Decision Analytics Journal 3 (2022) 100041

Contents lists available at ScienceDirect

Decision Analytics Journal

journal homepage: www.elsevier.com/locate/dajour

Smart farming using Machine Learning and Deep Learning techniques

Senthil Kumar Swami Durai [a,*], Mary Divya Shamili [b]

[a] School of Engineering, Presidency University, Bangalore, India
[b] Department of CSE, School of Engineering, Presidency University, Bangalore, India

ARTICLE INFO

Keywords:
Machine learning
Precise
Deep learning
Recommendation

ABSTRACT

The practice of cultivating the soil, producing crops, and keeping livestock is referred to as farming. Agriculture is critical to a country's economic development. Nearly 58 percent of a country's primary source of livelihood is farming. Farmers till date had adopted conventional farming techniques. These techniques were not precise thus reduced the productivity and consumed a lot of time. Precise farming helps to increase the productivity by precisely determining the steps that needs to be practiced at its due season. Predicting the weather conditions, analyzing the soil, recommending the crops for cultivation, determine the amount of fertilizers, pesticides that need to be used are some elements of precision farming. Precise Farming uses advanced technologies such as IOT, Data Mining, Data Analytics, Machine Learning to collect the data, train the systems and predict the results. With the help of technologies Precise farming helps to reduce manual labor and increase productivity. Farmers have been facing various challenges in these recent times, this includes crop failure due to less rainfall, infertility of soil and so on. Due to the changes taking place in the environment the proposed work helps to identify how to manage crops and harvest in a smart way. It guides an individual for smart farming. The aim of this work is to help an individual cultivate crops efficiently and hence achieve high productivity at low cost. It also helps to predict the total cost needed for cultivation. This would help an individual to pre-plan the activities before cultivation resulting in an integrated solution in farming.

1. Introduction

Agriculture, or farming as it is commonly known, is the practice of growing crops and raising cattle. It contributes greatly to a country's economy. Many raw materials and food products are produced by agriculture. Raw materials such as cotton, jute is used by industries for manufacturing various products that is used in day-to-day life. Agriculture not only helps for food production but also produces resources needed for creating commercial products. Agriculture used traditional techniques for cultivation of crops. Conventional or traditional farming is mostly practiced all over the world. It involves techniques suggested by experienced farmers. These techniques are not precise hence results in hard labor and time consumption.

The application of digital technologies which includes robots, electronic devices, sensor and automation technologies is associated with Precision Agriculture. This technology aims to reduce workloads, increase profitability and decision management. Precision agriculture additionally referred to as precision farming is a farming control system that provides a comprehensive approach to deal with the spatial and temporal crop and soil variability to maximize profitability, optimize yield, improve quality of production [1]. Precision Agriculture is an efficient way to improvise the yield. On discussing about the adoption rate of precision agriculture, the high value enterprise farms adoption rate was more compared to low value enterprise farms. The adoption rate of precision agriculture also depends on the country and the geological locations. The adoption rate of Precision Agriculture in mountain zone is less compared to farmers in the valley [2]. The variation of adoption rate is due to the high investments needed. Hence there needs a way to reduce cost on machines hence all farm-size can adopt precision agriculture.

Precision agriculture is aided by advanced technologies such as IoT, Data Mining, Artificial Intelligence, and Data Science. The Internet of Things (IOT) is a network of interconnected computational things like sensors and smart gadgets that can communicate with one another and share data [3]. In agronomic applications, wireless sensor networks are being used to remotely monitor ambient and soil characteristics in order to predict crop health. Using WSN as a forecasting approach, the watering schedule of agricultural fields can be predicted. Wireless Sensor Networks acquire data from external variables such as pressure, humidity, and temperature, as well as soil moisture, salinity, and conductivity [4].

Machine learning makes agricultural applications incredibly efficient and simple. Data acquisition, model building, and generalization are the three stages of the machine learning process. The majority of cases, machine learning algorithms are used to deal with complex

* Corresponding author.
E-mail addresses: harisen1234@yahoo.co.in (S.K.S. Durai), mary.20202dsc0001@presidencyuniversity.in (M.D. Shamili).

https://doi.org/10.1016/j.dajour.2022.100041
Received 21 December 2021; Received in revised form 17 March 2022; Accepted 29 March 2022
Available online 2 April 2022

problems when human competence is insufficient. Machine learning may be used in agriculture to forecast soil parameters like organic carbon and moisture content, as well as crop yield prediction, disease and weed identification in crops, and species detection [5]. Traditional machine learning is improved by Deep Learning by adding additional complexity to the model and changing the input with various functions that allow data representation in a hierarchical manner, through multiple levels of abstraction, depending on the network architecture employed. A significant benefit of Deep Learning is feature learning, or indeed the automatic extraction of features from original data. The ability to identify unknown things such as anomalies rather than just a collection of existing items is a key aspect of the Deep learning model, which uses the homogeneous properties of an agricultural field to discover faraway, badly obstructed, and unknown objects [6].

Blockchain has swiftly become a key technology in a variety of precision agriculture applications. The requirement for smart peer-to-peer systems capable of verifying, securing, monitoring, and analyzing agricultural data has prompted researchers to consider developing blockchain-based IoT systems in precision agriculture. Blockchain plays a critical role in transforming traditional methods of storing, sorting, and distributing agricultural data into a digital format a way that is more dependable, immutable, transparent, and decentralized. The combination of the Internet of Things and blockchain in precision farming results in a network of smart farms. More autonomy and flexibility are attained as a result of this pairing [7].

The above-mentioned technologies such as IOT, Data Science, Machine Learning, Deep Learning, Blockchain deals mostly with data which are very useful for understanding and providing great insights of data. Hence, these advanced technologies are used in various agriculture practices such as identifying the best crop for a particular location, identifying factors that would destroy the crops such as weeds, insects and crop diseases to obtain insights about the crop growth and help in decision making.

Agriculture can be divided into 7 important steps that includes Land Management, Soil Preparation, Water Monitoring, Identifying the weeds, Pesticides Recommendation, identifying diseased crops, and cost estimation. Land Management refers to the monitoring physical features that includes weather conditions, geological characteristics. This is important since there are variations in climatic conditions across the globe which would affect the crops. Rainfall is an important aspect of the earth's climate, and its unpredictability has a direct impact on agriculture, water management systems, and biological systems [8]. As a result, tools that assist in predicting rainfall in advance are required so that crop management can be simplified.

Soil is an essential component of agriculture. Rooting, moisture and nutrient storage, mineral reserve, anchoring, and a variety of other variables that affect plant growth are all determined by soil depth [9]. The initial step for Soil preparation is testing the soil. It involves identifying the soil's current nutrient levels and the suitable amount of nutrients to be feed to a certain soil based on its fertility and crop demands. The values from the soil test report are being used to categorize a number of key soil parameters, notably Phosphorus, Potassium, Nitrogen, Organic Carbon, Boron, as and soil ph [10]. Irrigation is a type of agriculture that plays an important role in water and soil conservation. Complicated data could be used to maintain irrigation performance and consistency when assessing systems with respect to water, soil, climate, and crop facts [11].

Weeds are plants that is grown where it is not needed. It includes plants that are not intentionally sown. Weeds compete for water, nutrients, light, and space with agricultural plants, lowering crop yields. Weeds can diminish the commercial worth of agricultural regions by lowering the quality of farm products, causing irrigation water loss, and making harvesting machinery harder to run. To control weeds, farmers often spray homogeneous herbicide spraying throughout the field twice or three times during the growth season. However, this method has resulted in the uncontrolled use of large volumes of herbicides, which is harmful to humans, non-target animals, and the environment [12].

Plant diseases can have a devastating influence on food safety, as well as a considerable loss in both the quality and quantity of agricultural goods. Plant diseases can potentially prevent grain harvesting entirely in severe circumstances. As a result, in the field of agricultural information, computerized identification and diagnosis of plant diseases is widely needed. Many approaches for doing this problem have been offered, with deep learning emerging as the preferred method because to its excellent performance [13].

Hence this work focuses on the steps involved in cultivation of crop. It uses Deep Learning and Machine Learning algorithms to deliver solutions to various challenges faced during cultivation. It mainly focuses on recommending the crops based on weather parameters, suggesting the nutrients requirements and specifying the Growing Degree Days. It also helps in identifying the weeds and recommending herbicides for the same. Many insects ruin the crops hence pesticides are recommended based on the insects that are present in the field. And finally cost estimation is very much needed in these recent times. Crisis, uncertainties would result in great loss. Hence forecasting the cost for cultivating a crop is necessary to plan for future uncertain events. This work specifies various costs in cultivation for future years.

2. Literature review

Crop growth is primarily influenced by the soil's macronutrient and trace mineral content of the soil. Soil being the broad representation of several environmental factors including rainfall, humidity, sunlight, temperature and soil ph. The use of a support vector machine and decision tree algorithm to distinguish the type of crop based on micronutrients and meteorological characteristics has been presented as an efficient means of predicting the crop. Three crops where selected such as rice, wheat and sugarcane. Based on certain observations details about micronutrients where been obtained. These details where feed into the classifier model that in turn predicted the crop based on the passed values. There are many Machine Learning algorithms that works in a different manner. Hence selecting only two models will not provide the required output. The accuracy score of SVM was greater than decision tree algorithm with a sore of 92% [14]. In this work best out of two algorithms is selected. But there are various algorithms dedicated for classification tasks. There is a need for working on other models such as K Neighbors classifier, Logistic Regression, Ensemble classifiers. These algorithms are indeed applied in proposed research work. The [14] predicts only a crop based on the values entered into the SVM model. Data is most valuable. Hence more information can be obtained apart from using them for prediction. The proposed research work not only recommends the crops and also uses the data to obtain various information that would provide a detailed view about the predicted crops this includes specifying the Growing Degree Days such as heat units, amount of heat needed for the crop growth and the amount of nitrogen, phosphorous and potassium content need to be supplied for the growth per 200 lb. fertilizer.

Machine Learning algorithms such as SVM and decision tree classifier was used [14] but in this work Machine Learning algorithms such as Decision Tree, K Nearest Neighbor, Linear Regression model, Neural Network, Naïve Bayes and Support Vector Machine was used for recommending a crop to the user. It has provided an exposure to other algorithms compared to [14]. Linear Regression model was used to predict the production value against the climatic parameters such as rainfall, temperature and humidity. The scores of all these algorithms were below 90% [15]. This work was just a model implementation using the dataset. Web interface needs to be implemented so that even common people can use it efficiently. All the values need to be provided manually for the model to predict the crop. The proposed work helps in extracting temperature and humidity values using Web Scraping. Hence manually entering the values are not needed. The proposed work provides an interactive web interface where the user specifies the average rainfall and soil Ph value. The temperature and

humidity details are extracted automatically and feed into the best model that includes 10 algorithms with hyper parameter tuning. The proposed work tends to achieve an accuracy of 95.45% with hyper parameter tuning the algorithms which was not included in [14]. The predicted results along with certain information are displayed in the web interface which makes the user to understand the results more efficiently.

Base temperature of a given crop can be used to calculate the GDD Growing Degree Days. The main aim of this study is to come up with easy and mathematically acceptable formulas for calculating GDD's base temperature. Temperature data for snap beans, sweet corn, and cowpea are used to propose, prove, and test mathematical formulas. These new mathematical formulae, in comparison to earlier approaches, can produce the base temperature quickly and correctly. These formulas can be used to calculate the GDD base temperature for every crop at any developmental stage [16]. This work provides a formula to calculate the GDD for the crops. Hence the formula specified in [16] was applied to the predicted crop to estimate their GDD in the proposed work.

Weeds grown along with soybean can be detected using K-means and CNN model. K-means were used for identifying the features of the images and convolutional neural network for was used for classifying the weeds and soybean. It also suggests that accuracy can we improved by fine tuning the CNN model. CNN model provides an efficient way to detect the weeds present among crops. When used along with K-means initially the images and its augmentations are clustered and on using CNN model helps to precisely identify the weed [17]. The proposed work uses the pretrained model such as Resnet152V2 hence it has important layers such as skip layer and identity layer. The main goal of these layer is to make sure that the output image is same as the input. This increases the accuracy and the predictions are correct. Not only predicting the image the proposed model also helps to provide details about the herbicides that can be used which is an additional information for the user.

Existing deep learning techniques are used for weed detection. This study provides information of various ML and Deep Learning algorithms that can be used for identifying weeds. It mainly emphasis on pre-trained models. It suggests that pre-trained models as lot of benefits and hence can be used to image classification. It also provides guidance of how to work on datasets and make the datasets efficient for building the models. Many public datasets are available on various platforms that can be used for this purpose. It specifies Image Resizing, data augmentation, image segmentation some of the techniques would bring about accurate classifications and tendency of increasing the accuracy is also more in pre-trained models [18]. Since this study provides directions to perform deep learning techniques the proposed model has opted certain techniques preprocessing steps such as Image Resizing, data augmentation is opted before building the actual deep learning model to predict the weeds.

Another algorithm that can be used for identifying weeds in vegetable plantation is the CenterNet. CenterNet is used for weed identification. It includes two stages. In first stage the Bok choy images were collected and detected. In the second stage, color-index based segmentation were performed on the images collected to identify the weeds present in the dataset. The images were collected from Nanjing, China. The images were augmented to increase the dataset size and images were annotated. CenterNet algorithm was used for both training and testing the images. It is a ground-based weed identification technique. More optimization would lead to better results was suggested [19]. CenterNet algorithm is simple yet there is a need an algorithm that strives to get correct prediction. The proposed work uses Resnet152V2 algorithm that strives to achieve more accuracy since it has special layers such as skip layer and identity layer that tries to get input image as output itself. Hence predictions would be absolutely correct. Hence Resnet152V2 algorithm is selected to obtain accurate prediction and based on the prediction obtain the list of herbicides.

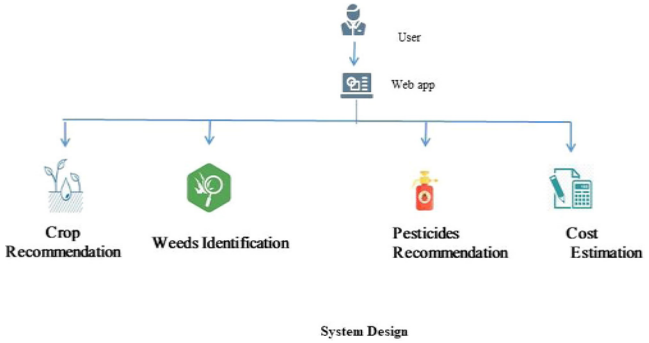

Fig. 3.1. System architecture.

Farmers face a challenging task in identifying crop insects since pest infestation destroys a substantial portion of the crop and affects its quality. The use of highly skilled taxonomists to correctly identify insects based on their physical traits is a shortcoming of traditional insect identification. Experiments were conducted using image characteristics and ml algorithms such as neural networks, support vector machine, k-nearest neighbors, naive bayes, and convolutional neural network model to identify twenty-four insects from the Wang and Xie dataset. To increase the performance of the classification models, 9-fold cross-validation was used. The CNN model had the greatest classification rates of 91.5 percent and 90 percent, respectively. The results revealed a considerable improvement in classification accuracy and computational time when compared to state-of-the-art classification algorithms [20]. This work [20] has used basic CNN model for classification as well as the same dataset used by various researchers. Hence the proposed model has used a different dataset called the Pests' dataset from Kaggle website. This dataset consists of 9 classes of insects. Each image is taken from different locations. This dataset was selected for the proposed model since the model is trained of images about various locations that gives more knowledge for the model to understand the image and distinguish them. The proposed model uses Resnet152V2 model for classification. The Resnet152V2 model is the basic model and top of which Global Average Pooling 2D, Dropouts and more hidden layers are been implemented. This refers to fine tuning the base pre-trained model. This helps in extracting more information and helps in efficient classification.

The association between the degree of difficulty in identifying insects and the identification key was investigated in this article. For a collection of 134 insects, the SPIPOLL database was utilized to generate 193 characteristic value pathways. Based on the average IES of all the insects with that of characteristic value was formulated. The CV's derived IES was then used to generate an estimated IES for each bug, resulting in a ranked list of insects. Finally, the anticipated bug ranking list was compared to the actual bug ranking list. The results showed a significant correlation between the estimated and actual truth IES, indicating that the CV can be used to estimate the IES of SPIPOLL insects [21]. This work has specified of how to consider the features of an image with respect to insects' dataset. Its main goal is to identify a key that helps in distinguishing the classes. This proposed work contributes in specifying that a key is important for distinguishing the insect classes. Hence the proposed work uses Resnet152V2 algorithm for this very reason. Resnet152V2 is a pre-trained model and it automatically picks the important features rather than manually defining them. The Resnet152V2 base model on addition with Dropouts helps in removing unnecessary hidden layers and selecting the relevant ones is an advantage. Identification of insects does not solve the problem completely. Suggesting Pesticides provides a complete solution. The proposed model helps to identify the insects as well as suggest Pesticides for the same.

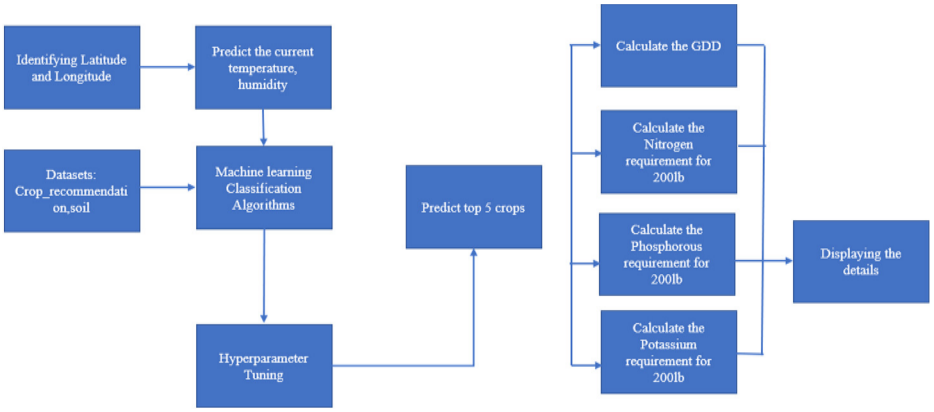

Fig. 3.1.1. Crop recommendation system architecture.

Attributes	Crop_Recommendation.csv	Soil.csv	Crop_names.csv
Source	https://www.kaggle.com/atharvaingle/crop-recommendation-dataset	https://www.kaggle.com/shekharyada/crop-soilcsv	https://www.kaggle.com/aj021977/crop-names
No.of samples	2200	43	35
Attributes	8	2	2
Used for	Classification	Classification	Classification
Labels Count	22	7	35

Fig. 3.1.2. Dataset details for crop recommendation module.

