Crop And Fertilizer Recommendation System Deployed On Cloud

Project report submitted in partial fulfillment of the requirement for the degree of Bachelor of Technology

in

Computer Science and Engineering

By

Arpit Mehta(181466)

Under the supervision of

Dr. Rajinder Sandhu

to

Department of Computer Science & Engineering and Information Technology

Jaypee University of Information Technology Waknaghat, Solan-173234, Himachal Pradesh

CERTIFICATE

I hereby declare that the work presented in this report entitled **“Crop and Fertilizer Recommendation System Deployed On Cloud ”** in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology** in **Computer Science and Engineering** submitted in the department of Computer Science & Engineering and Information Technology**,** Jaypee University of Information Technology Waknaghat is an authentic record of my own work carried out over a period from August 2021 to December 2021 under the supervision of **Dr. Rajinder Sandhu, Assistant Professor (SG), Department of Computer Science & Engineering and Information Technology.**

The matter embodied in the report has not been submitted for the award of any other degree or diploma.

Arpit Mehta
181466

This is to certify that the above statement made by the candidate is true to the best of my knowledge.

Dr. Rajinder Sandhu
Assistant Professor (SG)
Department of Computer Science & Engineering and Information Technology.
Dated:

ACKNOWLEDGEMENT

Firstly, I express my heartiest thanks and gratefulness to almighty God for his divine blessing makes us possible to complete the project work successfully. I really grateful and wish my profound my indebtedness to Supervisor Dr. Rajinder Sandhu, Assistant Professor (SG), Department of CSE Jaypee University of Information Technology, Wakhnaghat. Deep Knowledge & keen interest of my supervisor in the field of “Image Forgery Detection” to carry out this project. Her endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts and correcting them at all stage have made it possible to complete this project.

I would like to express my heartiest gratitude to Dr. Rajinder Sandhu, Department of CSE, for his kind help to finish my project. I would also generously welcome each one of those individuals who have helped me straight forwardly or in a roundabout way in making this project a win. In this unique situation, I might want to thank the various staff individuals, both educating and non-instructing, which have developed their convenient help and facilitated my undertaking.

Finally, I must acknowledge with due respect the constant support and patients of my parents.

Arpit Mehta

ABSTRACT

As we know that India is the second-largest populated country in the world and about 60% of people in India depends upon the agriculture field. Farmers are raising the same crops regularly without trying a new variety of crops and using the same traditional methods and are using fertilizers in irregular quantities without acknowledging the sufficient content. So, this is directly hitting on crop yield and also causes damages to the top layer of soil. To overcome these problems there is a need to introduce new methods and use of technology in farming in India. The use of technology in farming is common in western countries.

So, we have designed the system applying machine learning techniques for the farmers. The system will recommend the best fitting crop for specific land based on content and weather conditions. And also, the system provides knowledge regarding the required content of fertilizers. Hence by using the system farmers can farm a new variety of crops, may improve profit and can evade soil deterioration.

TABLE OF CONTENT

Content	Page No.

Chapter-1:INTRODUCTION

1.1 Introduction

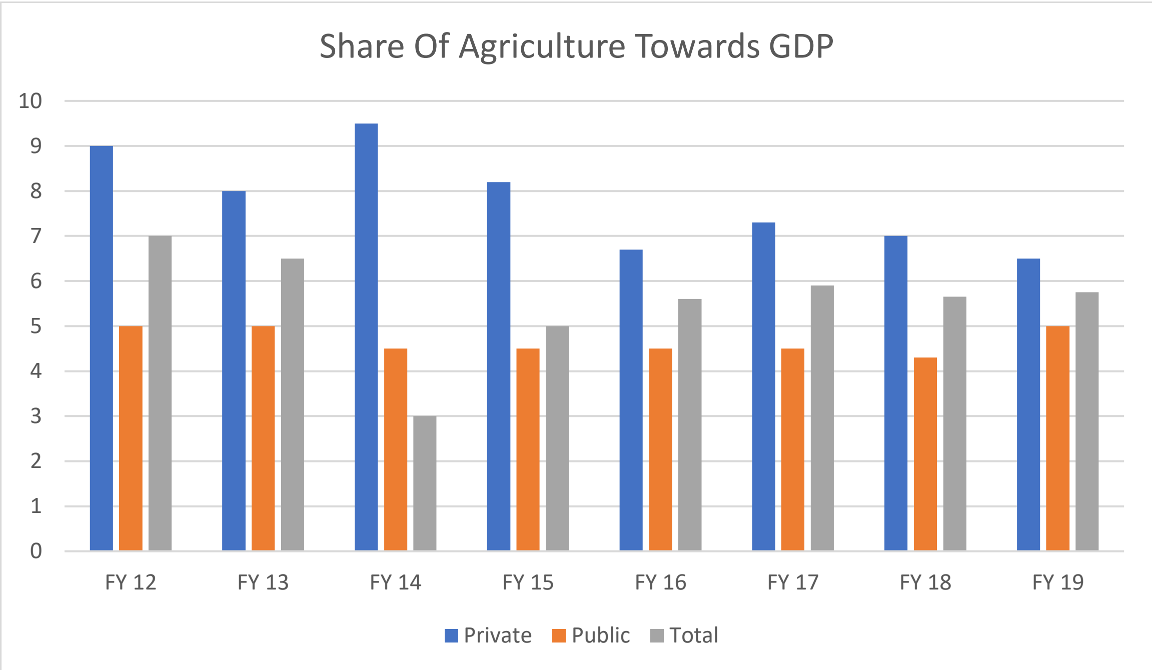
The history of agriculture in India records back to the Indus Valley civilization period and even before that in southern India. Agriculture in India has long been marked by problems like famine, drought and depletion. There were around 27 famines reported in the 17th and 18th centuries. Following freedom, agriculture in India is concentrated on the production of food grains to satisfy the expanding needs of a rapidly increasing population.The agriculture field in India is one of the most significant industries in the economy, which implies it is also a major employer. Agriculture is the prime source of livelihood for nearly 58% of India's population. Gross Value Added by agriculture sector was evaluated at Rs. 19.48 lakh crore in the Financial Year 2020 (FY20). The agriculture industry contributes to 20% of the GDP of India.India is the second-largest producer of fruits and vegetables and is the largest producer of mango and banana and is also the largest milk producer in the world. Globally India ranks 9th for agricultural exports.

There are several reasons causing misery to the farmers. The critical one is the increasing indebtedness caused by crop failures and the increasing expenses of farming.Millions of people are dependent on agriculture but the sector lacks efficiency and technology particularly in our country.The rate of growth of agricultural production in India is steadily declining in recent years. The contribution of agriculture to the GDP has been declining.Looking at the current circumstances faced by farmers, we have seen that there is an increase in the suicide rate over the years. The reasons behind this situation are weather conditions, debt, family problems. Sometimes farmers are not aware of the crop which befits their soil quality, soil nutrients and soil composition and rainfall value. Famers are using the traditional cultivation techniques. These traditional techniques have proved beneficial for many years, but there is need for adopting new agriculture technologies.

The utilization of information technology may change the condition of decision making and thus farmers may yield the best way. By using the concepts of Machine Learning we can attain some beneficial models that would help the farmers to enhance their yield. Applying ML techniques to historical weather and crop production data some predictions can be made based on knowledge collected which indeed can help in improving crop productivity. Machine learning could prove highly beneficial for the agriculture sector. Machine learning is part of artificial intelligence, using large datasets and high-performance computing to create new possibilities for the agrotechnology field.

By applying Machine learning techniques and analyzing agriculture datasets, we focus to implement a crop yield prediction system. So, we have designed the system using machine learning algorithms for the improvement of farmers. Our system will recommend the best suitable crop for specific land, based on content and weather conditions.

