CROP YIELD PREDICTION

A Project Report
submitted in partial fulfillment of the
requirements for the award of the degree of

Bachelor Of Technology
In
Computer Science & Engineering
by

M. Sowmya Sree(18H71A05G4) **K. Hima Soundarya(18H71A05D6)**
S.Akhila(18H71A05C1) **A. Sai Samanth(18H71A05F9)**

Under the Esteemed Guidance of
Mr. CH. Vinay Santhosh
Assistant Professor

Department of Computer Science & Engineering

DVR & Dr. HS
MIC College of Technology
(Autonomous)
Kanchikacherla– 521180, NTR Dist, Andhra Pradesh

June, 2022

DECLARATION

We here by declare that

- The work contained in this report is original and has been done by us under the guidance of our supervisor(s).
- The work has not been submitted to any other Institute for obtaining any degree or diploma.
- We have followed the guidelines provided by the Institute in preparing the report.
- We have conformed to the norms and guidelines given in the Ethical Code of Conduct of the Institute.
- Whenever We have used materials (data, theoretical analysis, figures, and text) from other sources, We have given due credit to them by citing in the text of the report and giving their details in the references.

Place:

Date:

M. Sowmya Sree(18H71A05G4)
K.Hima Soundarya(18H71A05D6)
S.Akhila(18H71A05C1)
A. Sai Samanth(18H71A05F9)

DVR & Dr. HS
MIC College of Technology
ISO 9001:2015 Certified Institute
(Approved by AICTE & Permanently Affiliated to JNTUK, Kakinada)
Kanchikacherla - 521180, Krishna Dist, A.P, India.

AUTONOMOUS
NBA-Accredited
B.Tech (CSE|ECE|ME)
Accredited by
NAAC
With Grade A

CERTIFICATE

This is to certify that the project report entitled **"Crop Yield Prediction"** submitted by **M. Sowmya Sree (18H71A05G4), K. Hima Soundarya (18H71A05D6), S. Akhila (18H71A05C1), A. Sai Samanth (18H71A05F9)** to the DVR & Dr. HS MIC College of Technology in partial fulfillment of the requirements for the award of the Degree of Bachelor of Technology in **Computer Science& Engineering** is a bonafide record of work. The contents of this report, in full or in parts, have not been submitted to any other Institute for the award of any Degree.

Supervisor **Head of the Department**

Principal

Internal Examiner **External Examiner**

ACKNOWLEDGEMENT

The satisfaction that accompanies the successful completion of any task would be incomplete without the mention of the people who made it possible and whose constant guidance and engagement crown all the efforts with success. We thank our college management and respected **Sri D. PANDURANGA RAO,** CEO for providing us the necessary infrastructure to carry out the project work.

We express our sincere thanks to our beloved Principal **Dr. K. Srinivas** who has been a great source of information for our work.

Mr. D. Prasad Associate Professor and Head, Department of CSE for extending his support to carry on this project work.

We are thankful to our Project Coordinator **Dr. G. Sai Chaitanya Kumar**, Associate Professor, Department of CSE for their internal support and professionalism who helped us in shaping the project into successful one.

We would like to thank my project supervisors, **Mr. CH. Vinay Santhosh**, Assistant Professor and Department of CSE for the guidance and support, especially for the valuable ideas and knowledge shared to me throughout the Project.

We take this opportunity to express our thanks to one and all who directly or indirectly helped us in bringing this effort to present form.

Finally, my special thanks go to my family for their continuous support and help throughout my academic years and for their continual support and encouragement for the completion of the project.

CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

WEKA	Waikato Environment for Knowledge Analysis
CNN	Convolutional Neural Networks
ML	Machine Learning
DNN	Deep Neural Networks
LSTM	Long Short Term Memory
DL	Deep Learning
RMSE	Root Mean Square Error
MSE	Mean Square Error
MAE	Mean Absolute Error
RF	Random Forest
SVM	Support Vector Machine
LR	Logistic Regression
NN	Neural Networks
SRS	Software Requirements Specification
NFR	Non-Functional Requirement
RAM	Random Access Memory
URL	Uniform Resource Locator
SSL	Secure Socket Layer
HTTP	Hyper Text Transfer Protocol
WSGI	Web Server Gateway Interface
REST	Representational State Transfer

ABSTRACT

Agriculture is the pillar of the Indian economy and more than 50% of India's population are dependent on agriculture for their survival. Variations in weather, climate, and other such environmental conditions have become a major risk for the healthy existence of agriculture. Machine learning plays a significant role as it has decision support tool for Crop Yield Prediction including supporting decisions on what crops to grow and what to do during the growing season of the crops.

Nowadays farmers facing so many challenges so to get rid of these problems we intended to build a good model for predicting crop yield using machine learning algorithms, those are linear regression, Random Forest.

The crop yield prediction is a method to achieve a high yield of the crop using previous available data like crop name, season, area, production, soil parameters continuously involves all features that used for high yield of the crop.

The prediction is based on analyzing a static set of data using Supervised Machine Learning techniques. This static dataset contains previous year’s data taken from the Yearbook of Agricultural Statistics.

CHAPTER 1

INTRODUCTION

1.1. Introduction

Agriculture is the backbone of every economy. In a country like India, which has ever increasing demand of food due to rising population, advances in agriculture sector are required to meet the needs. From ancient period, agriculture is considered as the main and the foremost culture practiced in India. Ancient people cultivate the crops in their own land and so they have been accommodated to their needs. Therefore, the natural crops are cultivated and have been used by many creatures such as human beings, animals and birds. The greenish goods produced in the land which have been taken by the creature leads to a healthy and welfare life. Since the invention of new innovative technologies and techniques the agriculture field is slowly degrading. Due to these, abundant invention people are been concentrated on cultivating artificial products that is hybrid products where there leads to an unhealthy life. Nowadays, modern people don’t have awareness about the cultivation of the crops in a right time and at a right place. Because of these cultivating techniques the seasonal climatic conditions are also being changed against the fundamental assets like soil, water and air which lead to insecurity of food. By analyzing all these issues and problems like weather, temperature and several factors, there is no proper solution and technologies to overcome the situation faced by us. In India there are several ways to increase the economical growth in the field of agriculture. There are multiple ways to increase and improve the crop yield and the quality of the crops. Data mining also useful for predicting the crop yield production.

More than 50% of India's population are dependent on agriculture for their survival. Most of the agricultural crops are being badly affected in terms of their performance over a period of the last two decades. Agricultural crop prediction depends on biology, climate, economy, geography. Crop yield is central to addressing food security challenges. So, to get rid of these problems we intended to build a model for predicting crop yield using machine learning algorithm, that is Random Forest. The prediction made by machine learning algorithms will help the farmers to come to a decision which crop to grow to induce the most yield. The prediction is based on analyzing a static set of historical data. The Application which we developed, runs the algorithm and shows the list of crops suitable for entered data with predicted yield value.

1.2 Motivation of the problem

- Healthy Existence of agriculture
- World’s food Demands
- Supporting farmers
- Recommend the crop based on the factors

1.3 Problem statement

The Problem Statement revolves around prediction of crop yield using Machine Learning Techniques. The goal of the project is to help the users choose a suitable crop to grow in order to maximize the yield and hence the profit. The system proposed tries to overcome the drawbacks of existing systems and make predictions by analyzing structured data. The solution we are proposing is to design a system taking into consideration the most influencing parameters to grow a crop and to get a better selection of crops which can be grown over the season. This would help reduce the difficulties faced by the farmers in selecting the crop to get high yield and thus maximize profits which in turn will reduce the suicide rates.

- Agriculture plays a critical role in the global economy.
- Crop yield prediction is an important agricultural problem.
- The Agricultural yield primarily depends on weather conditions(rain,temperature,etc).
- Accurate information about history of crop yield is important.
- To make decisions related to agricultural risk management and future predictions.

CHAPTER 2
LITERATURE SURVEY

2.1 Crops yield prediction and efficient use of fertilizers using machine learning.

This paper reports utilization of various information mining methods will anticipate rice trim yield for Maharashtra state, India. To this review, 27 regions of Maharashtra were picked on the establishment of accessible information from openly available Indian Administration records with different atmosphere and harvest parameters. Precipitation, least temperature, normal temperature, most extreme temperature,reference trim evapotranspiration, range, generation and yield for the Kharif season (June to November) were the parameters chosen for the study for the years 1998 to 2002. WEKA tool was used for dataset processing. The paper concludes that the rapid advances in sensing technologies and ML techniques will provide cost-effective and comprehensive solutions for better crop and environment state estimation and decision making.

2.2 Hybrid Deep Learning-based Models for Crop Yield Prediction

January 2022Applied Artificial Intelligence

Predicting crop yield is a complex task since it depends on multiple factors. Although many models have been developed so far in the literature, the performance of current models is not satisfactory, and hence, they must be improved. In this study, we developed deep learning-based models to evaluate how the underlying algorithms perform with respect to different performance criteria. The algorithms evaluated in our study are the XGBoost machine learning (ML) algorithm, Convolutional Neural Networks (CNN)-Deep Neural Networks (DNN), CNNXGBoost, CNN-Recurrent Neural Networks (RNN), and CNN-Long Short Term Memory (LSTM). For the case study, we performed experiments on a public soybean dataset that consists of 395 features including weather and soil parameters and 25,345 samples. The results showed that the hybrid CNN-DNN model outperforms other models, having an RMSE equal to 0.266, an MSE of 0.071, and an MAE of 0.199. The predictions of the model fit with an R2 of 0.87. The second-best result was achieved by the XGBoost model, which required less time to execute compared to the other DL-based models.

2.3 Mutual Information Feature Selection (MIFS) Based Crop Yield Prediction on Corn and Soybean Crops Using Multilayer Stacked Ensemble Regression (MSER)

Crop yield prediction model helps the farmers in order to make better decisions about the appropriate time to cultivate the crops and what types of crops to be cultivated based on environmental factors to produce better yield. Advanced ensemble regression crop yield prediction model is to predict the crop yield based on the phenotype factors includes precipitation, solar radiation, maximum temperature, minimum temperature etc. The corn and soybean crops dataset includes 38 years of yield performance data collected across 105 different locations. Comparative analysis made between correlation and mutual information on the basis of feature selection. Most related features towards crop yield and achieved better predictions on crop yield though mutual information. Our proposed mutual information based advanced ensemble regression technique involved in the prediction process of crop yield on corn and soybean crops and achieved good prediction accuracy based on phenotype factors. The predicted yield performance of advanced ensemble regression crop prediction model outperformed several supervised machine learning and advanced ensemble learning algorithms. Various regression accuracy parameter metrics such as mean absolute error, mean square error and root mean square error are also involved in performance measures. Our prediction results also ensure that weather and crop management parameters are most influential towards crop yield prediction rather than soil parameters.

2.4 Smart farming using machine learning

Our project mainly concentrates on crop forecasting. It helps the farmers to cultivate the best food crops and raise the right animals with accordance to environmental components. Also, the farmers can adapt to climate changes to some degree by shifting planting dates, choosing varieties with different growth duration, or changing crop rotations. For experimental analysis, the statistical numeric data related to agriculture is undertaken. The clustering — based techniques and supervised algorithms are utilized for managing the collected statistical data. Additionally, the suitable classification methods like Random Forest (RF), Support Vector Machine (SVM), Logistic Regression (LR), Neural Networks (NN) are employed for better classification outcome.

The advantages of the proposed system are:

- India’s farmers grow pulses, sugarcane, millets & also non – food items such as coffee, tea etc.

- These techniques will help in predicting the crop forecasting.

- A web enhancement is also provided which helps to predict the yield of the crop upon certain agricultural parameters.

2.5 Machine learning methods for crop yield prediction and Climate change impact assessment in agriculture

Crop yields are critically dependent on weather. A growing empirical literature models this relationship in order to project climate change impacts on the sector. We describe an approach to yield modeling that uses a semiparametric variant of a deep neural network, which can simultaneously account for complex nonlinear relationships in high-dimensional datasets, as well as known parametric structure and unobserved cross-sectional heterogeneity. Using data on corn yield from the US Midwest, we show that this approach outperforms both classical statistical methods and fully-nonparametric neural networks in predicting yields of years withheld during model training. Using scenarios from a suite of climate models, we show large negative impacts of climate change on corn yield, but less severe than impacts projected using classical statistical methods. In particular, our approach is less pessimistic in the warmest regions and the warmest scenarios.

