Chapter 4
Predictive Farmland Optimization and Crop Monitoring Using Artificial Intelligence Techniques

Antara Sahoo, Aniket Rathi, Shambhavi Bashishth, Sanghamitra Roy, and Chittaranjan Pradhan

Abstract India's economy is vastly affected by agriculture. Illiteracy has been a major concern among farmers which pulls them from identifying the crops that are perfect for their land and determining the type of diseases caused. Checking for diseases without any prior knowledge may result in overlooking some of the yields. The lack of awareness about certain crop diseases, land conditions and fertilizer has resulted in big losses in the past. Hence, the thought of automating the whole process using machine learning has been initiated. Crop selection, fertilizer selection and early disease prediction have been automated. In this research work, for crop yield and fertilizer prediction, various machine learning algorithms such as Support Vector Machine, Logistic Regression Random Forest, K-Nearest Neighbors and Artificial Neural Network (ANN) are used. The random forest algorithm for crop yield prediction gave an accuracy of 95.22% which is better compared to other algorithms, while in fertilizer prediction accuracy of 96% is achieved by ANN. The disease prediction model has been created using image datasets of wheat, maize and apple which were leveraged using convolution neural networks resulting in an average accuracy close to 98.5%.

Keywords Artificial intelligence · Crop monitoring · Deep learning · Fertilizer prediction · Image processing · Machine learning

A. Sahoo (✉) · A. Rathi · S. Bashishth · S. Roy · C. Pradhan
Kalinga Institute of Industrial Technology, Bhubaneswar, India

M. A. Ahad et al. (eds.), *Enabling Technologies for Effective Planning and Management in Sustainable Smart Cities*, https://doi.org/10.1007/978-3-031-22922-0_4

4.1 Introduction

Machine learning has made a significant impact in all industries and agriculture sectors. Several sectors in each domain try to use past data and apply intelligent machine learning algorithms for prediction and analysis purposes. With a value of INR 56,564 billion in 2019, the agriculture industry plays a vital role in the nation's economy in terms of employment and contribution to GDP. Moreover, with 18% of the world's population, the demand for agro-products has increased year by year [1, 2].

Farmers encounter some obstacles in using conventional methods of farming:

- In the agriculture life cycle, climatic factors like temperature, rainfall, and humidity are crucial. The rise in deforestation and pollution are leading to climatic changes, so it's increasingly demanding for farmers to determine how to prepare the soil, sow seeds, and harvest. India is a land with varying temperature ranges and rainfall levels, which play an essential factor in farm management. This shows how a variety of crops can be grown in agricultural fields.
- Every crop demands a specific type of nutrition for the soil. The three major nutrients essential in soil include nitrogen (N2), phosphorus (P) and potassium (K). The inadequacy of nutrients, as well as their excessive use, can lead to harvest failure.
- For crop protection, weeds play a significant role most of the time. If unregulated, it can directly affect crop yield and cease its growth. It can also absorb nutrients from the soil, which can cause a shortage of nutrition. Then preserving damaged plants on large acres of land may require a substantial financial setup.

In our research work, these crucial problems are considered where artificial intelligence algorithms have been applied to enhance agronomic management in India since agriculture is one of the major occupations of many rural Indians. In India, there exist around 394.6 million acres of arable land. So growing more agricultural products helps not only humans but also the entire environment. However, as India is a significant producer of wheat, rice, maize, cereals, etc., growing these in every farming land is impossible due to different climatic conditions. However, this increases the scope of producing varieties of crops with an excellent yield depending upon suitable weather conditions.

The topic of our research is based on agriculture yield amplification. The main aim is to help farmers increase their production with the help of suitable fertilizers and make the farmers aware of what diseases the staple crops of India like rice, wheat, apple and tomato possess. The importance lies in the fact that the farmers are unaware of the various properties of their crops due to a lack of thorough knowledge and are misinformed. This misinformation leads to many faulty courses of action, leading to a lower yield of crops [3]. In this scenario, if the crop is infected, it becomes unwanted and no longer edible, which even hampers nutrients present in the soil. Therefore, they need to destroy the previous ones to develop fresh crops. For a long time in India, farmers have been practicing stubble burning, leading to a

significant contribution to pollution and global warming. Here, an automated model envisages the crop yield above a specific soil type, and periodically checks the plant's health, whether it is infected, and its fertilizer content requirement [4].

With various machine learning techniques, one can feasibly predict the best type of crop that can be cultivated with a given set of parameters [5]. Further, the growth of the product can be moderated by fertilizer prediction and disease prediction models, which will help in the early detection of diseases and suggest a change in the type of fertilizers based on Nitrogen, Phosphorus Potassium (NPK) values and soil type. This will lead to proper crop management and yield amplification. Through AI techniques, the information will be unbiased and farmers will be able to make an informed decision regarding their net production [6]. This will help increase the profit margin of the farmers and hence, will increase the exports leading to an increase in the GDP of India. This research will have a more lasting impact than it currently appears to have.

The recent advancement of technology using the Internet of Things, machine learning and artificial technology is also applied in the agriculture domain for better productivity [7–9]. In our research, we have aimed at providing automation, where, using machine learning and neural network model, farmers can reduce their crop wastage rate. We divided our model into three subparts, i.e., crop prediction, early detection of plant diseases and fertilizer prediction. Our first model focuses on predicting an appropriate crop to be grown depending on climatic conditions and soil contents of the particular area. We have opted for the random forest classifier algorithm with hyper parameter tuning which helped improve the accuracy of the model and reduce the loss. Our next model is to identify any disease a crop can have before excessive damage using Convolutional Neural Network (CNN). CNNs are actively used for image detection purposes, and by feeding the picture of the diseased crop into the CNN model, through various layers of convolutional, dropout and pooling mechanisms, our model will identify the type of disease. Also, we have used the softmax activation function to reach maximum efficiency and reduced error. Consequently, after the disease is identified, our third model will suggest an appropriate kind of fertilizer that can be used to improvise the content of the rest of the crops depending on the nutrient content present in the soil. We have incorporated an Artificial Neural Network (ANN) model using Adam as an optimizer and regularization techniques to enhance the learning rate for predicting new inputs.

Using various algorithms, we did construct our model, whose detailed workflow is presented in the subsequent sections of the chapter. Section 4.2 contains the Literature Review on farmland optimization and crop monitoring techniques presented previously. Section 4.3 has an elaborated analysis of Related Works corresponding to our current research presented. Section 4.4 contains the Proposed Model which elicits a qualitative and quantitative analysis of our machine learning and ANN techniques proposed in this research. Section 4.5 has a Result Analysis that summarizes all our observations (subjective as well as pictorial) and computes various outputs based on tuning parameters of the algorithms. Section 4.6 contains the Conclusion that briefs on the three connected models that aid farmers to reduce crop wastage.

4.2 Literature Review

Agriculture, being a crucial sector of India, serves as the backbone of the economic system. The prediction of various crops under suitable conditions has been highlighted using data mining techniques [1]. The growth rate of three major crops in India is shown in Fig. 4.1 [2]. With exponential progress in agriculture, Singh et al. [3] reported a very sensitive issue of burning stubble and grasslands to control the growth of weeds, insects, plant infections, and excess crop residues that were vigorously practiced earlier. Even now in some parts of India, the stubble burning practice is still prevalent. Due to this quick, inexpensive and effective method, ease of seeding and other soil operations have been enhanced at a large scale. It is also the fact that it takes one-and-half months to decompose the wheat residue left by harvesters and when farmers don't have sufficient time to wait for sowing their fresh crops, stubble burning is practiced to prepare a new soil bed for their crops. But the dangers of burning wheat stubble can lead to major threats to the environment. This can directly harm the atmosphere, and indirectly the plants and humans as depicted in Fig. 4.2 [10].