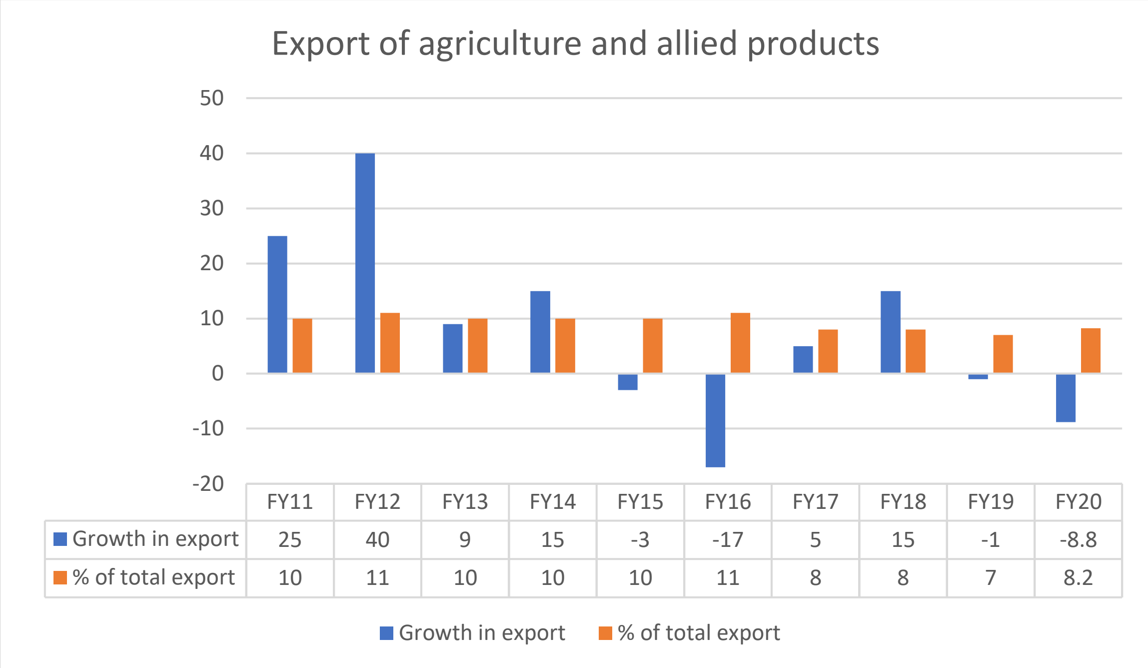
All the input data is supplied to the machine learning algorithms like Logistic Regression and Decision tree to recognize the pattern among input data. To evaluate the performance Accuracy is utilized. The two classifiers are compared with the values of Accuracy. The conclusion is based on a comparison between the two classifiers. A classifier with better accuracy is selected.



Graph 1. Falling Investment in Agriculture

Investment is the key to the development of any sector. The stats from the Graph 1 shows that the gross capital formation (GCF) of agriculture in the economy has dropped from

8.5 per cent in Financial Year 12 to 6.5 per cent in Financial Year 19. This is because the percentage of the private asset in agriculture has declined.



	FY11	FY12	FY13	FY14	FY15	FY16	FY17	FY18	FY19	FY20
Growth in export	25	40	9	15	-3	-17	5	15	-1	-8.8
% of total export	10	11	10	10	10	11	8	8	7	8.2

Graph 2. Falling Export of agriculture and allied products

Graph 2 shows the growth of the export of agriculture and products like dairy and poultry. However, no growth is visible. There is a sharp decline in the export of Agri based products.

1.2 Problem Statement

As described in the introduction, agriculture is the central component of the Indian economy. Thus, it’s very necessary to advance this field by assisting farmers to enhance their yield.

However, in India, agriculture has many problems. The majority of farmers has not been trained in the agriculture area. So, they use traditional agriculture methods. This is not efficient and leads to low yield. Moreover, the selections of crops and soil types are challenging to make for farmers.

To overcome these difficulties, we aim to propose a recommender system assisting the farmers on the cultivation techniques and guiding them to choose soil types, seasons and crops.

1.3 Objectives

The main objective of this project is to propose a system that would benefit the farmers by recommending crops and yield. Other specific objectives are:

- To guide farmers about crop yield based on previous year data.
- To guide farmers to choose proper quantity of fertilizers.
- To help farmers to increase their yield and profit.
- To deploy the system on the cloud service.

1.4 Methodology

- **Machine Learning**

  Machine learning is a kind of artificial intelligence, by which a computer learns about a task or so without being programmed for that fully. Various Machine Learning algorithms help the computer achieve this like Regression, KNN, SVM, Random Forest, etc.

  Machine Learning algorithms need a mapping function and training data to work. It learns from the training data using the mapping function and tries to predict the results. The better the training data the better algorithm performs.

  The Machine Learning system acclimates over time and tries to learn through experience, it improves over time.

- **Data Exploration**

  Data exploration is the plotting of data graphically to examine if there is any connection or relationship between data variables or if there are any patterns that could prove beneficial for us in designing our ML model.

  Data exploration is an important step to perform as it allows us to see if there are any missing or empty values. It helps to increase the performance of our model as the model performs well only if the data provided is adequate.

  Cleaning such values from data is necessary. And recognizing patterns helps in better feature selection thus better model performance.

- **K-Nearest Neighbor**

  The k-nearest neighbor algorithm is a classification algorithm. This algorithm takes a set of points and using these points it learns how to label the new points. It classifies the new input points according to the likeness of neighboring points.

- **Random Forest**

  A random forest is a type of supervised machine learning for solving classification and regression problems. Random forest algorithm is made up of multiple decision trees. It creates forests of many decision trees to evaluate the result. It takes 'K' data points and combines them to get accurate results.

- **Web Application**

  A web application is an application which we can use over the Internet utilizing a Web browser. Just like a program, there is user interaction and can are used for various purposes.

  Web applications are mostly coded in languages like HTML, CSS, JavaScript, etc. Some applications are static while some are dynamic. It depends on the web browser for rendering of the web application. Most of the websites we use are the web applications.

- **Support Vector Machine**

  Support Vector Machine is a supervised machine learning method. It is used for classification purposes. In this, we plot data items in the n-dimensional space. We achieve classification using a hyperplane that can easily classify two classes.

- **Flask**

  Flask is a web framework. It provides libraries and tools that help us in building a web application. It is written in Python. It is quite famous and easy to use to design web applications.

- **Cloud Computing**

  Cloud computing means storing and using the data and applications over the internet instead of system storage. It allows users to use the infrastructure provided by the cloud service provider. Users can use it on a pay-per-use basis even without owning the software or hardware. Services are of 3 types IaaS, PaaS and SaaS.

1.5 Organization

The organization of the report is as following:

- Chapter 1 – Introduction
- Chapter 2 – Literature Survey
- Chapter 3 – System Development
- Chapter 4 – Performance Analysis
- Chapter 5 – Conclusions

Chapter-2: LITERATURE SURVEY

In the agriculture domain, Machine Learning is recognized as an innovative field, as quality work has been produced with the help of machine learning techniques in the area of agriculture. There are different methods proposed and assessed by researchers throughout the world. The measures are taken to increase agriculture essentially includes ingraining technologies and devices to make the agriculture area more profitable for farmers by predicting the suitable crops using ML methods.

For our project work, we reviewed some of the research papers that presented some good techniques for crop prediction as well as fertilizer utilization.

1. **S. M. PANDE Et al. [1]**designed a mobile application that gives connectivity to farmers. The mobile application comprises various features that users can use for crop selection. The recommendation system supports the farmers to predict the yield of a provided crop. It allows examining the potential crops and their yield to take more accurate decisions.Machine learning algorithms such as Random Forest, SVM, and KNN were implemented and tested on the provided datasets. The different algorithms are compared using accuracy as a factor.

   The results achieved show that Random Forest is the best among tested algorithms applied on the provided datasets with an accuracy of 95%.

2. **D. Anantha Reddy Et al. [2]**proposed a crop recommendation system using ensemble learning. Different machine learning algorithms like K-nearest neighbor, decision tree and random forest are applied to the given dataset. Then using the concept of ensemble learning, using the Majority voting technique crop is recommended. Each classifier predicts a crop of its own, then using voting a single crop is recommended.

   In the end, the proposed system is able to recommend a crop with good accuracy of 92%.

3. **M. Keerthana Et al. [3]** implemented a system for crop yield prediction from previously assembled data. The machine learning algorithms with ensemble learning are used to achieve the result. Ensemble ofDecision Tree Regressor is applied to

predict the result. Different classifiers are used like linear regression, random forest, decisiontree. The accuracy of the decision tree was much better than other algorithms. AdaBoost regressor is used with the decision tree to boost the weak learner.The proposed system is able to recommend a crop with good accuracy of 92%.