CHAPTER 3
METHODOLOGY

3.1 Existing System

- Existing System uses Decision tree machine learning algorithm.
- It help the farmers to come to a decision which crop to grow.
- Decision Tree: Decision Tree is a Supervised learning technique that can be used for both classification and regression problems.
- Decision tree is a tree structure classifier where, Leaf node represents the outcome

Disadvantages

- Large datasets are not well suited.
- Accuracy is comparatively low.
- Trains only one decision tree.
- Overfitting can happen.

3.2 Proposed System

- We intend to build a machine learning model with Random Forest algorithm to provide high accuracy.
- Random Forest: It is a supervised learning algorithm, is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model.

Advantages

- Better Accuracy
- Can work with large dataset
- Multiple decision trees are trained parallelly for prediction.
- Possibility of overfitting is low.

3.2.1 Machine Learning

Machine Learning is undeniably one of the most influential and powerful technologies in today‘s world. Machine learning is a tool for turning information into knowledge. In the past 50 years, there has been an explosion of data . This mass of data is useless; we analyse it and find the patterns hidden within. Machine learning techniques are used to automatically find the valuable underlying patterns within complex data that we would otherwise struggle to discover. The hidden patterns and knowledge about a problem can be used to predict future events and perform all kinds of complex decision making. To learn the rules governing a phenomenon, machines have to go through a learning process, trying different rules and learning from how well they perform. Hence, why it‘s known as Machine Learning. “Traditionally, software engineering combined human created rules with data to create answers to a problem. Instead, machine learning uses data and answers to discover the rules behind a problem.” - Chollet, 2017

3.2.1.1 Basic Terminology

- Dataset: A set of data examples, which contain features important to solving the problem.
- Features: Important pieces of data that help us understand a problem. These are fed into a Machine Learning algorithm to help it learn.
- Model: The representation (internal model) of a phenomenon that a Machine Learning algorithm has learnt. It learns this from the data it is shown during training. The model is the output you get after training an algorithm. For example, a decision tree algorithm would be trained and produce a decision tree model.

3.2.1.2 Types of Machine Learning

There are multiple forms of Machine Learning; supervised, unsupervised, semi supervised and reinforcement learning. Each form of Machine Learning has differing approaches, but they all follow the same underlying process and theory.

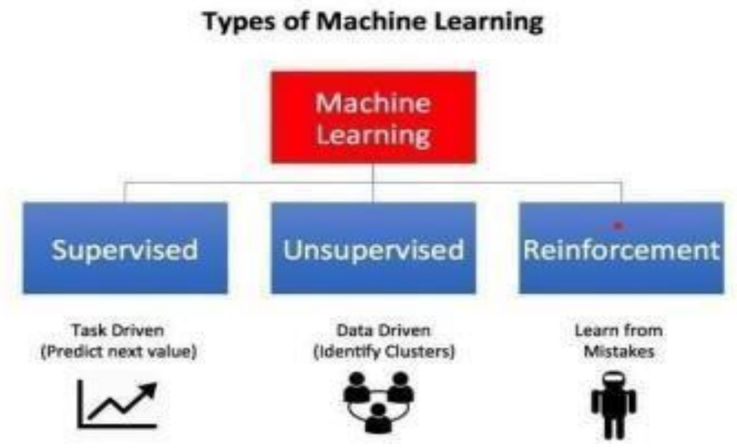

Types of Machine Learning
Machine Learning
Supervised
Unsupervised
Reinforcement
Task Driven (Predict next value)
Data Driven (Identify Clusters)
Learn from Mistakes

Fig 3.1 Types of Machine Learning

- **Supervised Learning:** It is the most popular paradigm for machine learning. Given data in the form of examples with labels, we can feed a learning algorithm these example-label pairs one by one, allowing the algorithm to predict the label for each example, and giving it feedback as to whether it predicted the right answer or not. Over time, the algorithm will learn to approximate the exact nature of the relationship between examples and their labels. When fully-trained, the supervised learning algorithm will be able to observe a new, never before-seen example and predict a good label for it.

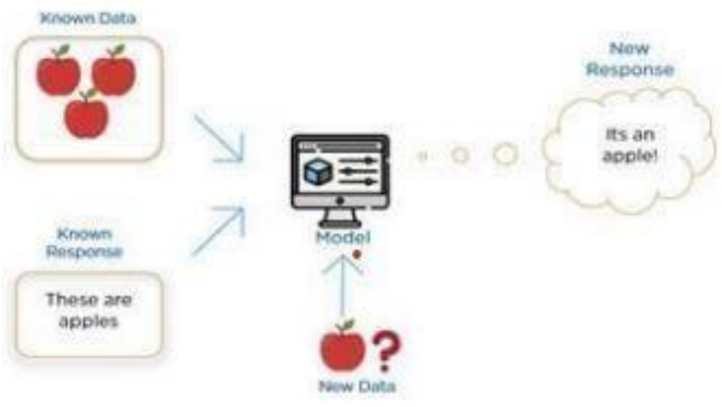


Fig 3.2 Supervised Learning

- **Unsupervised learning:** It is very much the opposite of supervised learning. It features no labels. Instead, the algorithm would be fed a lot of data and given the tools to understand the properties of the data. From there, it can learn to group, cluster, and organize the data in a way such that a human can come in and make sense of the newly organized data. Because unsupervised learning is based upon the data and its properties, we can say that unsupervised learning is data- driven. The outcomes from an unsupervised learning task are controlled by the data and the way it‘s formatted.

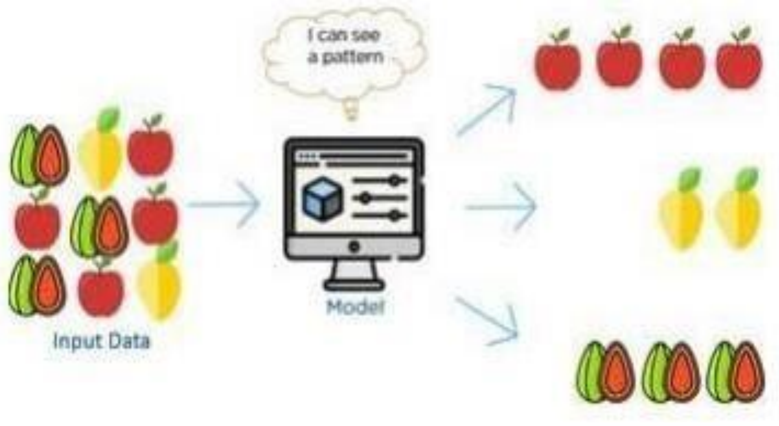


Fig 3.3 Unsupervised Learning

- **Reinforcement learning:** It is fairly different when compared to supervised and unsupervised learning. Reinforcement learning is very behavior driven. It has influences from the fields of neuroscience and psychology. For any reinforcement learning problem, we need an agent and an environment as well as a way to connect the two through a feedback loop. To connect the agent to the environment, we give it a set of actions that it can take that affect the environment. To connect the environment to the agent, we have it continually issue two signals to the agent: an updated state and a reward (our reinforcement signal for behavior).

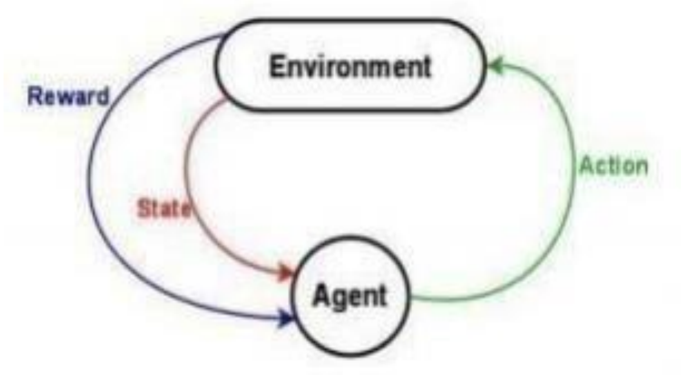
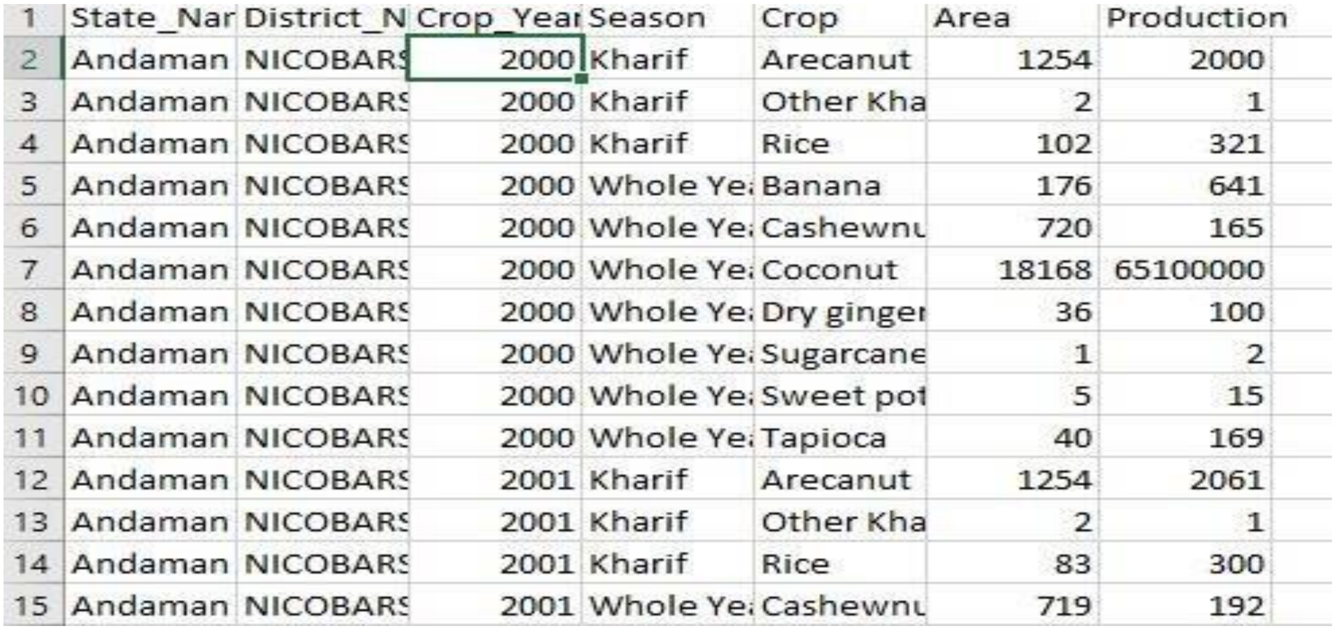


Fig 3.4 Reinforcement Learning

3.2.2 Data Exploration

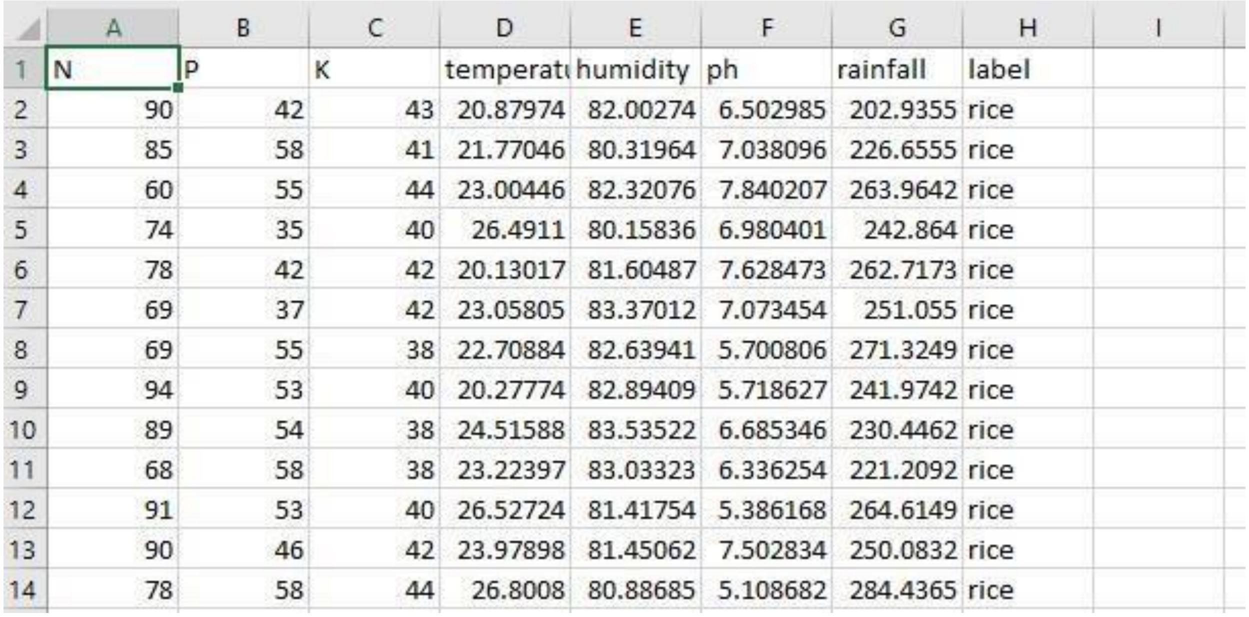
Machine Learning depends heavily on data. It‘s the most crucial aspect that makes algorithm training possible. It uses historical data and information to gain experiences. The better the collection of the dataset, the better will be the accuracy. The first step is Data Collection. For this project, we require two datasets. One predicting the yield algorithm and other for predicting crop recommendation .i.e. Average Rainfall and Average Temperature, Nitrogen, Phosphorous, Potassium. These parameters are used so as to be used as inputs for predicting them crop yield recommendation. For the experimental purpose, the statistical information is collected from Kaggle. We have taken a dataset consisting of historical data for crop yield prediction. The dataset have about 250000 instances.