The flaming can cause soil nutrient loss of organic carbon, nitrogen, phosphorous, and potassium and also deteriorate the ambient air quality [3]. Not only wheat but also burning of other agricultural residues discharge various trace gases like methane, oxides and ample quantities of suspended particulate matter causing adverse effects to humans. Till now there are many large-scale applications to tackle this deterioration. Many ways to conserve agriculture, mainly wheat-maize [5], can be effectively practiced if crop residue management plans are developed taking into consideration the demand, quality, feasibility, and economics of residue management which serves as an efficient way to preserve land. Figure 4.3 shows how important it is to take care of these leading crops like rice, wheat and maize due to their burning demands and exports. Still, there has been a minute sort of gaps in preserving soil as early as possible or early detection of any infection. Our work is to feature some improvement toward the conservation and preservation of natural soil.

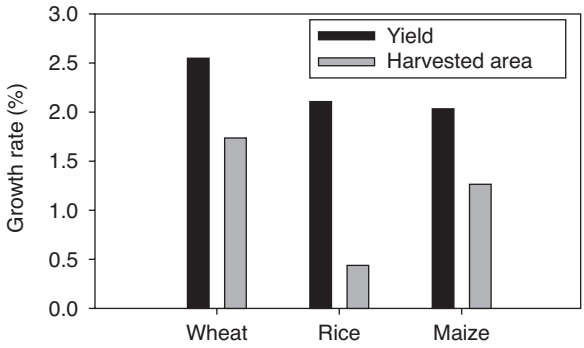


Fig. 4.1 Growth rate of 3 major crops in India [2]

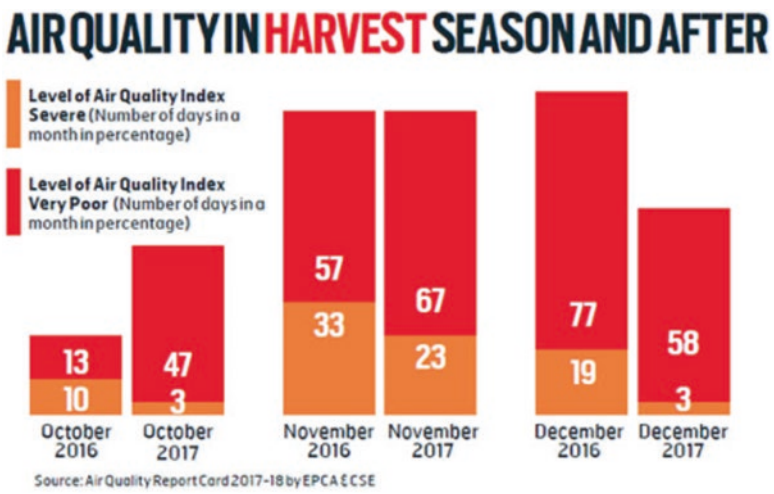


Fig. 4.2 Air quality in and after Harvest [10]

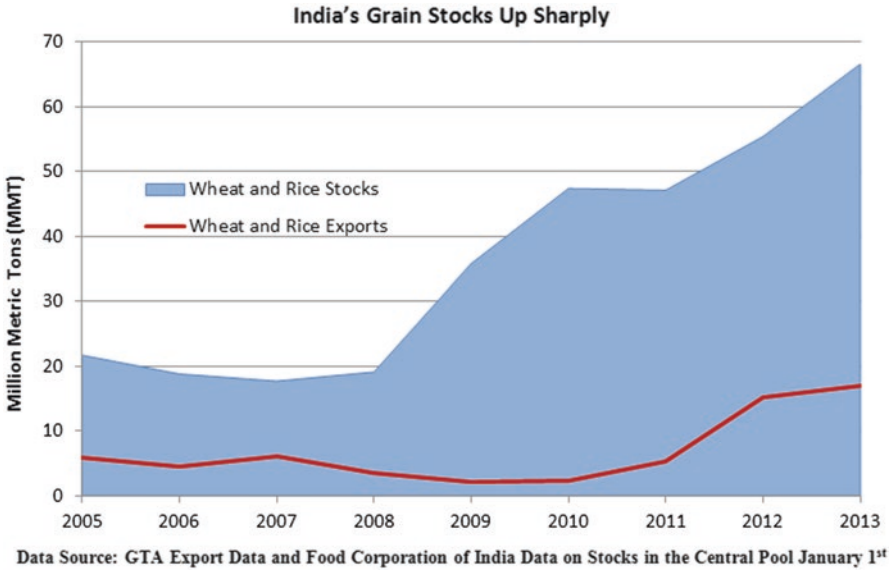


Fig. 4.3 A comparison of the wheat and rice stocks with their respective export

As soon as image processing is done to detect various infections, performing testing of soil samples can break the chain of spoil or destruction at a perfect time. There are many effective techniques to treat soil using suitable fertilizers as per requirements [11]. So this research is about performing tests of soil to know the present type of soil, weather conditions and contents of the fertilizer used. This way one can easily keep track and change fertilizers from time to time for a better

outcome. Literature [11] reported the regulating of fertilizers used after inspecting the current soil content. There exist various efficient tests to find co-integration between consumption of fertilizer and food grain production. Kumar and Indira have reported [12] the effective methods and tests to support the cointegration relation. As far as classifying infection is concerned, there are various recent machine learning approaches and techniques to detect diseases and pests in agricultural products [13].

During the second phase, apple being the cure to everything – a well-accurate model has been set up for early detection and classification of its infection. Turkoglu et al. [14], highlighted proper measures using a multi-model Long short-term memory (LSTM) with good accuracy and bagful features. Our chapter will further add on techniques of early detection of infections in apples with better accuracy levels using various kinds of activation functions.

4.3 Related Work

Crop yield mapping, yield estimation, matching of supply with demand, and crop management to increase productivity [15, 16] are essential steps to boost the economy. Machine learning provides low-cost and efficient solutions for crop yield prediction. There are many efficient ways of predicting a crop yield under weather conditions over the field [17]. As mentioned earlier, crop production in India solely depends on temperature and moisture content in a specific area of growth. The review [18], has clearly shown the dependence of crop yield w.r.t change in diurnal temperature range by which one can know the impact of time and temperature to boost up their yield. Similarly, our research deals with amplifying any crop, and its yield under suitable climatic conditions.

In our work, we have portrayed how any spare land can grow its worthy crop rather than being a lonely site or growing up a factory. Growing ample amounts of crops make a path to greenery and a pristine atmosphere which can directly improve polluted air around us. So, with the context of [17] keeping an eye on crop-climate relationships, we have created a model which predicts which crop is suitable to grow under the area depending on climatic conditions and soil content. After a crop is produced, depending upon the soil conditions fertilizer levels are given initially. While applying any fertilizer to the soil, crucial care must be taken of the contents in the soil. Haynes and Naidu [19] explained the influence of fertilizers and manures on the soil's organic matter content and how it changes the physical conditions of the mineral-rich soil. Many times due to improper irrigation facilities available to the poor farmers, the soil gets extensively waterlogged and some of its nutrients may be diluted which in turn will affect crop yield. In [20], it is explained how the soil due to waterlogging loses some nutrients and causes infection itself. Hence, from there the need for fertilizers comes. A complete analysis of fertilizer used in India for the last 20 years is reported by Kumar and Indira [12]. This showed a long-run correlation between fertilizer use and food grains consumption over the last

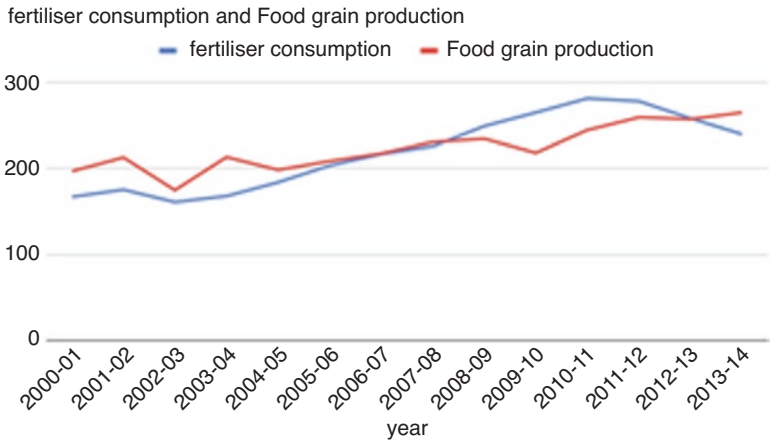


Fig. 4.4 Relation b/w fertilizer consumption and food grain production

20 years as in Fig. 4.4. They explained how farmers are increasing their production with increased fertilizer consumption without considering the environmental, health consequences, and sustainability of agriculture [12].