4. **S. Bhanumathi Et al. [4]** proposed an algorithm to efficiently predict crop yield and for the effective use of fertilizer. They found a suitable algorithm for both the crop yield and fertilizer content. Using the machine learning algorithms, they achieved the result. For the crop yield prediction, random forest algorithm is applied which gave good accuracy. And for the fertilizer utilization Back Propagation is used, in which the dataset is split into 80% for training and 20 % testing. And using the artificial neural network they are able to achieve excellent results. For future work, they aimed to develop a web application and deploy it.

S.No.	**Title**	**Year**	**Method Used**	**Accuracy**
1	S. M. PANDE Et al. [1]	2021	Random Forest	95%
2	D. Anantha Reddy Et al. [2]	2021	Ensemble learning	92%
3	M. Keerthana Et al. [3]	2021	Decision Tree Regressor	92%
4	S. Bhanumathi Et al. [4]	2019	Random forest& Back Propagation	93%

Table 1. Comparison of research papers

Chapter-3:SYSTEM DEVELOPMENT

3.1 Feasibility Study

The feasibility study is to evaluate the operational, technical, economic and organizational point of view that whether the project is viable i.e., it is possible to do efficiently and conveniently or not.

1. Technical Feasibility:

   The image forgery detection project has a very user-friendly UI and has been developed on PyCharm which is one of the most famous Integrated development environments (IDE) for Python and Google Colab which is one of the most trusted platforms for development. The technologies used in our project is feasible.

2. Operational Feasibility:

   The project is an operational feasible model and could benefit all the digital forensic labs. It can serve its purpose of detecting manipulation in images with good accuracy.

3. Economic Feasibility:

   The project has been developed using only open-source platforms and libraries. These all libraries are made available for free by the developers on the internet. So, project development does not involve any type of expenses.

4. Market Feasibility:

   The project has been developed by keeping the demands of the clients in mind. Clients like digital forensic labs and other news fast check experts. It checks all the constraints of market feasibility.

3.2 Requirements on Major Project

3.2.1 Functional Requirements

The functional requirements are the specific demands the system should offer to the end-user so that user could accomplish their tasks. The functional requirements of this project are mentioned below:

i. User-friendly environment for the customer with simple GUI.
ii. When the user browses the image to run the detection test GUI offers a button to run the test.
iii. Proper detection of the image manipulation.

3.2.2 Non-Functional Requirements

The non-functional requirements are the non-mandatory requirements that enhance the quality of the project. Some are listed below:

i. The project provides an accuracy of more than 95% which is quite an excellent number.
ii. The project can detect the manipulation in the image within few seconds.

3.3 Use Case Diagram

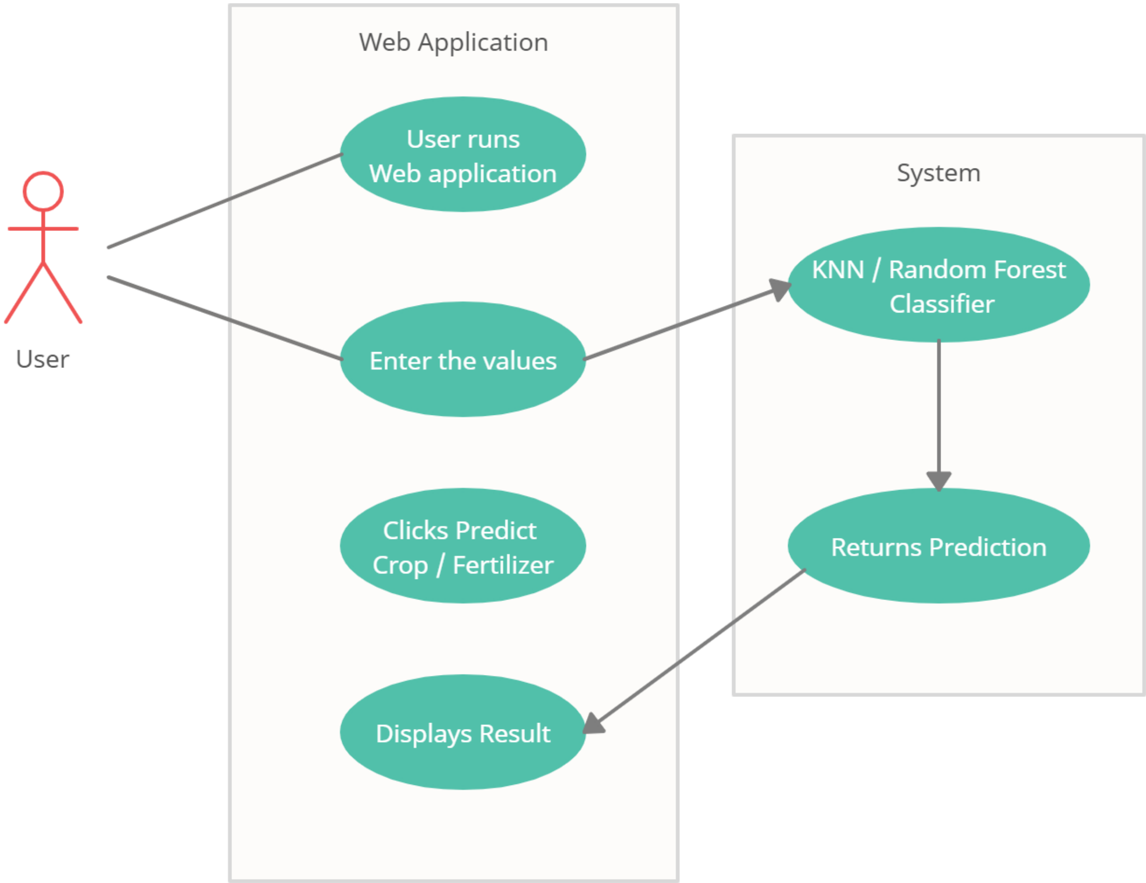


Figure 1. Use Case Diagram

3.4 DFD Diagram

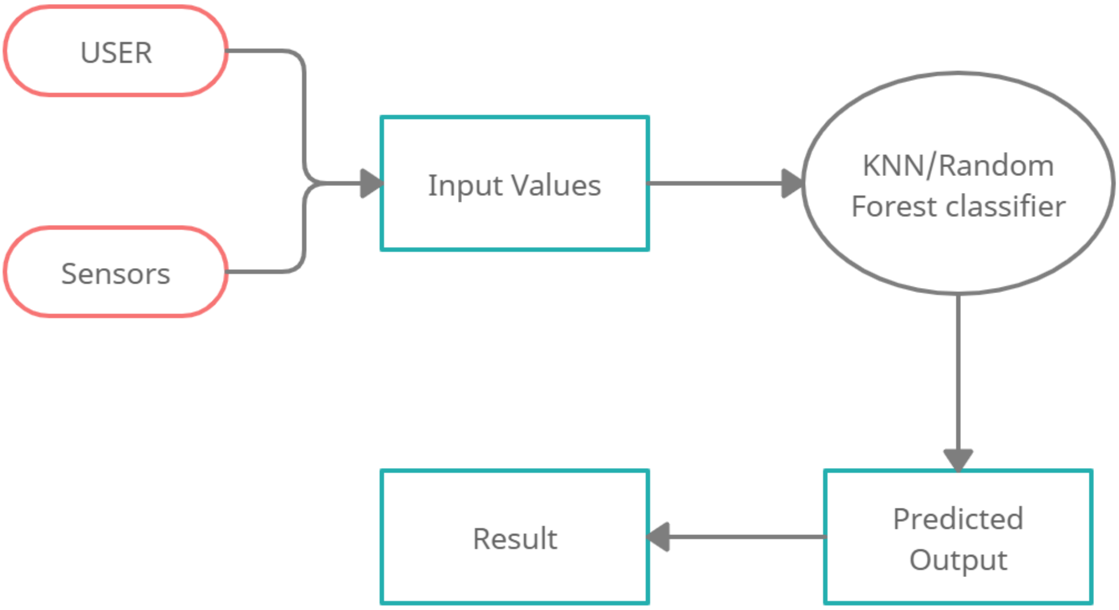


Figure 2. Data Flow Diagram

3.5 Implementation

3.5.1 Dataset

Obtaining the dataset for the agriculture domain in the Indian sub-terrain is challenging as there is no official collection of the needed datasets but distributed datasets are available on GitHub and Kaggle which could be used to obtain the desired result.