The yield prediction module dataset requires the following columns: State, District, Crop, Season, Average Temperature, Average Rainfall, Area and Production as these are the major factors that crops depend on. _Production' is the dependent variable or the class variable. There are eight independent variables and 1 dependent variable. We are considering all the states of India.



1	State_Nar	District_N	Crop_Year	Season	Crop	Area	Production
2	Andaman	NICOBARS	2000	Kharif	Arecanut	1254	2000
3	Andaman	NICOBARS	2000	Kharif	Other Kha	2	1
4	Andaman	NICOBARS	2000	Kharif	Rice	102	321
5	Andaman	NICOBARS	2000	Whole Ye	Banana	176	641
6	Andaman	NICOBARS	2000	Whole Ye	Cashewnu	720	165
7	Andaman	NICOBARS	2000	Whole Ye	Coconut	18168	65100000
8	Andaman	NICOBARS	2000	Whole Ye	Dry ginger	36	100
9	Andaman	NICOBARS	2000	Whole Ye	Sugarcane	1	2
10	Andaman	NICOBARS	2000	Whole Ye	Sweet pot	5	15
11	Andaman	NICOBARS	2000	Whole Ye	Tapioca	40	169
12	Andaman	NICOBARS	2001	Kharif	Arecanut	1254	2061
13	Andaman	NICOBARS	2001	Kharif	Other Kha	2	1
14	Andaman	NICOBARS	2001	Kharif	Rice	83	300
15	Andaman	NICOBARS	2001	Whole Ye	Cashewnu	719	192

Fig 3.5 Crop yield Prediction Dataset



	A	B	C	D	E	F	G	H	I
1	N	P	K	temperatu	humidity	ph	rainfall	label	
2	90	42	43	20.87974	82.00274	6.502985	202.9355	rice	
3	85	58	41	21.77046	80.31964	7.038096	226.6555	rice	
4	60	55	44	23.00446	82.32076	7.840207	263.9642	rice	
5	74	35	40	26.4911	80.15836	6.980401	242.864	rice	
6	78	42	42	20.13017	81.60487	7.628473	262.7173	rice	
7	69	37	42	23.05805	83.37012	7.073454	251.055	rice	
8	69	55	38	22.70884	82.63941	5.700806	271.3249	rice	
9	94	53	40	20.27774	82.89409	5.718627	241.9742	rice	
10	89	54	38	24.51588	83.53522	6.685346	230.4462	rice	
11	68	58	38	23.22397	83.03323	6.336254	221.2092	rice	
12	91	53	40	26.52724	81.41754	5.386168	264.6149	rice	
13	90	46	42	23.97898	81.45062	7.502834	250.0832	rice	
14	78	58	44	26.8008	80.88685	5.108682	284.4365	rice	

Fig 3.6 Crop Recommendation Dataset

3.2.3 Data Pre-Processing

In data pre–processing, we import the required packages, read the dataset and drop the unnecessary columns and check whether null values are present in our data or not. The isnull() method is used to check for null values. The .sum() method is used for getting the shortest output result. We categorize our data into two parts 80% for training & 20% for testing.

3.2.4 Machine Learning Model

Random Forest

Random forest is a flexible, easy to use machine learning algorithm that produces, even without hyper-parameter tuning, a great result most of the time. It is also one of the most used algorithms, because of its simplicity and diversity. It can be used for both classification and regression tasks. Random forest builds multiple decision trees and merges them together to get a more accurate and stable prediction. One big advantage of random forest is that it can be used for both classification and regression problems, which form the majority of current machine learning systems. Another great quality of the random forest algorithm is that it is very easy to measure the relative importance of each feature on the prediction. Sklearn provides a great tool for this that measures a feature's importance by looking at how much the tree nodes that use that feature reduce impurity across all trees in the forest. It computes this score automatically for each feature after training and scales the results so the sum of all importance is equal to one. The hyper parameters in random forest are either used to increase the predictive power of the model or to make the model faster. Python offers some built in random Forest functions which have the following hyper parameters. i To increase the predictive power:

- Firstly, there is the n_estimators hyper parameter, which is just the number of trees the algorithm builds before taking the maximum voting or taking the averages of predictions.
- Another important hyper parameter is max_features, which is the maximum number of feature random forest considers to split a node.
- The last important hyper parameter is min_sample_leaf. This determines the minimum number of leafs required to split an internal node.

ii To Increase the model's speed:

- The n_jobs hyper parameter tells the engine how many processors it is allowed to use.

If it has a value of one, it can only use one processor. A value of ―-1‖ means that there is no limit.

- The random_state hyper parameter makes the model‘s output replicable. The model will always produce the same results when it has a definite value of random_state and if it has been given the same hyper parameters and the same training data.
- The oob_score (also called oob sampling), which is a random forest crossvalidation method. In this sampling, about one-third of the data is not used to train the model and can be used to evaluate its performance. These samples are called the out-of-bag samples.

The working of Random Forest is as follows:

- Step 1 − First, start with the selection of random samples from a given dataset.
- Step 2 − Next, this algorithm will construct a decision tree for every sample. Then it will get the prediction result from every decision tree.
- Step 3 − In this step, voting will be performed for every predicted result.
- Step 4 − At last, select the most voted prediction result as the final prediction result.

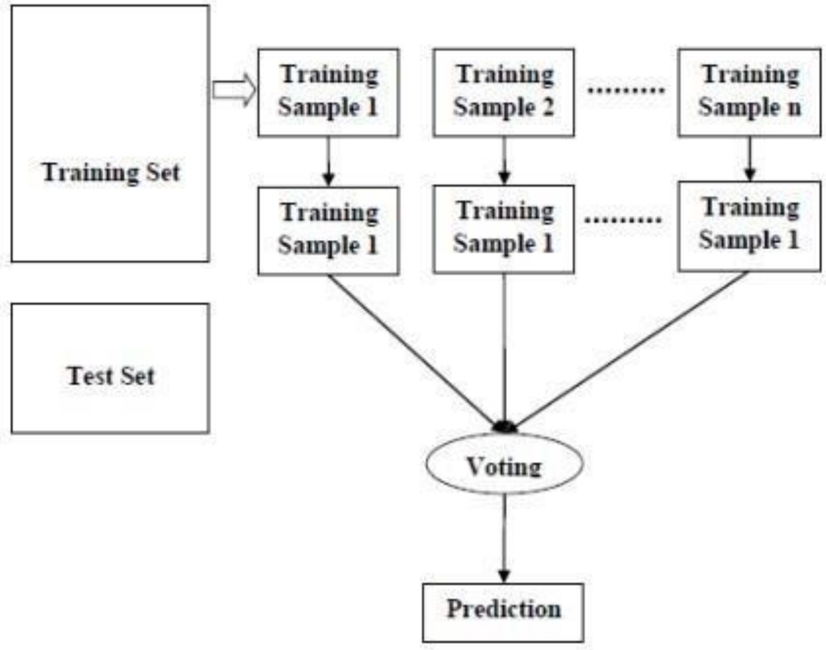


Fig 3.7 Random Forest Flow

CHAPTER 4
SYSTEM REQUIREMENTS SPECIFICATION

Software Requirement Specification(SRS)

A software requirements specification (SRS) is a description of a software system to be developed. It lays out functional and nonfunctional requirements, and may include a set of use cases that describe user interactions that the software must provide. It is very important in a SRS to list out the requirements and how to meet them. It helps the team to save upon their time as they are able to comprehend how are going to go about the project. Doing this also enables the team to find out about the limitations and risks early on.

A SRS can also be defined as a detailed description of a software system to be developed with its functional and non-functional requirements. It may include the use cases of how the user is going to interact with the software system. The software requirement specification document is consistent with all necessary requirements required for project development. To develop the software system we should have a clear understanding of Software system. To achieve this we need continuous communication with customers to gather all requirements.

A good SRS defines how the Software System will interact with all internal modules, hardware, and communication with other programs and human user interactions with a wide range of real life scenarios. It is very important that testers must be cleared with every detail specified in this document in order to avoid faults in test cases and its expected results.

Qualities of SRS

- Correct
- Unambiguous
- Complete
- Consistent
- Ranked for importance and/or stability
- Verifiable
- Modifiable
- Traceable

Some of the goals an SRS should achieve are to:

- Provide feedback to the customer, ensuring that the IT Company understands the issues the software system should solve and how to address those issues.
- Help to break a problem down into smaller components just by writing down the requirements.
- Speed up the testing and validation processes.
- Facilitate reviews.

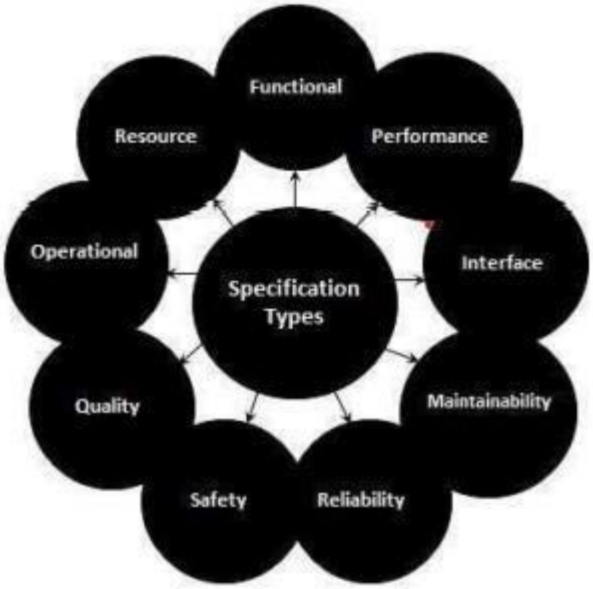


Fig 4.1 Types of Requirements in SRS

4.1 Functional Requirements

A Functional Requirement is a description of the service that the software must offer. It describes a software system or its component. A function is nothing but inputs to the software system, its behavior, and outputs. It can be a calculation, data manipulation, business process, user interaction, or any other specific functionality which defines what function a system is likely to perform. In software engineering and systems engineering, a Functional Requirement can range from the high-level abstract statement of the sender's necessity to detailed mathematical functional requirement specifications. Functional software requirements help you to capture the intended behavior of the system.

Benefits of functional requirements:

- Helps you to check whether the application is providing all the functionalities that were mentioned in the functional requirement of that application
- A functional requirement document helps you to define the functionality of a system or one of its subsystems.
- Functional requirements along with requirement analysis help identify missing requirements. They help clearly define the expected system service and behavior.
- Errors caught in the Functional requirement gathering stage are the cheapest to fix.
- Support user goals, tasks, or activities.

4.2 Non-Functional Requirements

Non-Functional Requirement (NFR) specifies the quality attribute of a software system. They judge the software system based on Responsiveness, Usability, Security, Portability and other non-functional standards that are critical to the success of the software system. Failing to meet non-functional requirements can result in systems that fail to satisfy user needs. Non-functional Requirements allows you to impose constraints or restrictions on the design of the system across the various agile backlogs. Example, the site should load in 3 seconds when the number of simultaneous users are > 10000. They specify the criteria that can be used to judge the operation of a system rather than specific behaviors. They may relate to emergent system properties such as reliability, response time and store occupancy. Non-functional requirements arise through the user needs, because of budget constraints, organizational policies, the need for interoperability with other software and hardware systems or because of external factors such as:- Product Requirements, Organizational Requirements, User Requirements, Basic Operational Requirement, etc.

