certain State_Name names to corresponding numeric values
train['State_Name'].unique()
array(['Andaman and Nicobar Islands', 'Andhra Pradesh', 'Assam', 'Goa',
'Karnataka', 'Kerala', 'Meghalaya', 'Puducherry', 'Tamil Nadu',
'West Bengal', 'Bihar', 'Chhattisgarh', 'Dadra and Nagar Haveli',
'Gujarat', 'Haryana', 'Madhya Pradesh', 'Maharashtra', 'Manipur',
'Rajasthan', 'Telangana ', 'Uttar Pradesh', 'Arunachal Pradesh',
'Himachal Pradesh', 'Jammu and Kashmir ', 'Nagaland', 'Odisha',
'Uttarakhand', 'Mizoram', 'Punjab', 'Tripura', 'Chandigarh',
'Jharkhand', 'Sikkim'], dtype=object)

#mapping certain State_Name names to corresponding numeric values
train.loc[train['State_Name']=='Andaman and Nicobar Islands', 'State_Name'] = 0
train.loc[train['State_Name']=='Andhra Pradesh', 'State_Name'] =1
train.loc[train['State_Name']=='Assam', 'State_Name'] =3
train.loc[train['State_Name']=='Goa', 'State_Name'] =4
train.loc[train['State_Name']=='Karnataka', 'State_Name'] =5
train.loc[train['State_Name']=='Kerala', 'State_Name'] =6
train.loc[train['State_Name']=='Meghalaya', 'State_Name'] =7
train.loc[train['State_Name']=='Puducherry', 'State_Name'] =8
train.loc[train['State_Name']=='Tamil Nadu', 'State_Name'] =9
train.loc[train['State_Name']=='West Bengal', 'State_Name'] =10
train.loc[train['State_Name']=='Bihar', 'State_Name'] =11
train.loc[train['State_Name']=='Chhattisgarh', 'State_Name'] =12
train.loc[train['State_Name']=='Dadra and Nagar Haveli', 'State_Name'] =13
train.loc[train['State_Name']=='Gujarat', 'State_Name'] =14
train.loc[train['State_Name']=='Haryana', 'State_Name'] =15
train.loc[train['State_Name']=='Madhya Pradesh', 'State_Name'] =16
train.loc[train['State_Name']=='Maharashtra', 'State_Name'] =17
train.loc[train['State_Name']=='Manipur', 'State_Name'] =18
train.loc[train['State_Name']=='Rajasthan', 'State_Name'] =19
train.loc[train['State_Name']=='Telangana ', 'State_Name'] =20
train.loc[train['State_Name']=='Uttar Pradesh', 'State_Name'] =21
train.loc[train['State_Name']=='Arunachal Pradesh', 'State_Name'] =22
train.loc[train['State_Name']=='Himachal Pradesh', 'State_Name'] =23
train.loc[train['State_Name']=='Jammu and Kashmir ', 'State_Name'] =24
train.loc[train['State_Name']=='Nagaland', 'State_Name'] =25
train.loc[train['State_Name']=='Odisha', 'State_Name'] =26
train.loc[train['State_Name']=='Uttarakhand', 'State_Name'] =27
train.loc[train['State_Name']=='Mizoram', 'State_Name'] =28
train.loc[train['State_Name']=='Punjab', 'State_Name'] =29
train.loc[train['State_Name']=='Tripura', 'State_Name'] =30
train.loc[train['State_Name']=='Chandigarh', 'State_Name'] =31
train.loc[train['State_Name']=='Jharkhand', 'State_Name'] =32
```

```
train.loc[train['State_Name']=='Sikkim', 'State_Name'] =33


#Getting the unique values from Crop column
train['Crop'].unique()
array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 15, 12, 13, 14], dtype=object)

#Getting the unique values from soil_type column
train['soil_type'].unique()
array([6, 3, 5, 0, 2, 4], dtype=object)

#Training the data based on State name, Crop and Soil type
X=train[["State_Name","Crop","Area","soil_type"]]
y=train["Production"]

#Splitting the data to train and test part , 80% for training and 20% for testing
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(X,y,test_size=0.2,random_state=0)

#Applied Random forest algorithm to get better accuracy (96%)
from sklearn.ensemble import RandomForestRegressor
regressor = RandomForestRegressor(n_estimators=16, random_state=0)
regressor.fit(X,y)

RandomForestRegressor(bootstrap=True, criterion='mse', max_depth=None,
max_features='auto', max_leaf_nodes=None,
min_impurity_decrease=0.0, min_impurity_split=None,
min_samples_leaf=1, min_samples_split=2,
min_weight_fraction_leaf=0.0, n_estimators=16,
n_jobs=None, oob_score=False, random_state=0, verbose=0,
warm_start=False)

#Applied K Nearest Neighbour algorithm to compare with algorithm gives better accuracy (90%)
from sklearn.neighbors import KNeighborsRegressor
regressor = KNeighborsRegressor(n_neighbors=5) or regressor = SVR(kernel='rbf', C=1.0, epsilon=0.1)
regressor.fit(X, y)


#Applied Artificial Neural Network algorithm to compare with algorithm gives better accuracy (86%)
from sklearn.neural_network import MLPRegressor
regressor = MLPRegressor(hidden_layer_sizes=(100, 50), activation='relu', solver='adam', random_state=0)
regressor.fit(X, y)

#Predicting the accuracy of the algorithms
print(regressor.score(X,y))
0.9684092471941279

#After predicting getting the values from X and Y
X
y

predictionss=regressor.predict([[0,0,1254.0,6]])
pred=format(int(predictionss[0]))

pred
'2024'

2064/1254.0
1.645933014354067

#Dumping this above model into pickle
import pickle
pickle.dump(regressor,open('yield.pkl','wb'))
model = pickle.load(open('yield.pkl', 'rb'))
```

Fig 1: Crop Recommendation System Dataset from Kaggle

	State_Name	Season	Crop	Area	Production	soil_type
0	Andaman and Nicobar Islands	Kharif	Arecanut	1254.0	2000.00	Laterite
1	Andaman and Nicobar Islands	Kharif	Arecanut	1254.0	2061.00	Laterite
2	Andaman and Nicobar Islands	Whole Year	Arecanut	1258.0	2083.00	Laterite
3	Andaman and Nicobar Islands	Whole Year	Arecanut	1261.0	1525.00	Laterite
4	Andaman and Nicobar Islands	Whole Year	Arecanut	1264.7	805.85	Laterite
...	...	...	...	...	...	...
94370	West Bengal	Whole Year	Turmeric	270.0	166.00	Loamy
94371	West Bengal	Whole Year	Turmeric	284.0	229.00	Loamy
94372	West Bengal	Whole Year	Turmeric	294.0	261.00	Loamy
94373	West Bengal	Whole Year	Turmeric	289.0	178.00	Loamy
94374	West Bengal	Whole Year	Turmeric	289.0	378.00	Loamy

94375 rows × 6 columns

Fig. 2 Types of Soils

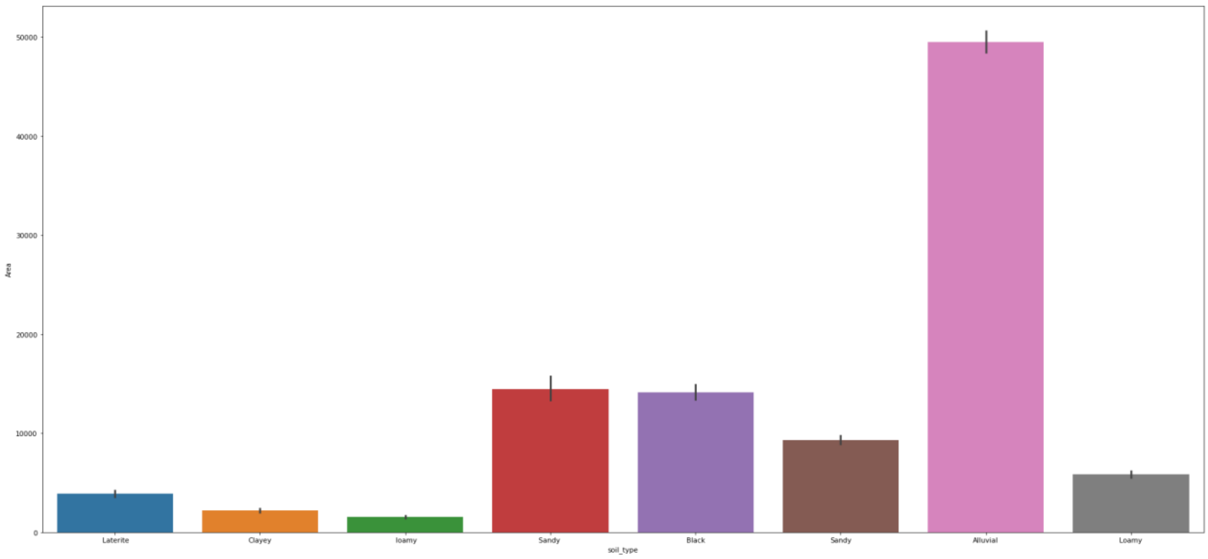
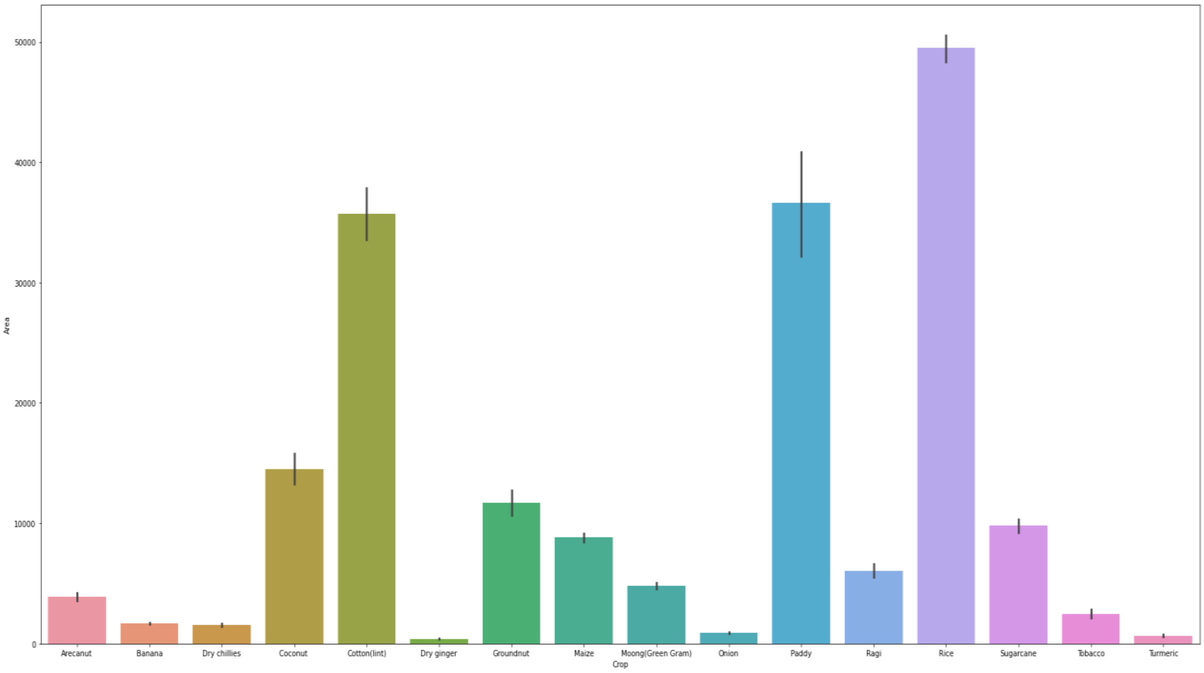


Fig 3. Types of Crops


50000
40000
30000
20000
10000
0
Area
Arecanut
Banana
Dry chillies
Coconut
Cotton(lint)
Dry ginger
Groundnut
Maize
Moong(Green Gram)
Onion
Paddy
Ragi
Rice
Sugarcane
Tobacco
Turmeric
Crop

Flask Code

`

python app.py