The following datasets are used in the project:

i. **Crop Recommendation Dataset.**

It is an open dataset made available on Kaggle [5]. The dataset is in CSV format and consists of 2200 records of 22 various fruits & vegetables. Dataset contains various attributes like Nitrogen, Phosphorous, Potassium, Temperature, humidity, pH and rainfall.

Out[4]:

	N	P	K	temperature	humidity	ph	rainfall	label
0	90	42	43	20.879744	82.002744	6.502985	202.935536	rice
1	85	58	41	21.770462	80.319644	7.038096	226.655537	rice
2	60	55	44	23.004459	82.320763	7.840207	263.964248	rice
3	74	35	40	26.491096	80.158363	6.980401	242.864034	rice
4	78	42	42	20.130175	81.604873	7.628473	262.717340	rice

Figure 3. Crop Recommendation Dataset

ii. **Fertilizer Prediction Dataset.**

It is an open dataset made available on Kaggle [6]. The dataset is in CSV format and consists of 100 records of various fertilizers. Dataset contains various attributes like Temperature, Humidity, Moisture, Soil Type, Crop, N, P, K, Fertilizer.

Out[79]:

	Temparature	Humidity	Moisture	Soil Type	Crop Type	Nitrogen	Potassium	Phosphorous	Fertilizer Name
0	26	52	38	Sandy	Maize	37	0	0	Urea
1	29	52	45	Loamy	Sugarcane	12	0	36	DAP
2	34	65	62	Black	Cotton	7	9	30	14-35-14
3	32	62	34	Red	Tobacco	22	0	20	28-28
4	28	54	46	Clayey	Paddy	35	0	0	Urea

Figure 4. Fertilizer Prediction Dataset

3.5.2 Data Exploration

```
In [6]:  df.shape

Out[6]:  (2200, 8)
```

```
In [7]:  df.columns

Out[7]:  Index(['N', 'P', 'K', 'temperature', 'humidity', 'ph', 'rainfall', 'label'], dtype='obje
         ct')
```

```
In [8]:  df['label'].unique()

Out[8]:  array(['rice', 'maize', 'chickpea', 'kidneybeans', 'pigeonpeas',
                'mothbeans', 'mungbean', 'blackgram', 'lentil', 'pomegranate',
                'banana', 'mango', 'grapes', 'watermelon', 'muskmelon', 'apple',
                'orange', 'papaya', 'coconut', 'cotton', 'jute', 'coffee'],
               dtype=object)
```

Figure 5. Fertilizer Prediction Dataset

This shows that there are 2200 rows and 8 columns in the dataset and shows different labels of the columns and all the unique crops in the dataset.

```
In [13]:  labels = data["label"].unique()
          data["label"].value_counts().plot(kind="bar")
          plt.show()
```

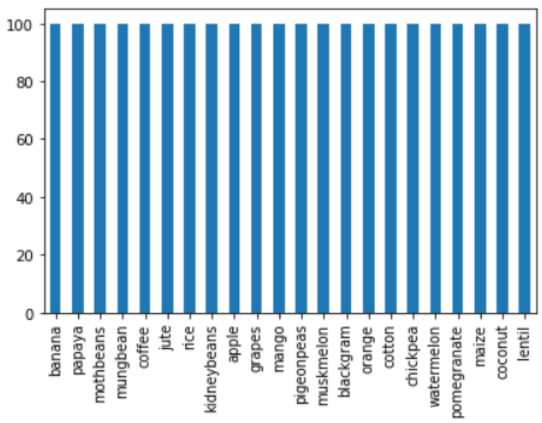


Figure 6. Dataset Plot

The plot shows that the dataset is balanced. There is no requirement to balance the dataset.

In [4]:

```
crop.describe()
```

Out[4]:

	N	P	K	temperature	humidity	ph	rainfall
count	2200.000000	2200.000000	2200.000000	2200.000000	2200.000000	2200.000000	2200.000000
mean	50.551818	53.362727	48.149091	25.616244	71.481779	6.469480	103.463655
std	36.917334	32.985883	50.647931	5.063749	22.263812	0.773938	54.958389
min	0.000000	5.000000	5.000000	8.825675	14.258040	3.504752	20.211267
25%	21.000000	28.000000	20.000000	22.769375	60.261953	5.971693	64.551686
50%	37.000000	51.000000	32.000000	25.598693	80.473146	6.425045	94.867624
75%	84.250000	68.000000	49.000000	28.561654	89.948771	6.923643	124.267508
max	140.000000	145.000000	205.000000	43.675493	99.981876	9.935091	298.560117

Figure 7. Dataset Describe

This shows that there are 2200 rows and 8 columns in the dataset and shows different labels of the columns and all the unique crops in the dataset.

In [11]:

```
sns.heatmap(df.corr(),annot=True)
```

Out[11]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7ff83ceb3950>
```

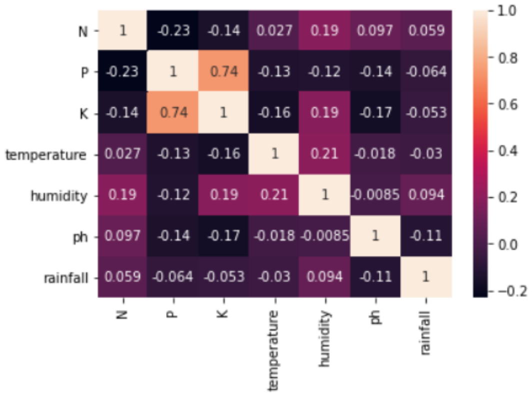


Figure 8. Heatmap

A heatmap comprises values depicting multiple shades of the same color for a specific value to be plotted.Heatmap makes it simple to visualize complicated data and understand it.

From the heatmap, we can observe that various clusters are formed based upon the features in the dataset.

3.6 Algorithm of the model

i. User Input the Values / Values are input from sensors.
ii. KNN and Random Forest classifier is used.
iii. Dataset is split in train and test.
iv. Models are saved in .pkl file.
v. Web application is designed using Flask.
vi. Web application is deployed over the cloud service.

3.7 Development

3.7.1 Importing Libraries

```
In [98]:
import os
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

from sklearn.preprocessing import StandardScaler, LabelEncoder
from sklearn.pipeline import make_pipeline
from sklearn.model_selection import train_test_split

from sklearn.neighbors import  KNeighborsClassifier
from sklearn.ensemble import RandomForestClassifier
import xgboost
from xgboost import XGBClassifier

from sklearn.metrics import accuracy_score, plot_confusion_matrix, confusion_matrix
import pickle

import warnings
warnings.filterwarnings("ignore")

%matplotlib inline
```

Figure 9. Importing necessary Libraires

3.7.2 Importing Datasets

```
14s [7] from google.colab import files

        uploaded = files.upload()

Choose Files Crop_reco...endation.csv
• Crop_recommendation.csv(application/vnd.ms-excel) - 150034 bytes, last modified: 12/19/2020 - 100% done
Saving Crop_recommendation.csv to Crop_recommendation.csv

0s [9] import io
       data = pd.read_csv(io.BytesIO(uploaded['Crop_recommendation.csv']))
```

```
[9] import io
    data = pd.read_csv(io.BytesIO(uploaded['Fertilizer_Prediction.csv']))
```

Figure 10. Importing both datasets

3.7.3 Feature Selection

```
[11] features = data[['N', 'P','K','temperature', 'humidity', 'ph', 'rainfall']]
     target = data['label']
```

```
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(features,target,test_size = 0.2,random_state =2)
```

Figure 11. Selecting Features

3.7.4 KNN Classifier

We used k-nearest neighbors algorithm (KNN) classifier. In a loop from 1 to 50, for every value error rate is plotted. We found that at K = 4 the error is least.