Benefits of Non Functional Requirements:

- The nonfunctional requirements ensure the software system follows legal and compliance rules.
- They ensure the reliability, availability, and performance of the software system.
- They ensure good user experience and ease of operating the software.
- They help in formulating security policy of the software system.

4.3 Hardware Requirements:

- **Processor :** Dual core
- **RAM :** 128MB
- **Hard Disk :** 128GB

4.4 Software requirements

- **Operating System:** Windows XP and above
- **Tools:** Google Colab, Visual Studio code
- **Python:** Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

Fig 4.2 Python

Python libraries Used

A Python library is a collection of related modules. It contains bundles of code that can be used repeatedly in different programs. It makes Python Programming simpler and convenient for the programmer. As we don't need to write the same code again and again for different programs. Python libraries play a very vital role in fields of Machine Learning, Data Science, Data Visualization, etc.

- **Numpy**

The name "Numpy" stands for "Numerical Python". It is the commonly used library. It is a popular machine learning library that supports large matrices and multidimensional data. It consists of in-built mathematical functions for easy computations. Even libraries like TensorFlow use Numpy internally to perform several operations on tensors. Array Interface is one of the key features of this library.

Installation of Numpy:

```
pip install numpy
```

- **Pandas**

Pandas are an important library for data scientists. It is an open-source machine learning library that provides flexible high-level data structures and a variety of analysis tools. It eases data analysis, data manipulation, and cleaning of data. Pandas support operations like Sorting, Re-indexing, Iteration, Concatenation, Conversion of data, Visualizations, Aggregations, etc.

Installation of Pandas:

```
pip install pandas
```

- **Requests**

Requests is a Python module that you can use to **send all kinds of HTTP requests**. It is an easy-to-use library with a lot of features ranging from passing parameters in URLs to sending custom headers and SSL Verification.

Installation of Requests:

```
pip install requests
```

- **Sklearn**

It is a famous Python library to work with complex data. Scikit-learn is an open-source library that supports machine learning. It supports variously supervised and unsupervised algorithms like linear regression, classification, clustering, etc. This library works in association with Numpy and SciPy.

Installation of Scikit-learn:

```
pip install scikit-learn
```

- **Flask :**

Flask is a lightweight WSGI web application framework. It is designed to make getting started quick and easy, with the ability to scale up to complex applications. It began as a simple wrapper around Werkzeug and Jinja and has become one of the most popular Python web application frameworks. It offers suggestions, but doesn't enforce any dependencies or project layout. It is up to the developer to choose the tools and libraries they want to use. There are many extensions provided by the community that make adding new functionality easy. Web Framework: Web Application Framework or simply Web Framework represents collection of libraries and modules that enables a web application developer to write applications without having to bother about lowlevel details such as protocols, thread management etc

Flask: Flask is a web application framework written in Python. It is developed by Armin Ronacher, who leads an international group of Python enthusiasts named Pocco. Flask is based on the Werkzeug WSGI toolkit and Jinja2 template engine. Both are Pocco projects.

WSGI: Web Server Gateway Interface (WSGI) has been adopted as a standard for Python web application development. WSGI is a specification for a universal interface between the web server and the web applications.

Jinja2 : Jinja2 which is its template engine. Flask is a lightweight Web Server Gateway Interface WSGI web application framework that was created to make getting started easy and making it easy for new beginners. With the tendency to scale up to complex applications. Flask was designed to be easy to use and extend. The idea behind Flask is to build a solid foundation for web applications of different complexity. From then on you are free to plug in any extensions you think you need. Also you are free to build your own modules. Flask is great for all kinds of projects. It's especially good for prototyping. Flask depends on two external libraries: the Jinja2 template engine and the Werkzeug WSGI toolkit.

Advantages of using Flask framework are:

- There is a built-in development server and a fast debugger provided.
- Lightweight
- Secure cookies are supported.
- Request dispatching using REST.
- Support for unit testing is built-in.
- higher flexibility.
- higher compatibility with latest technologies.
- high scalability for simple web applications.
- technical experimentation.
- customization.
- slightly higher framework performance.
- easier to use for simple cases.
- smaller size of the code base.

- ## HYPER TEXT MARKUP LANGUAGE

Hypertext Markup Language (HTML), the languages of the World Wide Web (WWW), allows users to produces Web pages that include text, graphics and pointer to other Web pages(Hyperlinks). HTML is not a programming language but it is an application of ISO Standard 8879, SGML (Standard Generalized Markup Language), but specialized to hypertext and adapted to the Web. The idea behind Hypertext is that instead of reading text in rigid linear structure, we can easily jump from one point to another point.

We can navigate through the information based on our interest and preference. A markup language is simply a series of elements, each delimited with special characters that define how text or other items enclosed within the elements should be displayed. Hyperlinks are underlined or emphasized works that load to other documents or some portions of the same document.

HTML can be used to display any type of document on the host computer, which can be geographically at a different location. It is a versatile language and can be used on any platform or desktop. HTML provides tags (special codes) to make the document look attractive.

HTML tags are not case-sensitive. Using graphics, fonts, different sizes, color, etc., can enhance the presentation of the document. Anything that is not a tag is part of the document itself.

.

Basic HTML Tags

<!-- -->	Specifies comments
<A>… ....</A>	Creates hypertext links
<B>… ....</B>	Formats text as bold
<BIG>............ </BIG>	Formats text in large font
<BODY>…</BODY>	Contains all tags and text in the

	HTML document
<CENTER>...</CENTER>	Creates text
<DD>…</DD>	Definition of a term
<DL>...</DL>	Creates definition lis
<FONT>…</FONT>	Formats text with a particular font
<FORM>...</FORM>	Encloses a fill-out form
<FRAME>...</FRAME>	Defines a particular frame in a set of frames
<H#>…</H#>	Creates headings of different levels
<HEAD>...</HEAD>	Contains tags that specify information about a document
<HR>...</HR>	Creates a horizontal rule
<HTML>…</HTML>	Contains all other HTML tags
<META>...</META>	Provides meta-information about a document
SCRIPT>…</SCRIPT>	Contains client-side or server-side script
<TABLE>…</TABLE>	Creates a table
<TD>…</TD>	Indicates table data in a table
<TR>…</TR>	Designates a table row
<TH>…</TH>	Creates a heading in a table

Advantages

- A HTML document is small and hence easy to sendsmall because it does not include formatted information.
- HTML is platform independent.
- HTML tags are not case-sensitive.

• Cascading Style Sheets

Cascading **S**tyle **S**heets, fondly referred to as CSS, is a simple design language intended to simplify the process of making web pages presentable.CSS handles the look and feel part of a web page. Using CSS, you can control the color of the text, the style of fonts, the spacing between paragraphs, how columns are sized and laid out, what background images or colors are used, layout designs, variations in display for different devices and screen sizes as well as a variety of other effects.CSS is easy to learn and understand but it provides powerful control over the presentation of an HTML document. Most commonly, CSS is combined with the markup languages HTML or XHTML.

Advantages of CSS

CSS saves time − You can write CSS once and then reuse same sheet in multiple HTML pages. You can define a style for each HTML element and apply it to as many Web pages as you want.

Pages load faster − If you are using CSS, you do not need to write HTML tag attributes every time. Just write one CSS rule of a tag and apply it to all the occurrences of that tag. So less code means faster download times.

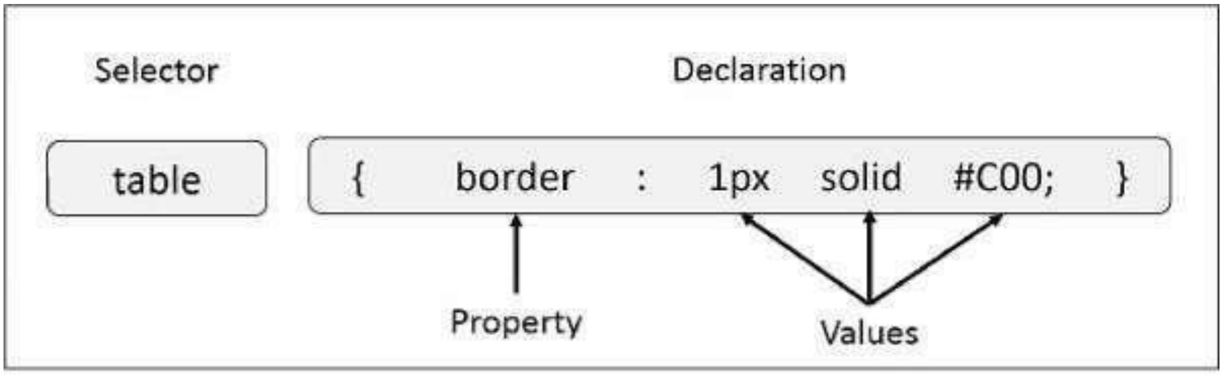
A CSS comprises of style rules that are interpreted by the browser and then applied to the corresponding elements in your document. A style rule is made of three parts −

Selector − A selector is an HTML tag at which a style will be applied. This could be any tag like <h1> or <table> etc.

Property − A property is a type of attribute of HTML tag. Put simply, all the HTML attributes are converted into CSS properties. They could be *color*, *border* etc.

Value −Values are assigned to properties. For example, *color* property can have value either *red* or *#F1F1F1* etc.

You can put CSS Style Rule Syntax as follows − selector {property: value}



• JAVA SCRIPT

JavaScript is a script-based programming language that was developed by Netscape Communication Corporation. JavaScript was originally called Live Script and renamed as JavaScript to indicate its relationship with Java. JavaScript supports the development of both client and server components of Web-based applications. On the client side, it can be used to write programs that are executed by a Web browser within the context of a Web page. On the server side, it can be used to write Web server programs that can process information submitted by a Web browser and then updates the browser’s display accordinglyEven though JavaScript supports both client and server Web programming, we prefer JavaScript at Client side programming since most of the browsers supports it. JavaScript is almost as easy to learn as

HTML, and JavaScript statements can be included in HTML documents by enclosing the statements between a pair of scripting tags

<SCRIPT>...</SCRIPT>.

SCRIPT LANGUAGE = Javascript> JavaScript statements

</SCRIPT>

Here are a few things we can do with JavaScript:

- Validate the contents of a form and make calculations.
- Animate images or rotate images that change when we move the mouse over them.
- Detect the browser in use and display different content for differentbrowsers.
- Detect installed plug-ins and notify the user if a plug-in is required. We can do much more
- with JavaScript, including creating entire application.

CHAPTER 5
SYSTEM ANALYSIS AND DESIGN

5.1 System Architecture

Basic Process

- **Data Collection:** Collect the data that the algorithm will learn from.
- **Data Preparation:** Format and engineer the data into the optimal format, extracting important features and performing dimensionality reduction.
- **Training:** Also known as the fitting stage, this is where the Machine Learning algorithm actually learns by showing it the data that has been collected and prepared.
- **Evaluation:** Test the model to see how well it performs.

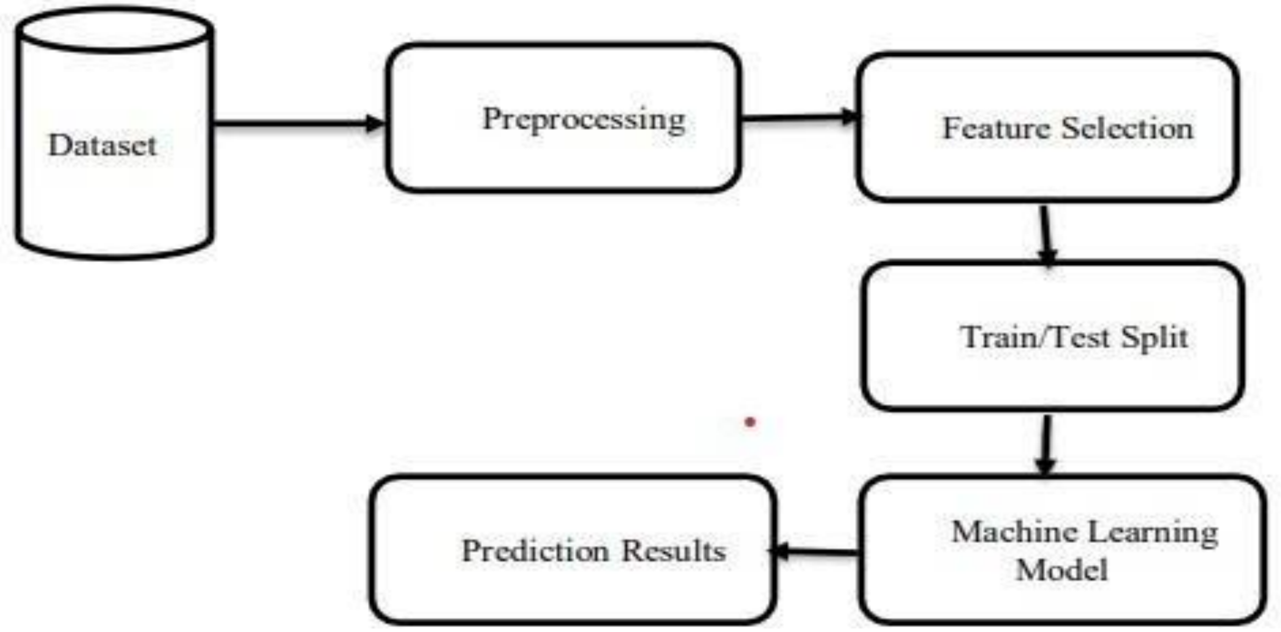


Fig 5.1 System Architecture

5.2 UML Diagrams

UML is a method for describing the system architecture in detail using the blue print. UML represents a collection of best engineering practice that has proven successful in the modeling of large and complex systems. The UML is very important parts of developing object -oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects. Using the helps UML helps project teams communicate explore potential designs and validate the architectural design of the software.