```
from flask import Flask, render_template, flash,redirect
import joblib
from flask import request
import numpy as np
import pickle
import pandas as pd
from flask import Flask, render_template, request, redirect, url_for
from weather import Weather, WeatherException

app = Flask(__name__)

app.config.from_pyfile('config/config.cfg')
w = Weather(app.config)

model = pickle.load(open('yield.pkl', 'rb'))
crop = pickle.load(open('crop.pkl', 'rb'))

@app.route('/')

@app.route('/index')
def index():
return render_template('index.html')
@app.route('/login')
def login():
return render_template('login.html')
@app.route('/chart')
def chart():
return render_template('chart.html')
@app.route('/upload')
def upload():
return render_template('upload.html')
@app.route('/preview',methods=["POST"])
def preview():
if request.method == 'POST':
dataset = request.files['datasetfile']
df = pd.read_csv(dataset,encoding = 'unicode_escape')
df.set_index('Id', inplace=True)
return render_template("preview.html",df_view = df)

@app.route('/yield_prediction')
def yield_prediction():
return render_template('yield_prediction.html')

@app.route("/predict",methods = ["POST"])
def predict():
if request.method == "POST":
print(request.form)
State_Name = request.form['State_Name']
Crop = request.form['Crop']
Area = request.form['Area']
Soil_type = request.form['soil_type']
pred_args = [State_Name,Crop,Area,Soil_type]
pred_args_arr = np.array(pred_args)
pred_args_arr = pred_args_arr.reshape(1,-1)
output = model.predict(pred_args_arr)
print(output)
pred=format(int(output[0]))
```

```
Yield= int(pred) / float(Area)
yields= Yield*1000

return render_template("yield_prediction.html",prediction_text=pred, yield_predictions= int(yields))
@app.route('/crop_prediction')
def crop_prediction():
return render_template('crop_prediction.html')

@app.route('/sandy',methods=['POST'])
def sandy():

int_features = [float(x) for x in request.form.values()]
final_features = [np.array(int_features)]
prediction = crop.predict(final_features)


preds=format((prediction[0]))

return render_template("crop_prediction.html",prediction_texts=preds)
@app.route('/homepage')
def homepage():
return render_template('homepage.html')

@app.route('/result', methods=['POST', 'GET'])
def result_page():
if request.method == 'POST':
location = request.form
w.set_location(location.get('location'))

try:
return render_template('result.html', data=w.get_forecast_data())
except WeatherException:
app.log_exception(WeatherException)
return render_template('error.html')
else:
return redirect(url_for('homepage'))
if __name__ == "__main__":
app.run(debug=True)
```

python weather.py

```
import requests


class Weather():

def __init__(self, config):
self.location = None
self.config = config

def set_location(self, location):
self.location = location

def get_location(self):
return self.location
```

```
def download_weather_data(self):
params = {
'q': self.location,
'appid': self.config['API_KEY'],
'units': 'metric'
}
try:
response = requests.get(self.config['API_URL'], params=params)
response.raise_for_status()
return response.json()
except requests.exceptions.HTTPError:
raise WeatherException("Weather for {} not found".format(self.location))

def get_forecast_data(self):
weather = self.download_weather_data()
w = weather['list']
current = w[0]['weather'][0]['description'], w[0]['main']['temp']
tomorrow = w[1]['weather'][0]['description'], w[1]['main']['temp']
dayafter = w[2]['weather'][0]['description'], w[2]['main']['temp']
Next = w[3]['weather'][0]['description'], w[3]['main']['temp']
return weather['city']['name'], current, tomorrow, dayafter, Next


class WeatherException(Exception):

def __init__(self, *args):
if args:
self.message = args[0]
else:
self.message = None

def __str__(self):
if self.message:
return self.message
else:
return "Weather not found!"
```

`

PHP Code

```php
<?php
// Check for empty fields
if(empty($_POST['name']) ||
empty($_POST['email']) ||
empty($_POST['phone']) ||
empty($_POST['message']) ||
!filter_var($_POST['email'],FILTER_VALIDATE_EMAIL))
{
echo "No arguments Provided!";
return false;
}
$name = strip_tags(htmlspecialchars($_POST['name']));
$email_address = strip_tags(htmlspecialchars($_POST['email']));
$phone = strip_tags(htmlspecialchars($_POST['phone']));
$message = strip_tags(htmlspecialchars($_POST['message']));
// Create the email and send the message
$to = 'yourname@yourdomain.com'; // Add your email address inbetween the '' replacing
yourname@yourdomain.com - This is where the form will send a message to.
$email_subject = "Website Contact Form: $name";
$email_body = "You have received a new message from your website contact form.\n\n"."Here are the
details:\n\nName: $name\n\nEmail: $email_address\n\nPhone: $phone\n\nMessage:\n$message";
$headers = "From: noreply@yourdomain.com\n"; // This is the email address the generated message will be
from. We recommend using something like noreply@yourdomain.com.
$headers .= "Reply-To: $email_address";
mail($to,$email_subject,$email_body,$headers);
return true;
?>
```

Web Code

index.html

```html
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
<meta name="description" content="">
<meta name="author" content="">
<title>crop prediction</title>
<!-- Bootstrap core CSS -->
<link href="../static/vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">
<!-- Fontawesome CSS -->
<link href="../static/css/all.css" rel="stylesheet">
<!-- Custom styles for this template -->
<link href="../static/css/style.css" rel="stylesheet">
</head>
<body>
<!-- Navigation -->
<nav class="navbar fixed-top navbar-expand-lg navbar-dark bg-light top-nav fixed-top">
<div class="container">
<a class="navbar-brand" href="index.html">
<img src="../static/images/footer-logo.png" alt="logo" />
</a>
```