```
In [77]:
error_rate = []
for i in range(1, 50):
    pipeline = make_pipeline(StandardScaler(), KNeighborsClassifier(n_neighbors = i))
    pipeline.fit(X_train, y_train)
    predictions = pipeline.predict(X_test)
    accuracy = accuracy_score(y_test, predictions)
    print(f"Accuracy at k = {i} is {accuracy}")
    error_rate.append(np.mean(predictions != y_test))

plt.figure(figsize=(10,6))
plt.plot(range(1,50),error_rate,color='blue', linestyle='dashed',
         marker='o',markerfacecolor='red', markersize=10)
plt.title('Error Rate vs. K Value')
plt.xlabel('K')
plt.ylabel('Error Rate')
print("Minimum error:-",min(error_rate),"at K =",error_rate.index(min(error_rate))+1)
```

```
Accuracy at k = 1 is 0.975
Accuracy at k = 2 is 0.9681818181818181
Accuracy at k = 3 is 0.975
Accuracy at k = 4 is 0.9795454545454545
Accuracy at k = 5 is 0.9772727272727273
```

Figure 12. KNN Classifier for crop recommendation

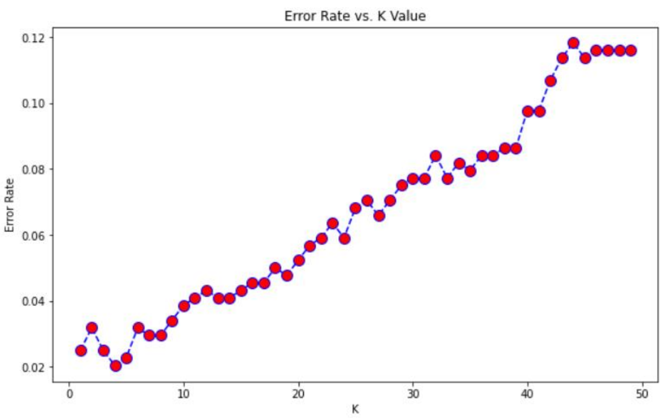


Figure 13. Error Rate VS K Value

As its clear from the plot that K = 4 is the best value, as accuracy is 97.9 and error is least.

Using KNN for the fertilizer prediction as well. We found K = 1 best with least error and accuracy of 93.5 %

In [102]:
```
error_rate = []
for i in range(1, 50):
    pipeline = make_pipeline(StandardScaler(), KNeighborsClassifier(n_neighbors = i))
    pipeline.fit(X_train, y_train)
    predictions = pipeline.predict(X_test)
    accuracy = accuracy_score(y_test, predictions)
    print(f"Accuracy at k = {i} is {accuracy}")
    error_rate.append(np.mean(predictions != y_test))

plt.figure(figsize=(10,6))
plt.plot(range(1,50),error_rate,color='blue', linestyle='dashed',
         marker='o',markerfacecolor='red', markersize=10)
plt.title('Error Rate vs. K Value')
plt.xlabel('K')
plt.ylabel('Error Rate')
print("Minimum error:-",min(error_rate),"at K =",error_rate.index(min(error_rate))+1)
```

Figure 14. KNN Classifier for fertilizer recommendation

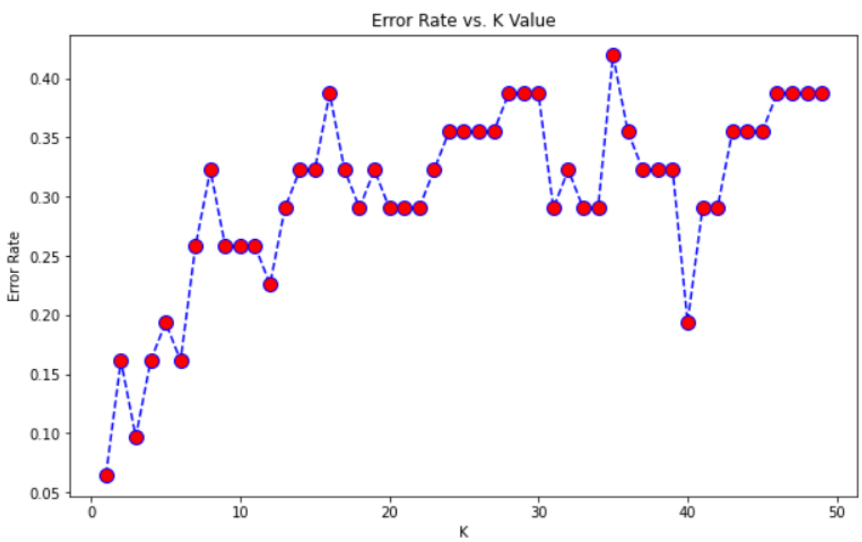


Figure 15. Error Rate VS K Value

3.7.5 Random Forest Classifier

Using the random forest algorithm classifier, it gives about 99% accuracy.

```
In [88]:
rf_pipeline = make_pipeline(StandardScaler(), RandomForestClassifier(random_state = 18))
rf_pipeline.fit(X_train, y_train)

# Accuray On Test Data
predictions = rf_pipeline.predict(X_test)
accuracy = accuracy_score(y_test, predictions)
print(f"Accuracy on Test Data: {accuracy*100}%")
plt.figure(figsize = (15,9))
sns.heatmap(confusion_matrix(y_test, predictions), annot = True)
plt.title("Confusion Matrix for Test Data")
plt.show()
```

```
Accuracy on Test Data: 99.77272727272727%
```

Figure 16. Random Forest Classifier for crop recommendation

```
In [107]:
rf_pipeline = make_pipeline(StandardScaler(), RandomForestClassifier(random_state = 18))
rf_pipeline.fit(X_train, y_train)

# Accuray On Test Data
predictions = rf_pipeline.predict(X_test)
accuracy = accuracy_score(y_test, predictions)
print(f"Accuracy on Test Data: {accuracy*100}%")
plt.figure(figsize = (15,9))
sns.heatmap(confusion_matrix(y_test, predictions), annot = True)
plt.title("Confusion Matrix for Test Data")
plt.show()
```

Figure 17. Random Forest Classifier for fertilizer recommendation

3.7.6 Saving Models

Saving the models in the .pkl file.

```
pickle.dump(rf_pipeline, open("rf_pipeline.pkl", "wb"))
pickle.dump(knn_pipeline, open("knn_pipeline.pkl", "wb"))
pickle.dump(fertname_dict, open("fertname_dict.pkl", "wb"))
pickle.dump(croptype_dict, open("croptype_dict.pkl", "wb"))
pickle.dump(soiltype_dict, open("soiltype_dict.pkl", "wb"))
```

Figure 18. Saving models as .pkl file

3.7.6 Web Application

Web application is designed using Flask and basic HTML and CSS.

```python
from flask import Flask, request
from flask_cors import CORS, cross_origin

import os
import json
import pickle
import numpy as np
from scipy import stats

app = Flask(__name__)

cors = CORS(app)

@app.after_request
def after_request(response):
    response.headers.add('Access-Control-Allow-Origin', '*')
    response.headers.add("Access-Control-Allow-Headers", "Origin, X-Requested-With, Content-Type, Accept")
    response.headers.add('Access-Control-Allow-Methods', 'GET,PUT,POST,DELETE,OPTIONS')
    response.headers.add('Access-Control-Allow-Credentials', 'true')
    return response

# Loading all Crop Recommendation Models
crop_rf_pipeline = pickle.load(
    open("C:/Users/chinm/Desktop/AgriAI_WebApp-main/AgriAI_WebApp-main/Flask_API/models/crop_recommendation/rf_pipeline.pkl", "rb")
)
crop_knn_pipeline = pickle.load(
    open("C:/Users/chinm/Desktop/AgriAI_WebApp-main/AgriAI_WebApp-main/Flask_API/models/crop_recommendation/knn_pipeline.pkl", "rb")
)
crop_label_dict = pickle.load(
    open("C:/Users/chinm/Desktop/AgriAI_WebApp-ma[illegible]ommendation/label_dictionary.pkl", "rb
)
```

Figure 19. Flask Code

```html
    </nav>
    <!-- End Navigation -->

<!-- Start Landing Page Section-->
    <!-- styling and adding img is done in the css file-->
        <div id="slides" class="carousel slide" data-ride="carousel">
            <ul class="carousel-indicators">
                <li data-target="#slides" data-slide-to="0" class="active"></li>
                <li data-target="#slides" data-slide-to="1"></li>
            </ul>
            <div class="carousel-inner">
                <div class="carousel-item active">
                    <img src="/static/img/agric_one.jpg"/>
                    <div class="carousel-caption">
                        <h1 class="display-2">Welcome to Smart Farm</h1>
                        <h3>Precision Agriculture Recommender System</h3>
                        <p><i>Using AI to improve agriculture</i></p>
                        <a class="btn btn-outline-light btn-lg" href="{{ url_for('login') }}">Login</a>
                        <a class="btn btn-outline-light btn-lg" href="{{ url_for('register') }}">Register</a>
                    </div>
                </div>
                <div class="carousel-item">
                    <img src="/static/img/agric_two.jpg"/>
                    <div class="carousel-caption">
                        <h1 class="display-2">Welcome to Smart Farm</h1>
                        <h3>Precision Agriculture Recommender System</h3>
                        <p><i>Using AI to improve agriculture</i></p>
                        <a class="btn btn-outline-light btn-lg" href="{{ url_for('login') }}">Login</a>
                        <a class="btn btn-outline-light btn-lg" href="{{ url_for('register') }}">Register</a>
                    </div>
                </div>
            </div>
        </div>
<!-- End Landing Page Section-->
</div>
<!-- End Home Section-->