5.2.1 Use Case Diagram

Use case diagram represents the functionality of the system. Use case focus on the behavior of the system from external point of view. Actors are external entities that interact with the system.

Use cases: A use case describes a sequence of actions that provide something of measurable value to an actor and is drawn as a horizontal ellipse.

Actors: An actor is a person, organization, or external system that plays a major role.

Include:

In one form of interaction, a given use case may include another. "Include is a Directed Relationship between two use cases, implying that the behavior of the included use case is inserted into the behavior of the including use case.The first use case often depends on the outcome of the included use case. This is useful for extracting truly common behaviors from multiple use cases into a single description. The notation is a dashed arrow with include label.

Generalization

A given use case may have common behaviors, requirements, constraints, and assumptions with a more general use case. In this case, describe them once, and deal with it in the same way, describing any differences in the specialized cases. The notation is a solid line ending in a hollow triangle drawn from the specialized to the more general use case (following the standard generalization notation.

Associations

Associations between actors and use cases are indicated in use case diagrams by solid lines. An association exists whenever an actor is involved with an interaction described by a use case. Associations are modeled as lines connecting use cases and actors to one another, with an optional arrowhead on one end of the line. The arrowhead is often used to indicating the direction of the initial invocation of the relationship or to indicate the primary actor within the use case.

Identified Use Cases

The user model view encompasses a problem and solution from the preservative of those individuals whose problem the solution addresses. The view presents the goals and objectives of the problem owners and their requirements of the solution. This view is composed of use case diagram These diagrams describe the functionality provided by a system to external integrators. These diagrams contain actors, use cases, and their relation.

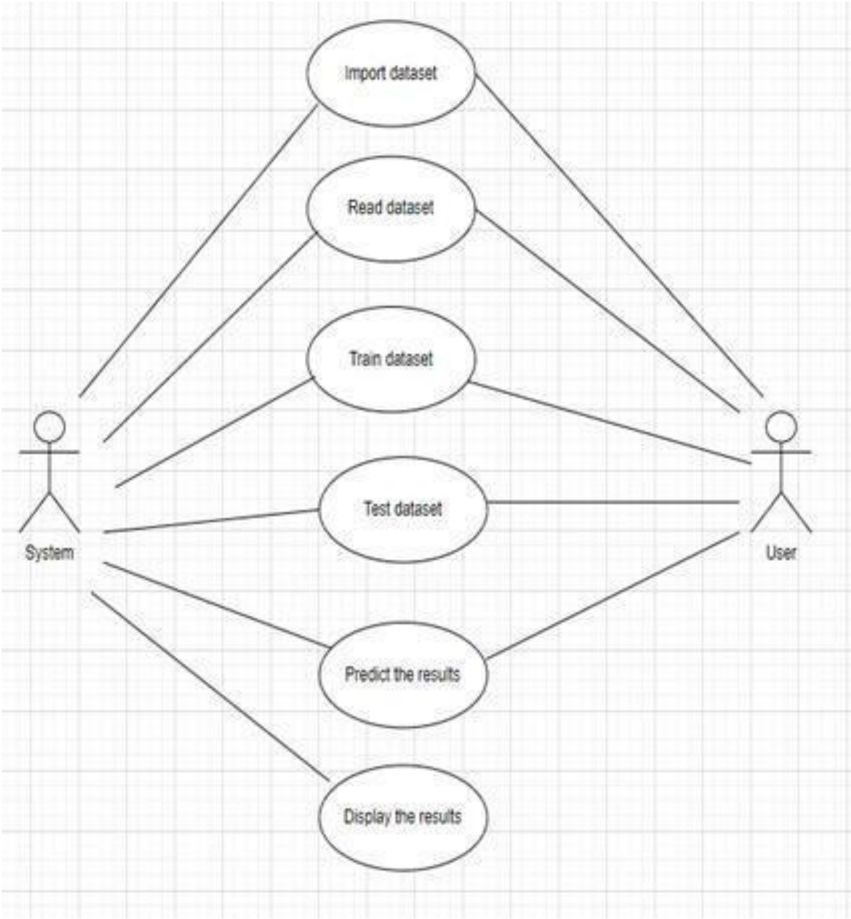


Fig 5.2: Use case diagram

5.2.2 Class Diagram

Class-based Modelling, or more commonly class-orientation, refers to the style of objectoriented programming in which inheritance is achieved by defining classes of objects; as opposed to the objects themselves (compare Prototype-based programming).

The most popular and developed model of OOP is a class-based model, as opposed to an objectbased model. In this model, objects are entities that combine state (i.e., data), behavior (i.e., procedures, or methods) and identity (unique existence among all other objects). The structure and behavior of an Object is defined by a class, which is a definition, or blueprint, of all objects of a specific type. An object must be explicitly created based on a class and an object thus created is considered to be an instance of that class. An object is similar toa structure, with the addition of method pointers, member access control, and an implicit data member which locates instances of the class (i.e, actual objects of that class) in the class hierarchy (essential for runtime inheritance features).

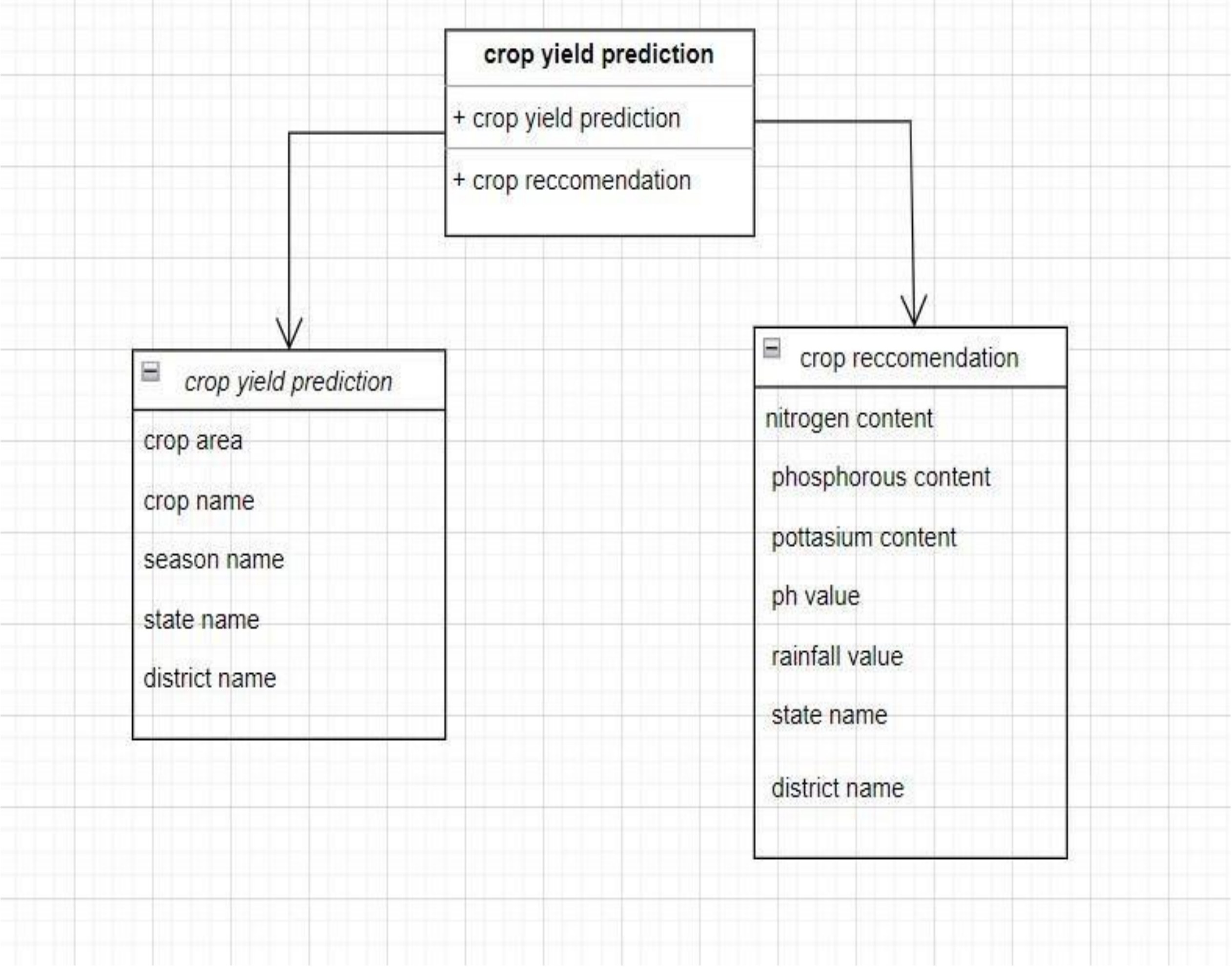


Fig 5.3: Class Diagram

5.2.3 Sequence Diagram

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams. A sequence diagram shows, as parallel vertical lines (lifelines), different processes or objects that live simultaneously, and, as horizontal arrows, the messages exchanged between them, in the order in which they occur. This allows the specification of simple runtime scenarios in a graphical manner. If the lifeline is that of an object, it demonstrates a role. Note that leaving the instance name blank can represent anonymous and unnamed instances. In order to display interaction, messages are used. These are horizontal arrows with the message name written above them. Solid arrows with full heads are synchronous calls, solid arrows with stick heads are asynchronous calls and dashed arrows with stick heads are return messages. This definition is true as of UML 2, considerably different from UML 1.x.

Activation boxes, or method-call boxes, are opaque rectangles drawn on top of life lines to represent that processes are being performed in response to the message (Execution Specifications in UML).

Objects calling methods on themselves use messages and add new activation boxes on top of any others to indicate a further level of processing. When an object is destroyed (removed from memory), an X is drawn on top of the lifeline, and the dashed line ceases to be drawn below it (this is not the case in the first example though). It should be the result of a message, either from the object itself, or another.