```html
<button class="navbar-toggler navbar-toggler-right" type="button" data-toggle="collapse" data-target="#navbarResponsive" aria-controls="navbarResponsive" aria-expanded="false" aria-label="Toggle navigation">
<span class="fas fa-bars"></span>
</button>
<div class="collapse navbar-collapse" id="navbarResponsive">
<ul class="navbar-nav ml-auto">
<li class="nav-item">
<a class="nav-link active" href="index.html">Home</a>
</li>
<li class="nav-item">
<a class="nav-link" href="login.html">Login</a>
</li>
<li class="nav-item">
<a class="nav-link" href="upload.html">Upload</a>
</li>
<li class="nav-item">
<a class="nav-link" href="preview.html">Preview</a>
</li>
</ul>
</div>
</div>
</nav>
<header class="slider-main">
<div id="carouselExampleIndicators" class="carousel slide carousel-fade" data-ride="carousel">
<ol class="carousel-indicators">
<li data-target="#carouselExampleIndicators" data-slide-to="0" class="active"></li>
<li data-target="#carouselExampleIndicators" data-slide-to="1"></li>
<li data-target="#carouselExampleIndicators" data-slide-to="2"></li>
</ol>
<div class="carousel-inner" role="listbox">
<!-- Slide One - Set the background image for this slide in the line below -->
<div class="carousel-item active" style="background-image: url('../static/images/slider-01.jpg')">
<div class="carousel-caption d-none d-md-block">
<h3>Crop Recommender System Using Machine Learning Approach</h3>
</div>
</div>
<!-- Slide Two - Set the background image for this slide in the line below -->
<div class="carousel-item" style="background-image: url('../static/images/slider-02.jpg')">
<div class="carousel-caption d-none d-md-block">
<h3>Crop Recommender System Using Machine Learning Approach</h3>
</div>
</div>
<!-- Slide Three - Set the background image for this slide in the line below -->
<div class="carousel-item" style="background-image: url('../static/images/slider-03.jpg')">
<div class="carousel-caption d-none d-md-block">
<h3>Crop Recommender System Using Machine Learning Approach</h3>
</div>
</div>
</div>
<a class="carousel-control-prev" href="#carouselExampleIndicators" role="button" data-slide="prev">
<span class="carousel-control-prev-icon" aria-hidden="true"></span>
<span class="sr-only">Previous</span>
</a>
<a class="carousel-control-next" href="#carouselExampleIndicators" role="button" data-slide="next">
<span class="carousel-control-next-icon" aria-hidden="true"></span>
<span class="sr-only">Next</span>
</a>
</div>
</header>
```

```html
<!-- Page Content -->
<!-- /.container -->
<!--footer starts from here-->
<footer class="footer">
<div class="container bottom_border">
<div class="row">
<div class="col-lg-3 col-md-6 col-sm-6 col">
<h5 class="headin5_amrc col_white_amrc pt2"></h5>
<!--headin5_amrc ends here-->
<!--footer_ul2_amrc ends here-->
</div>
<div class="col-lg-3 col-md-6 col-sm-6">
<h5 class="headin5_amrc col_white_amrc pt2"></h5>
<!--headin5_amrc-->
<!--footer_ul_amrc ends here-->
</div>
<div class="col-lg-3 col-md-6 col-sm-6 ">
<h5 class="headin5_amrc col_white_amrc pt2"></h5>
<!--headin5_amrc-->
<ul class="footer_ul_amrc">
<li class="media">
<div class="media-body">
<p></p>
<span></span>
</div>
</li>
<li class="media">
<div class="media-body">
<p></p>
<span></span>
</div>
</li>
<li class="media">
<div class="media-body">
<p><p>
<span></span>
</div>
</li>
</ul>
<!--footer_ul_amrc ends here-->
</div>
</div>
</div>
<div class="container">
<div class="footer-logo">
<a href="#"><img src="images/footer-logo.png" alt="" /></a>
</div>
<!--foote_bottom_ul_amrc ends here-->
<!--social_footer_ul ends here-->
</div>
</footer>
<!-- Bootstrap core JavaScript -->
<script src="../static/vendor/jquery/jquery.min.js"></script>
<script src="../static/vendor/bootstrap/js/bootstrap.bundle.min.js"></script>
</body>
</html>
```

chart.html

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
<meta name="description" content="">
<meta name="author" content="">
<title>chart </title>
<!-- Bootstrap core CSS -->
<link href="/static/vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">
<!-- Fontawesome CSS -->
<link href="/static/css/all.css" rel="stylesheet">
<!-- Custom styles for this template -->
<link href="/static/css/styles.css" rel="stylesheet">
</head>
<body>
<!-- Navigation -->
<nav class="navbar fixed-top navbar-expand-lg navbar-dark bg-light top-nav fixed-top">
<div class="container">
<a class="navbar-brand" href="index.html">
<img src="/static/images/footer-logo.png" alt="logo" />
</a>
<button class="navbar-toggler navbar-toggler-right" type="button" data-toggle="collapse" data-
target="#navbarResponsive" aria-controls="navbarResponsive" aria-expanded="false" aria-label="Toggle
navigation">
<span class="fas fa-bars"></span>
</button>
<div class="collapse navbar-collapse" id="navbarResponsive">
<ul class="navbar-nav ml-auto">
<li class="nav-item">
<a class="nav-link " href="index.html">Home</a>
</li>
<li class="nav-item">
<a class="nav-link" href="login.html">Login</a>
</li>
<li class="nav-item">
<a class="nav-link " href="yield_prediction.html">Yield Prediction</a>
</li>
<li class="nav-item">
<a class="nav-link" href="crop_prediction.html">Crop Recommendation</a>
</li>
<li class="nav-item">
<a class="nav-link" href="homepage.html">Weather Forecast </a>
</li>
<li class="nav-item">
<a class="nav-link " href="homepage.html">Weather Report </a>
</li>
<li class="nav-item">
<a class="nav-link active" href="chart.html"> chart</a>
</li>
</ul>
</div>
</div>
</nav>
<header class="slider-main">
<div id="carouselExampleIndicators" class="carousel slide carousel-fade" data-ride="carousel">
<div class="carousel-inner" role="listbox">
<!-- Slide One - Set the background image for this slide in the line below -->
```

```
<div class="carousel-item active" style="background-image: url('../static/images/slider-13.jpg')">
<div class="carousel-caption d-none d-md-block">
<h2>Crop Recommender System Using Machine Learning Approach</h2>
</div>
</div>
<!-- Slide Two - Set the background image for this slide in the line below -->
<!-- Slide Three - Set the background image for this slide in the line below -->
</div>
</div>
</header>
<!-- Page Content -->
<div class="container">
<div class="services-bar">
<center><h1 class="my-4">chart </h1></center>
<!-- Services Section -->
<div class="row">
<!-- Vendor CSS Files -->


<div class="col-md-5 col-sm-5 col-xs-5" style="margin-left:180px;">
<html>
<head>
<script type="text/javascript" src="https://www.gstatic.com/charts/loader.js"></script>
<script type="text/javascript">
google.charts.load("current", {packages:['corechart']});
google.charts.setOnLoadCallback(drawChart);
function drawChart() {
var data = google.visualization.arrayToDataTable([
["crop", "Area", { role: "style" } ],
["Rice", 5000, "#b87333"],
["Paddy", 4000, "#76A7FA"],
["cotton", 3500, "gold"],
["coconut", 2000, "color: #703593"],
["sugarcane", 2260, "color: #0000FF"]
]);

var view = new google.visualization.DataView(data);
view.setColumns([0, 1,
{ calc: "stringify",
sourceColumn: 1,
type: "string",
role: "annotation" },
2]);

var options = {
title: "",
width: 600,
height: 400,
bar: {groupWidth: "95%"},
legend: { position: "none" },
};
var chart = new google.visualization.ColumnChart(document.getElementById("columnchart_values"));
chart.draw(view, options);
}
</script>
<center><div id="columnchart_values" style="width: 900px; height: 300px;"></div> </center>
</body>
</html>
</div>
</br>
```