<!-- footer Section -->
<div class="offset text-center">
<footer class="container">
  <p>&copy; <i>SpefacTech</i> 2019</p>
</footer>
</div>
```

Figure 20. HTML

3.8 Deployment

Web application made using Flask is now deployed on the cloud service. After deployment it is ready to use.

Using Crop Recommender:

The User need to input the values of N, P, K, temp, etc. It would recommend the crop with percentage using both KNN and Random Forest.

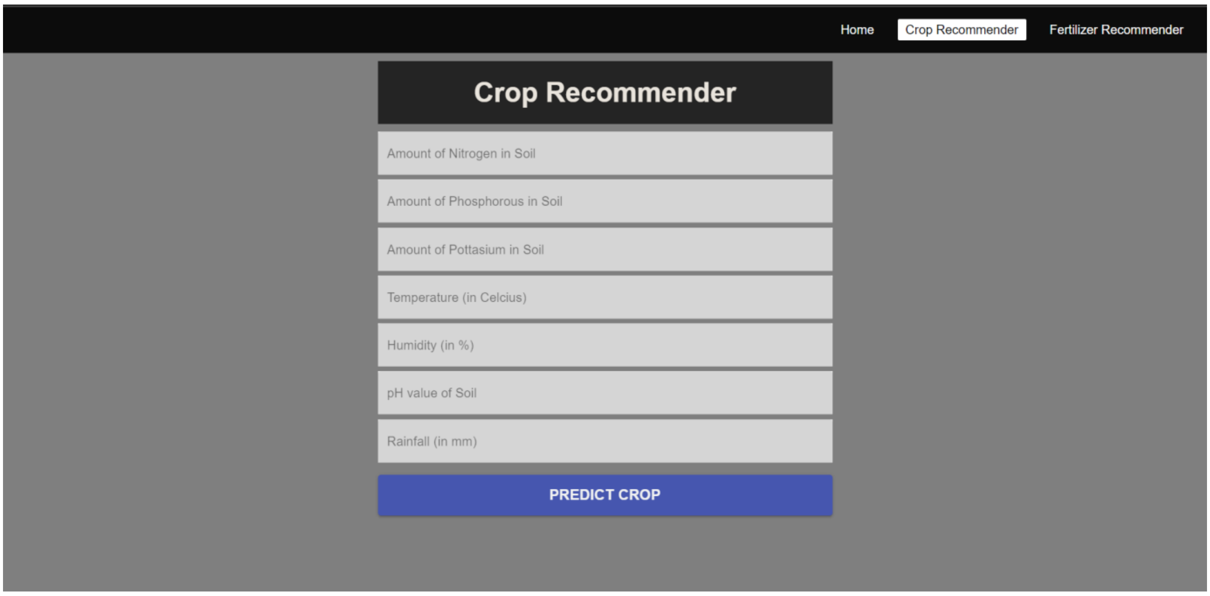


Figure 21. Crop Recommender

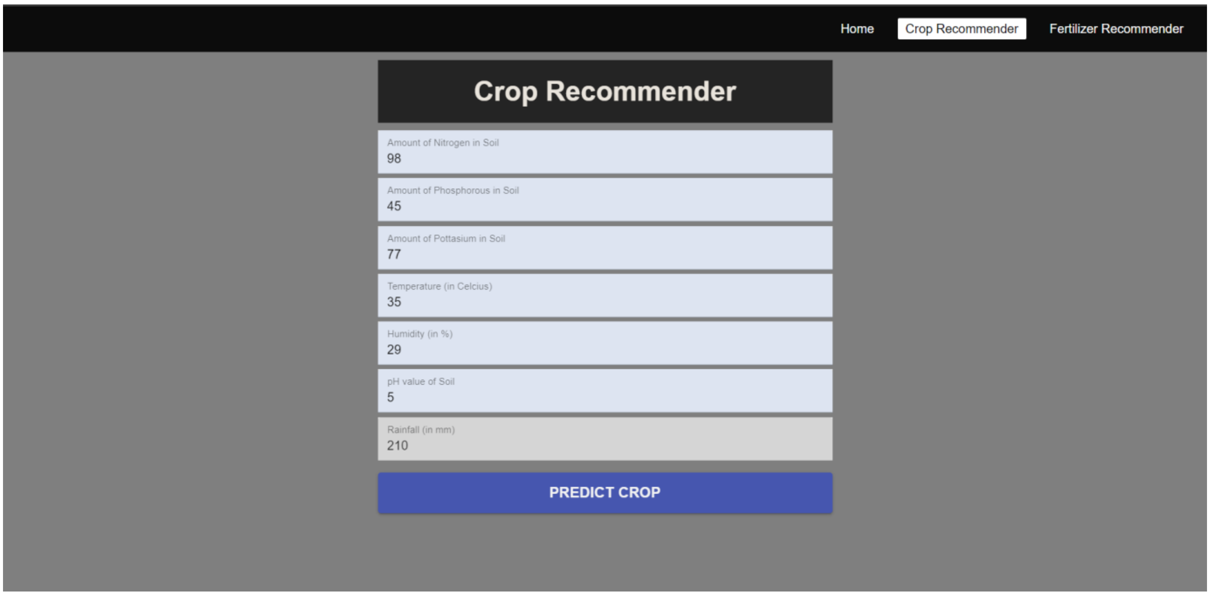


Figure 22. Inputting in Crop Recommender

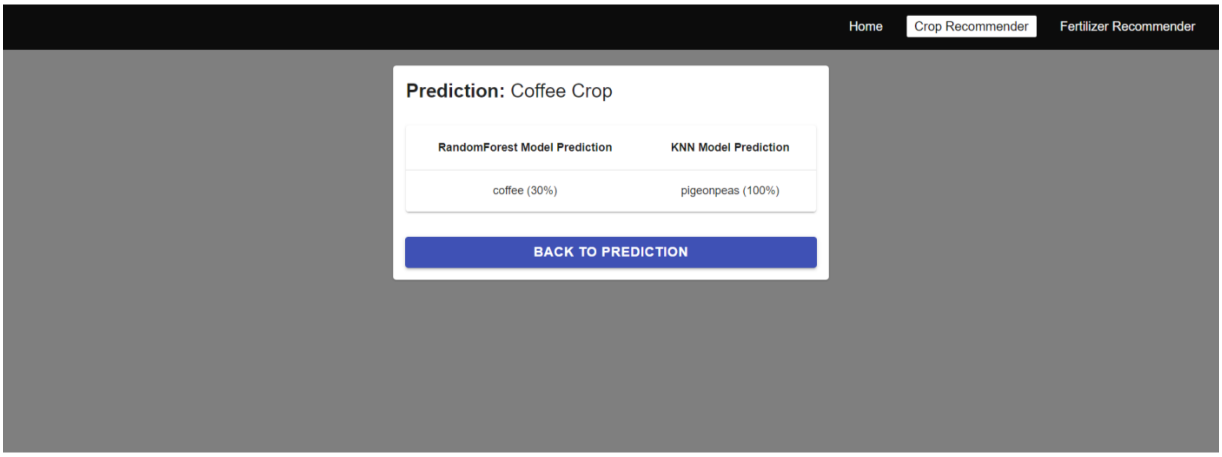


Figure 23. Crop Recommender Result

Here in the fig.23, it recommends coffee crop according to the input values. With the name of crop, it also displays the probability from both the classifier.

Using Fertilizer Recommender:

The User need to input the values of N, P, K, temp, etc. along with that it requires additional input i.e., Soil Type and Crop farmer wants to grow. It would recommend the fertilizer with percentage using both KNN and Random Forest. Taking soil type and crop it would recommend the fertilizer.

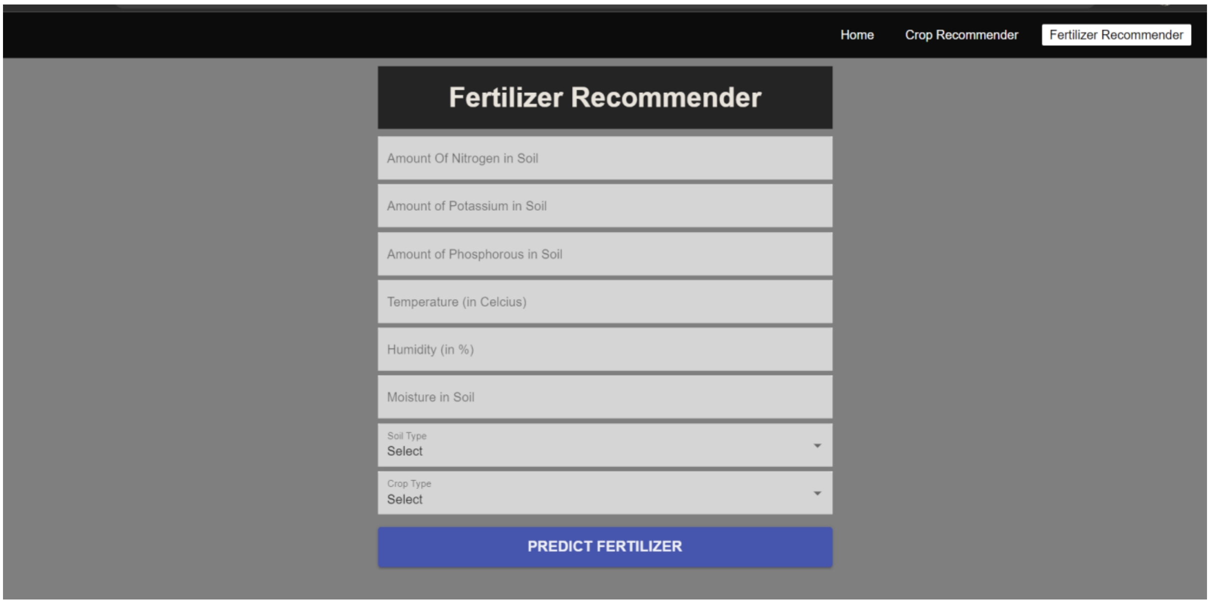


Figure 24. Fertilizer Recommender

Figure 25. Inputting in Fertilizer Recommender

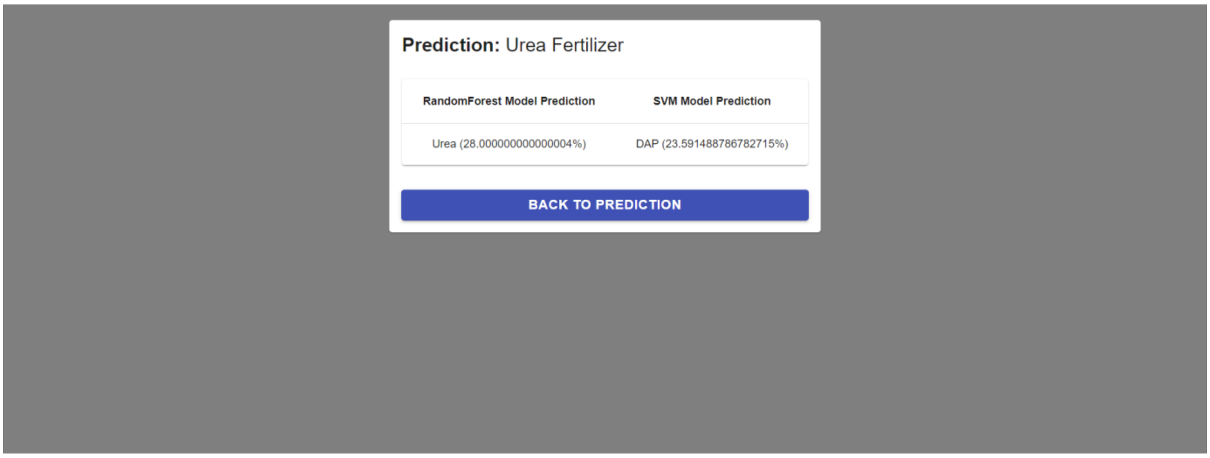
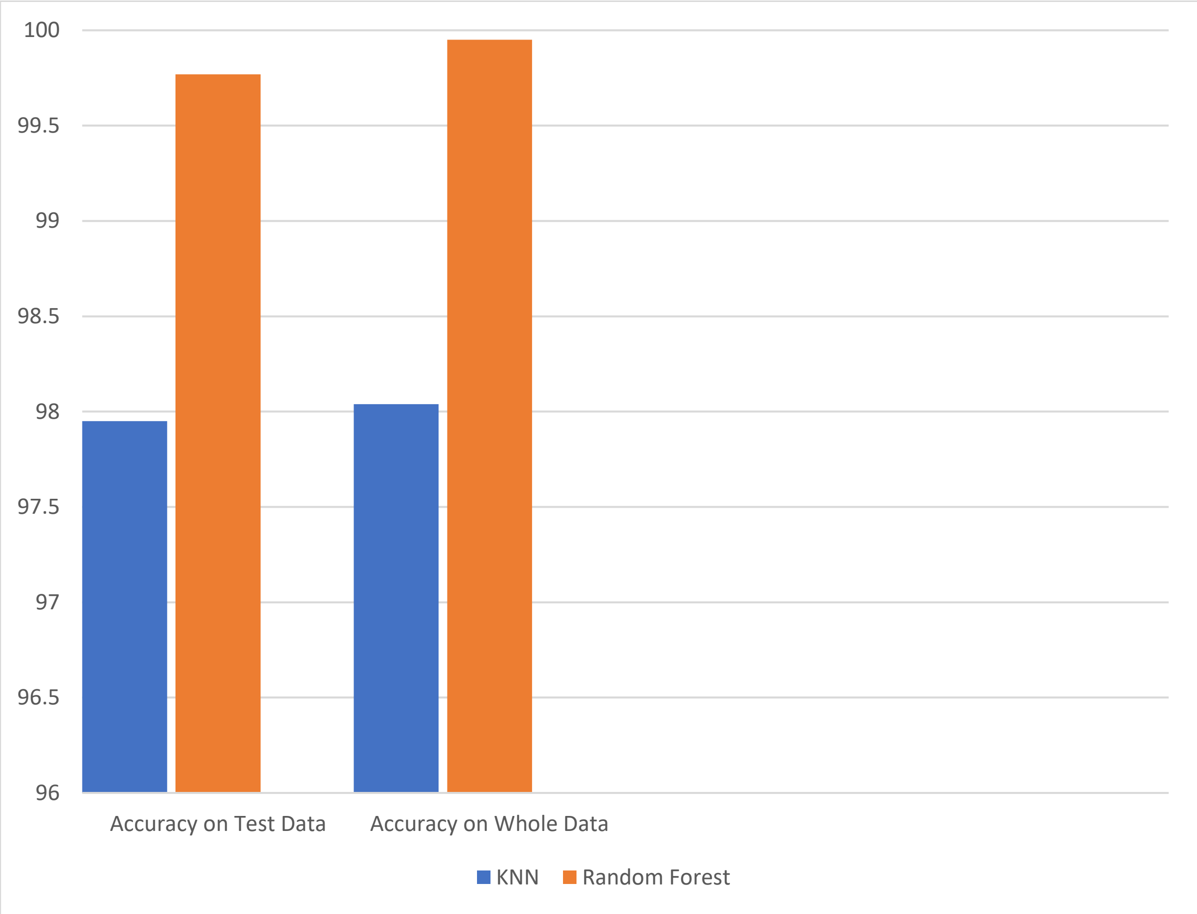


Figure 26. Fertilizer Recommender Result

Here in the fig.26, it recommends urea fertilizer according to the input values. With the name of fertilizer, it also displays the probability from both the classifier.

Chapter-4: PERFORMANCE ANALYSIS

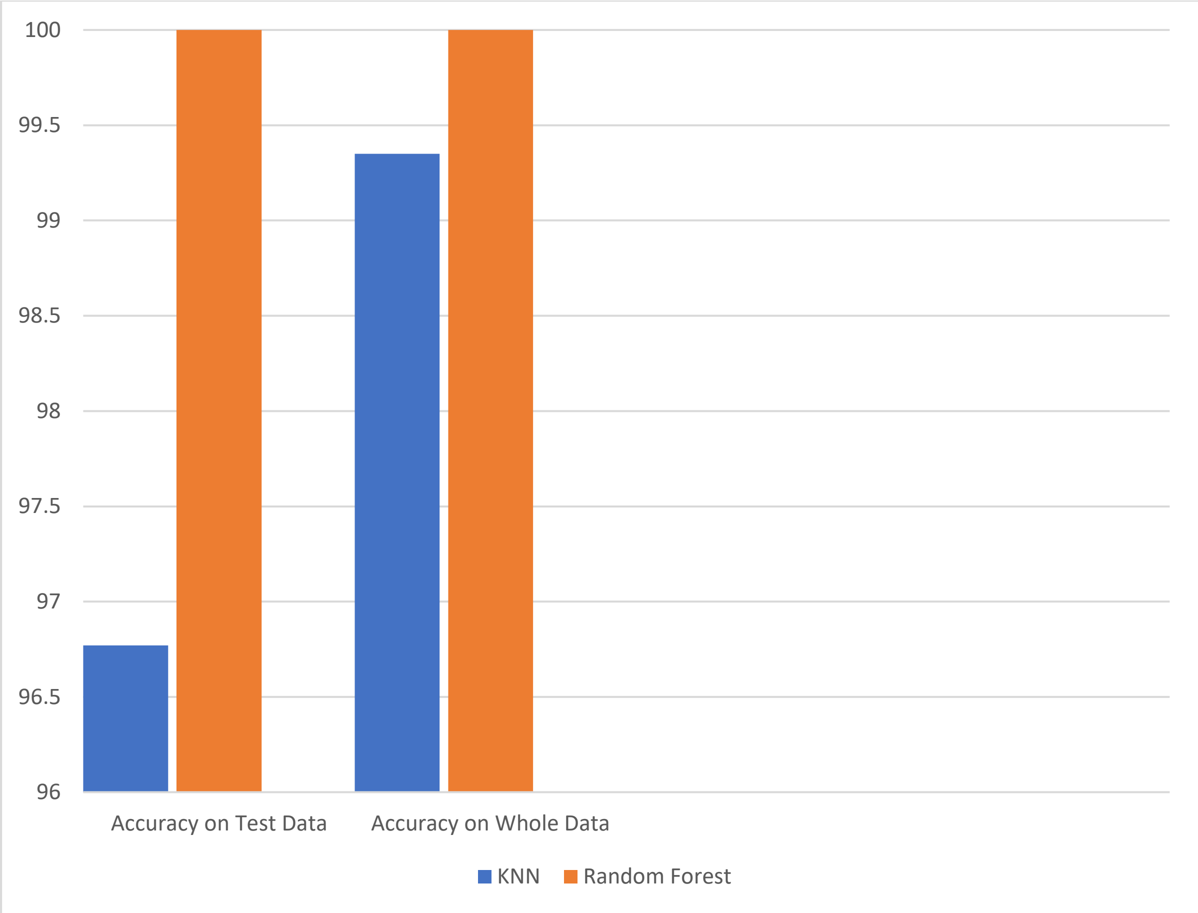
To predict crop, ML algorithms such as KNN and Random Forest is used. Both KNN and Random Forest classifier provides a good result. Accuracy was greater than 90% in both cases. Random Forest classifier proves to be better in terms of accuracy.



Graph 3. KNN vs Random Forest for Crop Recommendation

For test data as well as for the whole dataset, random forest proves to be much better algorithm for crop predication. Random Forest performed quite well and reaching accuracy of 99%.

Forfertilizer recommendation, again the same ML algorithms KNN and Random Forest is used.



Graph 4. KNN vs Random Forest for Fertilizer Recommendation

Chapter-5: CONCLUSIONS

5.1 Conclusion