Various elements must be considered when estimating the cost of a crop. It divides agricultural costs into five categories and provides calculations for each. It also gives examples of how to figure out how much a crop cost. It is a theoretical article that always guide the implementation of estimating the cost of cultivation [22]. This theoretical study was used in the proposed model to calculate the cost of cultivation. It was very helpful as it provided elementary description to calculate the costs for cultivation. The formulas proposed in this study was used in the proposed system to estimate the costs till the year 2028.

From 2004 to 2015, the goal of this research is to evaluate the gap between various costs and gross value of output (GVO), as well as the trends of input utilization and critical factors for gross value of output of gram crop across top production states. The findings demonstrate so after 2009–2010, all states' GVO and overall costs increased significantly. The commencement of the Government of India's agricultural waiver scheme in 2008–2009 was found to be the cause of a large increase in operational costs from 2009 to 2010. It was also obvious that the compound annual growth rate was larger in 2009–2010 than in 2014–2015 when comparing 2004–2005 to 2007–2008. Profit margins were high in Madhya Pradesh and Rajasthan, indicating a cost-cutting trend [23]. This work provided a comparative study about the costs between different states of India. The proposed work has used Ensemble regression algorithms that is used to forecast the costs till 2028. It provides a comparative study of a crop for a specific state from 2010–2028. Hence the user would be able to identify the trends of costs from the year 2010–2028. The forecasting is explicitly applied on the India's cost of cultivation survey data from 2010–2018. This provides an elaborative view of operational cost, fixed cost, total cost, Cost Concepts were displayed in form of a graph for better understanding of the trends of the costs.

3. Methodology

There are four modules proposed in this work Fig. 3.1 such as crop recommendation, weed identification, Pesticide recommendation, crop cost estimation. The proposed work is a Web application developed through Django framework. The Web Interface starts with the User login page. In order to access these modules, users need to initially register with their basic details such as their name, Address, Country, State, Pin code, Phone number, Username and Password. Once the account is created, they are redirected to the login page where the user needs to login using their credentials. The following sections describes the modules in detail.

3.1. Module 1: Crop recommendation

Datasets Used: For Crop Recommendation module the dataset used are Crop recommender.csv, soil.csv, scientific_names.csv. All these datasets were obtained from Kaggle website. Fig. 3.1.2 gives the summary of the datasets used in this module (see Fig. 3.1.1).

The Crop recommendation was used for training model since it contains attributes such as temperature, humidity, average rainfall, soil Ph, nitrogen requirement ratio, potassium requirement ratio and phosphorous requirement ratio essential for predicting a crop. The datasets such as Soil names and Crop names are used after prediction to obtain the soil type and scientific name of the predicted crops.

Steps involved in Crop Recommendation module are as follows

Step 1: Importing Libraries and Dataset

In order to utilize Machine Learning algorithms and preprocessing tools specific libraries needs to be imported. Using these libraries, the model building and prediction would be performed efficiently. The libraries such as NumPy, pandas, pickle, matplotlib, seaborn, Label

	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
...	...	...	...	...	...	...	...	...
2195	107	34	32	26.774637	66.413269	6.780064	177.774507	coffee
2196	99	15	27	27.417112	56.636362	6.086922	127.924610	coffee
2197	118	33	30	24.131797	67.225123	6.362608	173.322839	coffee
2198	117	32	34	26.272418	52.127394	6.758793	127.175293	coffee
2199	104	18	30	23.603016	60.396475	6.779833	140.937041	coffee

Fig. 3.1.3. Crop recommendation dataset sample values.

```
missing values
N              0
P              0
K              0
temperature    0
humidity       0
ph             0
rainfall       0
label          0
dtype: int64
```

Fig. 3.1.4. Missing value details.

```
dtypes
N                int64
P                int64
K                int64
temperature    float64
humidity       float64
ph             float64
rainfall       float64
label           object
dtype: object
```

Fig. 3.1.5. Datatypes of each column.

Encoder, train_test_split were imported. The models such as Naïve bayes, Logistic Regression, SVM, Decision Tree Classifier, Bagging Classifier, Random Forest Classifier, AdaBoost Classifier, Gradient Boosting Classifier, XGBoost Classifier, LGBM Classifier and KNN was imported. The dataset called crop recommendation was used initially for training and testing the models. Fig. 3.1.3 shows the glimpse of the crop recommendation dataset. Each crop has a set of values for temperature, humidity, rainfall, Nitrogen, potassium, phosphorous.

Step 2: Descriptive Analysis

To obtain a best predictive model descriptive analytics is to be performed in prior. Descriptive analytics provides an idea of how the dataset looks like and helps to draw new insights. Once the dataset is imported, missing values per attribute is checked. For crop recommendation dataset the attributes are free of missing values, results are displayed in Fig. 3.1.4 Once identifying that there are no missing values, in Fig. 3.1.5 the datatype of the attributes are identified followed by listing the unique values in the dependent variable, i.e., Label attribute Fig. 3.1.6.

Step 3: Data Visualization

Once the basic details about the dataset is obtained data visualization is performed to analyze the dataset in a visual format. A correlation matrix is a relationship lattice is basically a table appearance that specifies the correlation coefficients between attributes. Here, the attributes are addressed in the first line, and in the first column. Fig. 3.1.7 shows the correlation matrix for the dataset.

Fig. 3.1.8 displays the distribution of crops instances in the dataset. The fig signifies that the dataset has all crops equally distributed in the dataset. Each attribute in the dataset is plotted against the dependent variable such as Label column. Taking into consideration Nitrogen requirement per each crop the dataset specifies that cotton requires more nitrogen for its growth compared to all other crops. These details are obtained from Fig. 3.1.9. In the same way Fig. 3.1.10 specifies the Potassium requirement for each crop in which grapes and apple being the highest and orange consumption being the least followed by Fig. 3.1.11 specifying the Phosphorous requirement for each crop in which grapes and apple being the highest.

With respect to Temperature requirement for each crop papaya requires more temperature (Fig. 3.1.12), rice needs more rainfall compared to all other crops (Fig. 3.1.13) and coconut requires more humidity compared to other crops is demonstrated in Fig. 3.1.14. When analyzing the soil Ph requirement per crop almost all crops require more Ph value which is demonstrated in Fig. 3.1.15.

All the attributes except Label attribute are numerical in nature. Distribution plots is primarily used for univariant sets of data and depicts information using a histogram. Hence, distribution plots were plotted to identify the distribution of data throughout the dataset. In Fig. 3.1.16 the values of Phosphorous attribute as not equally distributed. There are ups and downs in the distribution of data which is specified by the blue line in the graph. With respect to Fig. 3.1.17 the potassium column has more zero values. Values lies between 20–45 and 75–120 in Fig. 3.1.18 that describes the distribution of data for Nitrogen.

Fig. 3.1.19 demonstrates distribution of data for Temperature Column most of the values lies between 20–35. It also shows normal distribution of data. In Fig. 3.1.20 the values are not much distributed

```
unique crops
['rice' 'maize' 'chickpea' 'kidneybeans' 'pigeonpeas' 'mothbeans'
 'mungbean' 'blackgram' 'lentil' 'pomegranate' 'banana' 'mango' 'grapes'
 'watermelon' 'muskmelon' 'apple' 'orange' 'papaya' 'coconut' 'cotton'
 'jute' 'coffee']
```

Fig. 3.1.6. Crops present in the dataset.

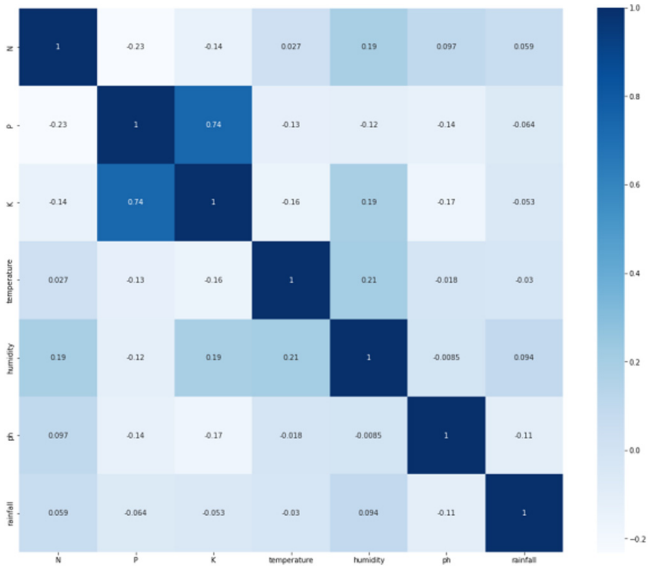

Fig. 3.1.7. Correlation matrix.

```
rice           100
maize          100
jute           100
cotton         100
coconut        100
papaya         100
orange         100
apple          100
muskmelon      100
watermelon     100
grapes         100
mango          100
banana         100
pomegranate    100
lentil         100
blackgram      100
mungbean       100
mothbeans      100
pigeonpeas     100
kidneybeans    100
chickpea       100
coffee         100
Name: label, dtype: int64
```

Fig. 3.1.8. Count of each Crop in the dataset.

in rainfall. With respect to Fig. 3.1.21 the values are distributed more within the range 5.5 to 7.5 for soil Ph column. All the distribution plots depicts that the data for each column is not normally distributed and its value counts vary between ranges.

Step 4 : Outlier Detection and Outlier Treatment

An outlier is a data point that is discovered to be considerably different from the other values for a given set. These outliers may corrupt the dataset and may provide wrong predictions. Hence before passing the data into the model outliers must be detected and tested. In order to detect the outliers, the proposed model has used box plots and IQR Technique. Box plots are used to visualize the outliers in an attribute. Fig. 3.1.22 depicts the box plot for Soil PH attribute. The figure shows that the values above 8.5 and below 4.5 are been categorized as outliers. Inter-Quantile Range Technique (IQR) is used to detect the outliers using a quantile range which specifies the percentage of data that is outside the quantile range between 0.75–0.25. In some cases, outliers can be removed or may be included because it is important in the business perspective. In crop recommendation dataset the outliers detected are useful and must be included because the observations or tuples in the dataset is obtained using experiments and these observations corresponds to a particular crop growth detail. As a result, this work has included all the observations present in the dataset since it is very important for prediction.

Step 5: Label Encoding

The dependent attribute or variable 'Label' in the dataset is Label encoded. This attribute has categorical and non-numerical values. The 'Label' attribute contains names of the crops. Since it is non-numerical the values need to be encoded into numerical values as many Classification models does not encourage the use of non-numerical values. Hence, these values are encoded into numerical values and then fed into the model for future predictions.

Step 6: Splitting the data into Train and test sets

After detecting the outliers and visualizing the attributes, the first step for building the model is splitting the dataset into training and testing set. The initial data required for training machine learning algorithms is referred to as training data. Machine learning algorithms are fed training datasets to learn how to make accurate predictions or accomplish a desired activity. Testing set defines a set of data used to provide an accurate evaluation of a finalized model fit on the training sample. The training and testing set split ratio is 50:50.

Step 7: Model Building

This work aims to identify the crops using Machine Learning Classification algorithms. The proposed work uses 10 classification algorithms to find the best model for future prediction. The steps performed for model building are as follows:

- The model is being imported from the library.
- Model is being defined.
- The training and testing data are fitted into the model.
- After training the model, the model is being tested over the testing dataset.
- Confusion Matrix and evaluation metrics are calculated.

Fig. 3.1.23 provides results about Random Forest Classifier. The predicted values of the testing set are displayed in form of a list pointed by y_predictions are then followed by training set accuracy score, test set accuracy score and the accuracy score of that model for the crop recommendation dataset. Classification report is depicting the

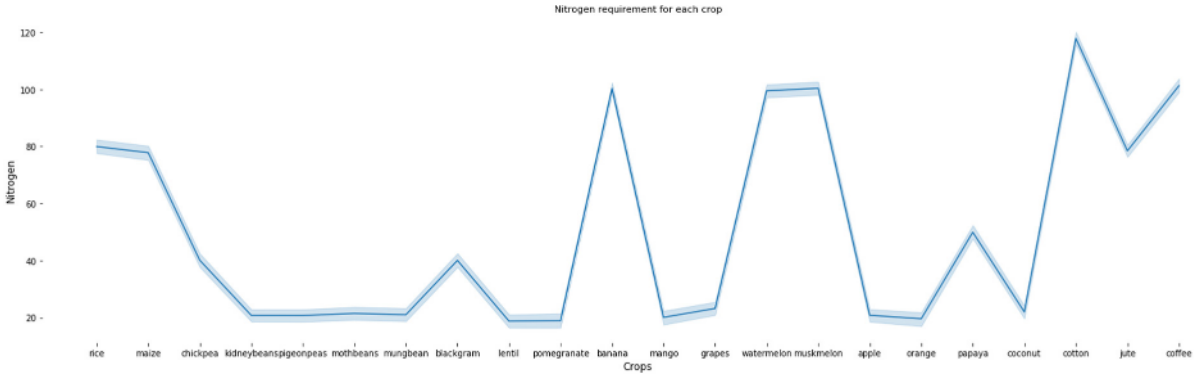

Fig. 3.1.9. Nitrogen requirement for each crop.

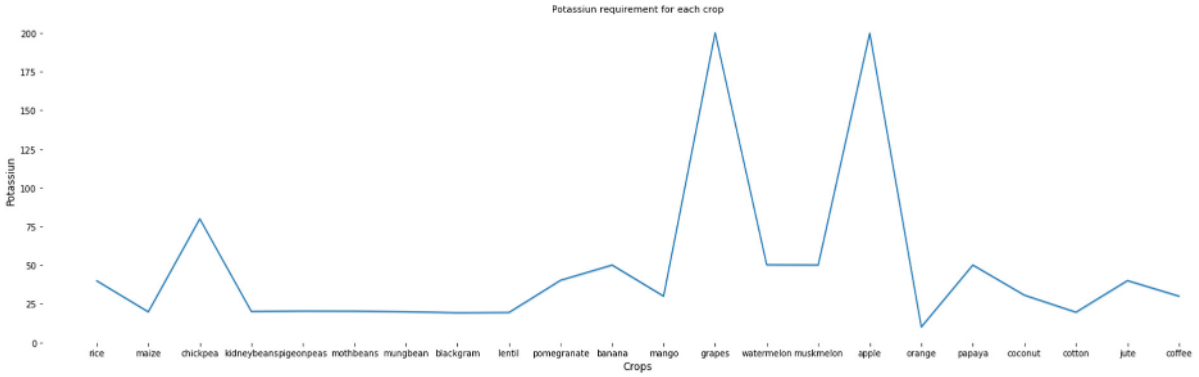

Fig. 3.1.10. Potassium requirement for each crop.

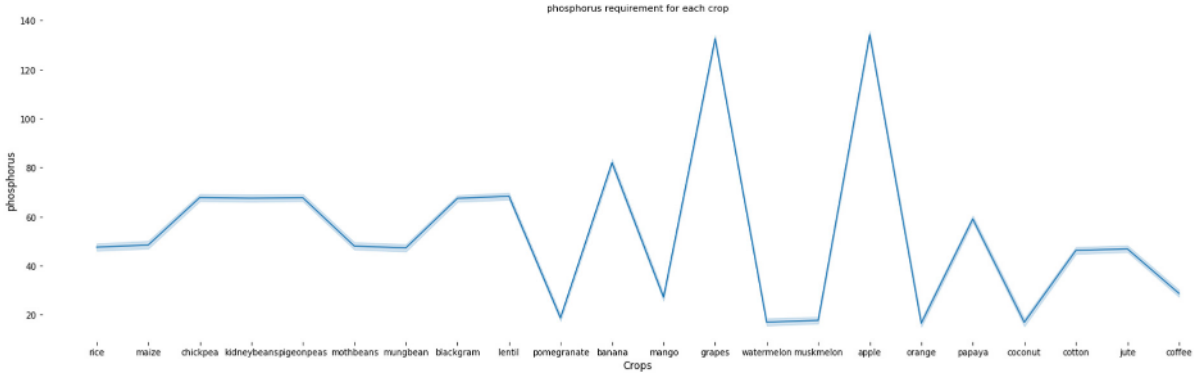

Fig. 3.1.11. Phosphorous requirement for each crop.

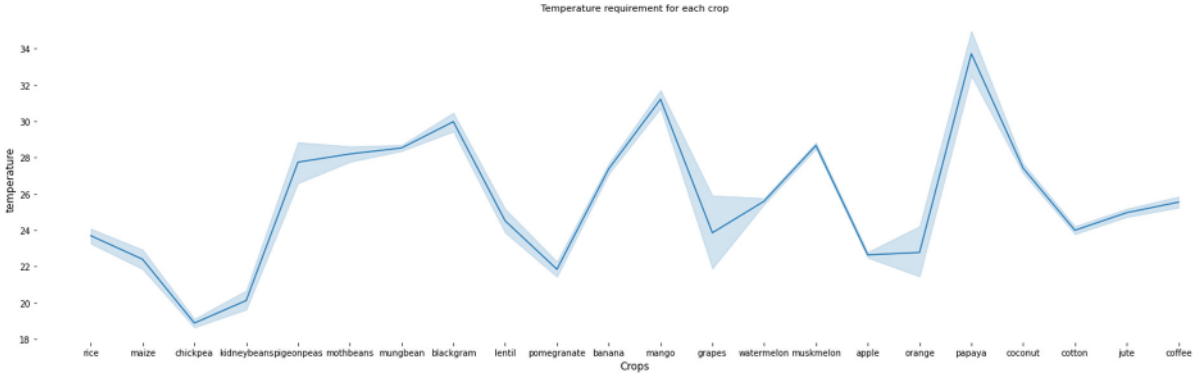

Fig. 3.1.12. Temperature requirement for each crop.

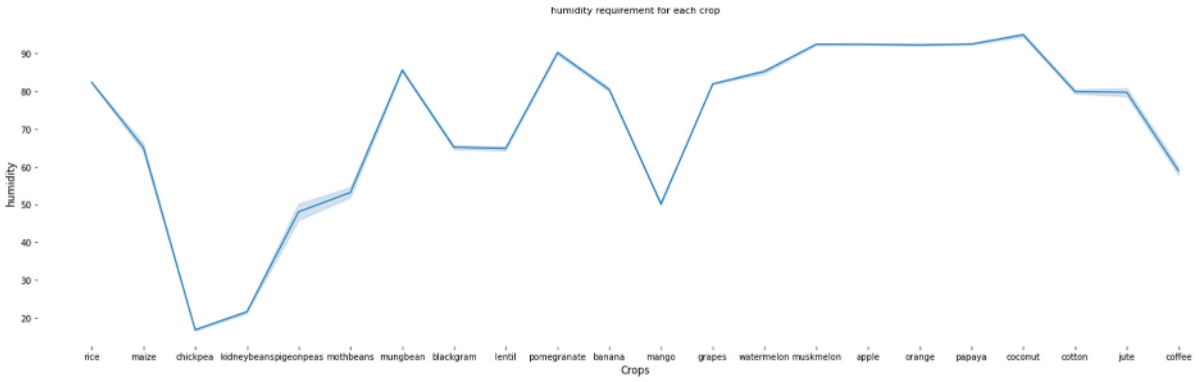

Fig. 3.1.13. Humidity requirement of each crop.