India is the leading producer of wheat and maize whose yield mostly depends on monsoon and temperature. Being the sole grain for all beings, producing an ample amount of the crop is necessary. A comparative analysis of their yield based on various machine learning techniques is elaborated in [15]. High-yielding varieties such as maize and wheat demanded more usage of fertilizers which is supported by the subsidy policy on fertilizers [12]. With vast acres of land and ample crop yield, the crop should be taken care of any blight. So our model is to timely inspect the condition of the crop with high accuracy to minimize any error.

Many plant leaf diseases were studied including maize (corn) with northern leaf blight, common rust and gray spot (shown in Fig. 4.5) with a well-detailed analysis [21–23]. The model proposed in this work has achieved a higher accuracy of 98.82% than that reported by Geetharamani and Pandian [22, 24].

Singh and Arora [23] have used a convolution neural network method to distinguish between healthy and unhealthy wheat. The unhealthy wheat crops affected by leaf rust and stem rust are shown in Fig. 4.6. In this work, we were successful in enhancing our prediction with an accuracy of 98.7% by learning effective algorithms and activation functions that one should use to pump up the accuracy than that reported in the literature [21, 23].

Fruit diseases can cause a literal economical loss if not controlled on time. An example has been taken to show how the apple scab, rust, and black rot diseases [22] affect the apple leaf and crop (shown in Fig. 4.7). Alharbi and Arif [25] have used CNN to detect and classify apple diseases. In our work, we have included some concepts and better algorithms to predict diseases with better accuracy of 98.93%.

If necessary action is not taken on time, this will result in wastage of crops, and fertilizers leading to financial loss and poor resource management. If the waste is

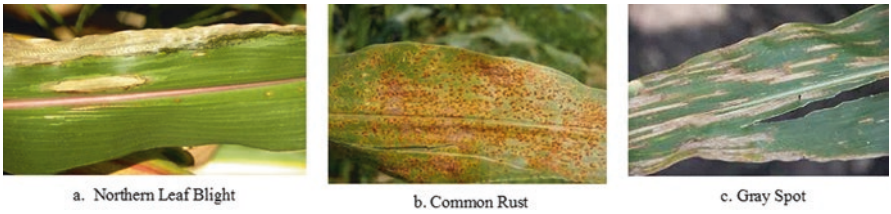


Fig. 4.5 Diseases found in maize

Fig. 4.6 Diseases found in wheat

Fig. 4.7 Diseases found in apple

huge due to late inspection, this can lead to burning the residues to grow fresh crops which will directly affect the entire environment drastically. Our research will keep the four most important features – temperature, rainfall, moisture, and pH level of the soil to envisage a particular crop.

4.4 Proposed Model

India's prime source of economy is agricultural produce, because of the vast areas of cultivable land it is blessed with. As stated earlier, due to biochemical and environmental factors the crop yield may get affected. Hence, we have proposed three interconnected models, which when run in parallel can help the farmers produce

more yield. Our first model is the crop prediction module, where based on parameters like temperature, humidity, pH and rainfall, we predict which crop should be ideally grown in a particular area. Then after the crop yield, a snapshot can be taken to check whether the particular crop has been affected or not. If the crop is healthy, then we continue with timely crop monitoring to prevent any crop loss. But, if the crop is affected, we can pass it through our model and predict which disease the crop is affected by. After the prediction is made, based on conditions of the soil one check which suitable fertilizer should be used to prevent the crops from being affected by any disease, based on parameters of the soil like the temperature of the surrounding, humidity levels of the region, the soil type, the crop type grown and the concentrations of nitrogen, potassium and phosphorus in that particular land. With the help of these parameters, an appropriate remedy can be provided to the farmer. Figure 4.8 gives a diagrammatic representation flow of our proposed model, where the relational connection between the various modules such as the crop prediction module, disease classification module and fertilizer prediction module is represented step by step.

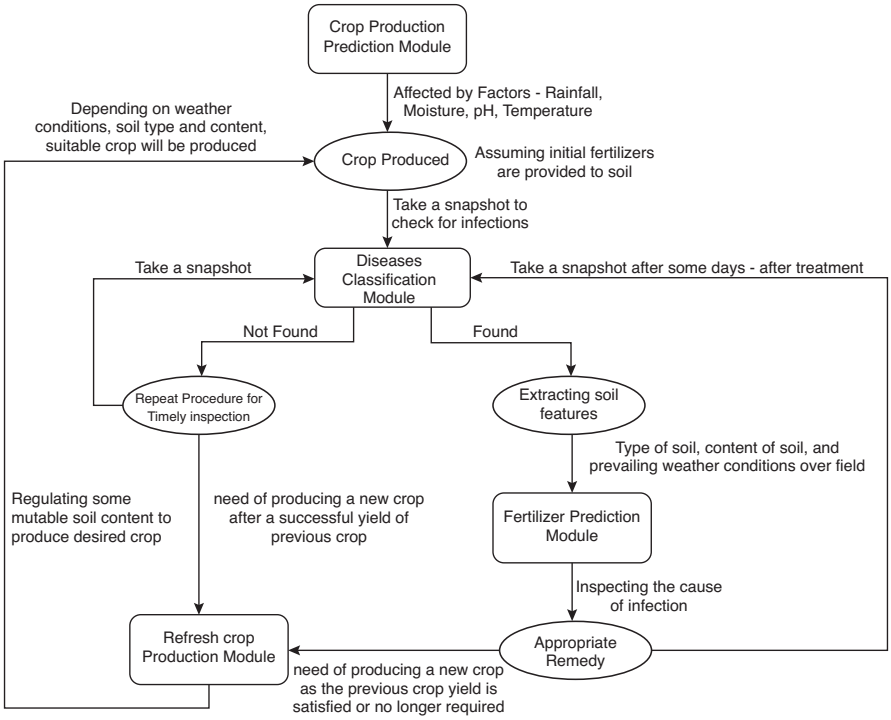


Fig. 4.8 Proposed model architecture

4.4.1 *Machine Learning Approach for Crop Prediction*

The use of statistics and mathematical models clubs together to build a machine learning model. Data and statistics help any model or network to learn relationships, dependencies, and equations among their participating factors. The development of this form of correlations between dependent and independent variables has been made. In [15], comprehensive research is presented on how machine learning is of importance in agriculture. To ensure the machine learning (ML) model is successful, we need to tune every step towards betterment. Figure 4.9 shows a layout of how a machine learns and analyzes patterns.

Here some trained data passes through which a model analyzes and learns all co-relations by a specific algorithm. Then the model infers some prediction or classification rules to predict new examples or future data. In Fig. 4.10, a detailed chart is presented on how the ML model works.