`

```
</br>
</br>
</br>
<div class="col-md-5 col-sm-5 col-xs-5" style="margin-left:220px;">
<html>
<head>
<script type="text/javascript" src="https://www.gstatic.com/charts/loader.js"></script>
<script type="text/javascript">
google.charts.load('current', {'packages':['corechart']});
google.charts.setOnLoadCallback(drawChart);

function drawChart() {

var data = google.visualization.arrayToDataTable([
['soil_type', 'Area'],
['Clayey', 11],
['alluvial', 2],
['Red ', 2],
['Black ', 2],
['sandy ', 7]
]);

var options = {
title: ' Soil Occupied Area'
};

var chart = new google.visualization.PieChart(document.getElementById('piechart'));

chart.draw(data, options);
}
</script>
</head>
<body>
<div id="piechart" style="width: 900px; height: 500px;"></div>
</body>
</html>


</div>


</div>
<!-- /.row -->
</div>
<!-- Portfolio Section -->
<hr>
<!-- Get In Touch Now Section -->
</div>
<!-- /.container -->
<!--footer starts from here-->
<!-- Bootstrap core JavaScript -->
<script src="/static/vendor/jquery/jquery.min.js"></script>
<script src="/static/vendor/bootstrap/js/bootstrap.bundle.min.js"></script>
</body>
</html>
```

crop_prediction.html

```
<!DOCTYPE html>
<html lang="en">
```

```html
<head>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
<meta name="description" content="">
<meta name="author" content="">
<title>crop prediction </title>
<!-- Bootstrap core CSS -->
<link href="../static/vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">
<!-- Fontawesome CSS -->
<link href="../static/css/all.css" rel="stylesheet">
<!-- Custom styles for this template -->
<link href="../static/css/styles.css" rel="stylesheet">
</head>
<body>
<!-- Navigation -->
<nav class="navbar fixed-top navbar-expand-lg navbar-dark bg-light top-nav fixed-top">
<div class="container">
<a class="navbar-brand" href="index.html">
<img src="../static/images/footer-logo.png" alt="logo" />
</a>
<button class="navbar-toggler navbar-toggler-right" type="button" data-toggle="collapse" data-
target="#navbarResponsive" aria-controls="navbarResponsive" aria-expanded="false" aria-label="Toggle
navigation">
<span class="fas fa-bars"></span>
</button>
<div class="collapse navbar-collapse" id="navbarResponsive">
<ul class="navbar-nav ml-auto">
<li class="nav-item">
<a class="nav-link " href="index.html">Home</a>
</li>
<li class="nav-item">
<a class="nav-link" href="login.html">Login</a>
</li>
<li class="nav-item">
<a class="nav-link " href="yield_prediction.html">Yield Prediction</a>
</li>
<li class="nav-item">
<a class="nav-link active" href="crop_prediction.html">Crop Recommendation</a>
</li>
<li class="nav-item">
<a class="nav-link" href="homepage.html">Weather Forecast </a>
</li>
<li class="nav-item">
<a class="nav-link" href="homepage.html">Weather Report </a>
</li>
</ul>
</div>
</div>
</nav>
<header class="slider-main">
<div id="carouselExampleIndicators" class="carousel slide carousel-fade" data-ride="carousel">
<div class="carousel-inner" role="listbox">
<!-- Slide One - Set the background image for this slide in the line below -->
<div class="carousel-item active" style="background-image: url('../static/images/slider-08.jpg')">
<div class="carousel-caption d-none d-md-block">
<h2>Crop Recommender System Using Machine Learning Approach</h2>
</div>
</div>
<!-- Slide Two - Set the background image for this slide in the line below -->
<!-- Slide Three - Set the background image for this slide in the line below -->
```

```
</div>
</div>
</header>
<!-- Page Content -->
<div class="container">
<div class="services-bar">
<center><h1 class="my-4">Crop Recommendation Prediction</h1></center>
<!-- Services Section -->
<div class="row">
<form class="form-horizontal" action="{{ url_for('sandy')}}" method="POST">
</div>
<div class="span5">
<div class="control-group">
<center><label class="control-label" for="inputMessage"><b>Enter your Details</b></label></br></center>
</br>
</br>
<center> <div class="controls">
State_Name: <select type="text" name="State_Name" required="required" placeholder="State_Name">
<option value="0">Andaman and Nicobar Islands</option>
<option value="1">Andhra Pradesh</option>
<option value="3">Assam</option>
<option value="4">Goa</option>
<option value="5">Karnataka</option>
<option value="6">Kerala</option>
<option value="7">Meghalaya</option>
<option value="8">Puducherry</option>
<option value="9">Tamil Nadu</option>
<option value="10">West Bengal</option>
<option value="11">Bihar</option>
<option value="12">Chhattisgarh</option>
<option value="13">Dadra and Nagar Haveli</option>
<option value="14">Gujarat</option>
<option value="15">Haryana</option>
<option value="16">Madhya Pradesh</option>
<option value="17">Maharashtra</option>
<option value="18">Manipur</option>
<option value="19">Rajasthan</option>
<option value="20">Telangana</option>
<option value="21">Uttar Pradesh</option>
<option value="22">Arunachal Pradesh</option>
<option value="23">Himachal Pradesh</option>
<option value="24">Jammu and Kashmir</option>
<option value="25">Nagaland</option>
<option value="26">Odisha</option>
<option value="27">Uttarakhand</option>
<option value="28">Mizoram</option>
<option value="29">Punjab</option>
<option value="30">Tripura</option>
<option value="31">Chandigarh</option>
<option value="32">Jharkhand</option>
<option value="33">Sikkim</option>
</select>
</div></center>
</br>
<div class="controls">
                                   
```

```
                                    
                                    
soil_type: <select type="text" name="soil_type" required="required" placeholder="soil_type">
<option value="0">Sandy</option>
<option value="1">Red</option>
<option value="2">Black</option>
<option value="3">Clayey</option>
<option value="4">Alluvial</option>
<option value="5">Loamy</option>
<option value="6">Latterite</option>
</select>
</div>
</br>
<center> Area : <input type="text" name="Area" placeholder="Area" required="required" /></center>
</br>
</br>
<center><input type="submit" class="btn btn-success" value="submit" style="margin-left: 10px" ></center>
</div>
</div>
</form>
<h3>Crop Recommendation is : {{prediction_texts}}</h3>
</div>
<!-- /.row -->
</div>
<!-- Portfolio Section -->
<hr>
<!-- Get In Touch Now Section -->
</div>
<!-- /.container -->
<!--footer starts from here-->
<footer class="footer">
<div class="container bottom_border">
<div class="row">
<div class="col-lg-3 col-md-6 col-sm-6 col">
<h5 class="headin5_amrc col_white_amrc pt2"></h5>
<!--headin5_amrc ends here-->
<!--footer_ul2_amrc ends here-->
</div>
<div class="col-lg-3 col-md-6 col-sm-6">
<h5 class="headin5_amrc col_white_amrc pt2"></h5>
<!--headin5_amrc-->
<!--footer_ul_amrc ends here-->
</div>
<div class="col-lg-3 col-md-6 col-sm-6 ">
<h5 class="headin5_amrc col_white_amrc pt2"></h5>
<!--headin5_amrc-->
<ul class="footer_ul_amrc">
<li class="media">
<div class="media-body">
<p></p>
<span></span>
</div>
</li>
<li class="media">
<div class="media-body">
<p></p>
```

```
<span></span>
</div>
</li>
<li class="media">
<div class="media-body">
<p><p>
<span></span>
</div>
</li>
</ul>
<!--footer_ul_amrc ends here-->
</div>
</div>
</div>
<div class="container">
<div class="footer-logo">
<a href="#"><img src="images/footer-logo.png" alt="" /></a>
</div>
<!--foote_bottom_ul_amrc ends here-->
<!--social_footer_ul ends here-->
</div>
</footer>
<!-- Bootstrap core JavaScript -->
<script src="../static/vendor/jquery/jquery.min.js"></script>
<script src="../static/vendor/bootstrap/js/bootstrap.bundle.min.js"></script>
</body>
</html>
```

homepage.html

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
<meta name="description" content="">
<meta name="author" content="">
<title>weather report </title>
<!-- Bootstrap core CSS -->
<link href="../static/vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">
<!-- Fontawesome CSS -->
<link href="../static/css/all.css" rel="stylesheet">
<!-- Custom styles for this template -->
<link href="../static/css/styles.css" rel="stylesheet">
</head>
<body>
<!-- Navigation -->
<nav class="navbar fixed-top navbar-expand-lg navbar-dark bg-light top-nav fixed-top">
<div class="container">
<a class="navbar-brand" href="index.html">
<img src="../static/images/footer-logo.png" alt="logo" />
</a>
<button class="navbar-toggler navbar-toggler-right" type="button" data-toggle="collapse" data-
target="#navbarResponsive" aria-controls="navbarResponsive" aria-expanded="false" aria-label="Toggle
navigation">
<span class="fas fa-bars"></span>
</button>
<div class="collapse navbar-collapse" id="navbarResponsive">
<ul class="navbar-nav ml-auto">
```