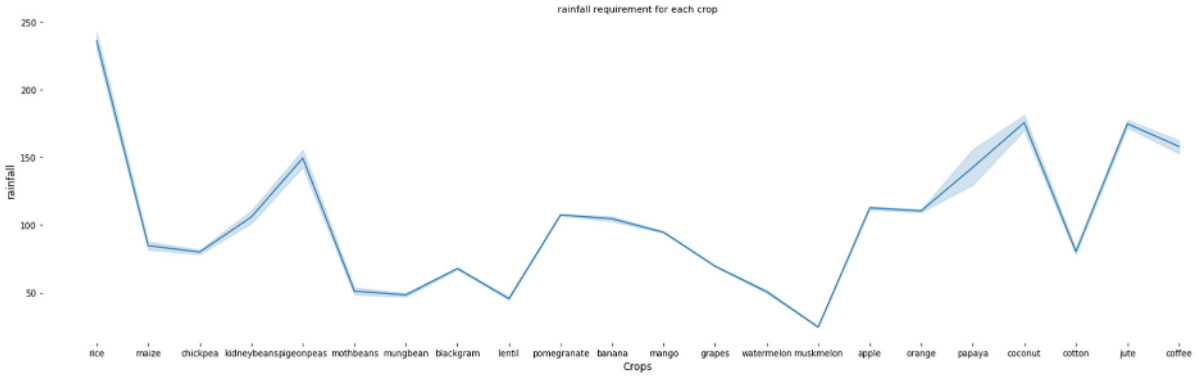

Fig. 3.1.14. Rainfall requirement of each crop.

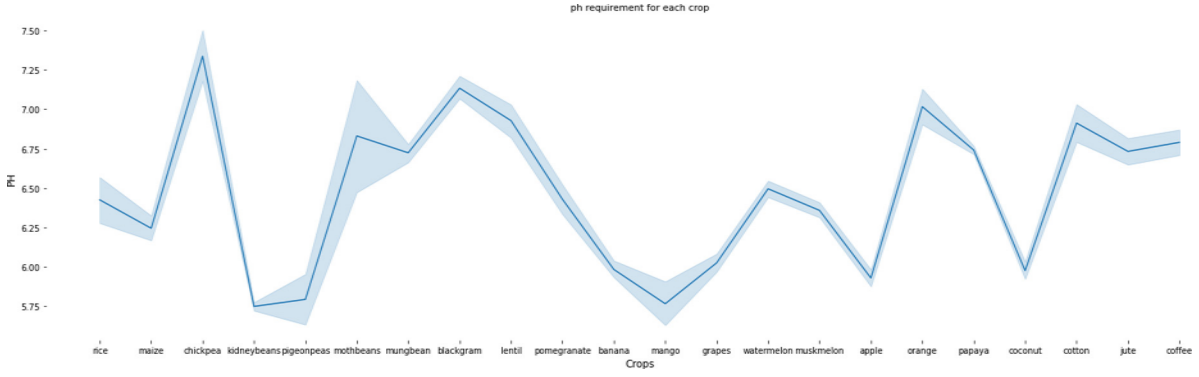

Fig. 3.1.15. Soil Ph requirement of each crop.

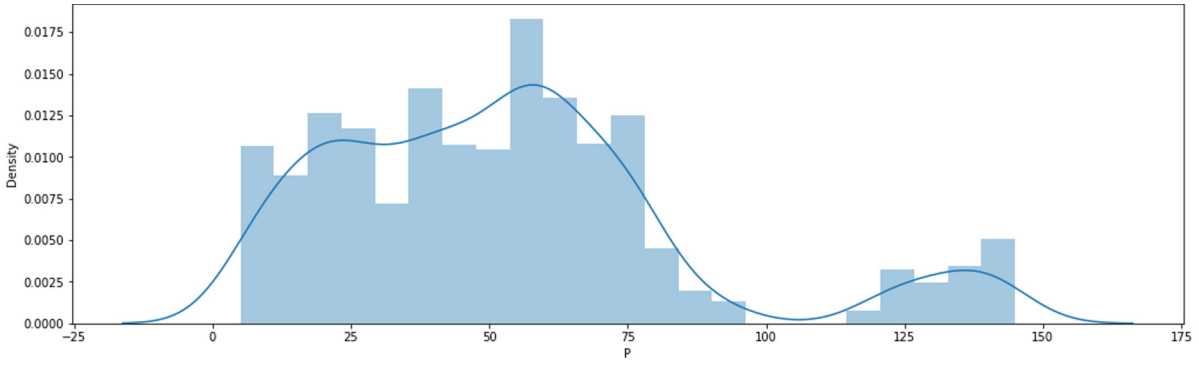

Fig. 3.1.16. Distribution of data for Phosphorous Column.

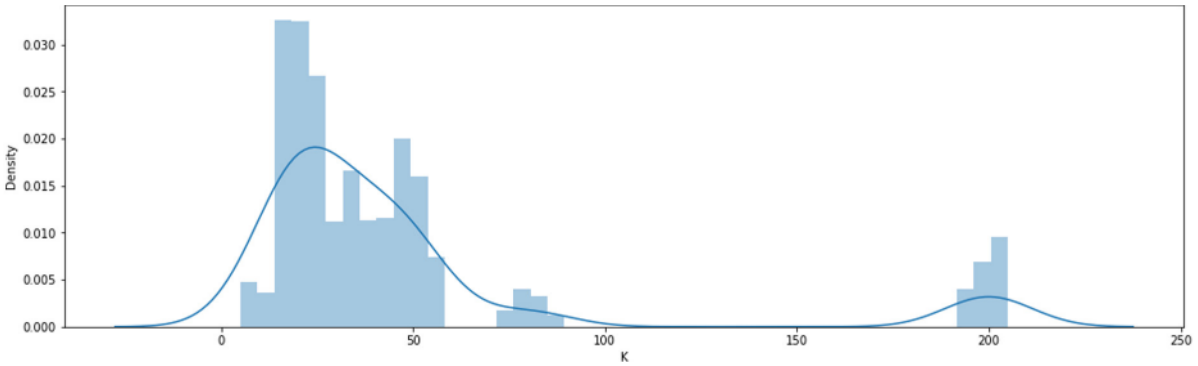

Fig. 3.1.17. Distribution of data for Potassium Column.

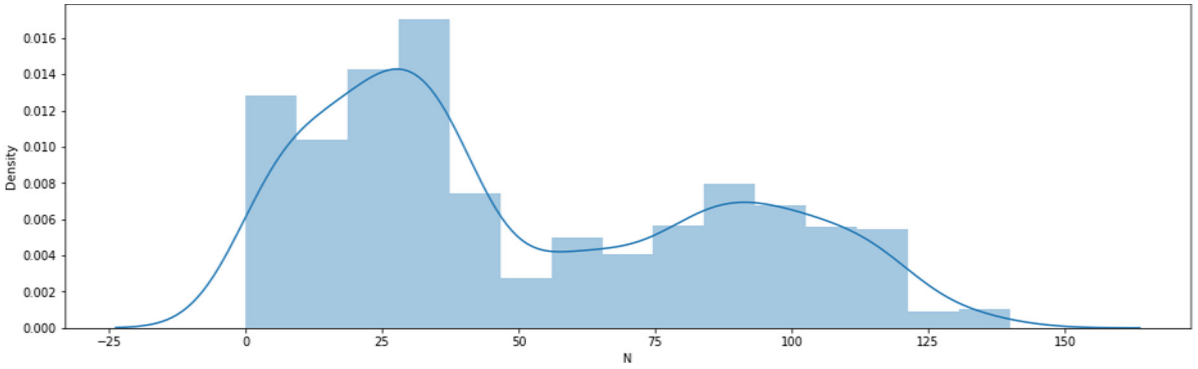

Fig. 3.1.18. Distribution of data for Nitrogen Column.

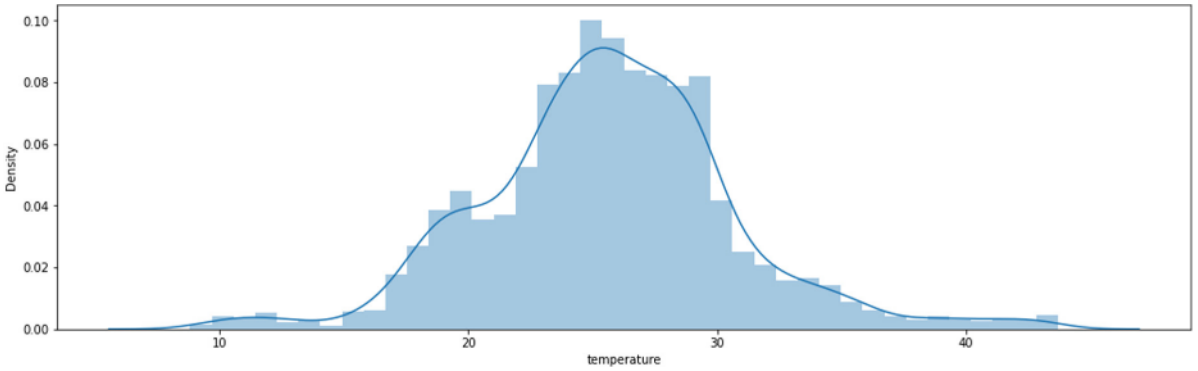

Fig. 3.1.19. Distribution of data for Temperature Column.

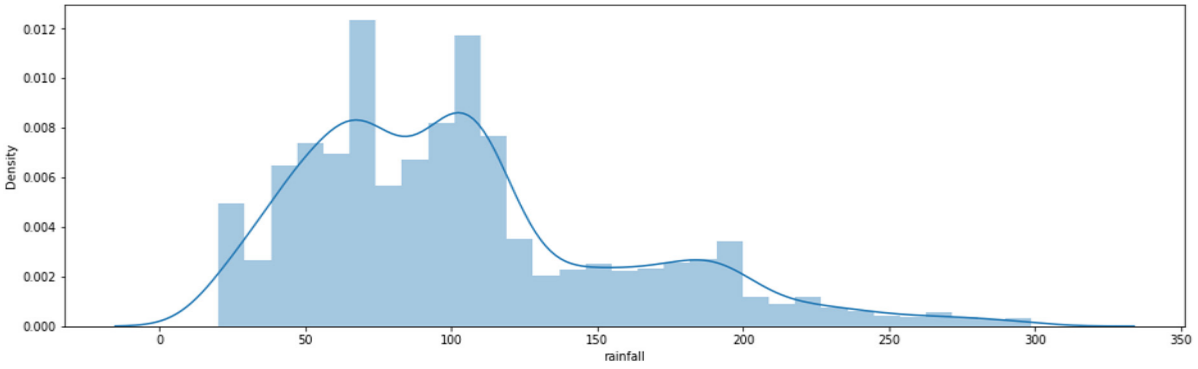

Fig. 3.1.20. Distribution of data for Rainfall Column.

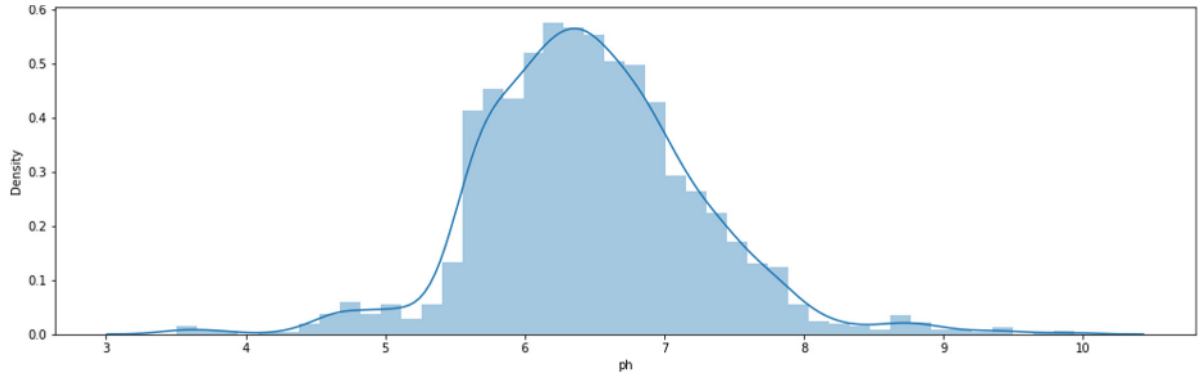

Fig. 3.1.21. Distribution of data for PH Column.

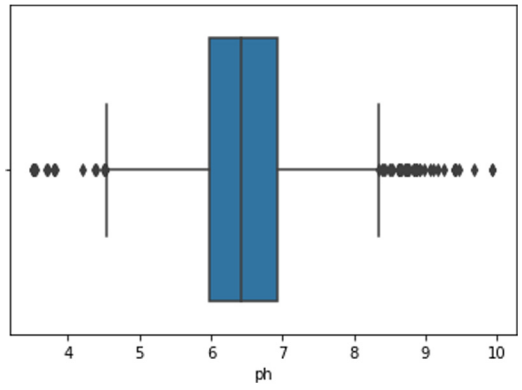

Fig. 3.1.22. Outliers in Soil PH attribute.

```
y_prediction: [15 21 17 ...  2 14 21]
Training set accuracy score: 100.0 %
Test set accuracy score: 94.72727272727272 %
Accuracy =  0.9254545454545454
              precision    recall  f1-score   support

           0       0.85      1.00      0.92        56
           1       0.94      0.94      0.94        49
           2       0.86      1.00      0.92        48
           3       1.00      1.00      1.00        53
           4       1.00      1.00      1.00        56
           5       0.87      1.00      0.93        52
           6       0.90      0.90      0.90        49
           7       0.96      0.98      0.97        49
           8       0.87      0.98      0.92        49
           9       1.00      1.00      1.00        57
          10       0.88      1.00      0.94        44
          11       0.95      0.83      0.89        47
          12       0.89      1.00      0.94        51
          13       0.98      0.83      0.90        54
          14       0.98      0.98      0.98        46
          15       1.00      1.00      1.00        46
          16       0.75      0.85      0.80        47
          17       0.94      0.92      0.93        51
          18       0.97      0.70      0.81        53
          19       0.89      0.63      0.74        54
          20       0.97      0.84      0.90        43
          21       0.98      1.00      0.99        46

    accuracy                           0.93      1100
   macro avg       0.93      0.93      0.92      1100
weighted avg       0.93      0.93      0.92      1100
```

Fig. 3.1.23. Model building results.

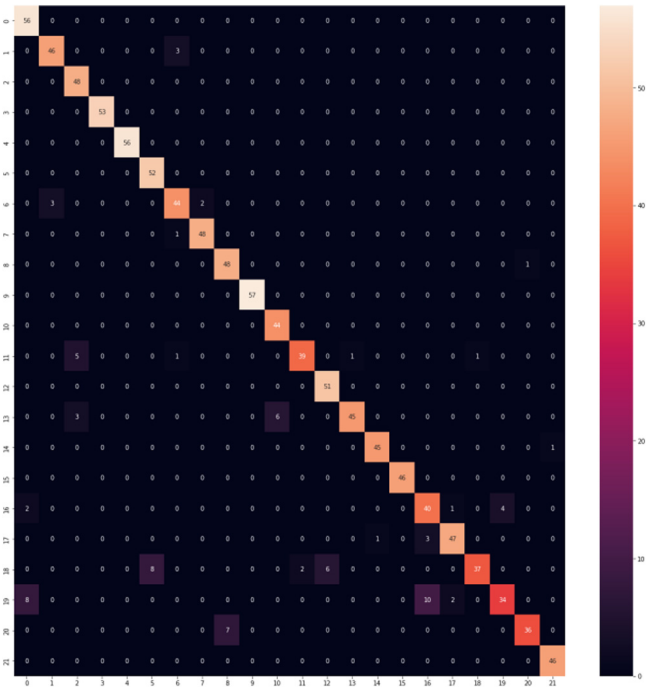

Fig. 3.1.24. Confusion matrix.

$$\text{Accuracy} = \frac{(\text{True Positive} + \text{True Negative})}{(\text{True Positive} + \text{False Positive} + \text{True Negative} + \text{False Negative})}$$

Fig. 3.1.25. Accuracy formula.

	Algorithm	model	Train_score	Test_score	accuracy
0	KNN	knn	89.636364	84.272727	84.272727
1	NaiveBayes	nb	96.363636	94.727273	94.727273
2	LogisticRegression	lr	66.454545	63.909091	63.909091
3	SVM	svm	69.454545	65.181818	65.181818
4	DecisionTreeClassifier	dt	69.454545	65.181818	92.181818
5	BaggingClassifier	bg	99.454545	92.545455	92.545455
6	RandomForestClassifier	rf	100.000000	94.727273	92.545455
7	AdaBoostClassifier	ad	14.363636	12.909091	12.909091
8	GradientBoostingClassifier	gb	95.727273	90.454545	90.454545
9	XGBClassifier	xg	96.363636	91.727273	91.727273
10	lbgmClassifier	lbgm	100.000000	93.454545	93.454545

Fig. 3.1.26. Model performance summary.