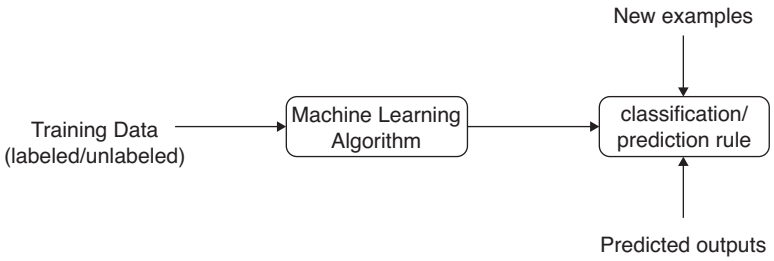
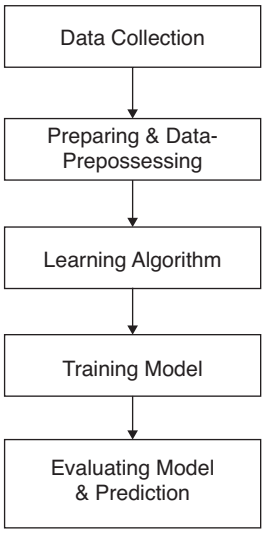


Fig. 4.9 Basic architecture of a machine learning model

Fig. 4.10 Working steps of a machine learning model



4.4.1.1 Data Collection

Gathering data is crucial for a model as the quality and quantity of data that we collect directly determines how good and accurate your predictive model can be. Mathematically, the amount of training data is directly proportional to model accuracy. Our source of data has been extracted from Kaggle datasets having 3100 entries with attributes temperature, rainfall, pH, and the humidity of the field to predict the desired crop that can be grown. Figures 4.11 and 4.12 depict the range of rainfall and temperature present in our data where the optimum range of temperature and rainfall for most of the crops produced is 14–38 °C and 20–250 cm, respectively.

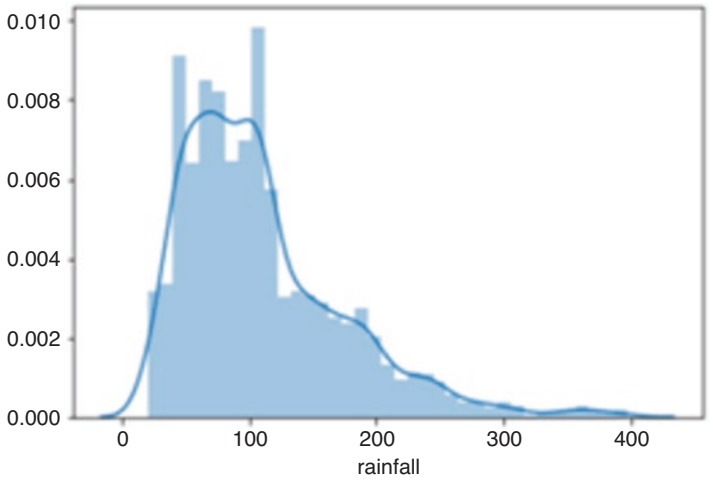


Fig. 4.11 Rainfall range for crop prediction in India

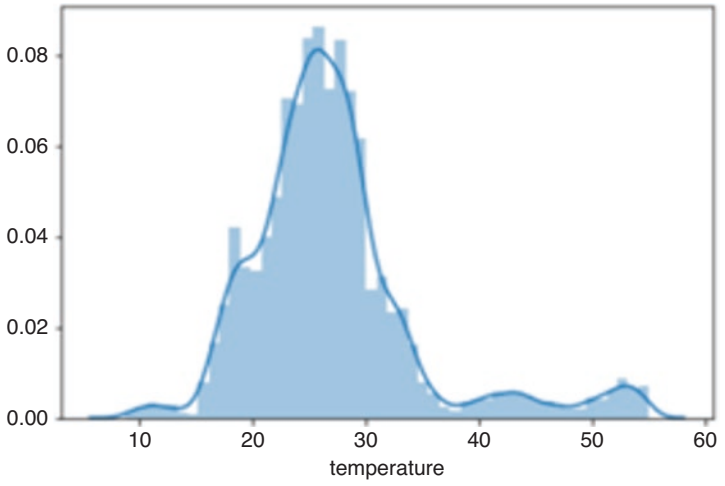


Fig. 4.12 Temperature range for crop prediction in India

4.4.1.2 Data Preparing and Pre-processing

Firstly in data preparation we read and load our data into a suitable set and prepare it for training. Pre-processing sometimes becomes very important because the data we collect can be irregular, absurd, and erroneous due to which data needs other forms of adjusting and manipulation like deduping, normalization, error correction, etc. Raw data should always be cleaned before feeding into the machine to enhance overall accuracy. Moreover, Bhaya and Wesam [26] have discussed various beneficial methods of preprocessing techniques in detail, which are used for data mining.

The next phase explains the need of splitting data into training and testing sets to get validated by the learning algorithm. To prepare this data, the test set is taken as 25% of the whole dataset with a random state tuned to 1. The random state is used in the data splitting module to ensure that the generated splits are reproducible. Feature scaling in entire data is needed in almost all models to give unbiased importance to every factor. In Table 4.1, a data frame is shown with varying data ranges in all columns.

Due to this irregularity, there can be a possibility of the machine giving utmost importance to a factor with a high data range and neglecting others due to low data range irrespective of their real contributing nature. So we need to standardize all variables in the same range. There are mainly 2 processes of normalizing as given in Eqs. 4.1 and 4.2.

$$X_{new} = \frac{X - X_{mean}}{Standard\ Deviation} \tag{4.1}$$

$$X_{new} = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \tag{4.2}$$

A broader perspective has been provided in [27] highlighting various methods of normalization and their influence. For our work, we have used a standard scalar to normalize all values under each attribute to develop equal importance as in Table 4.2. Using this scalar all values range from −3 to +3.

Table 4.1 Dataset sample for crop prediction

Temperature	Humidity	PH	Rainfall
51.395179	46.579188	6.332919	105.272329
51.750697	54.662403	6.511772	166.146187
53.017400	49.864205	5.299104	65.959049
53.211092	61.440867	5.322864	64.152838
50.875089	52.118891	7.377994	163.452682

Table 4.2 Normalized dataset sample

Temperature	Humidity	PH	Rainfall
−0.26197908	−0.16490876	−0.27393577	−0.48244424
−0.11746592	0.62523333	−0.05613165	−0.69880365
−0.28625737	−0.50470681	0.84887115	0.29740972

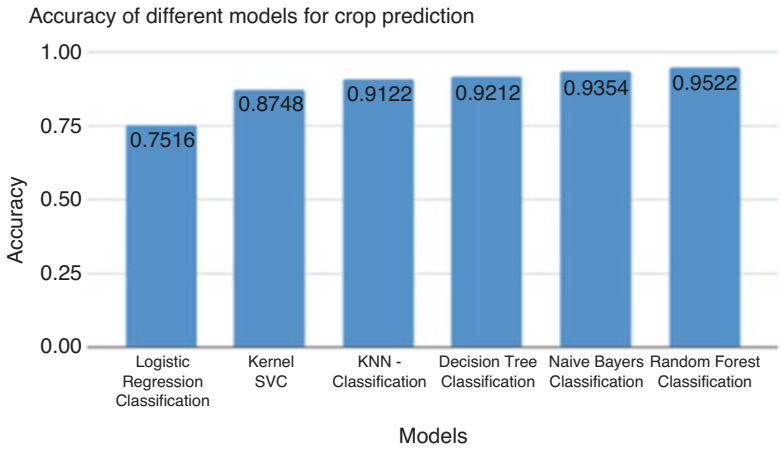


Fig. 4.13 Accuracy of different models

4.4.1.3 Learning Algorithm

Machine learning is backed by algorithms that predict output values within an acceptable range after analyzing input data. These algorithms are programmed to learn and optimize their calculations for better performance and efficiency when new data is allocated, developing intelligence over time. Machine learning marks the use of a wide range of algorithms to get a good correlation among the variables. We have used supervised algorithms like Logistic Regression, Naive Bayes classification, K-Nearest Neighbor Classification, and Support Vector Classification to make the machine learn correlations among variables [28–30]. Apart from Random Forest Classification which is derived from the Decision Tree was found to be at maximum accuracy (Fig. 4.13) with a great learning rate.