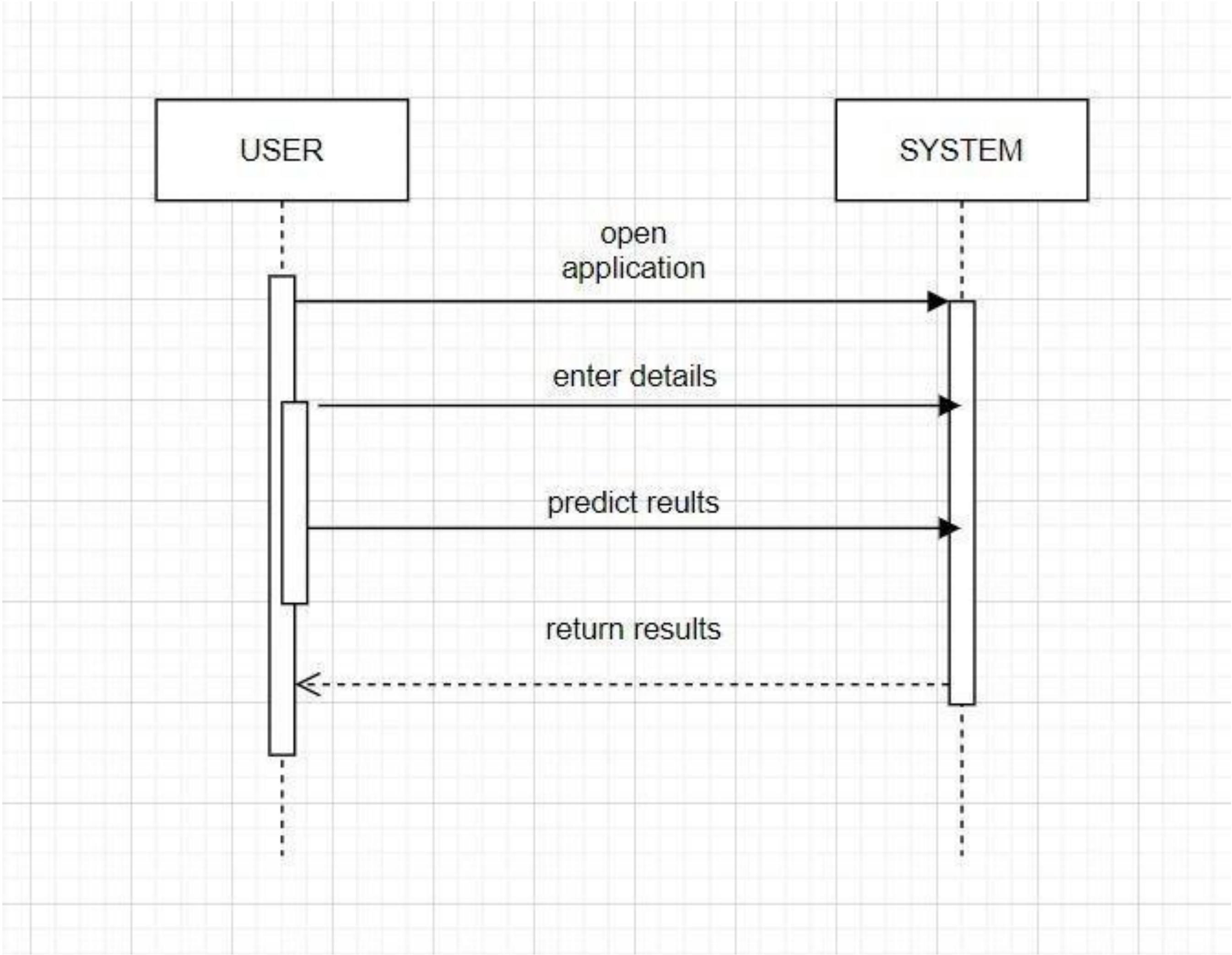


Figure 5.4: Sequence Diagram

5.2.4 Activity diagram

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step work-flows of components in a system. An activity diagram shows the overall flow of control. Activity diagrams are constructed from a limited repertoire of shapes, connected with arrows. The most important shape types:

- Rounded rectangles represent activities;
- Diamonds represent decisions;
- Bars represent the start (split) or end (join) of concurrentactivities;
- A black circle represents the start (initial state) of the workflow;
- An encircled black circle represents the end (final state)

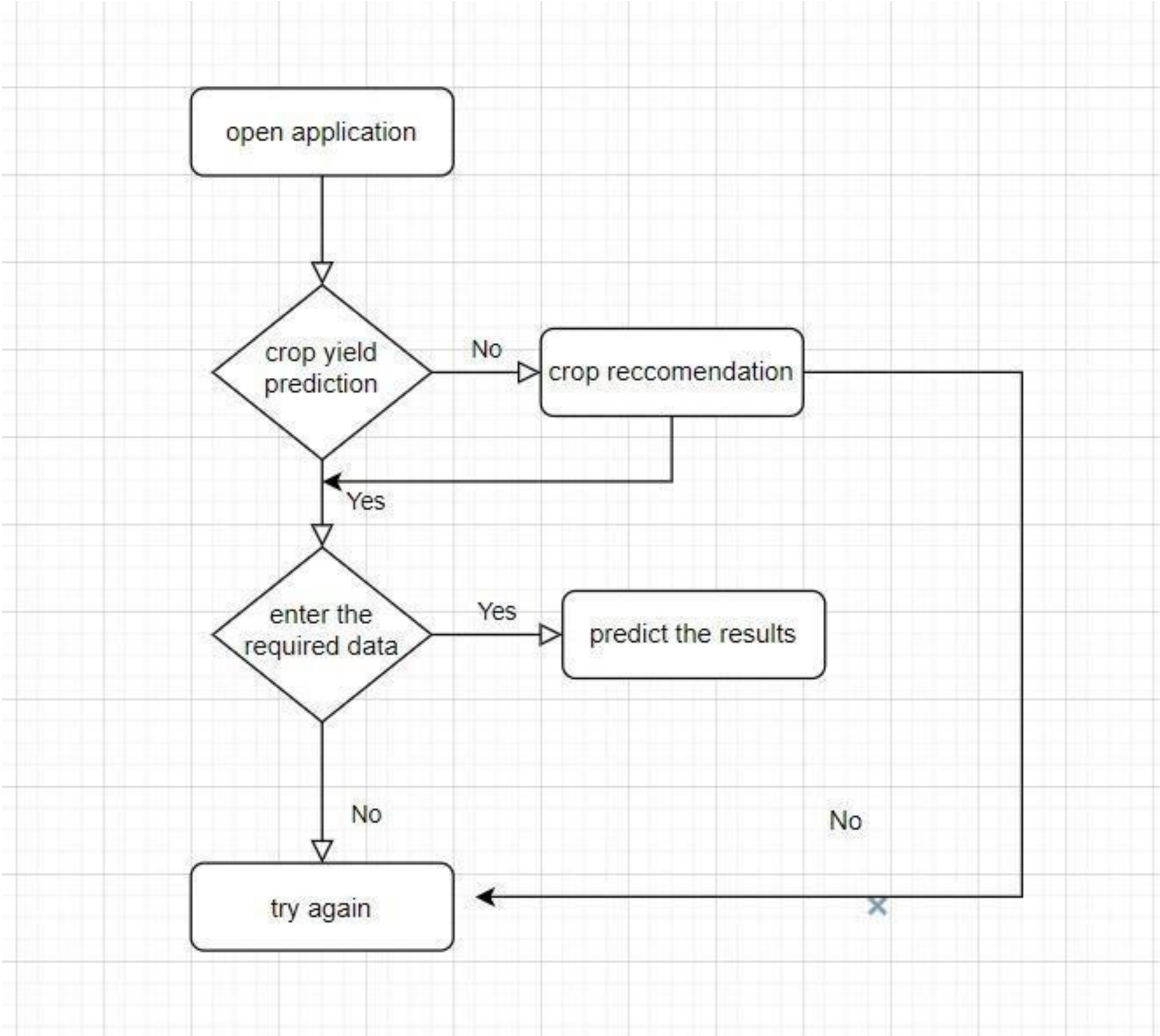


Fig 5.5 Activity Diagram

CHAPTER 6
IMPLEMENTATION

Implementation

The implementation of the project was divided into two .i.e. crop yield prediction and crop recommendation. User can select any one of the options according to the user's choice.

User can check with different inputs and predict the yield of the crop and recommend a crop by the crop recommendation.

Crop Yield Prediction

This module returns the predicted production of yield of crops based on the user's input. If the user wants to know the production of a particular crop, the system takes the crop as the input as well. The output of the crop yield prediction will be in tons. This means the crop prediction will written in tons. These are the following steps of the algorithm implemented:

- Step 1 : Choose the functionality i.e., crop yield prediction.
- Step 2 : Take crop ,district ,state ,season and area as inputs.
- Step 3 These values are given as input to the random forest implementation in the backend and the corresponding predictions are returned.
- Step 4 : The algorithm returns the crop production predicted.

Crop Recommendation

This module returns the predicted crop that can be suitable for area based on the user's input. If the user wants to know the recommendation of a crop, the system takes the several inputs as parameters. The output of the crop recommendation will be crop name. This means the crop recommendation will written the suitable crop name for cultivation. These are the following steps of the algorithm implemented:

- Step 1 : Choose the functionality i.e., crop yield prediction.
- Step 2 : Take district ,state , nitrogen, phosphorous, potassium, rainfall value and PH value as inputs.
- Step 3 : These values are given as input to the random forest implementation in the backend and the corresponding predictions are returned.
- Step 4 : The algorithm returns the recommended crop that is predicted.

CHAPTER 7
RESULTS AND DISCUSSIONS

7.1 Sample Code

App.py

```
from flask import Flask, render_template, request

import requests

import pickle

import numpy as np

import pandas as pd

import config

from sklearn.preprocessing import LabelEncoder

from sklearn.ensemble import RandomForestRegressor

from sklearn.model_selection import train_test_split

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

app = Flask(__name__)

def weather_fetch(city_name):

    """

    Fetch and returns the temperature and humidity of a city

    :params: city_name

    :return: temperature, humidity

    """

          api_key

      =config.weather_api_key
```

```
     base_url = "http://api.openweathermap.org/data/2.5/weather?"
    complete_url = base_url + "appid=" + api_key + "&q=" + city_name
    response = requests.get(complete_url)
    x = response.json()
    if x["cod"] != "404":
        y = x["main"]
        temperature = round((y["temp"] - 273.15), 2)
        humidity = y["humidity"]
        return temperature, humidity
    else:
            return None
    @app.route('/')
    def main():
        return render_template('index.html')
@app.route('/predict')
def man():
    return render_template('home.html')
@ app.route('/crop-recommend')
def crop_recommend():
    return render_template('crop_rec.html')
```

```
@ app.route('/crop-predict', methods=['POST'])
def crop_prediction():
    if request.method == 'POST':
        N = int(request.form['nitrogen'])
        P = int(request.form['phosphorous'])
        K = int(request.form['pottasium'])
        ph = float(request.form['ph'])
        rainfall = float(request.form['rainfall'])
        # state = request.form.get("stt")
        city = request.form.get("city")
        if weather_fetch(city) != None:
            temperature, humidity = weather_fetch(city)
            data = np.array([[N, P, K, temperature, humidity, ph, rainfall]])
            my_prediction = model.predict(data)
            final_prediction = my_prediction[0]
            return render_template('crem.html', prediction=final_prediction)
        else:
            return render_template('try_again.html')
@ app.route('/crop-pred')
def crop_pred():
    return render_template('cred.html')
```