```
<li class="nav-item">
<a class="nav-link " href="index.html">Home</a>
</li>
<li class="nav-item">
<a class="nav-link" href="login.html">Login</a>
</li>
<li class="nav-item">
<a class="nav-link " href="yield_prediction.html">Yield Prediction</a>
</li>
<li class="nav-item">
<a class="nav-link" href="crop_prediction.html">Crop Recommendation</a>
</li>
<li class="nav-item">
<a class="nav-link active" href="homepage.html">Weather Forecast </a>
</li>
<li class="nav-item">
<a class="nav-link" href="homepage.html">Weather Report </a>
</li>
</ul>
</div>
</div>
</nav>
<header class="slider-main">
<div id="carouselExampleIndicators" class="carousel slide carousel-fade" data-ride="carousel">
<div class="carousel-inner" role="listbox">
<!-- Slide One - Set the background image for this slide in the line below -->
<div class="carousel-item active" style="background-image: url('../static/images/slider-11.jpg')">
<div class="carousel-caption d-none d-md-block">
<h2>Crop Recommender System Using Machine Learning Approach</h2>
</div>
</div>
<!-- Slide Two - Set the background image for this slide in the line below -->
<!-- Slide Three - Set the background image for this slide in the line below -->
</div>
</div>
</header>
<!-- Page Content -->
<div class="container">
<div class="services-bar">
<center><h1 class="my-4">Weather Forecast </h1></center>
<!-- Services Section -->
<div class="row">
<div class="col-md-5 col-sm-5 col-xs-5" style="margin-left:350px;">
<form action="/result" method="POST" id="search">
<center><div class="input-group mb-3">
<input type="text"
name="location"
class="form-control"
id="txtArea"
placeholder="Search for a Location"
aria-label="Location"
aria-describedby="button-addon2"
required
>
<div class="input-group-append">
<button class="btn btn-outline-secondary" id="button-addon2" type="submit">
Forecast
</button></center>
</div>
</div></center>
```

```
</form>
</div>
</div>
<!-- /.row -->
</div>
<!-- Portfolio Section -->
<hr>
<!-- Get In Touch Now Section -->
</div>
<!-- /.container -->
<!--footer starts from here-->
<footer class="footer">
<div class="container bottom_border">
<div class="row">
<div class="col-lg-3 col-md-6 col-sm-6 col">
<h5 class="headin5_amrc col_white_amrc pt2"></h5>
<!--headin5_amrc ends here-->
<!--footer_ul2_amrc ends here-->
</div>
<div class="col-lg-3 col-md-6 col-sm-6">
<h5 class="headin5_amrc col_white_amrc pt2"></h5>
<!--headin5_amrc-->
<!--footer_ul_amrc ends here-->
</div>
<div class="col-lg-3 col-md-6 col-sm-6 ">
<h5 class="headin5_amrc col_white_amrc pt2"></h5>
<!--headin5_amrc-->
<ul class="footer_ul_amrc">
<li class="media">
<div class="media-body">
<p></p>
<span></span>
</div>
</li>
<li class="media">
<div class="media-body">
<p></p>
<span></span>
</div>
</li>
<li class="media">
<div class="media-body">
<p><p>
<span></span>
</div>
</li>
</ul>
<!--footer_ul_amrc ends here-->
</div>
</div>
</div>
<div class="container">
<div class="footer-logo">
<a href="#"><img src="images/footer-logo.png" alt="" /></a>
</div>
<!--foote_bottom_ul_amrc ends here-->
<!--social_footer_ul ends here-->
</div>
</footer>
<!-- Bootstrap core JavaScript -->
```

`

```
<script src="../static/vendor/jquery/jquery.min.js"></script>
<script src="../static/vendor/bootstrap/js/bootstrap.bundle.min.js"></script>
</body>
</html>
```

login.html

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
<meta name="description" content="">
<meta name="author" content="">
<title>Login </title>
<!-- Bootstrap core CSS -->
<link href="../static/vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">
<!-- Fontawesome CSS -->
<link href="../static/css/all.css" rel="stylesheet">
<!-- Custom styles for this template -->
<link href="../static/css/styles.css" rel="stylesheet">
</head>
<body>
<!-- Navigation -->
<nav class="navbar fixed-top navbar-expand-lg navbar-dark bg-light top-nav fixed-top">
<div class="container">
<a class="navbar-brand" href="index.html">
<img src="../static/images/footer-logo.png" alt="logo" />
</a>
<button class="navbar-toggler navbar-toggler-right" type="button" data-toggle="collapse" data-
target="#navbarResponsive" aria-controls="navbarResponsive" aria-expanded="false" aria-label="Toggle
navigation">
<span class="fas fa-bars"></span>
</button>
<div class="collapse navbar-collapse" id="navbarResponsive">
<ul class="navbar-nav ml-auto">
<li class="nav-item">
<a class="nav-link" href="index.html">Home</a>
</li>
<li class="nav-item">
<a class="nav-link active" href="login.html">Login</a>
</li>
<li class="nav-item">
<a class="nav-link" href="upload.html">Upload</a>
</li>
<li class="nav-item">
<a class="nav-link" href="preview.html">Preview</a>
</li>
</ul>
</div>
</div>
</nav>
<header class="slider-main">
<div id="carouselExampleIndicators" class="carousel slide carousel-fade" data-ride="carousel">
<div class="carousel-inner" role="listbox">
<!-- Slide One - Set the background image for this slide in the line below -->
<div class="carousel-item active" style="background-image: url('../static/images/slider-04.jpg')">
<div class="carousel-caption d-none d-md-block">
```

```
<h2>Crop Recommender System Using Machine Learning Approach</h2>
</div>
</div>
<!-- Slide Two - Set the background image for this slide in the line below -->
<!-- Slide Three - Set the background image for this slide in the line below -->
</div>
</div>
</header>
<!-- Page Content -->
<div class="container">
<div class="services-bar">
<center><h1 class="my-4">LOGIN </h1></center>
<!-- Services Section -->
<div class="row">
<head>

<style>


</style>
<script>
addEventListener("load", function () {
setTimeout(hideURLbar, 0);
}, false);

function hideURLbar() {
window.scrollTo(0, 1);
}

function login(){
var uname = document.getElementById("uname").value;
var pwd = document.getElementById("pwd").value;

if(uname == "admin" && pwd == "admin")
{
alert("Login Success!");
window.location = "upload.html";
return false;
}
else
{
alert("Invalid Credentials!")
}
}



</script>
</head>

<body id="page-top">


<!-- Portfolio Section -->
<section class="page-section portfolio" id="portfolio">
<br>
<br>
<!-- Portfolio Section Heading -->
<!-- Icon Divider -->

<!-- Portfolio Grid Items -->
```

`

```
<div class="row">

<!-- Portfolio Item 1 -->
<div class="col-md-6 col-lg-4" style="margin-left:480px">
<div class="control-group">
<!-- Username -->
<label class="control-label" for="username"><b>Username</b></label>
<div class="controls">
<input type="text" id="uname" name="uname" placeholder="" class="form-control">
</div>
</div>
</br>
<div class="control-group">
<!-- Password-->
<label class="control-label" for="password"><b>Password</b></label>
<div class="controls">
<input type="password" id="pwd" name="pwd" placeholder="" class="form-control">
</div>
</div>
<div class="col-md-6 col-lg-4" style="margin-left:-190px">
<div class="control-group">
<!-- Button -->
<br>
<div class="controls">
<input type="button" class="btn btn-success" value="Login" style="margin-left: 230px" onclick="login()">
</div>
</div>
</div>
</div>
</div>
</section>



</body>
</div>
<!-- /.row -->
</div>
<!-- Portfolio Section -->
<hr>
<!-- Get In Touch Now Section -->
</div>
<!-- /.container -->
<!--footer starts from here-->
<footer class="footer">
<div class="container bottom_border">
<div class="row">
<div class="col-lg-3 col-md-6 col-sm-6 col">
<h5 class="headin5_amrc col_white_amrc pt2"></h5>
<!--headin5_amrc ends here-->
<!--footer_ul2_amrc ends here-->
</div>
<div class="col-lg-3 col-md-6 col-sm-6">
<h5 class="headin5_amrc col_white_amrc pt2"></h5>
<!--headin5_amrc-->
<!--footer_ul_amrc ends here-->
</div>
<div class="col-lg-3 col-md-6 col-sm-6 ">
<h5 class="headin5_amrc col_white_amrc pt2"></h5>
<!--headin5_amrc-->
```

```
<ul class="footer_ul_amrc">
<li class="media">
<div class="media-body">
<p></p>
<span></span>
</div>
</li>
<li class="media">
<div class="media-body">
<p></p>
<span></span>
</div>
</li>
<li class="media">
<div class="media-body">
<p><p>
<span></span>
</div>
</li>
</ul>
<!--footer_ul_amrc ends here-->
</div>
</div>
</div>
<div class="container">
<div class="footer-logo">
<a href="#"><img src="images/footer-logo.png" alt="" /></a>
</div>
<!--foote_bottom_ul_amrc ends here-->
<!--social_footer_ul ends here-->
</div>
</footer>
<!-- Bootstrap core JavaScript -->
<script src="../static/vendor/jquery/jquery.min.js"></script>
<script src="../static/vendor/bootstrap/js/bootstrap.bundle.min.js"></script>
</body>
</html>
```

preview.html

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
<meta name="description" content="">
<meta name="author" content="">
<title>Preview </title>
<!-- Bootstrap core CSS -->
<link href="../static/vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">
<!-- Fontawesome CSS -->
<link href="../static/css/all.css" rel="stylesheet">
<!-- Custom styles for this template -->
<link href="../static/css/styles.css" rel="stylesheet">
</head>
<body>
<!-- Navigation -->
<nav class="navbar fixed-top navbar-expand-lg navbar-dark bg-light top-nav fixed-top">
<div class="container">
<a class="navbar-brand" href="index.html">
```