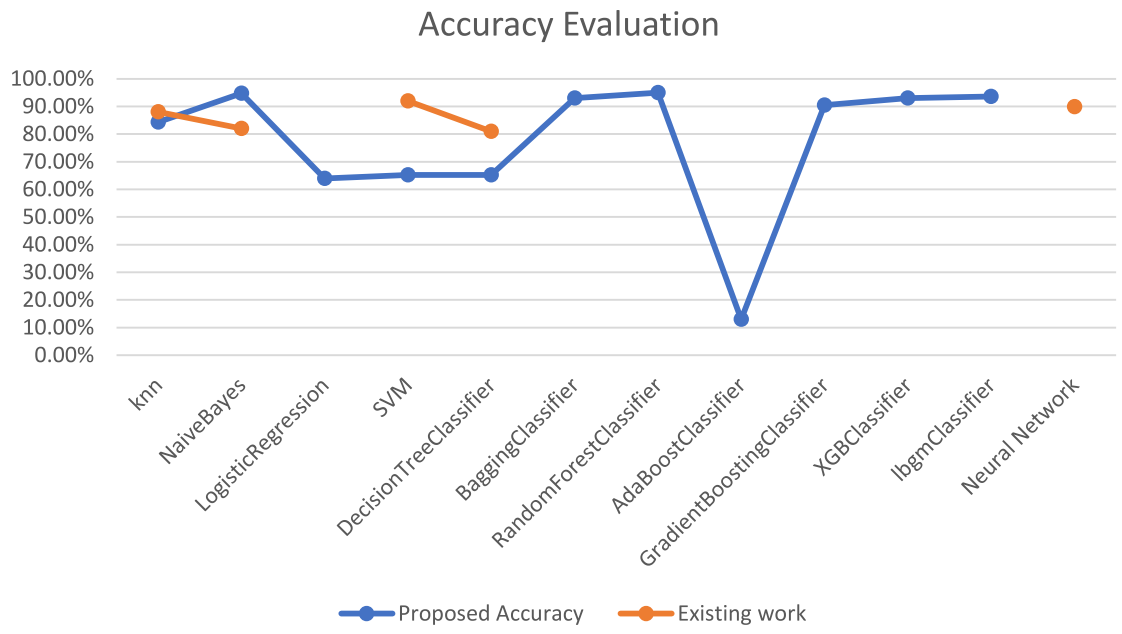

Fig. 3.1.27. Accuracy comparison.

	Algorithm	model	Train_score	Test_score	accuracy
1	NaiveBayes	nb	96.363636	94.727273	94.727273
4	DecisionTreeClassifier	dt	69.454545	65.181818	92.181818
5	BaggingClassifier	bg	99.454545	92.545455	92.545455
6	RandomForestClassifier	rf	100.000000	94.727273	92.545455
8	GradientBoostingClassifier	gb	95.727273	90.454545	90.454545
9	XGBClassifier	xg	96.363636	91.727273	91.727273
10	lbgmClassifier	lbgm	100.000000	93.454545	93.454545

Fig. 3.1.28. Models with accuracy above 90%.

Classification Algorithm	Hyperparameters used
K-nearest neighbor	• neighbours: [3, 4, 5, 10] • weights: ['uniform', 'distance'] • algorithm: ['auto', 'ball tree', 'kd_tree', 'brute'] • leaf size : [10, 20, 30, 50]
Naïve Bayes	Var smoothing: np.logspace(0,-9, num=100)
Decision Tree	• max_leaf_nodes: range between 2-100 • min_samples_split: [2, 3, 4]
Bagging Classifier	• base_estimator__max_depth: [1, 2, 3, 4, 5] • max_samples: [0.05, 0.1, 0.2, 0.5]
Random Forest Classifier	• bootstrap: [True, False] • max_depth: [10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110, None] • max_features: ['auto', 'sqrt'] • min_samples_leaf: [1, 2, 4] • min_samples_split: [2, 5, 10] • estimators: [200, 400, 600, 800, 1000, 1200, 1400, 1600, 1800, 2000]
AdaBoost Classifier	• estimators:[500,1000,2000] • learning rate:[.001,0.01,.1]
Gradient Boosting Classifier	• estimators=53 • learning rate=0.25 • max_features=2 • max_depth=2 • random state=0
XGBoost Classifier	• estimators: random number between 150-100 • learning rate: (0.01, 0.59) • subsample: (0.3, 0.6) • max_depth: [3, 4, 5, 6, 7, 8, 9] • colsample_bytree: (0.5, 0.4) • min_child_weight: [1, 2, 3, 4]

Fig. 3.1.29. Summary of all the hyperparameters used for the respective algorithm.

	Algorithm	model	accuracy
0	GSCV KNN	grid_obj_knn	85.545455
1	GSCV NB	grid_obj_nb	94.727273
2	GSCV DT	grid_obj_dt	93.818182
3	GSCV BG	grid_obj_bg	90.636364
4	GSCV NB_1	grid_obj_nb	94.727273
5	RSCV RF	rf_random	95.454545
6	GSCV RF	grid_search_rf	94.545455
7	GSCV ADA	grid_obj_ada	56.181818
8	GSCV GBC	grid_obj_gbc	93.272727
9	GSCV XGB	xgb_tune	92.090909

Fig. 3.1.30. Model performance summary with hyperparameter tuning.

precision, recall and F1 score for the applied model. The same steps are followed for all the 11 algorithms and the corresponding predicted values, training set accuracy, testing set accuracy, classification report is obtained. Fig. 3.1.24 shows the confusion matrix for the random forest classifier. The diagonal of the confusion matrix shows the how many times the prediction was correct. Accuracy is calculated using this formula (Fig. 3.1.25) using the formulas obtained in the confusion matrix.

Fig. 3.1.29 gives the summary of the performance of all the models used in the proposed work. According to this summary it is been observed that most of the algorithms has accuracy above 90%. In order to obtain best model for prediction all the models whose accuracy above 90% is picked and hyper parameter tuning is performed.

Fig. 3.1.31 shows a comparison of accuracy between Proposed work and Existing work. The existing work has used less algorithms and their accuracy are displayed in color orange. Any algorithm is been measured using accuracy to obtain a best model. Based on the model evaluation all the algorithms used by existing authors seems to have less accuracy. The proposed work has used 11 algorithms in total to obtain a best model for further analysis. To further extract the model's capability this work has used hyperparameter tuning.

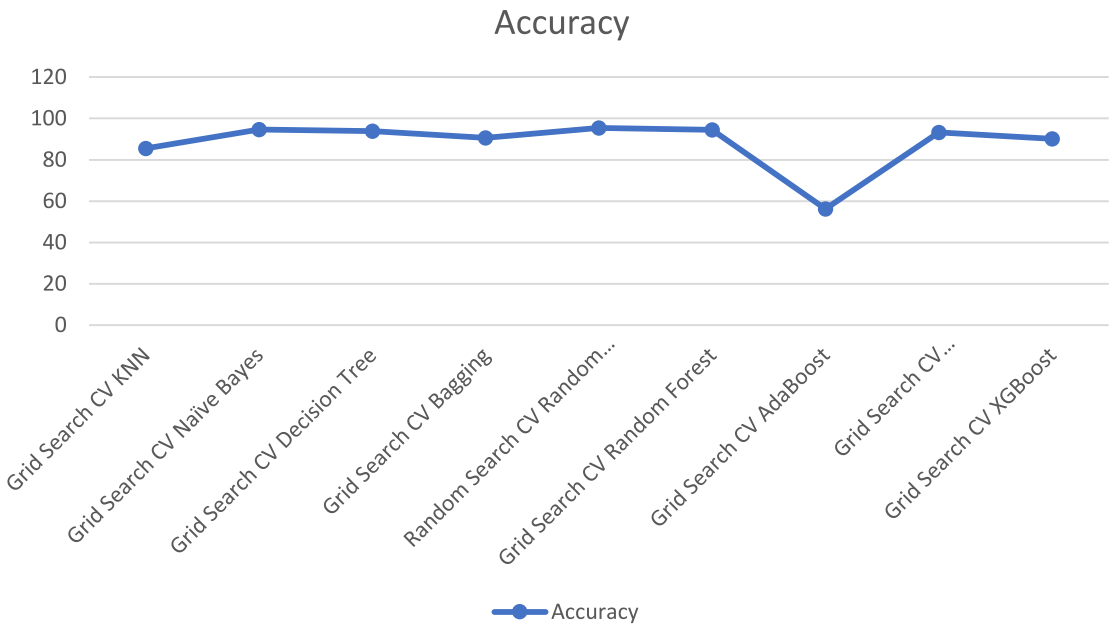

Fig. 3.1.31. Model performance summary with hyperparameter tuning.

Step 8: Hyperparameter Tuning

Hyperparameters are the key factors that determine the model architecture, and hyperparameter tuning seems to be the process of discovering the optimum model architecture. Hyperparameters were used to tune that filtered algorithms.

Fig. 3.1.28 depicts the hyperparameters used for the algorithms above 90%. Different algorithms have different parameters to be defined before going for training the algorithms. Once these parameters are defined the model is defined along with adding these parameters to it using Randomized CVS or Grid Search CV algorithm. Once the model is defined using these parameters as shown in Fig. 3.1.28 training data is used for training the hyper tuned model followed by testing them using the test set. Accuracy for the hyper tuned algorithms is calculated and hence used as metric for deciding which algorithm to choose for future predictions (see Fig. 3.1.29).

Some algorithms after hyperparameter tuning its accuracy decreased but when Random Forest Classifier was hyper tuned using RandomSearchCV increased the accuracy to 95.4545% which is shown in Fig. 3.1.29 (see Fig. 3.1.30).

The Fig. 3.1.26 depicts the accuracy comparison among algorithms with hyperparameter tuning. This shows that most of the algorithms when hyper tuned tends to provide more accuracy. Hence Random Forest Classifier hyper tuned using Randomized Search CV is been picked and used for further analysis.

Step 9: Predicting the crop

On identifying the best model for predicting the crop. The model is stored in a pickle file. Pickle file is used to serialize Python objects architectures, which is the method of transforming an object in memory to a byte stream that can be stored on drive as a binary file. This binary file can be de-serialized back to a Python object when we load it into a Python code. Hence, the model is stored into a pickle file so just on calling a pickle file we can predict the crops.

Values of temperature, humidity, average rainfall and soil Ph need to be passed into the model to predict the crops. In order to obtain the climate conditions such as temperature and humidity the latitude and longitude of the current location is identified using Web Scraping the website 'https://ipinfo.io/'. This is done by requesting the program to open this website which uses the IP address of the device to obtain the details on the current location. The values extracted are longitude and latitude values. Extracted values are passed as input to the OpenWeatherAPI which provides the details about the current

Case 1: minimum temperature <= base temperature

$$GDD = \frac{(\text{base temperature} + \text{maximum temperature})}{2} - \text{base temperature}$$

Case 2: minimum temperature > base temperature

$$GDD = \frac{(\text{minimum temperature} + \text{maximum temperature})}{2} - \text{base temperature}$$

Case 3: maximum temperature > base maximum temperature

$$GDD = \frac{(\text{minimum temperature} + \text{base maximum temperature})}{2} - \text{base temperature}$$

Fig. 3.1.32. GDD value calculation.

$$\text{Nitrogen concentration for 200 lb. Fertilizer} = 200 * \frac{(\text{Mean value of Nitrogen ratio})}{100}$$

$$\text{Phosphorous concentration for 200 lb. Fertilizer} = 200 * \frac{(\text{Mean value of Nitrogen ratio})}{100}$$

$$\text{Potassium concentration for 200 lb. Fertilizer} = 200 * \frac{(\text{Mean value of Nitrogen ratio})}{100}$$

Fig. 3.1.33. Nutrient concentration calculation.

location. Temperature, humidity, place, region details are fetched. The average rainfall and soil Ph value is entered by the user through a Web interface. On integrating all the parameters and is passed into the model for prediction. The model prediction probability is been fetched and ordered in descending order and the crops with highest probability is been selected and values are decoded to get the actual crop names. Thus, the model displays the Top 5 crop names that can be grown in the current location. This work not only provides recommendations of crops but also suggests the GDD-Growing degree days and the amount of NPK — Nitrogen, potassium, phosphorous required for crop growth.

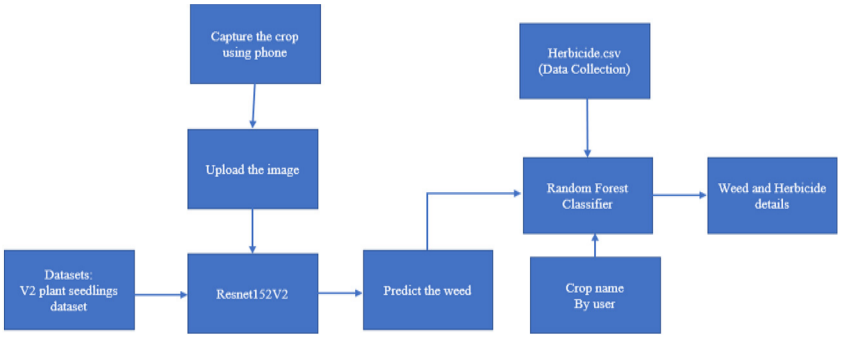

Fig. 3.2.1. Weed Identification system architecture.

Fig. 3.2.2. Sample images of weeds used in the model building.

Attributes	V2_plant seedlings
Source	https://www.kaggle.com/vbookshelf/v2-plant-seedlings-dataset
No.of samples	4823
Type	Image Dataset
Used for	Classification
Labels Count	9

Fig. 3.2.3. Dataset details for Weed Identification module.

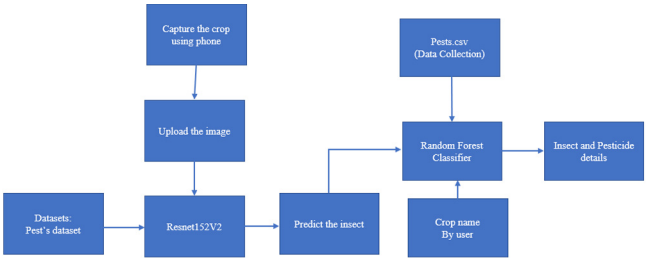

Fig. 3.3.1. Pest Identification system architecture.

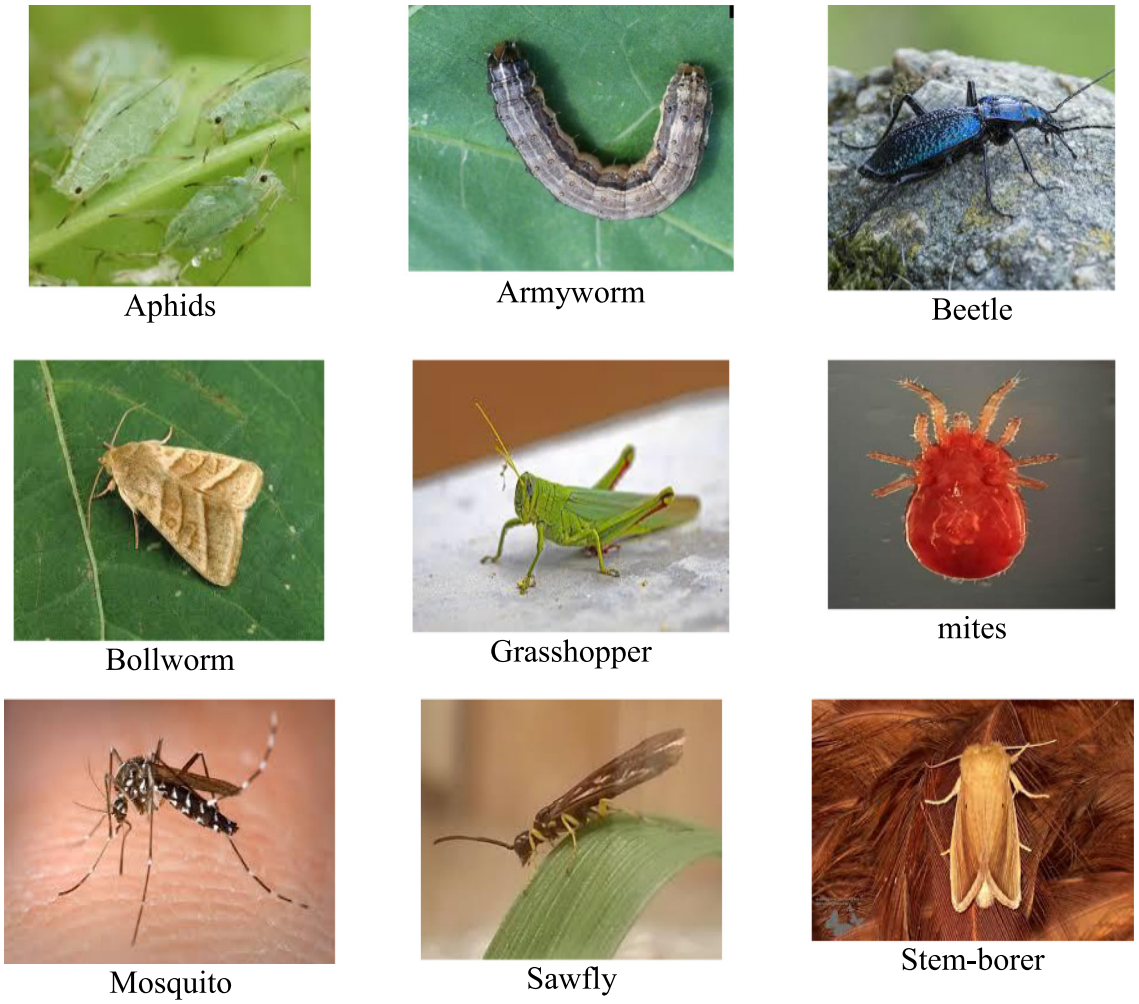

Fig. 3.3.2. Sample images of pests dataset.

Attributes	Pests Dataset
Source	https://www.kaggle.com/simranvolunesia/pest-dataset
No.of samples	3150
Type	Image Dataset
Used for	Classification
Labels Count	9

Fig. 3.3.3. Dataset details for Pest Identification and Pesticide Recommendation module.

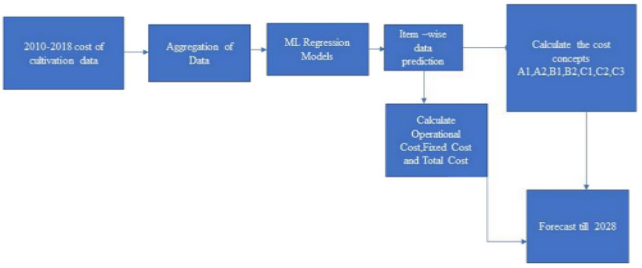

Fig. 3.4.1. Cost estimation system architecture.

Growing degree days (GDD) are a weather-based marker that can be used to gauge crop development. It is a crop producer calculation that is a measure of temperature levels that is used to anticipate plant and pest growth rates, such as the time that a crop achieves maturity. The calculation of Growing Degree Days allows farmers to forecast the rate at which their plants will mature. Fig. 3.1.27 demonstrates of how the GDD values are calculated for each predicted crop.

Respective Minimum and maximum values are obtained for a crop from the dataset. Each crop possesses minimum and maximum base temperatures. These temperature values were been obtained from official agricultural websites and were stored as a dataset. These values were been applied to get the GDD value for a crop (see Fig. 3.1.31).

Nitrogen, Phosphorous, Potassium content is necessary for crop growth. Mean value of nitrogen, Phosphorous and potassium was extracted for a particular crop and was divided by 100 and was multiplied by 200 to obtain the nutrient concentration for 200 lb. fertilizer per hectare is shown in Fig. 3.1.28 (see Figs. 3.1.32 and 3.1.33).

3.2. Module 2: Weed identification

Dataset: The dataset used for this module is v2 plant seedlings dataset from Kaggle (see Fig. 3.2.1).