Since the best result is achieved using a decision tree with entropy, the criterion for the random forest is also chosen as entropy. This gave an accuracy of 95.22% and is the best among all other models.

4.4.1.4 Training the Model

As in the earlier section, it concluded that applying Random Forest Algorithm with the Decision Tree algorithm as its origin has increased the predictive nature of the model effectively with an accuracy of around 95%. In [17], a well-designed approach has been researched on crop prediction using Decision trees. So here in this work, we chose a random forest as our proposed algorithm. Figure 4.14 shows the working of the Random Forest Algorithm for any model.

We can understand the working of the Random Forest algorithm with the help of the following steps. These processes go over desired iterations to make the predictive model perfect and efficient.

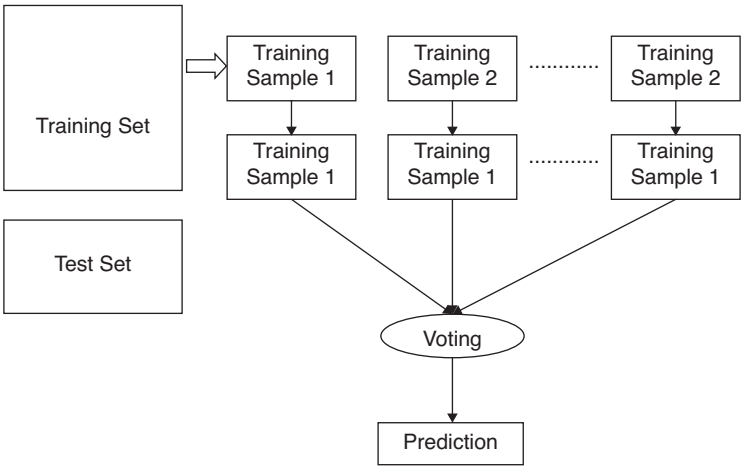


Fig. 4.14 Working of random forest classifier

```
from sklearn.ensemble import RandomForestClassifier
classifier = RandomForestClassifier(n_estimators = 1000,
                                    criterion = 'entropy',
                                    random_state = 1,
                                    max_depth=100)
classifier.fit(X_train, y_train)
```

Fig. 4.15 Parameters considered under random forest classification

- Step 1 – For a given data, a random number of samples are selected.
- Step 2 – For every sample, a decision tree is constructed which gives the prediction result.
- Step 3 – For every result, voting is performed.
- Step 4 – The most voted result is selected as the final result.

In Fig. 4.15, the code snippet shows the initiation of a random forest algorithm and fitting that classifier to learn co-relations among attributes for the prediction of future data.

This classifier object fits the training set to learn the relations, then validates with the testing test over iterations to figure out the best accuracy which is discussed in the section below. This final model is then used to predict future data in real-time scenarios.

4.4.1.5 Evaluating the Model and Prediction

To evaluate our model's performance [29], we use different types of evaluation metrics to improve the power of prediction before performing predictions on unseen data. To evaluate our models, we used metrics like f1-score, recall, and precision to

validate at their best. In [23, 31], a proper dependency of evaluation/accuracy metrics is shown w.r.t. various predictive models.

Precision: When the model predicts a class, precision is used to determine how many times the model is predicting that class correctly as in Eq. 4.3

$$Precision = \frac{TP + TN}{TP + TN + FP + FN} \tag{4.3}$$

Where, TP = True Positive, TF = True Negative, FN=False Negative, FP=False Positive.

Recall: When the value is true, recall tells us how many times the model can predict it's true as in Eq. 4.4

$$True\ Positive \frac{Rate}{Recall} = \frac{TP}{TP + FN} \tag{4.4}$$

F1-score: The harmonic mean of precision and recall is F1-score, where its best value is 1 which means perfect precision and recall and the worst is 0 in Eq. 4.5

$$F1 = 2 \times \frac{Precision \times recall}{Precision + recall} \tag{4.5}$$

The study of data analysis [23] inferred that the higher the F1 score, the better is the predictive model, with 0 being the worst possible and 1 being the best. Taking into consideration all crops, the average of all evaluation metrics tends toward 1 which indicates our predictive model is well-developed and good.

The average of all evaluation components was found to be:

Precision = 0.9519354839
Recall = 0.9451612903
F1_score = 0.9464516129

These components generally should be near to 1 for any model to fit perfectly. The model accuracy for crop production amplification was found to be 95.22% with the proposed algorithm as Random Forest Classifier using entropy index.

4.4.2 Disease Detection Prediction

A Convolutional Neural Network (CNN) is a type of deep learning algorithm used for image classification. It can be termed a multi-layer neural network designed for analyzing visual inputs to perform tasks such as image classification, segmentation and object detection. So every CNN follows a flowchart where it can learn the pixels step by step and with that knowledge, it can predict and classify any image. The following steps are depicted in Fig. 4.16.

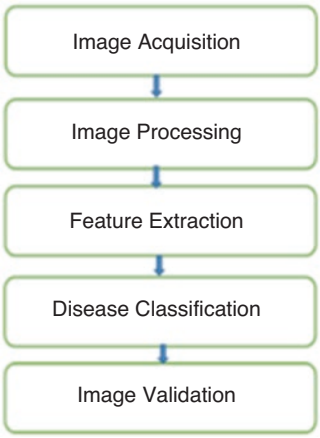


Fig. 4.16 The working principle behind a CNN model

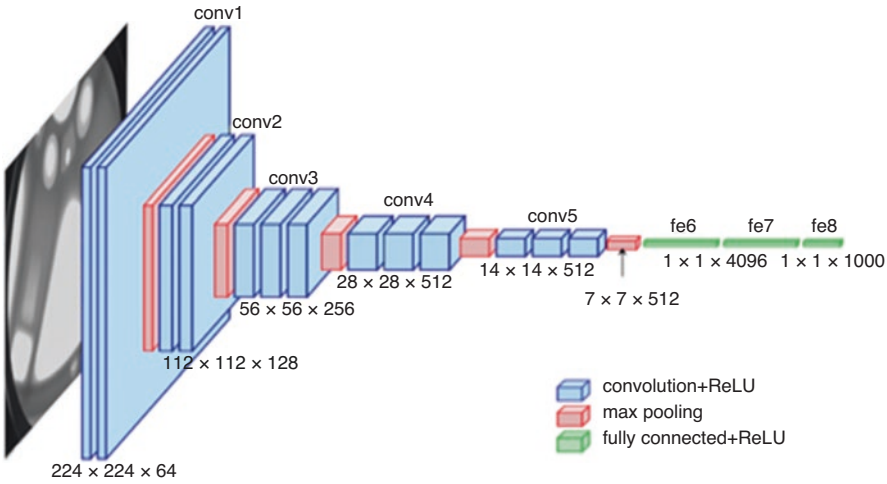


Fig. 4.17 Basic architecture of a CNN model [32]

In this research work, we would like to brief you on a basic CNN overview in Fig. 4.17. The initial pixels have been reduced for better machine interpretation. Abetting the pixels removes the complexity of any image and allows algorithms to learn images effectively. So now we will follow a step-by-step procedure where an applied algorithm can detect and classify diseases caused by maize, wheat and apple.

4.4.2.1 Image Acquisition

Image datasets for maize and apple diseases have been taken from the Kaggle data source and tested on real field image data. This data set is reformed using offline augmentation from the original data set. For wheat disease classification, images

Table 4.3 Number of training and testing sets of all types of each crop

Crop name	Type od disease	Number of training samples	Number of testing samples
Maize	Healthy	1859	465
	Northen blight	1908	477
	Cercospora gray spot	1642	410
	Common rust	1907	477
Wheat	Healthy	562	134
	Leaf-rust	562	134
	Stem-rust	562	134
Apple	Healthy	2008	502
	Cedar apple rust	1760	440
	Black rot	1987	497
	Apple scab	2016	504

come from a variety of sources. Some of the data is also from public images under Google.