```
<img src="../static/images/footer-logo.png" alt="logo" />
</a>
<button class="navbar-toggler navbar-toggler-right" type="button" data-toggle="collapse" data-
target="#navbarResponsive" aria-controls="navbarResponsive" aria-expanded="false" aria-label="Toggle
navigation">
<span class="fas fa-bars"></span>
</button>
<div class="collapse navbar-collapse" id="navbarResponsive">
<ul class="navbar-nav ml-auto">
<li class="nav-item">
<a class="nav-link" href="index.html">Home</a>
</li>
<li class="nav-item">
<a class="nav-link" href="login.html">Login</a>
</li>
<li class="nav-item">
<a class="nav-link" href="upload.html">Upload</a>
</li>
<li class="nav-item">
<a class="nav-link active" href="preview.html">Preview</a>
</li>
<li class="nav-item">
<a class="nav-link" href="yield_prediction.html">Yield Prediction</a>
</li>
</ul>
</div>
</div>
</nav>
<header class="slider-main">
<div id="carouselExampleIndicators" class="carousel slide carousel-fade" data-ride="carousel">
<div class="carousel-inner" role="listbox">
<!-- Slide One - Set the background image for this slide in the line below -->
<div class="carousel-item active" style="background-image: url('../static/images/slider-06.jpg')">
<div class="carousel-caption d-none d-md-block">
<h2>Crop Recommender System Using Machine Learning Approach</h2>
</div>
</div>
<!-- Slide Two - Set the background image for this slide in the line below -->
<!-- Slide Three - Set the background image for this slide in the line below -->
</div>
</div>
</header>
<!-- Page Content -->
<div class="container">
<div class="services-bar">
<center><h1 class="my-4">Preview</h1></center>
</br>
</br>
</br>
</br>
<!-- Services Section -->
<div class="row">
<body id="page-top">

<!-- Navigation -->


<!-- Contact Section -->
<section class="page-section" id="contact">
<div class="container">
```

```html
<br>
<br>
<!-- Contact Section Heading -->
<h2 class="text-center text-uppercase text-secondary mb-0"> </h2>

<!-- Icon Divider -->
<div class="divider-custom">
<div class="divider-custom-line"></div>
<div class="divider-custom-icon">
</div>
<div class="divider-custom-line"></div>
</div>

<!-- Contact Section Form -->
<div class="row" style="margin-left:10px">
<div class="col-lg-8 mx-auto">
<!-- To configure the contact form email address, go to mail/contact_me.php and update the email address in
the PHP file on line 19. -->
{{ df_view.to_html(classes="table striped",na_rep="-") | safe}}
</div>
</div>
</section>
<div class="form-group" style="padding:0px 250px 10px 40px;height:200px">
<input style="margin-left:300px" type="button" onclick="hideLoader()" class="btn btn-primary" value="Click
to Train | Test" />
<div id="loading" style="display:None;margin-top:552950px"></div>
</div>


<!-- Copyright Section -->
<section class="copyright py-4 text-center text-white">
<div class="container">
</div>
</section>

<!-- Scroll to Top Button (Only visible on small and extra-small screen sizes) -->
<div class="scroll-to-top d-lg-none position-fixed ">
<a class="js-scroll-trigger d-block text-center text-white rounded" href="#page-top">
<i class="fa fa-chevron-up"></i>
</a>
</div>


<!-- Bootstrap core JavaScript -->
<script type='text/javascript'
src='https://ajax.googleapis.com/ajax/libs/jquery/2.2.4/jquery.min.js'></script>
<script type='text/javascript'>
function hideLoader() {
$('#loading').show(0).delay(1000).hide(0,function(){
alert("Training finished!");
window.location = "yield_prediction.html";
});
}



</script>

</body>
</div>
<!-- /.row -->
```

```
</div>
<!-- Portfolio Section -->
<hr>
<!-- Get In Touch Now Section -->
</div>
<!-- /.container -->
<!--footer starts from here-->
<footer class="footer">
<div class="container bottom_border">
<div class="row">
<div class="col-lg-3 col-md-6 col-sm-6 col">
<h5 class="headin5_amrc col_white_amrc pt2"></h5>
<!--headin5_amrc ends here-->
<!--footer_ul2_amrc ends here-->
</div>
<div class="col-lg-3 col-md-6 col-sm-6">
<h5 class="headin5_amrc col_white_amrc pt2"></h5>
<!--headin5_amrc-->
<!--footer_ul_amrc ends here-->
</div>
<div class="col-lg-3 col-md-6 col-sm-6 ">
<h5 class="headin5_amrc col_white_amrc pt2"></h5>
<!--headin5_amrc-->
<ul class="footer_ul_amrc">
<li class="media">
<div class="media-body">
<p></p>
<span></span>
</div>
</li>
<li class="media">
<div class="media-body">
<p></p>
<span></span>
</div>
</li>
<li class="media">
<div class="media-body">
<p><p>
<span></span>
</div>
</li>
</ul>
<!--footer_ul_amrc ends here-->
</div>
</div>
</div>
<div class="container">
<div class="footer-logo">
<a href="#"><img src="images/footer-logo.png" alt="" /></a>
</div>
<!--foote_bottom_ul_amrc ends here-->
<!--social_footer_ul ends here-->
</div>
</footer>
<!-- Bootstrap core JavaScript -->
<script src="../static/vendor/jquery/jquery.min.js"></script>
<script src="../static/vendor/bootstrap/js/bootstrap.bundle.min.js"></script>
</body>
</html>
```

`

result.html

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
<meta name="description" content="">
<meta name="author" content="">
<title>weather Report </title>
<!-- Bootstrap core CSS -->
<link href="../static/vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">
<!-- Fontawesome CSS -->
<link href="../static/css/all.css" rel="stylesheet">
<!-- Custom styles for this template -->
<link href="../static/css/styles.css" rel="stylesheet">
</head>
<body>
<!-- Navigation -->
<nav class="navbar fixed-top navbar-expand-lg navbar-dark bg-light top-nav fixed-top">
<div class="container">
<a class="navbar-brand" href="index.html">
<img src="../static/images/footer-logo.png" alt="logo" />
</a>
<button class="navbar-toggler navbar-toggler-right" type="button" data-toggle="collapse" data-
target="#navbarResponsive" aria-controls="navbarResponsive" aria-expanded="false" aria-label="Toggle
navigation">
<span class="fas fa-bars"></span>
</button>
<div class="collapse navbar-collapse" id="navbarResponsive">
<ul class="navbar-nav ml-auto">
<li class="nav-item">
<a class="nav-link " href="index.html">Home</a>
</li>
<li class="nav-item">
<a class="nav-link" href="login.html">Login</a>
</li>
<li class="nav-item">
<a class="nav-link " href="yield_prediction.html">Yield Prediction</a>
</li>
<li class="nav-item">
<a class="nav-link" href="crop_prediction.html">Crop Recommendation</a>
</li>
<li class="nav-item">
<a class="nav-link" href="homepage.html">Weather Forecast </a>
</li>
<li class="nav-item">
<a class="nav-link active" href="homepage.html">Weather Report </a>
</li>
<li class="nav-item">
<a class="nav-link " href="chart.html"> chart</a>
</li>
</ul>
</div>
</div>
</nav>
<header class="slider-main">
<div id="carouselExampleIndicators" class="carousel slide carousel-fade" data-ride="carousel">
<div class="carousel-inner" role="listbox">
```

```html
<!-- Slide One - Set the background image for this slide in the line below -->
<div class="carousel-item active" style="background-image: url('../static/images/slider-12.jpg')">
<div class="carousel-caption d-none d-md-block">
<h2>Crop Recommender System Using Machine Learning Approach</h2>
</div>
</div>
<!-- Slide Two - Set the background image for this slide in the line below -->
<!-- Slide Three - Set the background image for this slide in the line below -->
</div>
</div>
</header>
<!-- Page Content -->
<div class="container">
<div class="services-bar">
<center><h1 class="my-4">Weather Report </h1></center>
<!-- Services Section -->
<div class="row">
<div class="col-md-5 col-sm-5 col-xs-5" style="margin-left:400px;">
<section class="result">
<h2><small class="text-muted">Forecast for</small> {{data[0]}}</h2>
<div class="container-fluid">
<div class="card border-secondary mb-3" style="max-width: 20rem;">
<div class="card-header"><strong>Today</strong></div>
<div class="card-body">
<article>
<p>{{data[1][0]}}</p>
<em>temperature: {{data[1][1]}}°C</em>
</article>
</div>
</div>
<div class="card border-secondary mb-3" style="max-width: 20rem;">
<div class="card-header"><strong>Tomorrow</strong></div>
<div class="card-body">
<article>
<p>{{data[2][0]}}</p>
<em>temperature: {{data[2][1]}}°C</em>
</article>
</div>
</div>
<div class="card border-secondary mb-3" style="max-width: 20rem;">
<div class="card-header"><strong>Day after tomorrow</strong></div>
<div class="card-body">
<article>
<p>{{data[3][0]}}</p>
<em>temperature: {{data[3][1]}}°C</em>
</article>
</div>
</div>
<div class="card border-secondary mb-3" style="max-width: 20rem;">
<div class="card-header"><strong>Next Day</strong></div>
<div class="card-body">
<article>
<p>{{data[4][0]}}</p>
<em>temperature: {{data[4][1]}}°C</em>
</article>
</div>
</div>
</div>
</section>
</div>
```

```
<h1>If no heavy rains predicted in the upcoming days, you can add fertilizers</h1>
</br>
<h1>If heavy rains predicted in the upcoming days, you should not add fertilizers</h1>
</br>
</br>
</br>
</div>
<!-- /.row -->
</div>
<!-- Portfolio Section -->
<hr>
<!-- Get In Touch Now Section -->
</div>
<!-- /.container -->
<!--footer starts from here-->
<footer class="footer">
<div class="container bottom_border">
<div class="row">
<div class="col-lg-3 col-md-6 col-sm-6 col">
<h5 class="headin5_amrc col_white_amrc pt2"></h5>
<!--headin5_amrc ends here-->
<!--footer_ul2_amrc ends here-->
</div>
<div class="col-lg-3 col-md-6 col-sm-6">
<h5 class="headin5_amrc col_white_amrc pt2"></h5>
<!--headin5_amrc-->
<!--footer_ul_amrc ends here-->
</div>
<div class="col-lg-3 col-md-6 col-sm-6 ">
<h5 class="headin5_amrc col_white_amrc pt2"></h5>
<!--headin5_amrc-->
<ul class="footer_ul_amrc">
<li class="media">
<div class="media-body">
<p></p>
<span></span>
</div>
</li>
<li class="media">
<div class="media-body">
<p></p>
<span></span>
</div>
</li>
<li class="media">
<div class="media-body">
<p><p>
<span></span>
</div>
</li>
</ul>
<!--footer_ul_amrc ends here-->
</div>
</div>
</div>
<div class="container">
<div class="footer-logo">
<a href="#"><img src="images/footer-logo.png" alt="" /></a>
</div>
<!--foote_bottom_ul_amrc ends here-->
```

```
<!--social_footer_ul ends here-->
</div>
</footer>
<!-- Bootstrap core JavaScript -->
<script src="../static/vendor/jquery/jquery.min.js"></script>
<script src="../static/vendor/bootstrap/js/bootstrap.bundle.min.js"></script>
</body>
</html>
```