Both modelswere successful in predicting crops and recommending fertilizer. The models achieved a training accuracy of approx. 98-99%. Both KNN and Random Forest performed quite well. But random forest performed a little more than KNN with 3-4 % more accuracy. The dataset was balanced and didn’t cause any problem however entries in the dataset were less. Although the accuracy is very high, it would be much better if there was a much better datasets available.

The UI of the web application is quite simple and easy to use. With the help of the sensors in the agriculture land there would be no need to enter the values manually. The data from the sensors will automatically feed to the system and it would recommend the crops and fertilizers.

It would definitely benefit the farmers in decision making. It would consider the rainfall and soil type. Farming in India mainly depends upon the rainfall season i.e., monsoon season. Bad monsoon causes a lot of problem to the farmers, so taking rainfall as a factor is a plus point.

5.2 Application of the Project

As we know the farmers face a lot of problems in India. There is a serious need to make some technological advancements in this field. With in introduction of the Machine Learning techniques in this field it would be very much beneficial to the farmers.

Sensors should be introduced in farming lands. The cost of sensors is quite low and would act as one time investment. The sensors would automatically take the data and feed it to the system that would help the decision making of the farmers.

Apart from sensors the UI of the system is quite simple. If farmer have done soil testing of his/her land they can manually enter the data in the system.

Advance systems like this could change the face of the farming sector in India. It would make it more cost effective and would help to produce a good yield.

5.3 Limitation of the Minor Project

Even though system is quite good enough to accurately recommend crop and fertilizer however it is still having some limitations.

There is no proper dataset for such models available in India. Most of datasets are either old or having less data entries. With much better datasets it would be very beneficial for training a much more accurate model for recommending crops.

Majority of farmers are illiterate and would not able to use the system. Most of them don’t posses a good knowledge to use the system. Apart from that farmer in India follow traditional method only for cultivating crops. So, using system like this make be a new thing for them.