Here for extracting images from source to machine, class_mode is assigned as 'sparse' due to multinomial variations of diseases for each crop. The following is some information and basics of how infections grow. We have highlighted all types of diseases focused on them in this research. As mentioned earlier, maize and wheat are important – utmost care should be taken to its cause and precautions adding to timely checkups.

The diseases specified in Table 4.3 are generally caused by wet springs and humid weather conditions. This disease may not kill the host but is involved in fruit deformation and premature fruit drop. The spread of these infections across the fields can also be determined by the windblown spores of the fungi which can carry disease long distances.

4.4.2.2 Image Pre-processing

Image pre-processing involves re-scaling or transforming every pixel value from the range (0, 255) to (0,1). Some images have a high pixel range and some have a low pixel range, due to which it can be quite perplexing for a machine to recognize its complete features. In this case, if we treat all the images in the same manner, the neural network module will also consider all images equally important. Also, scaling every image to the same range of (0,1) will make images contribute more evenly to the total loss existing. Moreover, a brief study about image processing is provided in [33].

From Keras, we import the Image Data Generator library to perform the enhancement of an image as shown in Fig. 4.18.

Here, in our proposed methodology, some mini processing steps have been taken for a better enhancement of images: such as every image being resized into 64*64 target size. The training data is processed in a way the machine can withstand any

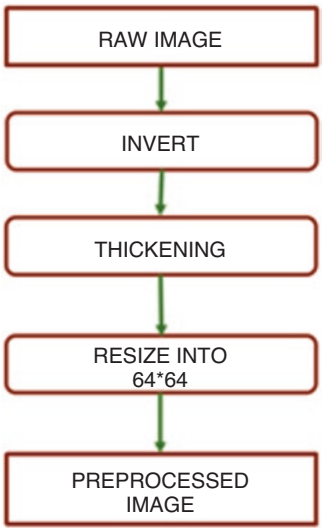


Fig. 4.18 Step wise procedure involved in image processing

image flips i.e., rotations – left and right and also preserve the true classification of any crop disease. Also for better feature extraction in further steps, all images have been processed with a prior zoom limit of 0.2.

As mentioned above data augmentation has been used in all possible forms to make the learning better for the neural network which can help to increase the amount of relevant data in any dataset. Images of all crops for every disease after re-scaling are shown in Fig. 4.19.

In Fig. 4.20, it is shown how the whole image processing is well connected to image acquisition, and feature extraction processes extending data validation and classification. These procedural steps give clarity about which image has been in input, from noise removal to final classification via perfect validation accuracy.

4.4.2.3 Feature Extraction

To recognize any image, CNN must scan every image deeply. All the existing features in any image should be known for better extraction of key features to enrich the classification accuracy. Feature extraction has convolution layers followed by max-pooling and an activation function. The working principle of this step includes treating the pre-trained network as an arbitrary feature extractor, allowing the input image to propagate forward, stopping at the pre-specified layer, and features are the output of that layer. Also, Keiron and Ryan [34] have reported an overview of the working of CNN architecture.

The accuracy of learned models is increased using feature extraction by extracting features from input data. This phase not only helps in enhancing the final

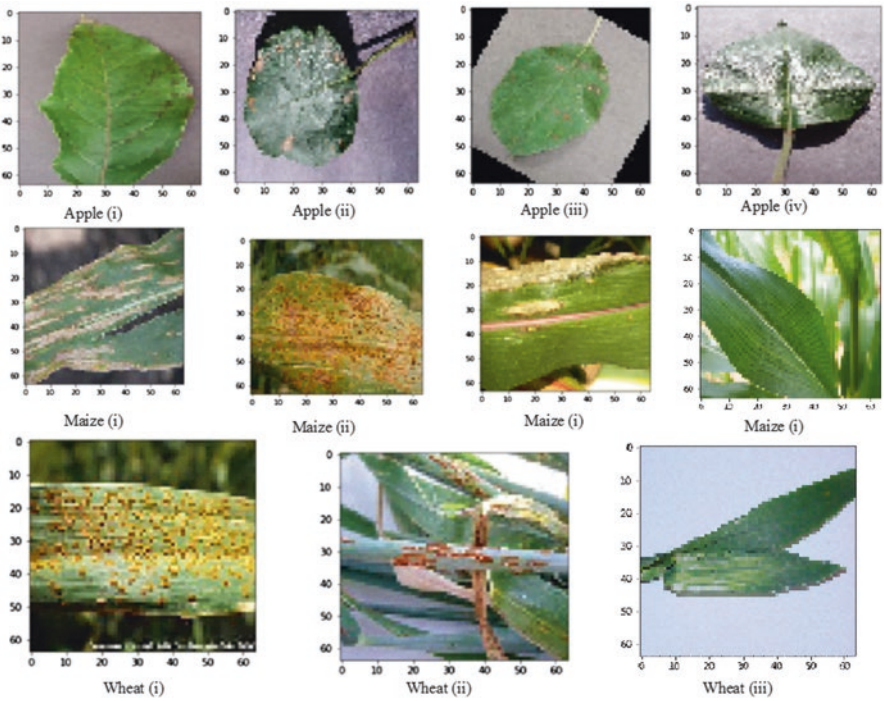


Fig. 4.19 Re-scaled images of all crops diseases after pre-processing

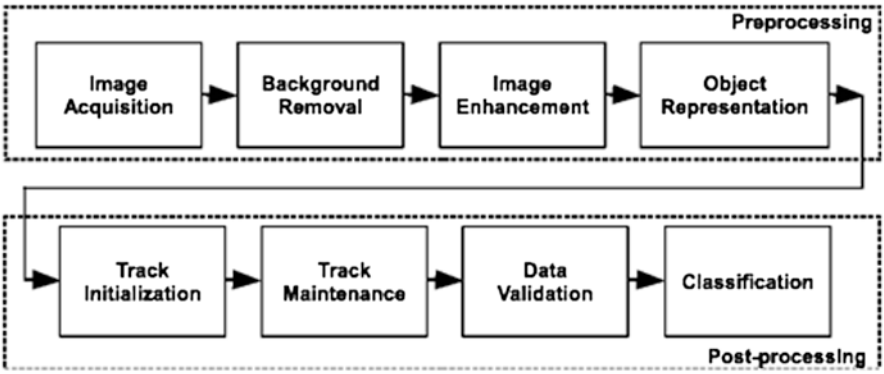


Fig. 4.20 Cycle of pre-processing and post-processing

accuracy but also removes redundant data and hence reduces the dimensionality of data. Hence, it increases training and inference speed.

In this research for the feature extraction process, we have created two deep convolutional layers, followed by two pooling layers and an activation function 'reLU' for each crop having the same parameters as shown in Table. 4.4. We found our

Table 4.4 Parameters used in the feature extraction of the CNN architecture

	Conv 2D				Max Pooling	
	Filters	Kernel size	Input shape	Activation	Pool size	Strides
Layer 1	32	(3, 3)	[64, 64, 3]	Relu	(2, 2)	2
Layer 2	32	(3, 3)	[64, 64, 3]	Relu	(2, 2)	2

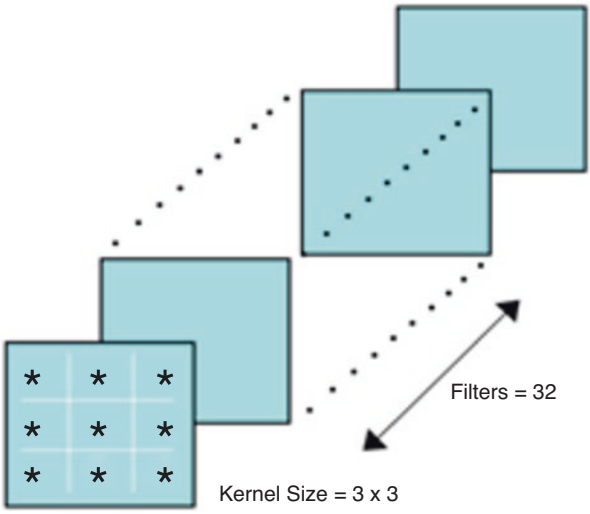


Fig. 4.21 Kernel size and filters used in conv2D model

model to be in a good perspective using 2-layers in feature extraction. A good amount of research has been done in [25] based on feature extraction. In agreement with their work, our model has been regenerated and optimized with greater accuracy.