upload.html

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
<meta name="description" content="">
<meta name="author" content="">
<title>Upload </title>
<!-- Bootstrap core CSS -->
<link href="../static/vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">
<!-- Fontawesome CSS -->
<link href="../static/css/all.css" rel="stylesheet">
<!-- Custom styles for this template -->
<link href="../static/css/styles.css" rel="stylesheet">
</head>
<body>
<!-- Navigation -->
<nav class="navbar fixed-top navbar-expand-lg navbar-dark bg-light top-nav fixed-top">
<div class="container">
<a class="navbar-brand" href="index.html">
<img src="../static/images/footer-logo.png" alt="logo" />
</a>
<button class="navbar-toggler navbar-toggler-right" type="button" data-toggle="collapse" data-
target="#navbarResponsive" aria-controls="navbarResponsive" aria-expanded="false" aria-label="Toggle
navigation">
<span class="fas fa-bars"></span>
</button>
<div class="collapse navbar-collapse" id="navbarResponsive">
<ul class="navbar-nav ml-auto">
<li class="nav-item">
<a class="nav-link" href="index.html">Home</a>
</li>
<li class="nav-item">
<a class="nav-link" href="login.html">Login</a>
</li>
<li class="nav-item">
<a class="nav-link active" href="upload.html">Upload</a>
</li>
<li class="nav-item">
<a class="nav-link" href="preview.html">Preview</a>
</li>
</ul>
</div>
</div>
</nav>
<header class="slider-main">
<div id="carouselExampleIndicators" class="carousel slide carousel-fade" data-ride="carousel">
<div class="carousel-inner" role="listbox">
<!-- Slide One - Set the background image for this slide in the line below -->
```

```
<div class="carousel-item active" style="background-image: url('../static/images/slider-07.jpg')">
<div class="carousel-caption d-none d-md-block">
<h2>Crop Recommender System Using Machine Learning Approach</h2>
</div>
</div>
<!-- Slide Two - Set the background image for this slide in the line below -->
<!-- Slide Three - Set the background image for this slide in the line below -->
</div>
</div>
</header>
<!-- Page Content -->
<div class="container">
<div class="services-bar">
<center><h1 class="my-4">Upload </h1></center>
<!-- Services Section -->
<div class="row">
<body id="page-top">

<!-- About Section -->
<section class="col-md-12 col-sm-12 col-xs-12" id="about">
<!-- About Section Heading -->

<!-- Icon Divider -->
<!-- About Section Content -->
<div class="col-md-12 col-sm-12 col-xs-12" style="margin-left:480px;">
<form action="http://localhost:5000/preview" name="fs" id="fs" method="post" enctype=multipart/form-data>
<br/>
<input type="file" name="datasetfile" id="file1" required />
<br/>
<br/>
<br/>
<input type="submit" style="margin-right:250px" class="btn btn-success btn-large" value="Upload">
</form>
</div>
</section>

</body>
</div>
<!-- /.row -->
</div>
<!-- Portfolio Section -->
<hr>
<!-- Get In Touch Now Section -->
</div>
<!-- /.container -->
<!--footer starts from here-->
<footer class="footer">
<div class="container bottom_border">
<div class="row">
<div class="col-lg-3 col-md-6 col-sm-6 col">
<h5 class="headin5_amrc col_white_amrc pt2"></h5>
<!--headin5_amrc ends here-->
<!--footer_ul2_amrc ends here-->
</div>
<div class="col-lg-3 col-md-6 col-sm-6">
<h5 class="headin5_amrc col_white_amrc pt2"></h5>
<!--headin5_amrc-->
<!--footer_ul_amrc ends here-->
</div>
<div class="col-lg-3 col-md-6 col-sm-6 ">
```