This dataset was used since it has images of various weed growth stages. This is very important since it has images of all weed stages. Any weed at its initial stage can be captured and uploaded into the system.

Attributes	Cost of Cultivation
Source	https://eands.dacnet.nic.in/Cost_of_Cultivation.htm
No.of documents	8
Used for	Forecasting

Fig. 3.4.2. Dataset details for cost estimation module.

```
Index(['Year', 'Crop', 'State', 'Cultivation Cost A1 ', 'Cultivation Cost A2',
       'Cultivation Cost B1', 'Cultivation Cost B2', 'Cultivation Cost C1',
       'Cultivation Cost C2', 'Cultivation Cost C2 Revised',
       'Production Cost A1', 'Production Cost A2', 'Production Cost B1',
       'Production Cost B2', 'Production Cost C1', 'Production Cost C2',
       'Production Cost C2 Revised', 'Production Cost C3',
       'Value of Main Product', 'Value of By- Product (Rs./Hectare)',
       'Material Input of Seed', 'Material Input of Fertilizer',
       'Material Input of Manure', 'Labour Input of Human',
       'Labout Input of Animal', 'Rate per Unit of Seed',
       'Rate per Unit of Fertilizer', 'Rate per Unit of Manure',
       'Rate per Unit of Human Labour', 'Rate per Unit of Animal Labour',
       'Implicit Rate', 'Number of Holdings in Sample',
       'Number of Tehsils in Sample', 'Derived Yield of Crop',
       'Family Human Labour Hours', 'Attached Human Labour Hours',
       'Casual Human Labour Hours', 'Total Human Labour Hours',
       'Operational Cost', 'Operational Cost of Family Human Labour',
       'Operational Cost of Attached Human Labour',
       'Operational Cost of Casual Human Labour',
       'Operational Cost of Total Human Labour',
       'Operational Cost of Hired Animal Labour',
       'Operational Cost of Owned Animal Labour',
       'Operational Cost of Total Animal Labour',
       'Operational Cost of Hired Machine Labour',
       'Operational Cost of Owned Machine Labour',
       'Operational Cost of Total Machine Labour', 'Operational Cost of Seed',
       'Operational Cost of Fertilizer', 'Operational Cost of Manure',
       'Operational Cost of Fertilizer & Manure',
       'Operational Cost of Insecticides',
       'Operational Cost of Irrigation Charges', 'Crop Insurance',
       'Payment to Contractor', 'Miscellaneous Operational Cost',
       'Operational Cost of Interest on Working Capital', 'Fixed Costs',
       'Fixed Cost on Rental Value of Owned Land',
       'Fixed Cost on Rent Paid for Leased-in-Land',
       'Fixed Cost on Land Revenue, Taxes & Cesses',
       'Fixed Cost of Depreciation on Implements & Farm Building',
       'Fixed Cost of Interest on Fixed Capital', 'Total Cost [11+12]'],
      dtype='object')
```

Fig. 3.4.3. Represents the Columns present in the dataset.

$$\text{Mean squared error} = \frac{\sum (\text{predicted value} - \text{actual value})^2}{2}$$

$$\text{Mean absolute error} = \frac{\sum |\text{predicted value} - \text{actual value}|}{2}$$

$$\text{Root mean squared error} = \sqrt{\text{mean squared error}}$$

Fig. 3.4.4. Formulas for estimating the model.

The proposed work efficiently categorizes the weed type. This is due to the model trained by the v2 plantings dataset.

Steps involved in Weed Identification module are as follows

Step 1: Importing Libraries and Dataset

In order to utilize Deep Learning algorithms and preprocessing tools specific libraries needs to be imported. Using these libraries, the model building and prediction would be performed efficiently. The libraries such as NumPy, pandas, train_test_split, Resnet152V2, TensorFlow, keras were imported. Fig. 3.2.3 shows the glimpse of the V2_Plant_seedlings dataset. After importing the dataset, the file names are been saved in a data frame along with its class name. Some of the images present in each class are depicted using Fig. 3.2.2. There are nine weed classes present in the dataset. Each class has more than 200 images.

Step 2: Splitting the Training, Testing set and Validation set

The dataset is split into 3 parts namely training, testing and validation data. The Validation data is used while executing the epochs.

Training set contains 2701 images and Testing set contains 1447 images. Validation set has 675 images. All the images were resized to 224X224 size.

Data Augmentation meaning increasing the number of images for training was performed. This included in rotating, flipping and rescaling the image. Additional images were obtained which is then passed into the model for classification.

Step 3: Model Building

To classify the images Resnet152V2 pre-trained keras model is been used. Deep Residual Networks are networks with convolutional, pooled, activating, and fullyconnected layers mounted on top of each other. The only structure that converts an accordance with the established into a recurrent neural network is the identity link between the levels. To address the problem of the curse of dimensionality, this architecture adds a conception of the Residual Network. In this network, we use a technique known as skip connections. The skip connection skips several phases of training and connects output pin. The advantage of incorporating this manner of residual connections is that regularization will bypass any layer that reduces architecture performance. As both a corollary, incredibly deep learning models can be trained without the problems that vanishing/exploding gradients cause. Due to its advantages Resnet152V2 was opted for the proposed work to classify the images.

In the proposed work Resnet152V2 was defined followed by Global Average Pooling2D, Dropouts and additional hidden layers.4 Hidden layers was introduced above resnet152V2. First Hidden layer had 1024 neurons, second and third hidden layer with 512 neurons and fourth hidden layer with 32 neurons. All the hidden layers had activation function RELu. Dropout were introduced in between each hidden layer so that it removes unnecessary neurons and tries to select the best parameters to identify the images. The output layer had 1 neuron with activation function SoftMax for classification.

The epochs executed for this module is 25 with batch size 32. The epochs are trained using the training data and are validated using validation data. Checkpoints are being defined and these checkpoints are stored for future weights updation.

Step 4: Herbicide Recommendation

Not only predicting the image is important but also recommending Herbicides for the predicted weeds would be very useful for the user. Herbicides are generally used for killing the weeds. Manually removing the weeds are mostly impossible for a large farm. Hence herbicides are optimal way for destroying the weeds from the land.

The Herbicide details were collected by referring to various agricultural websites. To find a suitable herbicide the chemical component to kill them is identified from agricultural websites. Based on the chemical components the herbicide having it has an active component is being picked. The elaborate details regarding dosage, well-suited crops, well-suited soil, active ingredients, identifying whether it is a pre-emergent

- **Item-wise Estimation**
- **Total_Human_Labour** = *Operational_Cost_Attached_Labour +Operational_Cost_Casual_Labour*
- **Animal_Labour** = *Operational_Cost_hired_Animal_Labour+Operational_Cost_Owned_Animal_Labour*
- **Machine_Labour** = *Operational_Cost_Owned_Machine_Labour +Operational_Cost_Hired_Machine_Labour*
- **Fertilizer_and_Manure** = *Operational_Cost_On_Fertilizer +Operational_Cost_On_Manure*
- **Total_Operational_cost** = *Total_Family_Labour+Animal_Labour+Machine_Labour+Fertilizer_and_Manure+Operational_Cost_On_Insectides+Operational_Cost_On_Irrigation+ Operational_Cost_On_crop_Insurance+Operational_Cost_On_Payment_to_Contractor+Operational_Cost_On_Miscellaneous_Operational_Cost+Operat ional_Cost_Of_Interest_On_working_capital*
- **Fixed Cost** = *fc rental value of owned land +FC Rent Paid for Leased Land + FC Land revenue tax and cesses+Operational_Cost_Of_Interest_On_working_capital+FC_Depreciation_on_Implements_Farm_Building+FC_Interest_on_Fixed_Capital*
- **Total Cost**=*Total_Operational_cost +Fixed Cost*

Fig. 3.4.5a. Item-wise estimation formulas.

Cost A_1 : It includes –

1. Value of hired human labour
2. Value of hired and owned bullock labour
3. Value of hired and owned machine labour
4. Value of seed (both farm seed and purchased)
5. Value of manures (owned and purchased) and fertilizers
6. Depreciation
7. Irrigation charges
8. Land revenue
9. Interest on working capital
10. Miscellaneous expanses

Cost A_2 : Cost A_1 + rent paid for leased-in land

Cost B_1 : Cost A_1 + interest on fixed capital (excluding land)

Cost B_2 : Cost B_1 + rental value of owned land + rent for leased-in land

Cost C_1 : Cost B_1 + imputed value of family labour

Cost C_2 : Cost B_2 + imputed value of family labour

Cost C_3 : Cost C_2 + 10 per cent of cost C_2 as management cost.

Fig. 3.4.5b. Cost of cultivation formulas.

or post-emergent are obtained by a pdf file called 'Product Label'. This product label provides the required details. These details are stored as a dataset which is used for recommendation.

For all crops same herbicide cannot be used. It could harm the growth of the crops as well as affect the soil fertility. Hence to ensure accurate details crop specific herbicides are predicted using Random Forest Classifier.

The input parameters for Random Forest Classifier are crop name and the name of the weed. The crop name must be entered by the user. The name of the weed is identified by the Resnet152V2 fine tuning model when the image is been uploaded by the user. The Random Forest Classifier classifies the crops based on its highest probability herbicides are listed.

3.3. Module 3: Pests identification and pesticides recommendation

Dataset: The dataset used for this module is pests dataset from Kaggle (see Fig. 3.3.1).

This dataset was used since it has images of various insects captured at different locations. In many scenarios insects always merge with the crop and it is difficult to distinguish them. Hence this dataset is selected since it has insects in contact with crops. Any insect can be captured and uploaded into the system. The proposed work efficiently categorizes the insect. This is due to the model trained by the pest's dataset.

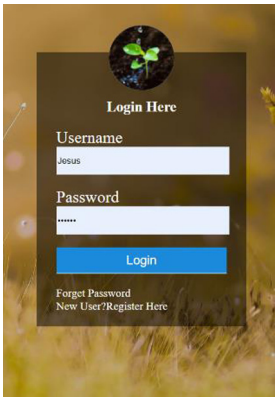

Fig. 4.1. Login page.

Steps involved in Pest Identification and Pesticides Recommendation module are as follows

Step 1: Importing Libraries and Dataset

In order to utilize Deep Learning algorithms and preprocessing tools specific libraries needs to be imported. Using these libraries, the model building and prediction would be performed efficiently. The libraries such as NumPy, pandas, train_test_split, Resnet152V2, TensorFlow, keras were imported. Fig. 3.3.3 shows the glimpse of the Pests dataset. After importing the dataset, the file names are been saved in a data frame along with its class name. Some of the images present in each class are depicted using Fig. 3.3.2. There are nine insect classes present in the dataset.

Step 2: Splitting the Training, Testing set and Validation set

The dataset is split into 3 parts namely training, testing and validation data. The Validation data is used while executing the epochs.

Training set contains 1764 images and Testing set contains 945 images. Validation set has 441 images. All the images were resized to 224X224 size.

Data Augmentation meaning increasing the number of images for training was performed. This included in rotating, flipping and rescaling the image. Additional images were obtained which is then passed into the model for classification.

Step 3: Model Building

To classify the images Resnet152V2 pre-trained keras model is been used. Deep Residual Networks are networks with convolutional, pooled, activating, and fully connected layers mounted on top of each other. The only structure that converts an accordance with the established into a recurrent neural network is the identity link between the levels. To address the problem of the curse of dimensionality, this architecture

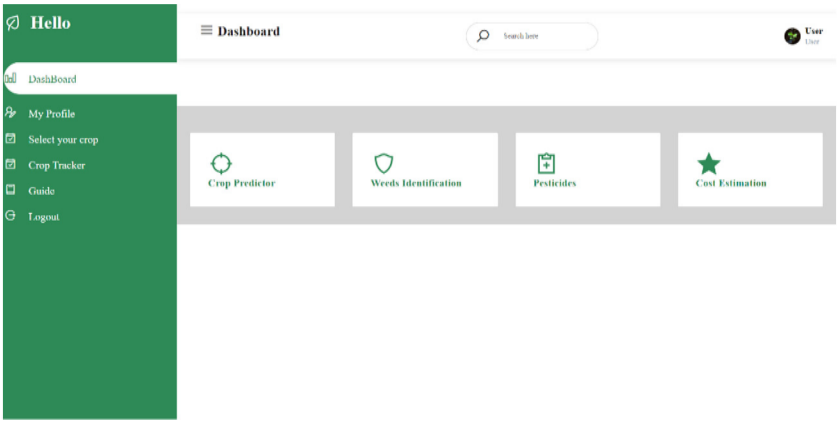

Fig. 4.2. User dashboard.

adds a conception of the Residual Network. In this network, we use a technique known as skip connections. The skip connection skips several phases of training and connects output pin. The advantage of incorporating this manner of residual connections is that regularization will bypass any layer that reduces architecture performance. As both a corollary, incredibly deep learning models can be trained without the problems that vanishing/exploding gradients cause. Due to its advantages Resnet152V2 was opted for the proposed work to classify the images.

In the proposed work Resnet152V2 was defined followed by Global Average Pooling2D, Dropouts and additional hidden layers.4 Hidden layers was introduced above resnet152V2. First Hidden layer had 1024 neurons, second and third hidden layer with 512 neurons and fourth hidden layer with 32 neurons. All the hidden layers had activation function RELu. Dropout were introduced in between each hidden layer so that it removes unnecessary neurons and tries to select the best parameters to identify the images. The output layer had 1 neuron with activation function SoftMax for classification.

The epochs executed for this module is 20 with batch size 32. The epochs are trained using the training data and are validated using validation data. Checkpoints are being defined and these checkpoints are stored for future weights updation. The accuracy obtained was 0.98 i.e., 98%.

Step 4: Pesticide Recommendation

Not only predicting the image is important but also recommending pesticides for the predicted insect would be very useful for the user. Pesticides are sprayed for killing the insects so that it does not destroy the crops and increase in number in the fields.

The Pesticide details were collected by referring to various agricultural websites. To find a suitable herbicide the chemical component to kill them is identified from agricultural websites. Based on the chemical components the pesticides having it has an active component is being picked. Other details regarding well-suited crops are obtained by a pdf file called 'Product Label'. This product label provides the required details. These details are stored as a dataset which is used for recommendation.

Same as herbicides for all crops same pesticides cannot be used. It could harm the crop growth as well as affect the soil fertility. To ensure correct pesticides crop names along with predicted insect are passed Random Forest Classifier. The crop name must be entered by the user. The name of the weed is identified by the Resnet152V2 fine tuning model when the image is been uploaded by the user. The Random Forest Classifier classifies the crops based on its highest probability pesticides are listed.

	Algorithm	model	Train_score	Test_score	accuracy
0	KNN	knn	89.636364	84.272727	84.272727
1	NaiveBayes	nb	96.363636	94.727273	94.727273
2	LogisticRegression	lr	66.454545	63.909091	63.909091
3	SVM	svm	69.454545	65.181818	65.181818
4	DecisionTreeClassifier	dt	69.454545	65.181818	92.181818
5	BaggingClassifier	bg	99.454545	92.545455	92.545455
6	RandomForestClassifier	rf	100.000000	94.727273	92.545455
7	AdaBoostClassifier	ad	14.363636	12.909091	12.909091
8	GradientBoostingClassifier	gb	95.727273	90.454545	90.454545
9	XGBClassifier	xg	96.363636	91.727273	91.727273
10	lbgmClassifier	lbgm	100.000000	93.454545	93.454545

Fig. 4.1.1. Summary details of all the models used.

3.4. Cost estimation

The main aim of cost estimation module is to predict the cost concepts and total cost needed for cultivation. This module is very important in recent times due to sudden crisis, natural calamities, price rise of cultivation resources. Indian cost of cultivation survey data is considered for this module. Eight years data was collected. From 2010–2018. The data was aggregated since each year was a separate file as shown in Fig. 3.4.2. In order to forecast values Machine Learning Regression models are been considered (see Fig. 3.4.1).

Discussing about the dataset the attributes are depicted in Fig. 3.4.3. The dataset can be viewed as two parts: Firstly, is the item wise break up attributes and cost of cultivation attributes. Item-wise breakup includes all the operational cost attributes and fixed cost attributes. Cost of cultivation attributes includes Cultivation A1, A2, B1, B2, C1, C2 and C3. To forecast the cost details, it is important to run each item-wise attribute separately. The year, state and crop attribute are used as the independent variables and each item-wise attribute separately.

For example, let us consider the Operational cost of Fertilizer & manure attribute. For predicting this attribute following steps are followed.

Step 1: Importing libraries and dataset

The necessary libraries such as NumPy, pandas, train_test_split, Label Encoder, Machine Learning Ensemble Regressors are being imported. The 8 separate datasets corresponding to years between 2010–2018 are been loaded.

Step 2: Data Preprocessing and Model Building

The imported datasets are aggregated to obtain a single dataset. The crop and state column are non-numerical in nature. Hence Label

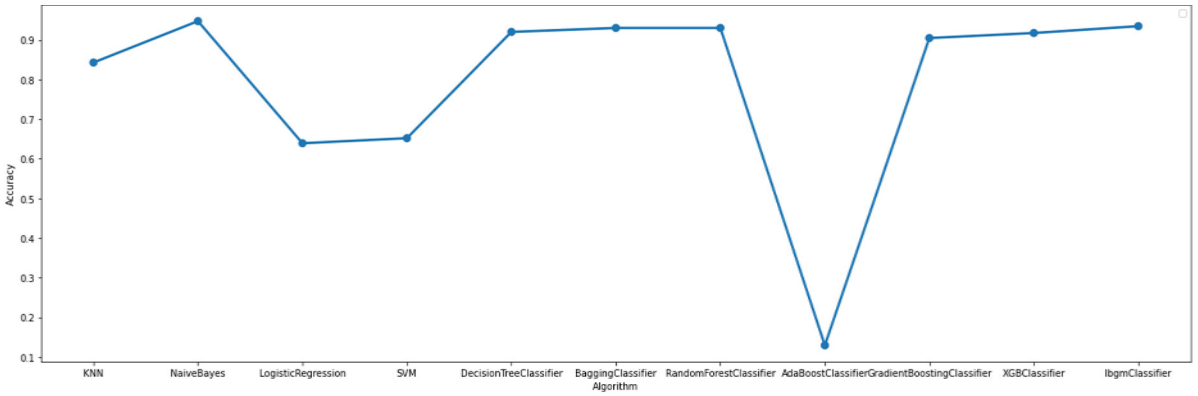

Fig. 4.1.2. Visual representation of algorithms with their accuracy.