4.4.2.3.1 Convolutional Layer

A conv2D layer has a filter or a kernel that consists of dimensions 3*3 kernel size and filters as 32 in our model as shown in Fig. 4.21. These are generally smaller than the input image so the whole image is covered. The area, called the receptive field, is where the filter is on the image area which is willing to consider or accept new suggestions and ideas.

There are three channels in an image mainly red, green, and blue through which the filter present in Conv2d extends with the possibility of each channel having different filters. Convolution is performed individually for each channel and then, they are integrated to get the final output called convoluted image. After the convolution operation, a feature map is received as an output of a filter.

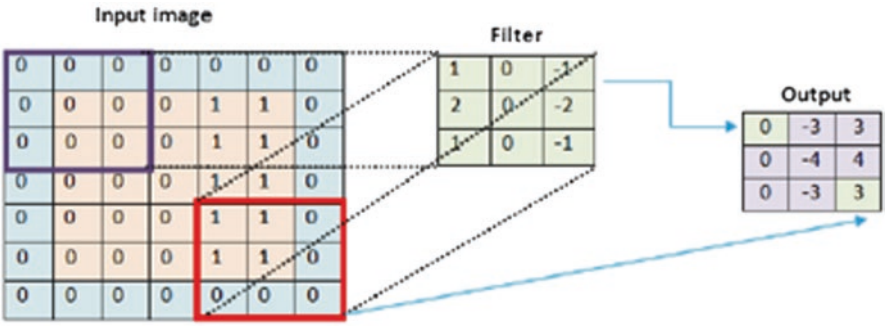


Fig. 4.22 Generic representation of a Conv2D network overlapping a filter dimension into input image to form the output shape [25]

Figure 4.22 represents a generic conv2D network used [25]. Convolution layer output is represented by Eq. 4.6.

$$M_j^p = f\left(\sum_{i \in M_j} M_i^{p-1} * k_{ij}^p + N_j^p\right) \tag{4.6}$$

Where p represents the pth layer, k_{ij} denotes convolutional kernel, N_j denotes bias and M_j denotes a set of input maps.

4.4.2.3.2 Pooling Layer

Extraction of sharp and smothered features is mainly done using the pooling layer. It is also done to reduce divergent features of data and computations for better estimation. Mathematically pooling is calculated as in Eq. 4.7.

$$\text{Output size} = \frac{\left(\text{Input size} - \text{Pool size} + 2 * \text{padding}\right)}{\text{stride}} + 1 \tag{4.7}$$

Gholamalinejad and Khosravi [35] have discussed various effective pooling methods in detail. In our work, we have used max-pooling represented as in Eq. 4.8. And a related diagram is shown to understand its working in Fig. 4.23.

$$V = \max_{i,j=1}^{h,w} S_{i,j} \tag{4.8}$$

4.4.2.4 Disease Classification

This is the last phase of the architecture where the disease is predicted. After the max-pooling feature, the process of flattening is done. The flattening method converts the output into a vector. The result of the classification can be obtained only

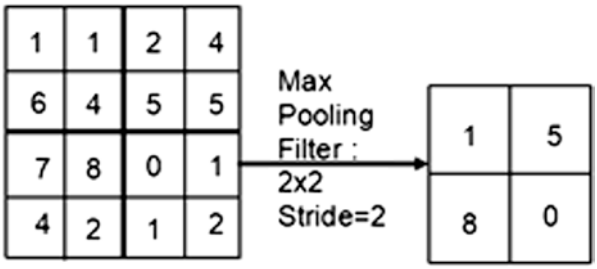


Fig. 4.23 Max pooling – Single Depth slice

Table 4.5 Number of units and activation function in a dense layer

	Dense Layer	
Number of Layers	Units	Activation
Layer 1	128	Relu
Layer 2	32	Relu
Layer 3	16	Relu
Layer 4	4	Softmax

through a vector. Hence, the conversion is necessary. Fully connected layers flatten the network's 2D spatial features into a 1D vector that represents image-level features for classification purposes [36].

Table 4.5 shows four multiple dense layers, a deeply connected neural network with the units and activation function used in each layer respectively. The hidden layers have used 'reLU' to increment the non-linearity in our images. The output layer has 4 units for each class of disease and softmax is used because it is suitable for mutually exclusive multi-class classification in the logistic regression model.

4.4.2.4.1 ReLU

The ReLU layer is used as an activation function here between the convolution layer and the feature maps (Fig. 4.24) [25] to convert all negative values to zero without affecting the size of the image and its dimensions.

So these 3 layers sum up to feature extraction giving a basic accuracy level. We added a dropout layer to achieve better accuracy. This regularization technique is used to prevent the model from overfitting as it randomly sets the input neural units to zero at each step during the training process [37]. It takes preventive measures to avoid complex co-adaptations on the training data which results in reduced overfitting.

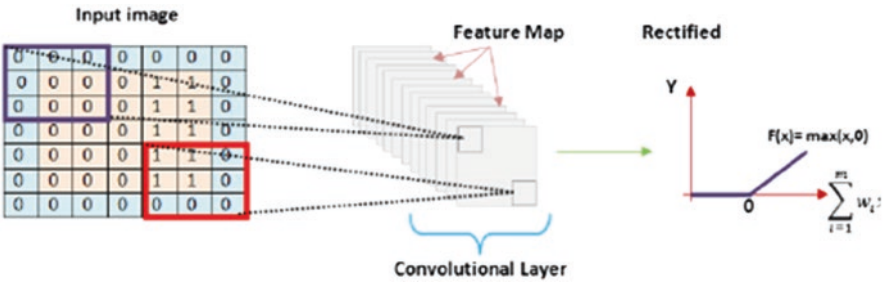
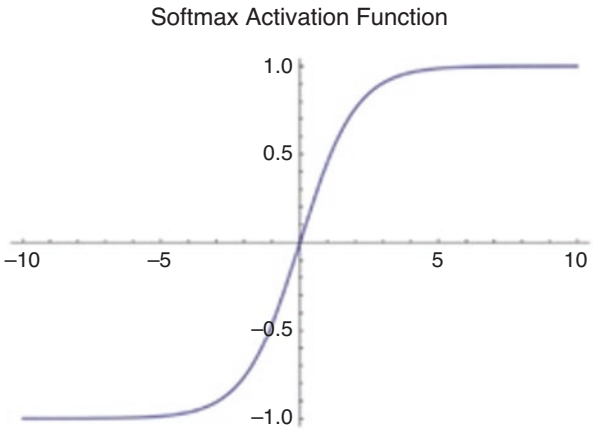


Fig. 4.24 Applying activation function Relu to feature maps of convolution layer [25]

Fig. 4.25 Graph of softmax activation function



4.4.2.4.2 Softmax

Softmax is a combination of multiple sigmoid functions. A brief discussion about various activation functions used in the deep learning technique is provided in Fig. 4.25 [38].

In the compilation process for our CNN model, we have used the Adam optimizer because it gives the best accuracy among other optimizers available.

Figure 4.26 represents the overview of the internal mechanism of our model (including all the layers). The fully connected layers are finally formed by the last few flattened layers of our CNN model. These layers are instrumental in predicting the final output, i.e., predicting a particular disease.