```
<h5 class="headin5_amrc col_white_amrc pt2"></h5>
<!--headin5_amrc-->
<ul class="footer_ul_amrc">
<li class="media">
<div class="media-body">
<p></p>
<span></span>
</div>
</li>
<li class="media">
<div class="media-body">
<p></p>
<span></span>
</div>
</li>
<li class="media">
<div class="media-body">
<p><p>
<span></span>
</div>
</li>
</ul>
<!--footer_ul_amrc ends here-->
</div>
</div>
</div>
<div class="container">
<div class="footer-logo">
<a href="#"><img src="images/footer-logo.png" alt="" /></a>
</div>
<!--foote_bottom_ul_amrc ends here-->
<!--social_footer_ul ends here-->
</div>
</footer>
<!-- Bootstrap core JavaScript -->
<script src="../static/vendor/jquery/jquery.min.js"></script>
<script src="../static/vendor/bootstrap/js/bootstrap.bundle.min.js"></script>
</body>
</html>
```

yield_prediction.html

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
<meta name="description" content="">
<meta name="author" content="">
<title>Yield Prediction</title>
<!-- Bootstrap core CSS -->
<link href="../static/vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">
<!-- Fontawesome CSS -->
<link href="../static/css/all.css" rel="stylesheet">
<!-- Custom styles for this template -->
<link href="../static/css/styles.css" rel="stylesheet">
</head>
<body>
<!-- Navigation -->
<nav class="navbar fixed-top navbar-expand-lg navbar-dark bg-light top-nav fixed-top">
<div class="container">
```

```
<a class="navbar-brand" href="index.html">
<img src="../static/images/footer-logo.png" alt="logo" />
</a>
<button class="navbar-toggler navbar-toggler-right" type="button" data-toggle="collapse" data-
target="#navbarResponsive" aria-controls="navbarResponsive" aria-expanded="false" aria-label="Toggle
navigation">
<span class="fas fa-bars"></span>
</button>
<div class="collapse navbar-collapse" id="navbarResponsive">
<ul class="navbar-nav ml-auto">
<li class="nav-item">
<a class="nav-link " href="index.html">Home</a>
</li>
<li class="nav-item">
<a class="nav-link" href="login.html">Login</a>
</li>
<li class="nav-item">
<a class="nav-link active" href="yield_prediction.html">Yield Prediction</a>
</li>
<li class="nav-item">
<a class="nav-link" href="crop_prediction.html">Crop Recommendation</a>
</li>
<li class="nav-item">
<a class="nav-link" href="homepage.html">Weather Forecast </a>
</li>
<li class="nav-item">
<a class="nav-link" href="homepage.html">Weather Report </a>
</li>
</ul>
</div>
</div>
</nav>
<header class="slider-main">
<div id="carouselExampleIndicators" class="carousel slide carousel-fade" data-ride="carousel">
<div class="carousel-inner" role="listbox">
<!-- Slide One - Set the background image for this slide in the line below -->
<div class="carousel-item active" style="background-image: url('../static/images/slider-10.jpg')">
<div class="carousel-caption d-none d-md-block">
<h2>Crop Recommender System Using Machine Learning Approach</h2>
</div>
</div>
<!-- Slide Two - Set the background image for this slide in the line below -->
<!-- Slide Three - Set the background image for this slide in the line below -->
</div>
</div>
</header>
<!-- Page Content -->
<div class="container">
<div class="services-bar">
<center><h1 class="my-4">Crop Yield Prediction </h1></center>
<!-- Services Section -->
<div class="row">
<h2></h2>
<form class="form-horizontal" action="{{ url_for('predict')}}" method="POST">
</div>
<div class="span5">
<div class="control-group">
<center><label class="control-label" for="inputMessage"><b>Enter your Details</b></label></br></center>
</br>
<center> <div class="controls">
```

```
State_Name: <select type="text" name="State_Name" required="required" placeholder="State_Name">
<option value="0">Andaman and Nicobar Islands</option>
<option value="1">Andhra Pradesh</option>
<option value="3">Assam</option>
<option value="4">Goa</option>
<option value="5">Karnataka</option>
<option value="6">Kerala</option>
<option value="7">Meghalaya</option>
<option value="8">Puducherry</option>
<option value="9">Tamil Nadu</option>
<option value="10">West Bengal</option>
<option value="11">Bihar</option>
<option value="12">Chhattisgarh</option>
<option value="13">Dadra and Nagar Haveli</option>
<option value="14">Gujarat</option>
<option value="15">Haryana</option>
<option value="16">Madhya Pradesh</option>
<option value="17">Maharashtra</option>
<option value="18">Manipur</option>
<option value="19">Rajasthan</option>
<option value="20">Telangana</option>
<option value="21">Uttar Pradesh</option>
<option value="22">Arunachal Pradesh</option>
<option value="23">Himachal Pradesh</option>
<option value="24">Jammu and Kashmir</option>
<option value="25">Nagaland</option>
<option value="26">Odisha</option>
<option value="27">Uttarakhand</option>
<option value="28">Mizoram</option>
<option value="29">Punjab</option>
<option value="30">Tripura</option>
<option value="31">Chandigarh</option>
<option value="32">Jharkhand</option>
<option value="33">Sikkim</option>
</select>
</div></center>
</br>
<center> <div class="controls">
Crop: <select type="text" name="Crop" required="required" placeholder="Crop">
<option value="0">Arecanut</option>
<option value="1">Banana</option>
<option value="2">Dry chillies</option>
<option value="3">Coconut</option>
<option value="4">Cotton</option>
<option value="5">Dry ginger</option>
<option value="6">Groundnut</option>
<option value="7">Maize</option>
<option value="8">Moong(Green Gram)</option>
<option value="9">Onion</option>
<option value="10">Paddy</option>
<option value="11">Ragi</option>
<option value="12">Sugarcane</option>
<option value="13">Tobacco</option>
<option value="14">Turmeric</option>
<option value="15">Rice</option>
</select>
</div></center>
</br>
<center> Area : <input type="text" name="Area" placeholder="Area" required="required" /></center>
</br>
```

`

```
<div class="controls">
                                    
                                    
                                    
soil_type: <select type="text" name="soil_type" required="required" placeholder="soil_type">
<option value="0">Sandy</option>
<option value="1">Red</option>
<option value="2">Black</option>
<option value="3">Clayey</option>
<option value="4">Alluvial</option>
<option value="5">Loamy</option>
<option value="6">Latterite</option>
</select>
</div>
</br>
</br>
<center><input type="submit" class="btn btn-success" value="submit" style="margin-left: 10px" ></center>
</div>
</div>
</form>
<p>Production would be <a href="#"> {{ prediction_text }} KGs</a></p>
<p>Yield would be <a href="#"> {{ yield_predictions }} KGs/acre</a></p>
<!-- /.row -->
</div>
<!-- Portfolio Section -->
<!-- Get In Touch Now Section -->
</div>
</div>
</br>
</br>
</br>
</br>
<!-- /.container -->
<!--footer starts from here-->
<footer class="footer">
<div class="container bottom_border">
<div class="row">
<div class="col-lg-3 col-md-6 col-sm-6 col">
<h5 class="headin5_amrc col_white_amrc pt2"></h5>
<!--headin5_amrc ends here-->
<!--footer_ul2_amrc ends here-->
</div>
<div class="col-lg-3 col-md-6 col-sm-6">
<h5 class="headin5_amrc col_white_amrc pt2"></h5>
<!--headin5_amrc-->
<!--footer_ul_amrc ends here-->
</div>
<div class="col-lg-3 col-md-6 col-sm-6 ">
<h5 class="headin5_amrc col_white_amrc pt2"></h5>
<!--headin5_amrc-->
<ul class="footer_ul_amrc">
<li class="media">
<div class="media-body">
<p></p>
```

```
<span></span>
</div>
</li>
<li class="media">
<div class="media-body">
<p></p>
<span></span>
</div>
</li>
<li class="media">
<div class="media-body">
<p><p>
<span></span>
</div>
</li>
</ul>
<!--footer_ul_amrc ends here-->
</div>
</div>
</div>
<div class="container">
<div class="footer-logo">
<a href="#"><img src="images/footer-logo.png" alt="" /></a>
</div>
<!--foote_bottom_ul_amrc ends here-->
<!--social_footer_ul ends here-->
</div>
</footer>
<!-- Bootstrap core JavaScript -->
<script src="../static/vendor/jquery/jquery.min.js"></script>
<script src="../static/vendor/bootstrap/js/bootstrap.bundle.min.js"></script>
</body>
</html>
```

Appendix - B

Bill Of Materials

To build an Crop Recommendation System machine learning (ML) model, we don't require a traditional bill of materials (BOM) as we would for physical products. It primarily involves the necessary hardware components and software tools required for the implementation and deployment of the ML model. Here are some key components that be included in the BOM:

1. **Hardware Components:**

- Central Processing Unit (CPU) or Graphics Processing Unit (GPU): The CPU or GPU serves as the computational powerhouse of the system. ML algorithms, particularly deep learning models, often require significant computational resources for training and inference. Choosing a powerful CPU or GPU ensures efficient execution of the ML algorithms.
- Memory: Sufficient RAM capacity (Min. 4 GB) is essential for storing and manipulating large datasets during training and testing. ML models with complex architectures and large parameter sizes can be memory-intensive, so having ample memory helps avoid bottlenecks and improves overall performance.
- Storage: A hard disk or solid-state drive (SSD) is required for storing datasets, trained ML models, and software tools. ML datasets can be substantial, and trained models can take up significant space, especially in the case of deep learning models. Sufficient storage capacity allows for efficient data management and model storage.
- Peripherals Standard peripherals like a keyboard, mouse, and monitor are necessary for setting up and operating the ML system. Additionally, other peripherals such as external storage devices or input devices may be required based on specific needs.

2. Software Tools:

- ML Frameworks: ML frameworks provide pre-built libraries and tools for developing and implementing ML models. TensorFlow, scikit-learn, and Keras are popular ML frameworks that offer a wide range of functionality, including neural networks, SVM, random forests, and other ML algorithms.
- Programming Languages: Python is widely used in the ML community due to its rich ecosystem of libraries and frameworks.
- Development Environment: An integrated development environment (IDE) or code editors like Jupyter Notebook or Visual Studio Code are commonly used for ML development. These tools provide a convenient interface for writing, debugging, and executing ML code. We are used Jupyter Notebook.
- Data Preprocessing and Analysis Tools: Libraries such as NumPy, Pandas, and SciPy are commonly used for data pre-processing, feature extraction, and exploratory data analysis tasks. These tools provide efficient and convenient methods for manipulating, transforming, and analysing data.
- ML Algorithms and Models: ML algorithms and models are the core components of an ML system. Neural networks used is CNN, support vector machines (SVM), logistic regression, and other algorithms are implemented using the chosen ML framework. These algorithms learn patterns from the input data and make predictions or classifications.
- Evaluation Metrics: To assess the performance of the ML models, evaluation metrics are used. Libraries like scikit-learn provide a wide range of metrics such as accuracy, precision, recall, F1 score. These metrics measure the effectiveness of the ML models in detecting and classifying arrhythmias.