	Algorithm	model	accuracy
0	GSCV KNN	grid_obj_knn	85.545455
1	GSCV NB	grid_obj_nb	94.727273
2	GSCV DT	grid_obj_dt	93.818182
3	GSCV BG	grid_obj_bg	90.636364
4	GSCV NB_1	grid_obj_nb	94.727273
5	RSCV RF	rf_random	95.454545
6	GSCV RF	grid_search_rf	94.545455
7	GSCV ADA	grid_obj_ada	56.181818
8	GSCV GBC	grid_obj_gbc	93.272727
9	GSCV XGB	xgb_tune	92.090909

Fig. 4.1.3. Models performance details after hyperparameter tuning.

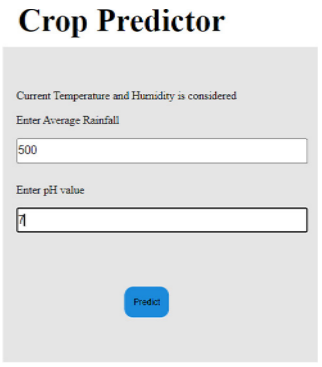

Fig. 4.1.4. Front end to enter the details average temperature and rainfall.

Encoding is performed to obtain numerical values for the same. Year, encoded crop name, encoded state name is been feed into the Machine Learning and Ensemble regressor models.

Step 3: Model Building

With regard to Regressor models XGBoost regressor is considered for explanation. Year, encoded crop name, encoded state name is been considered as the independent variable and Operational cost of Fertilizer & manure is considered as a dependent value. The dataset is split into test and training data. The regressor model is been defined and the training data is fitted into the model to train the model. The model is then tested using the testing score. The r2 score, root mean squared error, mean squared error, mean absolute error is been estimated for each model as shown in Fig. 3.4.4.

Step 4: Cost Calculation

Based on these values the best model is selected with the highest r2score. The same procedure is been practiced for all the item-wise attributes and the best models are stored in a pickle file. The pickle file is then used to forecast the values till 2028. Once all the attribute forecasting values are obtained then these values are stored dynamically in a data frame. This data frame is converted into a csv file and is used for future calculation.

When discussing about cost calculation the formulas are used to obtain the values of Operational cost, fixed cost, total cost, cost concepts values A1, A2, B1, B2, C1, C2 and C3. The forecasted values till 2028 are been applied to get the results. The formulas as depicted in Fig. 3.4.5a and Fig. 3.4.5b.

4. Results & discussion

The proposed work is a web interface through which the user can access the models efficiently. Fig. 4.1 represents the login page through which the user needs to login with their credentials to access the models. Fig. 4.2 shows the user dashboard after logging in, thus enabling users to access the models. The results of each model are discussed in each section.

4.1. Crop recommendation

Ten Algorithms were used for crop recommendation. The accuracy score above 90% was selected as shown in Fig. 4.1.1. The accuracy drifts are been clearly observed using Fig. 4.1.2. For these selected algorithms Hyperparameter Tuning was applied from which best model with highest accuracy was obtained as shown in Fig. 4.1.2

According to Fig. 4.1.3 the Random Forest classifier hyper tuned with Randomized CV is opted as best model since its accuracy is 95.45% and stored as a pickle file for further analysis.

Frontend:

The following figure shows Crop recommendation system. The user needs to provide details about the average rainfall and ph. The other parameters are derived. Initially longitude and latitude values are extracted from the current location using Web Scraping and is passed as parameters to the Weather API. The weather API provides the temperature, humidity details of the current location. Based on the parameters passed, the system predicts the 5 crops and provides details. According to Fig. 4.1.4 the user has entered the values 500 for average rainfall and 7 as the soil ph.

The page after clicking predict directs to the results page which contain the details provided section depicted by Fig. 4.1.6. This shows the longitude and latitude values of the current location. It also shows the place, region. Followed by the basic details the temperature, humidity obtained from passing longitude and latitude values are shown. Average rainfall and ph. values are entered by the user (see Fig. 4.1.5).

Details Provided

Attributes	Values
Longitude	77.5937
latitude	12.9719
place	Bengaluru
region	Karnataka
temperature	20
humidity	88
PH	7
Average Rainfall	500

Fig. 4.1.5. Details provided to the model.

Crop Details

Crops	pigeonpeas	apple	banana	blackgram	chickpea
Scientific Name	Cajanus cajan	Malus Pumila	Musa	Vigns mungo	Cicer arietinum
Soil type	['sandy' 'loamy' 'clayey']	['loamy']	['loamy']	['sandy' 'loamy']	['sandy' 'loamy' 'black' 'clayey']
Crop type	['Rabi Crop', 'Kharif Crop']	['Kharif Crop']	['Rabi Crop', 'Kharif Crop']	['Rabi Crop', 'Kharif Crop']	['Rabi Crop', 'Kharif Crop']
Growing Degree Days	9.648524160000001	15.516694609999998	12.459534894999997	5.021994705000001	9.010003044999998
Nitrogen Requirement for 200lb fertilizer	41.46	41.6	200.45999999999998	80.04	80.18
Potassium Requirement for 200lb fertilizer	40.58	399.78	100.1	38.48	159.84
Phosphorous Requirement for 200lb fertilizer	135.46	268.44	164.02	134.94	135.58

Fig. 4.1.6. Top 5 crop recommendations.

Based on the given details the model takes temperature, humidity and rainfall and ph. details to recommend crops. The system provides Top 5 Crops based on the details provided.

The above Fig. 4.1.7 shows the Top 5 crop recommendations followed by its scientific name. Once the model has predicted the crop name is passed on to identifying its scientific name. The system contains a dataset containing scientific names. In the same way the soil type is also obtained from soil.csv dataset for that specific crop. The Crop type specifies whether the given crop is Rabi, Kharif or Summer. The crop is been categorized based on its parameters such as Temperature, humidity and rainfall (see Table 4.1.1).

After identifying the crop type the system estimates the Growing Degree Days, Nitrogen requirement, potassium requirement, phosphorous section whose formulas are discussed in Section 3.1. It is observed that the GDD of apple is more and black gram being the least. With respect to nitrogen requirement banana requires the more nitrogen content to be present in 200 lb. fertilizer and apple and pigeon peas requires the same quantity. With respect to phosphorous and potassium apples requirement content is highest as shown in Fig. 4.1.7.

Table 4.1.1
Categorization of crops.

Crop type	Temperature	Humidity	Rainfall
Kharif Crop	14–37	18–100	100 and above
Summer Crop	25–37	40–85	
Rabi Crop	15–35	15–100	

The predictions as well as the details are more important for a user to analyze and select a crop. The existing research restricts the crop recommendation of only one. And does not provide any additional values. The proposed system predicts top 5 crops with its details which can be used by the user for carrying out their work in a smarter way.

The system just does not take up the inputs by the user but also checks if it is a valid input. For example the soil Ph range is between 0–14. If the user types a value greater than 14 or less than 0 the system displays a message that it is wrong and asks the user to re-enter correct

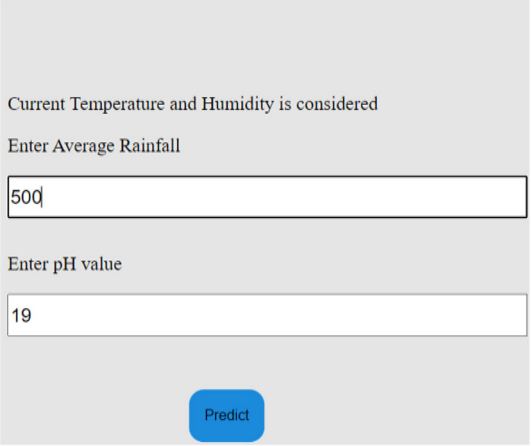

Fig. 4.1.7. Incorrect entries.

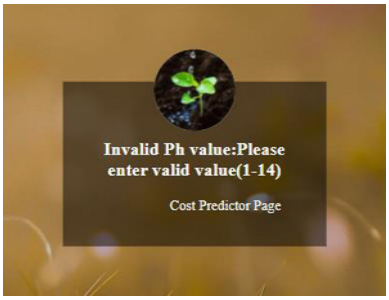

Fig. 4.1.8. Requesting the user to enter correct values.

	FilePath	Labels
2665	/content/drive/MyDrive/v2_plant_seedlings/Loos...	Loose Silky-bent
2609	/content/drive/MyDrive/v2_plant_seedlings/Loos...	Loose Silky-bent
3638	/content/drive/MyDrive/v2_plant_seedlings/Scen...	Scentless Mayweed
3792	/content/drive/MyDrive/v2_plant_seedlings/Scen...	Scentless Mayweed
2581	/content/drive/MyDrive/v2_plant_seedlings/Loos...	Loose Silky-bent
...	...	...
2895	/content/drive/MyDrive/v2_plant_seedlings/Loos...	Loose Silky-bent
2763	/content/drive/MyDrive/v2_plant_seedlings/Loos...	Loose Silky-bent
905	/content/drive/MyDrive/v2_plant_seedlings/Clea...	Cleavers
3980	/content/drive/MyDrive/v2_plant_seedlings/Shep...	ShepherdтАЩs Purse
235	/content/drive/MyDrive/v2_plant_seedlings/Blac...	Black-grass

4823 rows × 2 columns

Fig. 4.2.1. Storing File path and class labels in a data frame.

values. Fig. 4.1.7 displays the values entered by a user such as rainfall as 500 and soil Ph as 19.

The soil ph. value cannot be 19 as it does not lie in the range between 0–14. Hence the system displays the message stating to enter correct values as shown in Fig. 4.1.8

4.2. Weed identification

This module helps in identifying the weed present in farm and also suggest herbicides for the predicted weed. To predict the Weeds RESNET152V2 pre-trained algorithm was used. It resulted in an accuracy of 0.89.

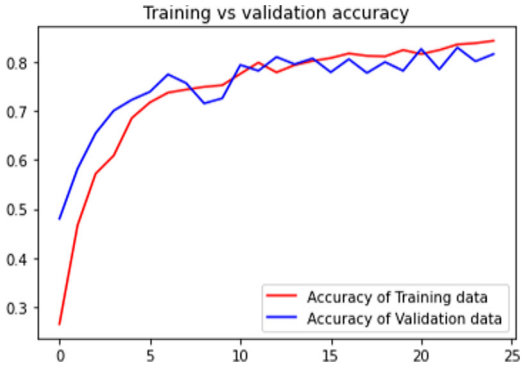

Fig. 4.2.2. Accuracy of training and validation data.

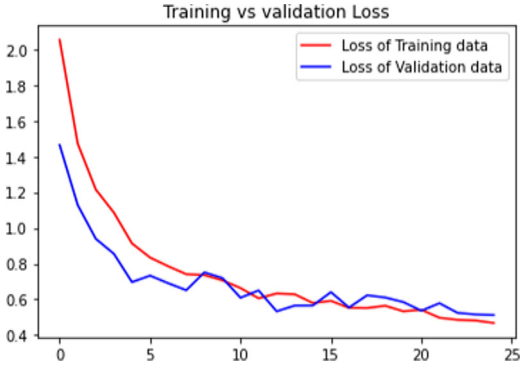

Fig. 4.2.3. Training and validation loss.

Fig. 4.2.1 represents the data frame consisting of file path and class labels. This data frame is shuffled and then split into training and test sets for model building and prediction. After fine tuning the base model Resnet152V2 and adding additional layers, epochs are executed for fitting the training data and validating against the validation data.25 Epochs are obtained with a training accuracy of 83.01% and validation accuracy with 82.52%. Fig. 4.2.2 shows how the training and validation accuracy fluctuated while iterating through each epoch. As the epochs increase the accuracy also increases. Fig. 4.2.3 the loss details are decreasing as the epochs increases. This shows that the model is learning progressively as the epochs increases.

After predicting the weed the next step is to predict the herbicides. In order to achieve this data is collected and stored into a data frame. Fig. 4.2.4 shows the sample data collected for herbicide recommendation.

The details obtained as shown in Fig. 4.2.4 are Weed name, concentration, Herbicide, Crop name, pre-emergent or post emergent, dose, soil type, Group, Weed growth stage in terms of leaves and Weeks. Concentration specifies the active ingredients present in the herbicide, followed by the herbicide name. Herbicides can be broadly classified into two types pre-emergent and post emergent herbicides' -emergent herbicides corresponds to application of herbicide in the field before cultivation. Weeks's column is related to pre-emergent which specifies the number of weeks prior to cultivation the herbicides need to be applied. Post-emergent herbicide specifies the application of herbicides once after the weed is been spotted on the field. Weeds needs to be killed during its initial stage. Hence Weed_stage_no_of_leaves specify the time to apply the herbicide based on the number of leaves grown.

The weed names and the crop names are non-numerical in nature hence it needs to be converted into numerical format for the model to use it. These two attributes are been Label encoded. The corresponding numerical values for these attributes chosen as the independent variables. The encode values details glimpse are shown in Fig. 4.2.5.

	Weed	Concentration	Herbicide	Crop	Pre_or_post_emergent	Dose mL/ha	Soil type	Group	Weed_stage_no_of_leaves	Weeks
0	black-grass	Flufenacet	Adama Herbicide	Wheat	Pre_emergent	0.3	Silty clay loam	12	0	4
1	black-grass	Flufenacet	Adama Herbicide	Wheat	Post_emergent	0.3	Silty clay loam	15	0	0
2	black-grass	Flufenacet	Adama Herbicide	Barley	Pre_emergent	0.3	Loamy Black soil	12	0	4
3	black-grass	Flufenacet	Adama Herbicide	Barley	Post_emergent	0.3	Sandy loam	12	0	0
4	black-grass	prosulfocarb	Clodinafop-propargyl	Wheat	Post_emergent	120.0	Silty clay loam	N	3	0

Fig. 4.2.4. Herbicide dataset details.

label	encoded
black-grass	0
charlock	1
cleavers	2
common chickweed	3
fat hen	4
loose silky-bent	5
scentless mayweed	6
sheperd's purse	7
small-flowered cranesbill	8

label	encoded
Apple	0
Barley	1
Beans	2
Blackberry	3
Cabbage	4
Carrot	5

label	encoded
Adama Herbicide	0
Alister Flex	1
Atlantis WG	2
Attribut	3
Avadex	4
Avadex Liquid	5
Avadex Microactiv	6
Axial 100 EC	7

Fig. 4.2.5. Encoded values for weeds, crops and Herbicide names.

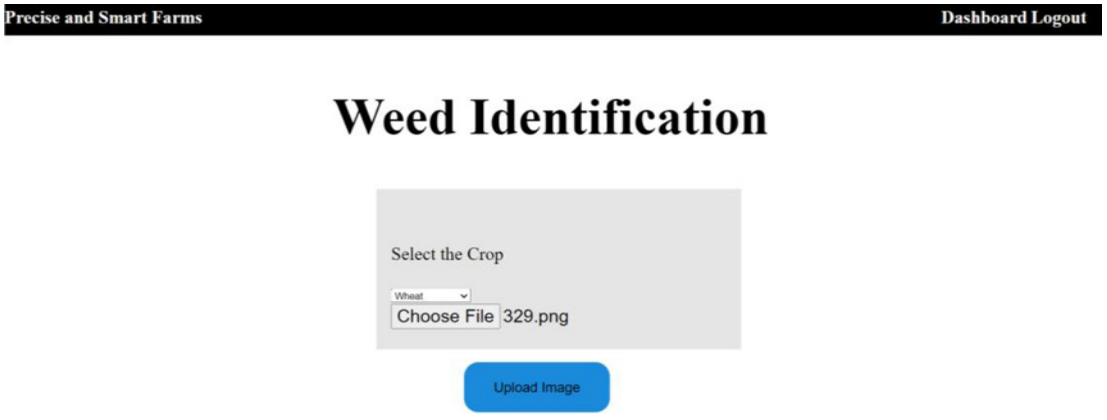

Fig. 4.2.6. Page to upload the image.

Herbicide attribute is also label encoded to obtain numerical values and are used as dependent variable. The dataset after Label encoding are split into two parts training and testing data. The training data is been fed into the random forest classifier model for prediction. Once the model learns the data the model is fed with testing data. The model is been stored into a pickle which is used for further analysis. Once the user specifies the crop name that he wants to cultivate and uploads the image of the weed found on the field the system predicts the Herbicide and displays the relevant details of the Herbicide.

Frontend:

For the user to identify the weed and get details of herbicides the user needs to specify the crop name and upload the image captured from the field as shown in Fig. 4.2.6. Fig. 4.2.7 shows a black grass image. This image is uploaded into the system and crop—Wheat is been selected.

Fig. 4.2.8 shows the description about the weed. The model has precisely predicted that it is a black grass. It shows the growing season

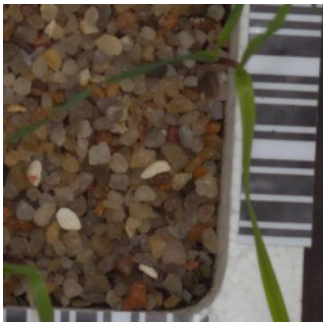

Fig. 4.2.7. Uploaded image is Black grass weed.

Description	Black-grass is a native annual grass weed that occurs mainly in the cereal growing areas and it is most abundant in winter crops. It is chiefly confined to heavy land, occurring only occasionally on sandy or gravely soil.Black-grass suffers from ergot (Claviceps purpurea) and this can result in contamination of the grain at harvest.		
Growing Season	September-October		
Herbicide	['Alister Flex']	['Cleranda']	['Broadway Star']
Concentration	['diflufenican 150 OD iodosulfuron-methyl-natrium 3 mesosulfuron-methyl 9 mefenpyr-diethyl (S)']	['picolinafin,pendimethalin,2,4-D,mecoprop-P,diflufencian']	['pyroxsulam 68.3 WG florasulam 22.8 clorquintocet-mexyl (S)']
Post/Pre emergence herbicides	['Post_emergent']	['Post_emergent']	['Post_emergent']
Dose	[100.]	[200.]	[200.]
Suited Soil type	['Loamy sand' 'Sandy Loam' 'Clayey Loam' 'Sandy loam' 'clay loam' 'Loamy Black soil' 'Moist soil']	['Sandy Loam' 'Loamy Black soil' 'Loamy soil' 'Clayey Loam']	['Clayey Loam' 'Sandy Loam']
Group	['17' '13']	['19']	['B']
Weed leaves	[5]	[9]	[3 0]
weeks	[0]	[0]	[0 3]

Fig. 4.2.8. Predicted details.

of the weed is from September–October. The Herbicides are listed with is an obtained by-passing crop name and weed name into Random Forest Classifier Model. Furthermore, the system categorizes it as pre-post emergent herbicide followed by its dosage mL/ha and suited soil types are also provided.

The Group represents the chemical solution that the herbicide belongs to. The weed leaves are the leaves stages during which the user is directed to spray the herbicides if it is a post-emergent. If Pre-emergent the system specifies the number of weeks before the herbicide needs to be sprayed.