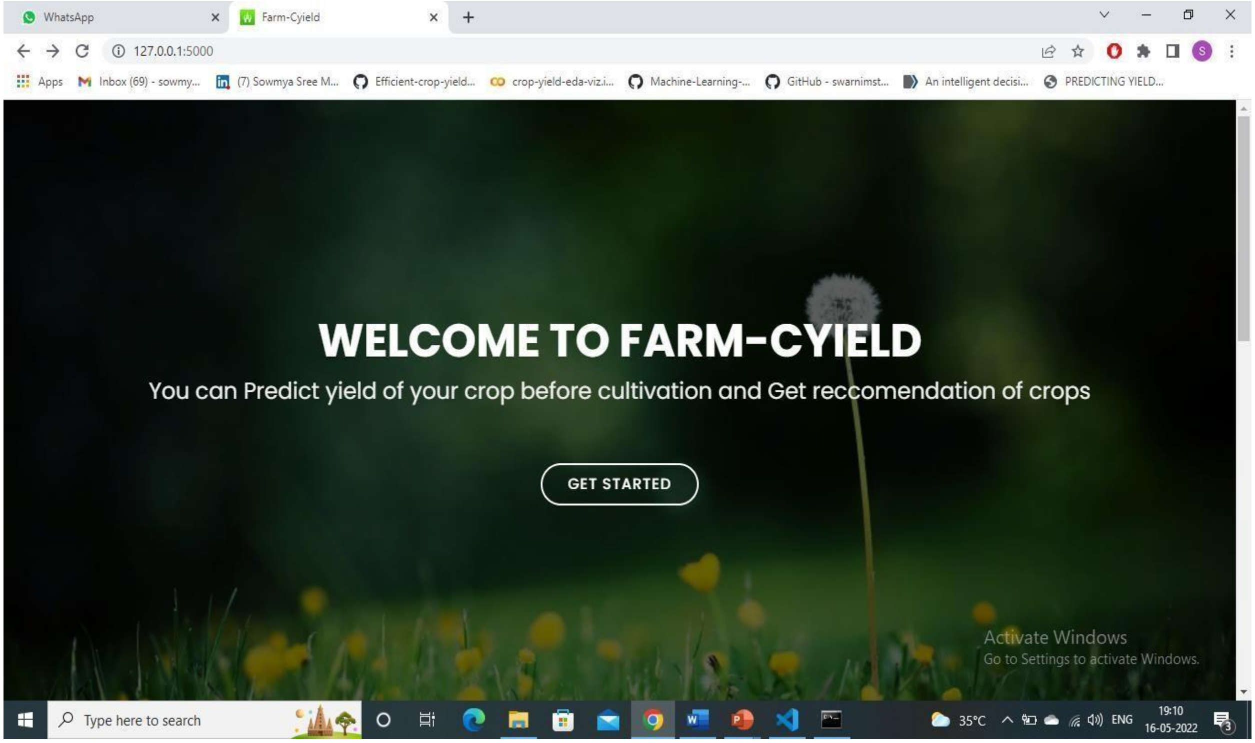
```
@ app.route('/crop-yield', methods=['POST'])
def crop_yield():
  if request.method == 'POST':
    crop_input = request.form.get("crop")
    area = int(request.form['area'])
    season = int(request.form['season'])
    state = request.form.get("stt")
    city = request.form.get("city")
     Dis = city.strip().upper()
    State = state.upper()
    data = pd.read_csv("crop_production.csv")
    data.head(7)
    df = data.copy()
    df.dropna(axis=0, inplace=True)
    df["Crop"].value_counts()
    crop_count = df["Crop"].value_counts()
    df = df.loc[df["Crop"].isin(crop_count.index[crop_count > 1500])]
    crop_name = crop_input.title()
    crop = df[(df["Crop"] == crop_name)]
    crop.head()
```

```
dt = crop.copy()
le = LabelEncoder()
dt['State_Name'] = dt['State_Name'].str.upper()
dt["district"] = le.fit_transform(dt["District_Name"])
dt['season'] = le.fit_transform(dt["Season"])
dt["state"] = le.fit_transform(dt["State_Name"])
X = dt[["Area", "district", "season", "state"]]
y = dt["Production"]
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.15)
model = RandomForestRegressor()
model.fit(X_train, y_train)
my_dict = pd.Series(dt.District_Name.values, index=dt.district).to_dict()
key_list = list(my_dict.keys())
val_list = list(my_dict.values())
position = val_list.index(Dis)
district_id = key_list[position]
state_id = dt[dt.State_Name == State]['state'].values[0]
x = [[area, district_id, season, state_id]]
ynew = model.predict(x)
prediction = ynew[0]
return render_template('cropres.html', prediction=prediction)
```

```
else:
    return render_template('try_again.html')
if __name__ == "__main__":
    app.run(debug=True)
```

7.2 Output Screens

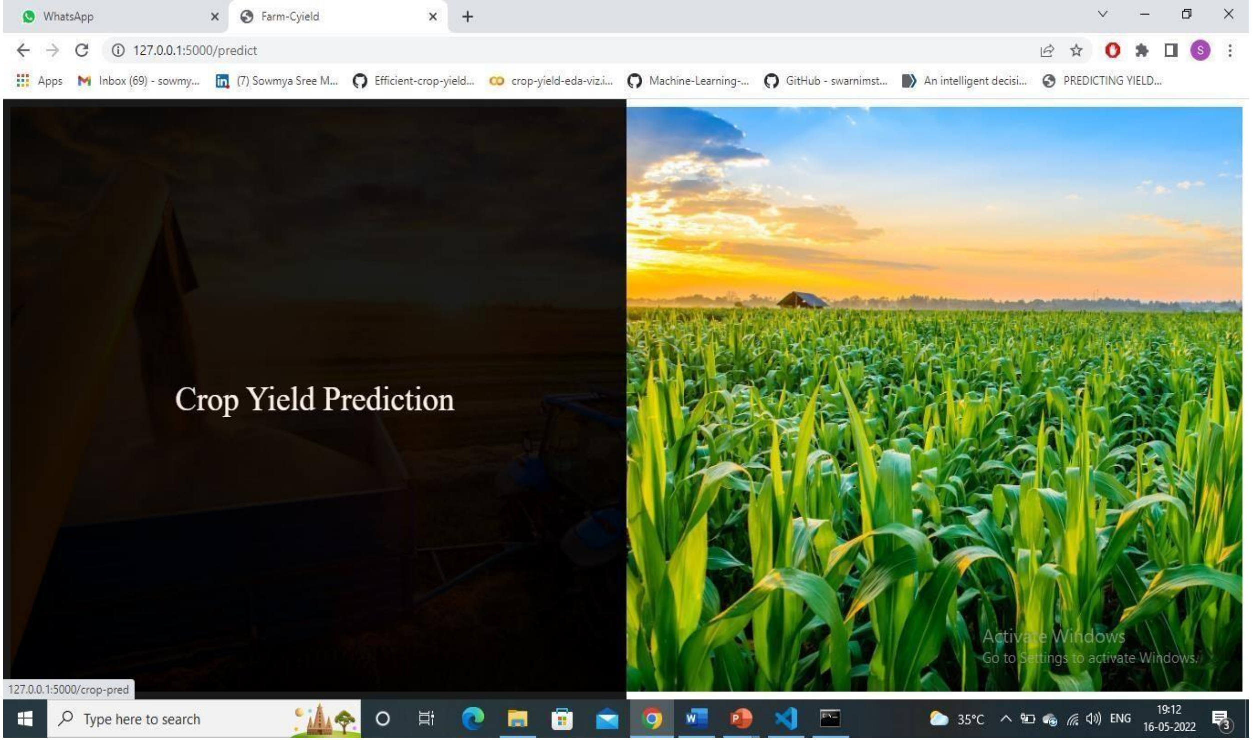
Home page



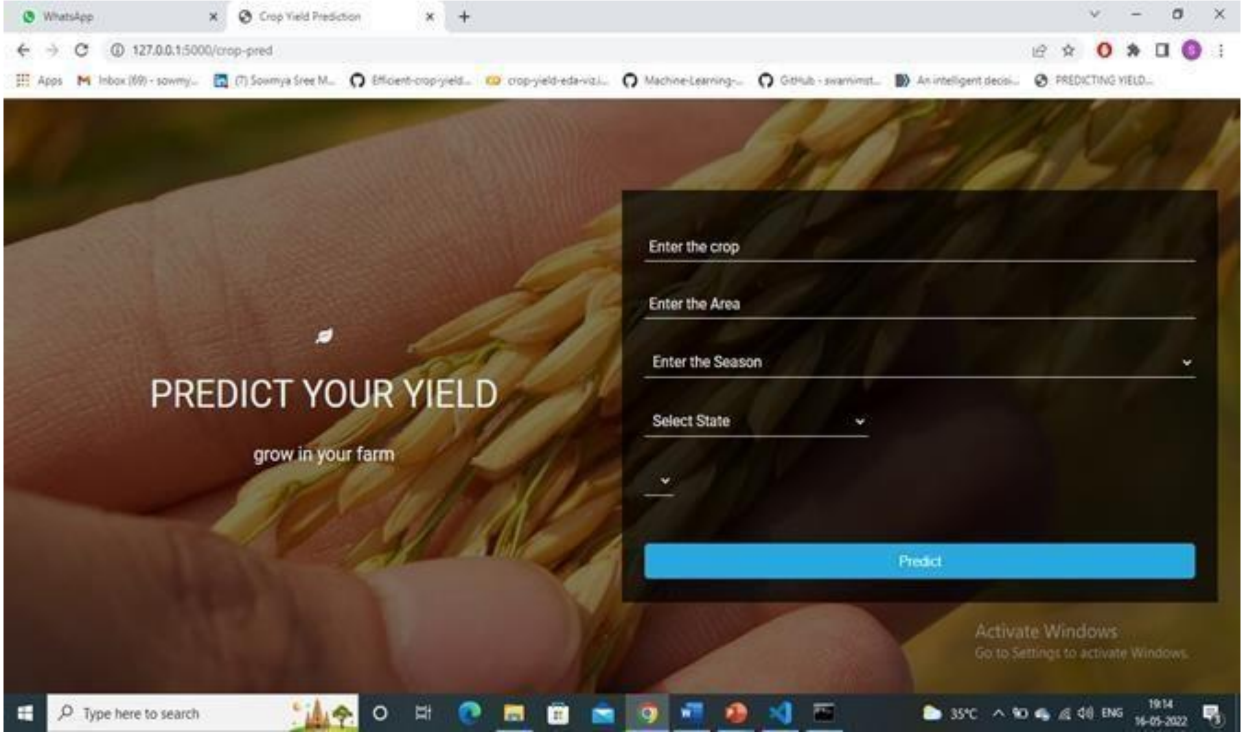
WELCOME TO FARM-CYIELD

You can Predict yield of your crop before cultivation and Get reccomendation of crops