5.4 Future Work

There is need to collect more data and make a better dataset with more data entries. This would help in training of the models.

The UI could be translated into the local languages so that every illiterate farmer that don’t know English would also be able to the system. It would be much more continent for the farmer to use such system in their native language.

There is need to test the system with hardware. Different sensors that would record the values and preset the results automatically.

References

[1] S. M. PANDE, P. K. RAMESH, A. ANMOL, B. R. AISHWARYA, K. ROHILLA and K. SHAURYA, "Crop Recommender System Using Machine Learning Approach," 2021 5th International Conference on Computing Methodologies and Communication (ICCMC), 2021, pp. 1066-1071, doi: 10.1109/ICCMC51019.2021.9418351.

https://ieeexplore.ieee.org/document/9418351

[2] Dasari, Anantha Reddy &Dadore, Bhagyashri&Watekar, Aarti. (2019). Crop Recommendation System to Maximize Crop Yield in Ramtek region using Machine Learning. International Journal of Scientific Research in Science and Technology. 485-489. 10.32628/IJSRST196172.

https://www.researchgate.net/publication/331426761_Crop_Recommendation_System_to_Maximize_Crop_Yield_in_Ramtek_region_using_Machine_Learning

[3] M. Keerthana, K. J. M. Meghana, S. Pravallika and M. Kavitha, "An Ensemble Algorithm for Crop Yield Prediction," 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV), 2021, pp. 963-970, doi: 10.1109/ICICV50876.2021.9388479.

https://ieeexplore.ieee.org/document/9388479

[4] S. Bhanumathi, M. Vineeth and N. Rohit, "Crop Yield Prediction and Efficient use of Fertilizers," 2019 International Conference on Communication and Signal Processing (ICCSP), 2019, pp. 0769-0773, doi: 10.1109/ICCSP.2019.8698087.

https://ieeexplore.ieee.org/abstract/document/8698087

[5] Crop Prediction Dataset
https://www.kaggle.com/atharvaingle/crop-recommendation-dataset

[6] Fertilizer Prediction Dataset
https://www.kaggle.com/gdabhishek/fertilizer-prediction

[7] https://www.ibef.org/industry/agriculture-india.aspx

[8] https://www.india.gov.in/topics/agriculture

[9]https://www.indiatoday.in/diu/story/seven-reasons-why-india-agriculture-sector-needs-a-fresh-churn-1750587-2020-12-17