4.3. Pesticides identification

This module helps in identifying the insects and pests present in farm and also suggest pesticides for the predict insect. To predict the Weeds RESNET152V2 pre-trained algorithm was used. It resulted in an accuracy of 0.98.

Fig. 4.3.1 represents the data frame consisting of file path and class labels. This data frame is shuffled and then split into training and test sets for model building and prediction. After fine tuning the base model Resnet152V2 and adding additional layers, epochs are executed for fitting the training data and validating against the validation data.20 Epochs are obtained with a training accuracy of 83.01% and validation accuracy with 98.2%. Fig. 4.3.2 shows how the training and validation accuracy fluctuated while iterating through each epoch. As the epochs increase the accuracy also increases. Fig. 4.3.3 the loss details are decreasing as the epochs increases. This shows that the model is learning progressively as the epochs increases.

After predicting the insect, the next step is to predict the pesticides. In order to achieve this data is collected and stored into a

	FilePath	Labels
2001	/content/drive/MyDrive/pest/train/grasshopper/...	grasshopper
943	/content/drive/MyDrive/pest/train/mites/jpg_91...	mites
1611	/content/drive/MyDrive/pest/train/beetle/jpg_6...	beetle
403	/content/drive/MyDrive/pest/test/aphids/jpg_15...	aphids
1301	/content/drive/MyDrive/pest/train/armyworm/jpg...	armyworm
...	...	...
2763	/content/drive/MyDrive/pest/train/bollworm/jpg...	bollworm
905	/content/drive/MyDrive/pest/train/mites/jpg_40...	mites
1096	/content/drive/MyDrive/pest/train/armyworm/jpg...	armyworm
235	/content/drive/MyDrive/pest/test/grasshopper/j...	grasshopper
1061	/content/drive/MyDrive/pest/train/armyworm/jpg...	armyworm

3150 rows × 2 columns

Fig. 4.3.1. Storing File path and class labels in a data frame.

data frame. Fig. 4.3.4 shows the sample data collected for pesticide recommendation.

The details obtained as shown in Fig. 4.3.4 are Pest name, Crop name and pesticide details. The pest names and the crop names are non-numerical in nature hence it needs to be converted into numerical format for the model to use it. These two attributes are been Label encoded. The corresponding numerical values for these attributes chosen as the independent variables. The encode values details glimpse are shown in Fig. 4.3.5. Pesticide attribute is also label encoded to obtain

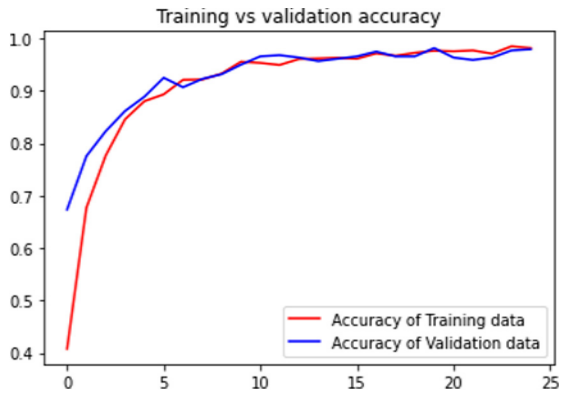

Fig. 4.3.2. Accuracy of Training and Validation Data.

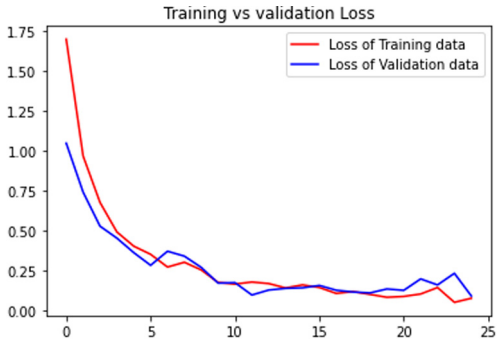

Fig. 4.3.3. Training and validation loss.

	Pest	Crop	Pesticide
0	aphids	Bean	Acephate 90 WDG
1	aphids	Sprouts	Acephate 90 WDG
2	aphids	Cauliflower	Acephate 90 WDG
3	aphids	Lettuce	Acephate 90 WDG
4	aphids	Peppers	Acephate 90 WDG

Fig. 4.3.4. Herbicide dataset details.

numerical values and are used as dependent variable. The dataset after Label encoding are split into two parts training and testing data. The training data is been fed into the random forest classifier model for prediction. Once the model learns the data the model is fed with testing data. The model is been stored into a pickle which is used for further analysis. Once the user specifies the crop name that he wants to cultivate and uploads the image of the insect found on the field the system predicts the Pesticide.

Frontend:

For the user to identify the pest and get details of pesticides the user needs to specify the crop name and upload the image captured from the field as shown in Fig. 4.3.6. Fig. 4.3.7 shows a stem borer image. This image is uploaded into the system and crop—Rice is been selected.

Fig. 4.3.8 shows the pesticides that can be used for killing stem borer, we can identify the image of the pest that the pest is same color of the stem. But the model proposed has precisely predicted that it is a stem borer pest.

4.4. Cost of cultivation

For this work Cost of cultivation data from 2010–2018 is considered. 11 Crops are been selected and are considered for analysis. They include Arhar, Bajra, Cotton, Groundnut, Jowar, Ragi, Paddy, Sugarcane, Maize, Potato and Wheat. All these attributes were fit into Regression models especially ensemble regression models such as XGBoost, GradientBoostingRegressor, Bagging Regressor and Decision Tree Regressor the models were predicted against each crop and state for the years starting from 2010–2028.

Considering in identifying a best model for Operational cost Fertilizer the following results are obtained.

The following algorithms listed in Fig. 4.4.1 were implemented to find the best model based on the r2 score. The algorithm with highest r2score is been opted as the model for fertilizer cost forecast's same algorithm is not implemented for other attributes. For every attribute 9 Regressor algorithms are executed and the model with highest r2 score is selected for them. For all the attributes mostly XGBoost Regressor, Random Forest Regressor and Bagging Regressor performs best hence they are saved as a pickle file and then used for forecasting values till year 2028.

These results were aggregated to obtain Operational Cost, Total Cost and Fixed Cost. In addition, these values were used for calculating Cost A1, A2, B1, B2, C1, C2 and C3.

In order to estimate the cost of cultivation the following columns can be split into two parts Operational Cost and Fixed Cost. Operational Cost consists of all the expenses such as purchase of seed, insecticides, animal labor, human labor, machine labor, cost on fertilizer and manure. And Fixed cost consists of fc rental value of owned land, FC Rent Paid for Leased in Land, FC Land revenue tax ceases, Operational Cost of Interest On working capital, Depreciation on Implements Farm Building Interest on Fixed Capital.

Cost Concepts splits the various expenses into 7 groups such as Cost A1, A2, B1, B2, C1, C2 and C3. Based on these formulas all the details are summed together to get the operational and fixed cost value. Later operational cost and fixed cost value are summed that results in total cost.

The user needs to specify the crop name and the select a state of India to get the results. The 11 crops and their corresponding states that the crop is grown is been listed in the drop down list as shown in Fig. 4.4.2.

Fig. 4.4.3 shows a glimpse of the forecast values till the year 2028 for all the attributes. These attributes are calculated based on the formulas in Fig. 3.4.5a.

Fig. 4.4.4 depicts the operational cost variation over the years 2010–2028. It is been observed at specific intervals there is sudden drift in the operational cost. It is been also observed at the cost are almost the cost reaches to 45000. Each year data consists of details such as Operational Cost of Attached, Casual, hired human Labor, hired and owned animal Labor, hired and owned Machine Labor, cost on Fertilizer, Manure, Insectides, Irrigation, crop Insurance, Payment to Contractor and Miscellaneous Cost. These data are summed up to obtain the operational cost.

In Fig. 4.4.5, with respect to fixed costs we have rental value of owned land, Rent Paid for Leased in Land, Land revenue tax and ceases, Depreciation on Implements Farm Building and Interest on Fixed Capital. All these attributes are forecasted and these values are summed up to obtain Fixed cost. The fixed cost shows an increase over the years till 2015. Followed by a drift in amount and forecasting states that it would remain the same at 13000.

Fig. 4.4.6 depicts the Total cost for cultivation. Total cost is the summation of Operational cost and Fixed cost. Since the operational cost and fixed cost are same after years. The total cost specifies the cost would be 65000.

Fig. 4.4.7 depicts the cost concepts cost A1. Concepts are mainly used by business units since it splits the expenses into each parameter.

encoded label	label
aphids	0
armyworm	1
beetle	2
bollworm	3
grasshopper	4
mites	5
mosquito	6
sawfly	7
stem_borer	8

label	
Almonds	0
Apple	1
BROCCOLI	2
Bananas	3
Barley	4

label	
ALPHA CYPERMETHRIN 250SC	0
Acephate 90 WDG	1
BIFENTHRIN 100 EC	2
Dimethoate 400 EC	3
Dursban WG	4
Guthion®	5
ORGANOPHOSPHATE	6
PYRETHRIN 7EC	7

Fig. 4.3.5. Encoded values for weeds, crops and Pesticide names.

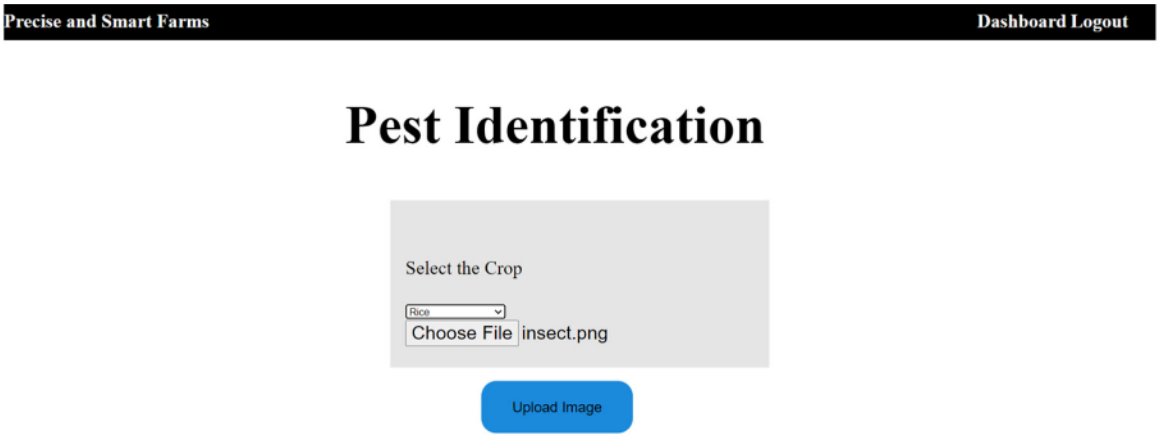

Fig. 4.3.6. Page to upload the image.

Fig. 4.3.7. Uploaded image is Stem borer.

This can be used for best decision making. Cost A1 has sudden peaks over the years resulting in the value of 38000 over the years.

Fig. 4.4.8 depicts the cost concepts cost A2. This cost value focuses on all the expenses along with the fixed cost on leased land.

Fig. 4.4.9 depicts the cost concepts cost B1. This cost value focuses on all the expenses along with interest on fixed capital.

Fig. 4.4.10 depicts the cost concepts cost B2. This cost value focuses cost B1 along with the rental value on owned land and rent on leased land.

Fig. 4.4.11 depicts the cost concepts cost C1. This cost value focuses cost B1 along with the imputed value of family labor. This value is minimum value of family labor.

Fig. 4.4.12 depicts the cost concepts cost C2. This cost value focuses cost B2 along with the imputed value of family labor. This value is minimum value of family labor.

Fig. 4.4.13 depicts the cost concepts cost C3. This cost value focuses cost C2 along 10 percent cost of C2 as management cost.

Based on these details farmers and business executives would be able to make things ready before cultivation.

5. Conclusion

Farming is a back bone of every country. Hence it needs to be monitored in a timely manner. The modules in the work provides a helping hand to farmers for identifying the crops that can be grown based on their place. Identifying the weeds and recommending herbicides is an important element. All the crops are prone to insects. Hence identifying the correct insect and recommending the pesticides for the same would be an efficient tool. Many farmers are unable to estimate the cost of cultivation. Due to some uncertainties, there might be loss for the farmers. It also provides estimation of cost of cultivation for each operation such as human labor, animal labor, cost of seeds, manures and fertilizers. It also forecasts the fixed costs. This provides an overview of how to plan the activities and do cultivation in a profitable manner.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Pest Identification and Pesticides Recommendation

Fig. 4.3.8. Pesticides recommendation.

	Algorithm	reg_name	train_Score	test_Score	r2score	mse	rmse	mae
0	linear Regression	lr	0.128958	0.155050	0.155050	1.381732e+07	3717.165299	2596.985101
1	DecisionTreeRegressor	dtr	0.998149	0.639880	0.639880	5.888976e+06	2426.721220	1343.692693
2	lasso	lasso	0.128958	0.155019	0.155019	1.381782e+07	3717.232959	2596.973486
3	Ridge	ridge	0.128958	0.155032	0.155032	1.381761e+07	3717.204336	2596.993632
4	BaggingRegressor	bagging	0.959432	0.602473	0.602473	6.500686e+06	2549.644350	1375.660185
5	RandomForestRegressor	rr	0.872400	0.582328	0.582328	6.830120e+06	2613.449759	1315.669355
6	AdaBoostRegressor	ar	0.398870	0.269382	0.269382	1.194767e+07	3456.540498	2569.057956
7	GradientBoostingRegressor	gr	0.799545	0.665001	0.665001	5.478177e+06	2340.550665	1416.091017
8	XGBoostRegressor	xg	0.796465	0.673183	0.673183	5.344378e+06	2311.791126	1454.907716

Fig. 4.4.1. Model performance details.

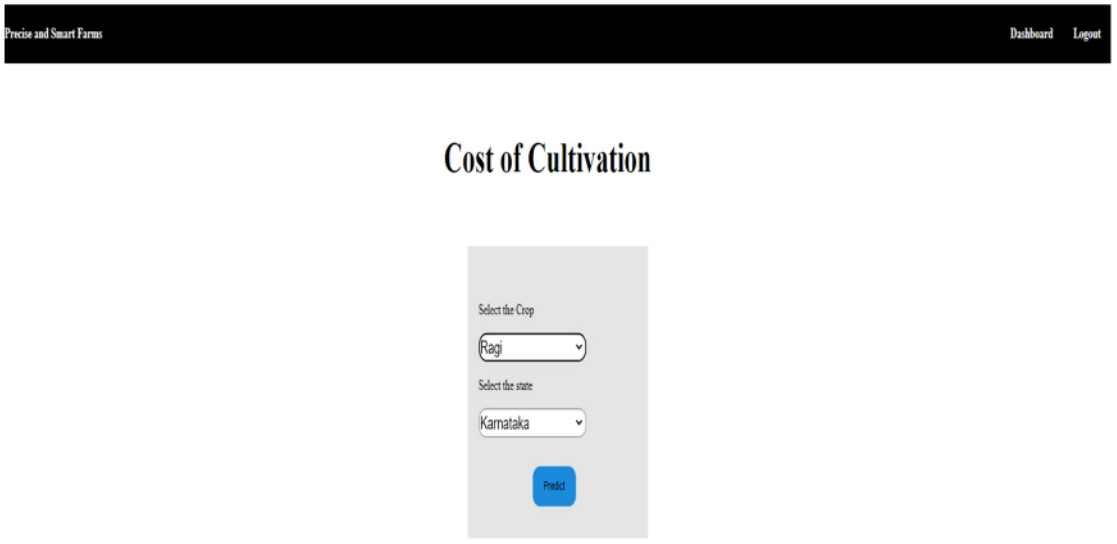

Fig. 4.4.2. Specifying the crop and state details for forecasting the cost.

Appendix. Supplementary material

Module 1: Crop Recommendation
https://www.kaggle.com/atharvaingle/crop-recommendation-dataset
https://www.kaggle.com/shekharyada/crop-soilcsv
https://www.kaggle.com/aj021977/crop-names

Module 2: Weed Identification
https://www.kaggle.com/vbookshelf/v2-plant-seedlings-dataset

Module 3: Pest Identification
https://www.kaggle.com/simranvolunesia/pest-dataset

Module 4: Cost Estimation
https://eands.dacnet.nic.in/Cost_of_Cultivation.htm

Cost of Cultivation Details

Year	2010	2011	2012	2013	2014	2015	2016	2017	2018
Family Labour	7766.86	10249.75	8025.26	11678.869999999999	14438.58	15012.79	15012.79	17028.3	12255.62
Animal_Labour	2560.35	5623.709999999999	5550.549999999999	5347.64	5478.79	2101.19	2101.19	12147.779999999999	11174.75
Machine_Labour	2786.6600000000003	3254.41	2997.72	2370.3900000000003	2112.81	1064.64	1064.64	5559.4400000000005	4156.41
Fertilizer_and_Manure	2325.379526102751	2114.647535639244	3157.100670992675	3751.981699217822	9507.547020332375	9507.547020332375	6623.0770203323755	6813.536330760619	5601.326551789704

2019	2020	2021	2022	2023	2024	2025	2026	2027	2028
12255.62	12255.62	12255.62	12255.62	12255.62	12255.62	12255.62	12255.62	12255.62	12255.62
11174.75	11174.75	11174.75	11174.75	11174.75	11174.75	11174.75	11174.75	11174.75	11174.75
4156.41	4156.41	4156.41	4156.41	4156.41	4156.41	4156.41	4156.41	4156.41	4156.41
5601.326551789704	5601.326551789704	5601.326551789704	5601.326551789704	5601.326551789704	5601.326551789704	5601.326551789704	5601.326551789704	5601.326551789704	5601.326551789704

Fig. 4.4.3. Cost of Cultivation details till 2028.