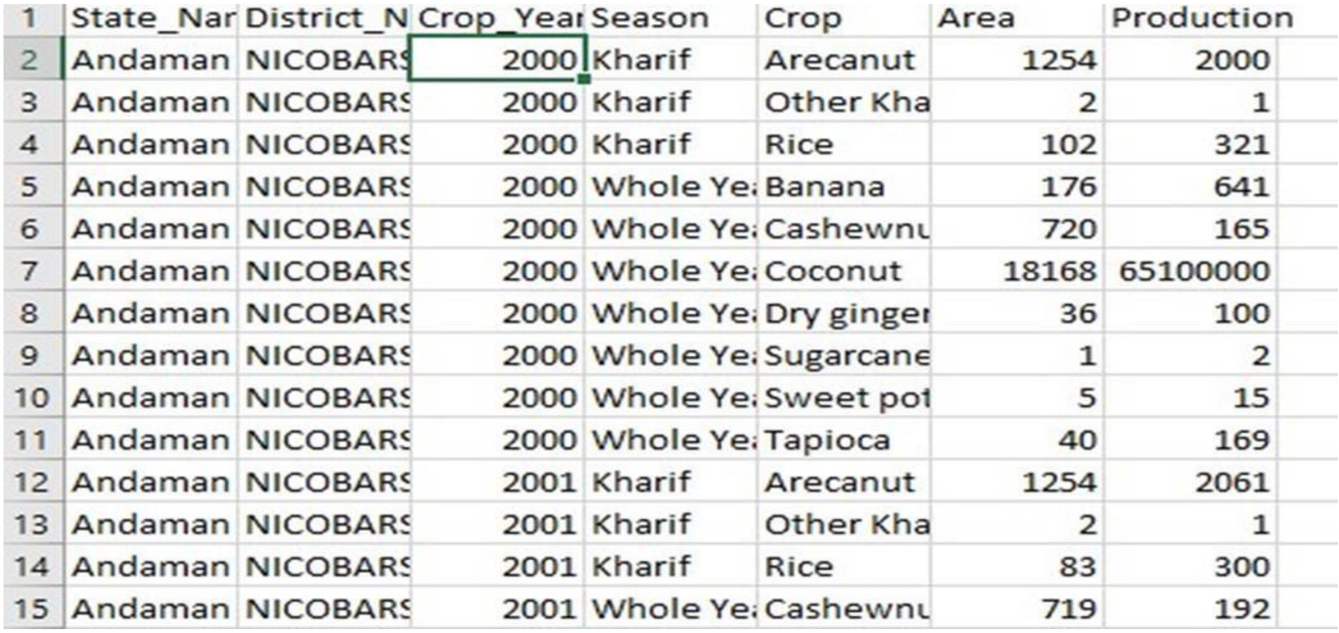
Prediction Page



Yield Prediction Page

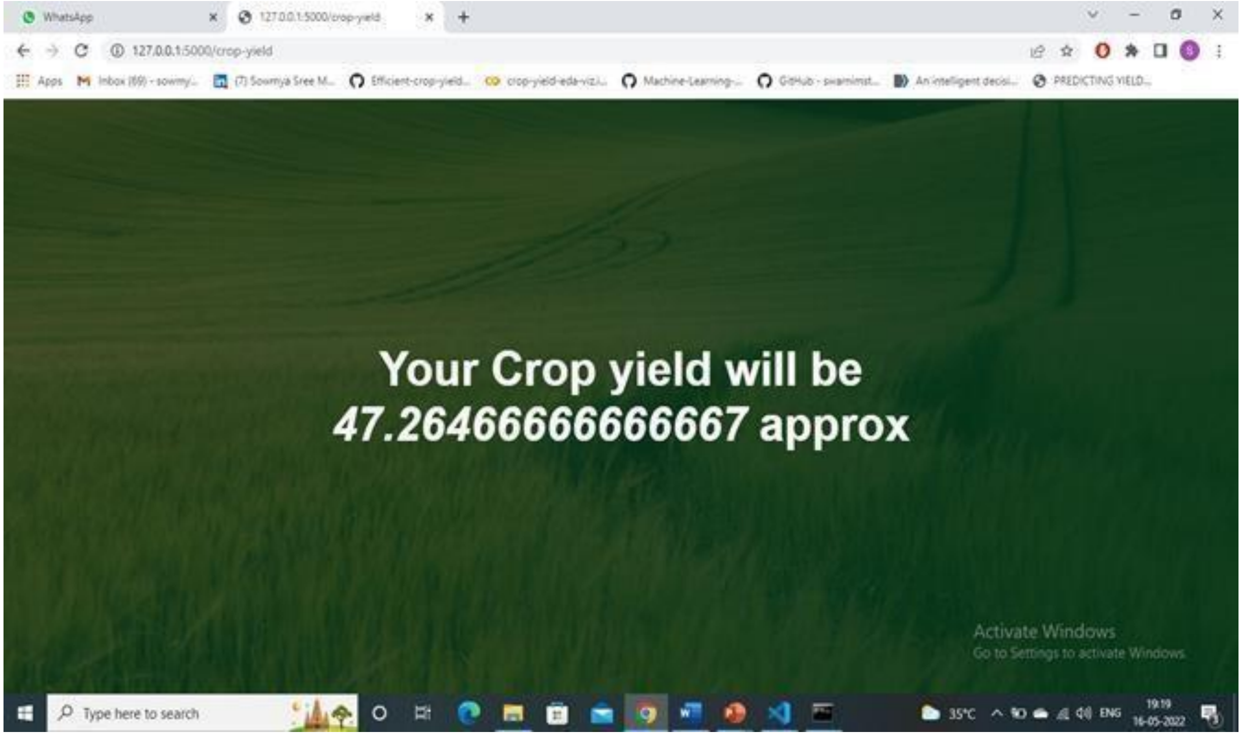


Crop Yield Prediction Dataset

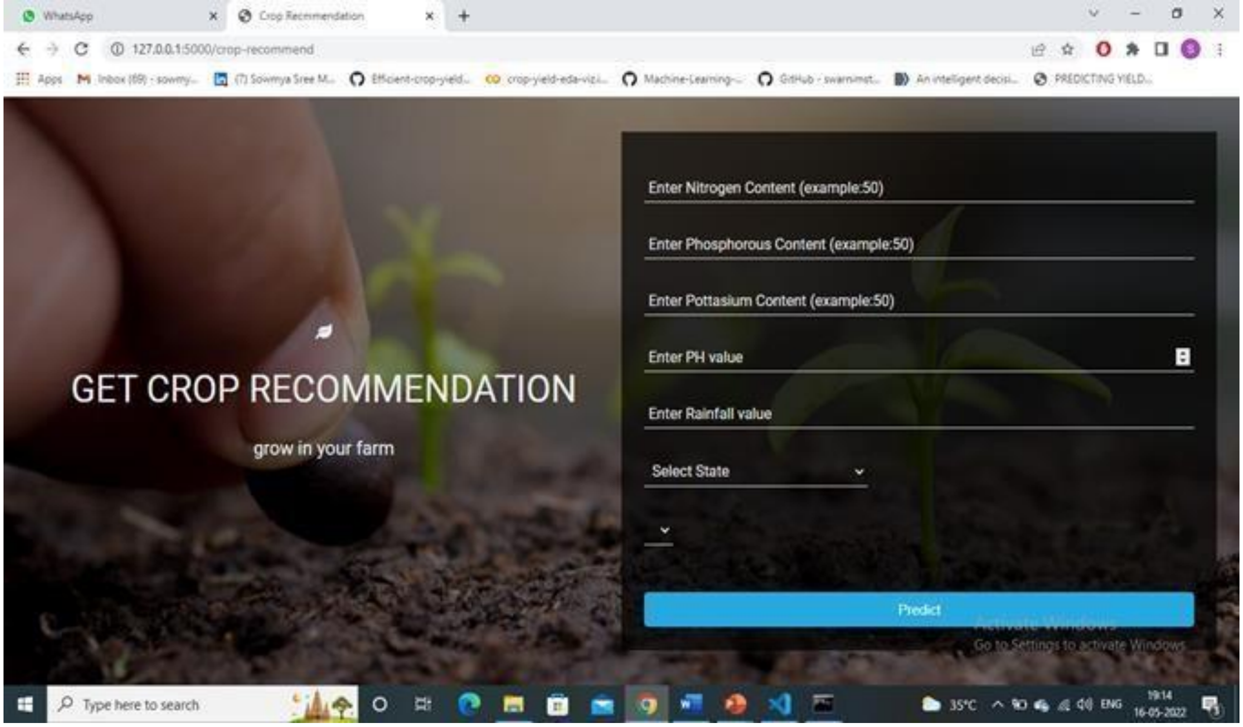


1	State_Nar	District_N	Crop_Year	Season	Crop	Area	Production
2	Andaman	NICOBARS	2000	Kharif	Arecanut	1254	2000
3	Andaman	NICOBARS	2000	Kharif	Other Kha	2	1
4	Andaman	NICOBARS	2000	Kharif	Rice	102	321
5	Andaman	NICOBARS	2000	Whole Ye	Banana	176	641
6	Andaman	NICOBARS	2000	Whole Ye	Cashewnu	720	165
7	Andaman	NICOBARS	2000	Whole Ye	Coconut	18168	65100000
8	Andaman	NICOBARS	2000	Whole Ye	Dry ginger	36	100
9	Andaman	NICOBARS	2000	Whole Ye	Sugarcane	1	2
10	Andaman	NICOBARS	2000	Whole Ye	Sweet pot	5	15
11	Andaman	NICOBARS	2000	Whole Ye	Tapioca	40	169
12	Andaman	NICOBARS	2001	Kharif	Arecanut	1254	2061
13	Andaman	NICOBARS	2001	Kharif	Other Kha	2	1
14	Andaman	NICOBARS	2001	Kharif	Rice	83	300
15	Andaman	NICOBARS	2001	Whole Ye	Cashewnu	719	192

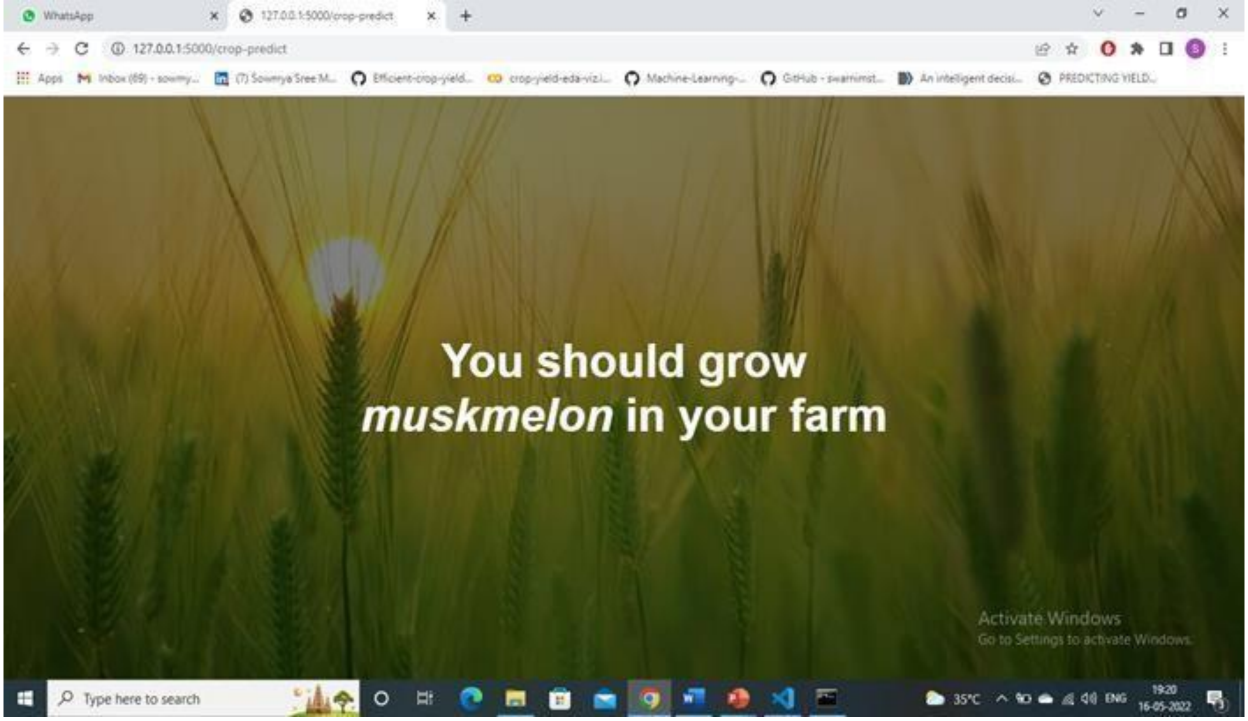
Yield Prediction Output Page


Your Crop yield will be *47.26466666666667* approx

Crop Recommendation Page



Crop Recommendation Output



You should grow *muskmelon* in your farm

CHAPTER 8

CONCLUSION

Conclusion

Nowadays farmers are facing so many challenges due to variations in weather ,climate and other such environmental conditions have become a major risk for the healthy existence of agriculture. This system is proposed to deal with the increasing rate of farmer suicides and to help them to grow financially stronger. The Crop Recommender system helps the farmers to predict the yield of a given crop and also helps them to decide which crop to grow.To solve these challenges and get rid of these problems we intended to build a good model for predicting crop yield using machine learning algorithms, those are linear regression, Random Forest.

Random Forest classifier was used for the crop yield prediction and recommendation of crop for chosen district and state. Implemented a system to crop yield prediction and recommendation of crop from the collection of past data. The proposed technique helps farmers in decision making of which crop to cultivate in the field. This work is employed to search out the gain knowledge about the crop that can be deployed to make an efficient and useful harvesting. By this model we can predict the yield of a crop by considering the crop ,season, area and other factors . This can be done by previous year statistics.

CHAPTER 9

REFERENCES

References

[1] Keita, N. Improving cost-effectiveness and relevance of agricultural censuses in africa: Linking population and agricultural censuses, 2018.

[2] Keita, N. Improving cost-effectiveness and relevance of agricultural censuses in africa: Linking population and agricultural censuses. 04 2019.

[3] Kuwata, K. and Shibasaki, R. Estimating crop yields with deep learning and remotely sensed data. In Geoscience and Remote Sensing Symposium (IGARSS), 2015 IEEE

Priya SRK, S. K. (2009). A study on pre-harvest forecast of sugarcane yield using climatic variables, Statistics and Applications. 1-8.

[4] R. Agrawal, R. J. (1999). Mining the most interesting rules, in: Proceedings of the 1999.SIGKDD International Conference on Knowledge Discovery and Data Mining, 145154.

[5] Rossana MC, L. D. (2013).A Prediction Model Framework for Crop Yield Prediction. Asia Pacific Industrial Engineering and Management Society Conference Proceedings Cebu, Phillipines, 185.

[6] Shivnath Ghosh, S. K. (2014). Machine Learning for Soil Fertility and Plant Nutrient Management. International Journal on Recent and Innovation Trends in Computing, 2 (2), 292-297.

[7] Utkarsha P, N. N. (2014). Evaluation of Modified K-Means Clustering Algorithm in Crop prediction. International Journal of Advanced Computer Research., 1–1.

[8] Groundnut Crop Yield Prediction Using Machine Learning Techniques by Vinita Shah and Prachi Shah.

[9] Developing regression model to forecast the rice yield at Raipur condition by A Jain, JL Chaudhary, MK Beck and Love Kumar

[10] Machine learning approach for forecasting crop yield based on climatic parameters by S.Veenadhari, Dr. Bharat Misra & Dr. CD Singh, International Conference on Computer Communication and Informatics (ICCCI -2014), Jan, 2014 .

[11] Predicting Yield of the Crop Using Machine Learning Algorithm by P.Priya, U.Muthaiah & M.Balamurugan, International Journal of Engineering Sciences & Research Technology (IJESRT), April, 2018.

[12] Heuristic Prediction of Crop Yield using Machine Learning Technique by S. Pavani, Augusta Sophy Beulet P, International Journal of Engineering and Advanced Technology (IJEAT) Volume-9, December 2019