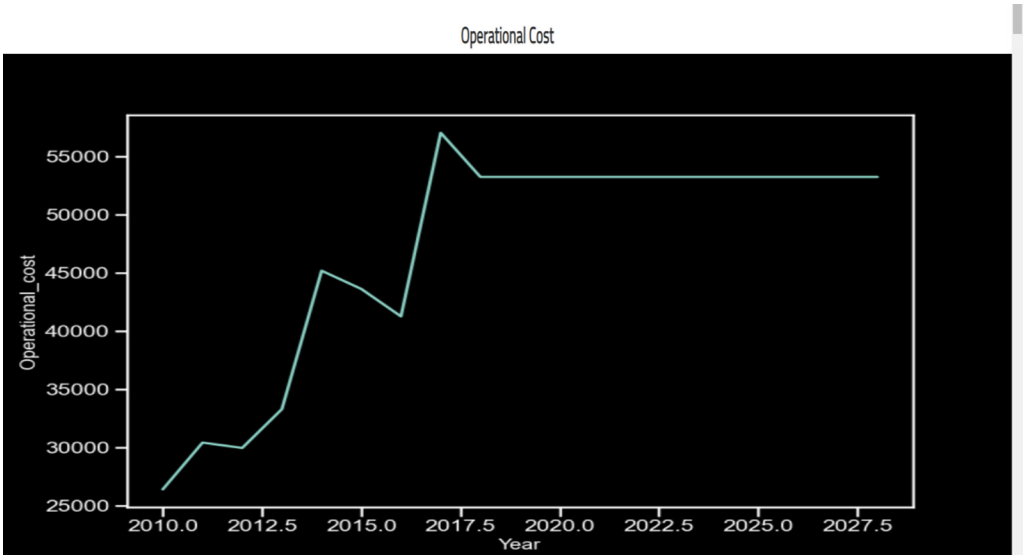

Fig. 4.4.4. Operational cost.

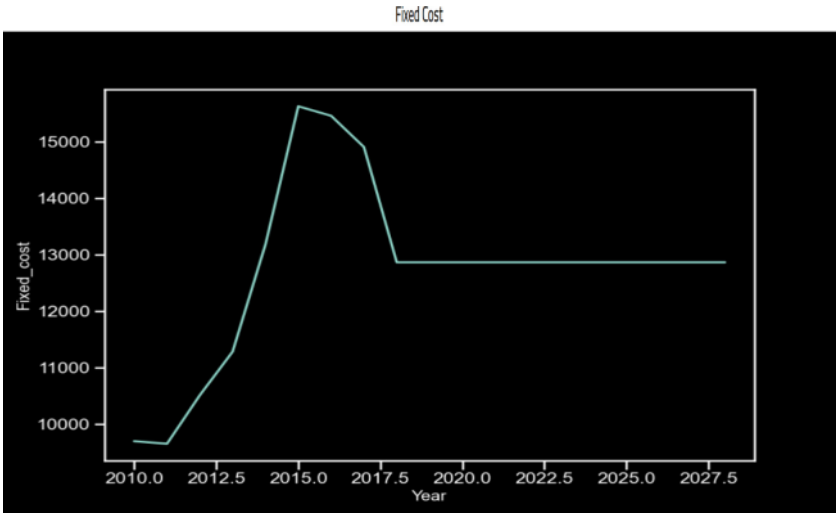

Fig. 4.4.5. Fixed cost.

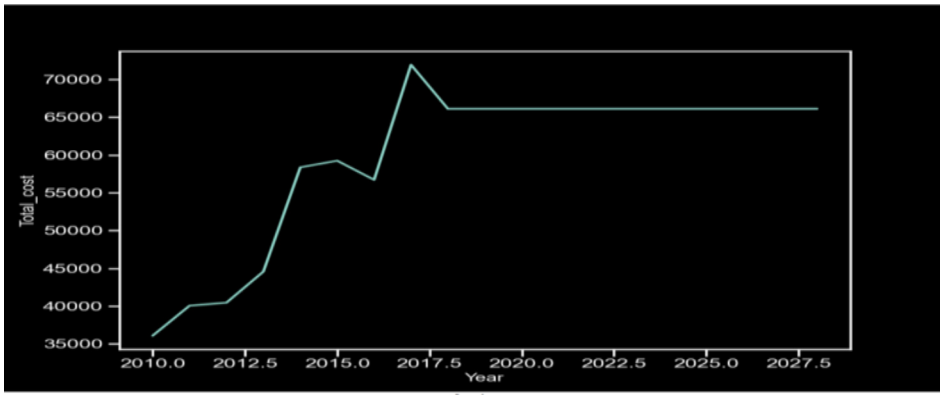

Fig. 4.4.6. Total cost.

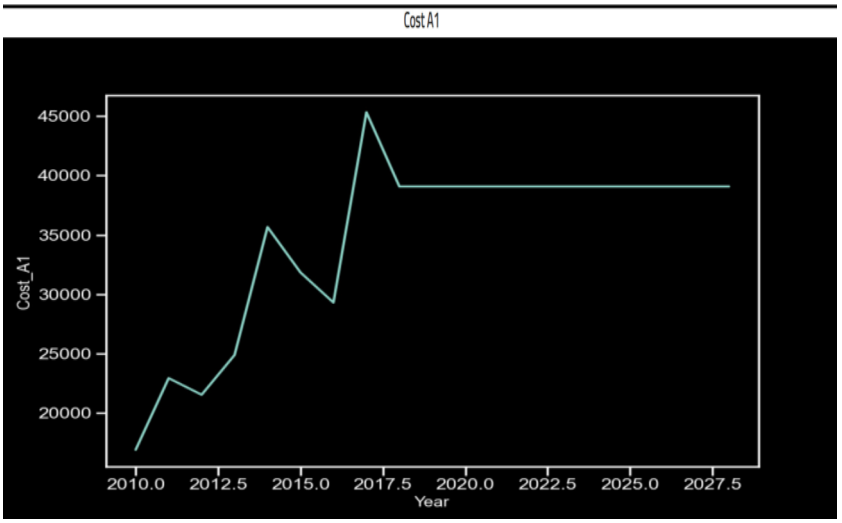

Fig. 4.4.7. Cost A1.

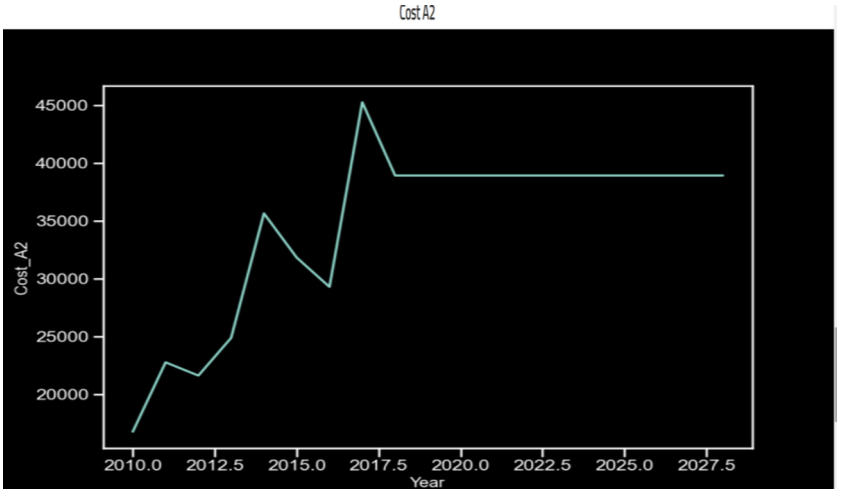

Fig. 4.4.8. Cost A2.

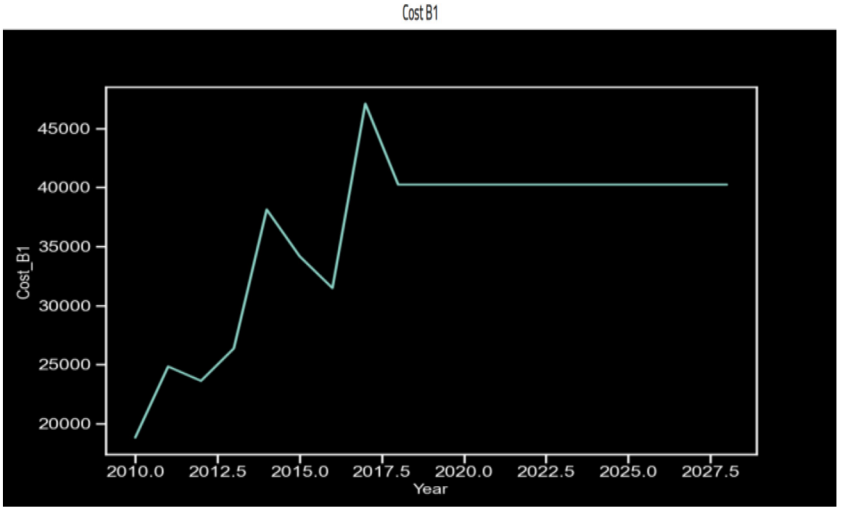

Fig. 4.4.9. Cost B1.

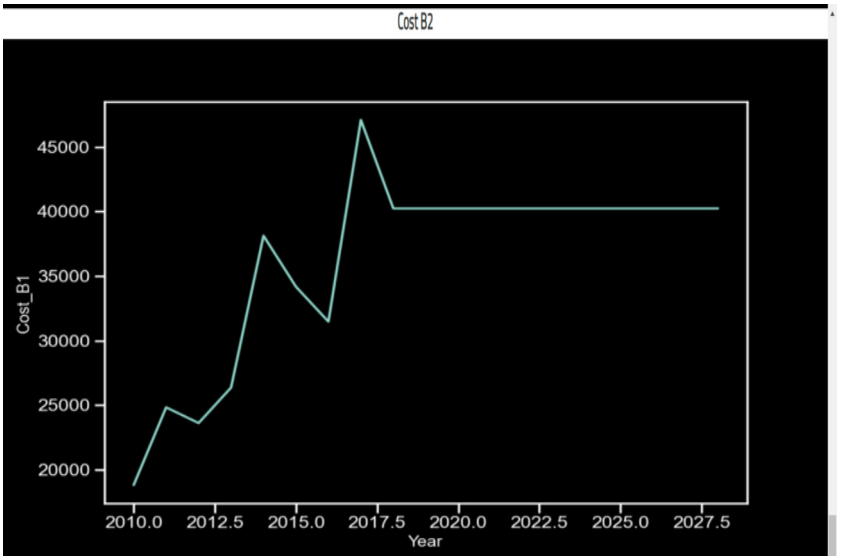

Fig. 4.4.10. Cost B2.

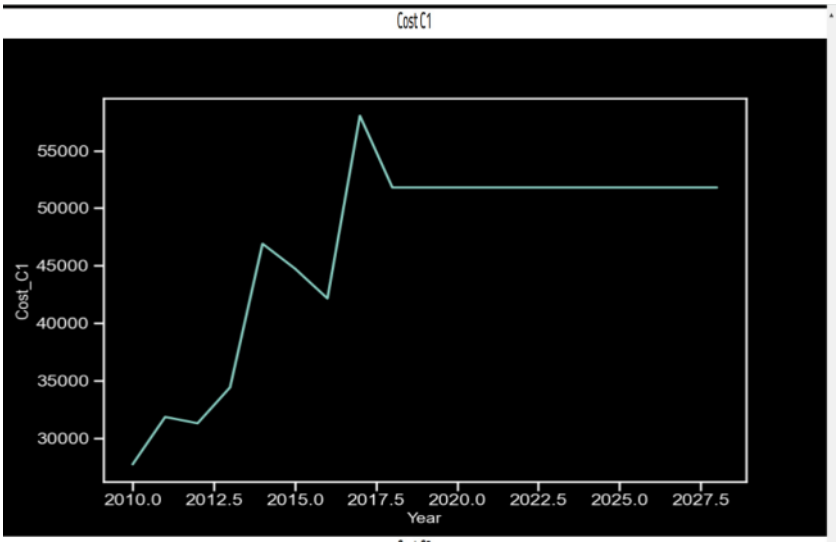

Fig. 4.4.11. Cost C1.

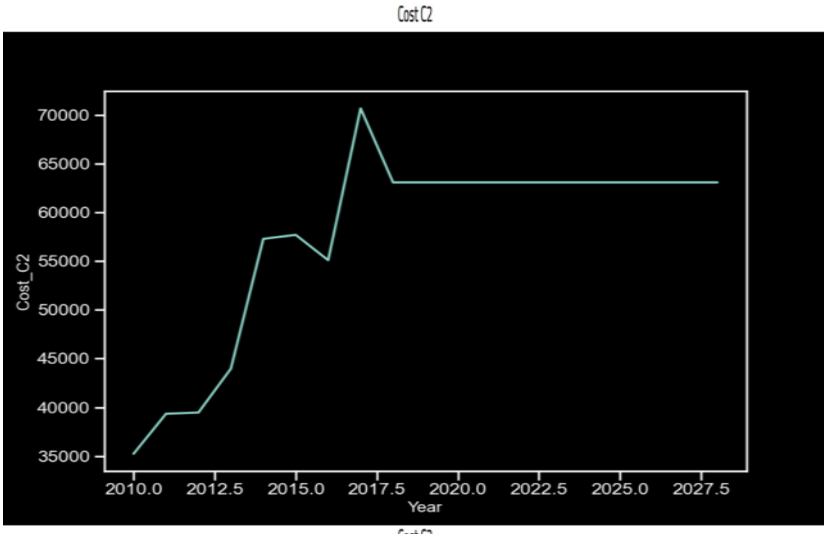

Fig. 4.4.12. Cost C2.

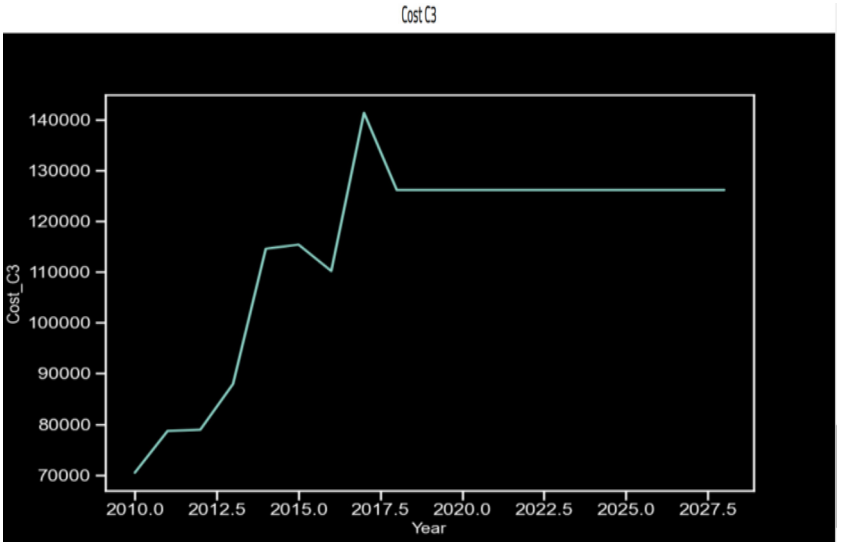

Fig. 4.4.13. Cost C3.

References

[1] Margit Paustian, Ludwig Theuvsen, Adoption of Precision Agriculture Technologies by German Crop Farmers, Springer, 2016.

[2] Tanja Groher, Katja Heitkämper, Achim Walter, Frank Liebisch, Christina Umstätter, Status Quo of Adoption of Precision Agriculture Enabling Technologies in Swiss Plant Production, Springer, 2020.

[3] Mohammed Riyadh Abdmezie, D. Tandjaoui, Imed Romdhani, Architecting the Internet of Things:State of the Art, Springer, 2016.

[4] Ravesa Akhter, Shabir AhmadSofi, Precision Agriculture using IoT Data Analytics and Machine Learning, Elsevier, 2021.

[5] Abhinav Sharma, Arpit Jain, Prateek Gupta, Vinay Chowdary, Machine Learning Applications for Precision Agriculture: A Comprehensive Review, IEEE, 2020.

[6] Andreas Kamilaris, F. Prenafeta-Boldu, Deep Learning in Agriculture: A Survey, Elsevier, 2018.

[7] Mohamed Torky, Abdoul Ella Hassanein, Integrating Blockchain and the Internet of Things in Precision Agriculture: Analysis, Opportunities, and Challenges, Elsevier, 2020.

[8] Sam Cramer, Michael Kampouridis, Alex A. Freitas, Antonis K. Alexandridis, An Extensive Evaluation of Seven Machine Learning Methods for Rainfall Prediction in Weather Derivatives, Elsevier, 2017.

[9] Jenifer L. Yost, Alfred E. Hartemink, How Deep Is the Soil Studied – An Analysis of Four Soil Science Journals, Springer, 2019.

[10] M.S. Suchithra, Maya L. Pai, Improving the Prediction Accuracy of Soil Nutrient Classification by Optimizing Extreme Learning Machine Parameters, Elsevier, 2020.

[11] Bright Keswani, Ambarish G. Mohapatra, Poonam Keswani, Ashish Khanna, Deepak Gupta, Joel Rodrigues, Improving Weather Dependent Zone Specific Irrigation Control Scheme in IoT and Big Data Enabled Self Driven Precision Agriculture Mechanism, Taylor and Francis, 2020.

[12] Borja Espejo-Garcia, Nikos Mylonas, Loukas Athanasakos, Spyros Fountas, Ioannis Vasilakoglou, Towards Weeds Identification Assistance Through Transfer Learning, Elsevier, 2020.

[13] Junde Chen, Jinxiu Chen, Defu Zhang, Yuandong Sun, Y.A. Nanehkaran, Using Deep Transfer Learning for Image-Based Plant Disease Identification, Elsevier, 2020.

[14] Ritesh Dash, Dillip Ku Dash, G.C. Biswal, Classification of Crop Based on Macronutrients and Weather Data using Machine Learning Techniques, Elsevier, 2021, p. 8.

[15] A. Priyadharshini, Swapneel Chakraborty, Aayush Kumar, O. Rajendra Pooniwala, Intelligent crop recommendation system using machine learning, 2021, p. 6.

[16] Senshan Yang, Joanne Logan, David L. Coffey, Mathematical Formulae for Calculating the Base Temperature for Growing Degree Days, Elsevier, 1995.

[17] JingLei Tang, Dong Wang, ZhiGuang Zhang, LiJun He, Jing Xin, Yang Xu, Weed Identification Based on K-Means Feature Learning Combined with Convolutional Neural Network, Elsevier, 2017.

[18] A.S.M. Mahmudul Hasan, Ferdous Sohel, Dean Diepeveen, Hamid Laga, Michael G.K. Jones, A Survey of Deep Learning Techniques for Weed Detection from Images, Elsevier, 2021.

[19] Xiaojun Jin, Jun Che, Yong Chen, Weed Identification using Deep Learning and Image Processing in Vegetable Plantation, IEEE, 2021.

[20] Thenmozhi Kasinathan, Dakshayani Singaraju, Srinivasulu Reddy Uyyala, Insect Classification and Detection in Field Crops using Modern Machine Learning Techniques, Elsevier, 2021.
[21] Zakaria Saoud, Can We Estimate Insect Identification Ease Degrees from their Identification Key Paths? Elsevier, 2020.
[22] Susheela Meena, I.P. Singh, Ramji Lal Meena, Cost of Cultivation and Returns on Different Cost Concepts Basis of Onion in Rajasthan, ND, 2016.
[23] Amit Mandal, Varying Profitability and Determinants of Gram Crop using Cost of Cultivation Data: A Fixed Effect Approach, Springer, 2021.