4.4.2.5 Image Validation

Several sample images were selected from different sources for validation for which the model predicted diseases accurately. This was achieved due to the high accuracy of our models which was accomplished by tuning various parameters to enhance for

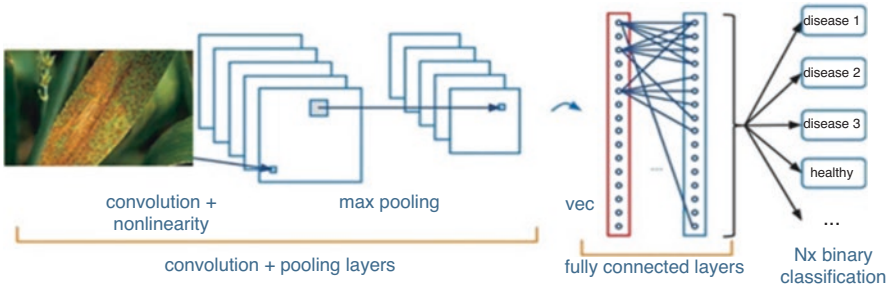


Fig. 4.26 Internal mechanism of how fully connected layers are formed using maize common rust image

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape
=================================================================
conv2d (Conv2D)              (None, 62, 62, 32)
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 31, 31, 32)
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 29, 29, 32)
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 14, 14, 32)
_________________________________________________________________
flatten (Flatten)            (None, 6272)
_________________________________________________________________
dense (Dense)                (None, 128)
_________________________________________________________________
dense_1 (Dense)              (None, 32)
_________________________________________________________________
dense_2 (Dense)              (None, 16)
_________________________________________________________________
dense_3 (Dense)              (None, 4)
=================================================================
```

Fig. 4.27 Proposed model summary for disease detection using CNN

a more precise outcome. Later in Sect. 4.5.2, a detailed analysis will be presented on validation tests and overall accuracy. Further, it is discussed how regularization techniques can improve performance in a supportive manner reducing the margin of error.

The various parameters of the CNN such as the number of layers needed, the number of units per layer, the kernel size and the apt activation function for our model, were carefully tuned after several considerations. After making changes, we obtained a good accuracy for our CNN model. Figure 4.27 describes the summary of the model. It shows the output shape of the layers and the number of layers after each change.

4.4.3 *Artificial Neural Networks for Fertilizer Prediction*

Artificial neural networks are structures used for computing various tasks and their working is inspired by the human brain. These networks are similar to simulating tasks like clustering, pattern recognition, and classification on the computer. Presently, the usability of these networks has also forayed into the world of agriculture, helping our farmers to increase their profits.

The ANNs learn by adjusting the weights and biases for all the input parameters, it tries learning and gives a prediction. The process of learning is summed up in the steps in Fig. 4.28.

The description of the diagrammatic representation of the above steps is discussed below.

4.4.3.1 Data Collection

Our data source for the research work is Kaggle. The parameters (Fig. 4.29) used for predicting an apt fertilizer are the temperature conditions, the humidity level (an absolute value), and the moisture content of the soil. Sandy, loamy, red, etc. were

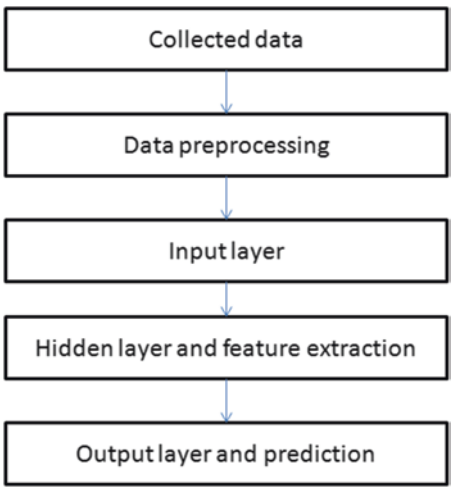


Fig. 4.28 Working of an ANN model

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

Fig. 4.29 First five readings of the fertilizer dataset

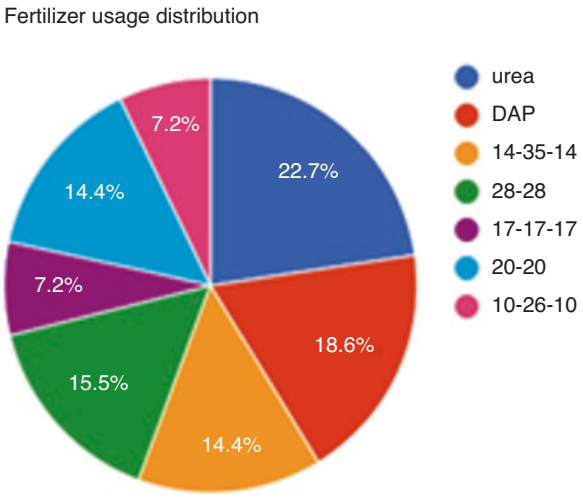


Fig. 4.30 Distribution of various fertilizers present in the dataset obtained

some of the soil types which were the determinants of the correct fertilizer. The amount of nitrogen, potassium, and phosphorus, some of the essential nutrients required for the good growth of the crop, are also included as parameters. Some of the fertilizers which can be predicted by our model are- Urea, 14-14, DAP, 28-28, etc. (Fig. 4.30) among which urea is found to be of maximum use in India.

4.4.3.2 Data Pre-processing

The collected data is labeled encoded. There might be a substantial difference in the values of some columns of the data which may lead to prioritizing the columns which have larger values, hence normalization is done. Among various normalization methods [26], the mathematical representation for some of the normalization techniques like Standard Scalar and Min-Max Scalar is given in Eqs. 4.1 and 4.2. We have used Standard Scalar in our model.

4.4.3.3 Input Layer

Artificial neural networks need data to be fed into our systems as input. Then, the data is randomly assigned with weights and forwarded to the next hidden layer. Figure 4.31 shows the input layer parameters where the neuron is represented by a circle to which all the inputs are added. Then, an activation function is applied followed by bias.

A bias can be considered to be analogous to adding a constant value in a linear equation and finally combining it to form a network. After this internal computation, a predicted value is obtained. This is an abstract overview of how the computation is done.

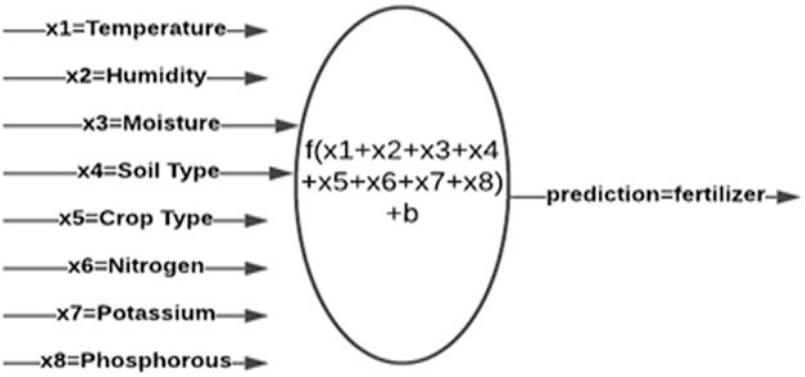


Fig. 4.31 Input layer showing various input parameters

Table 4.6 Hidden layers specification

	Units	Activation function
Layer 1	8	ReLU
Layer 2	8	ReLU

4.4.3.4 Hidden Layer and Feature Extraction

The hidden layers are used so that the machine gets enough to learn about the data and its co-relationship with other attributes which indeed helps in predicting the class for any future data. We have implemented our model using 2 hidden layers with 8 neurons each (Table 4.6).

When the input parameters are fed into the machine, due to the neurons in hidden layers some weights are multiplied by the input data to increase its learning rate. Finally, a bias is added to the result which serves as an additional parameter to adjust the output. Our developed neural network with hidden layers and its neuron is figured in Fig. 4.32.

Activation functions [38, 39] are generally used in deep learning models to introduce non-linearity. Arya and Ankit [40] have provided a brief analysis of the learning and recovery of 'reLU' function. The ReLU activation function (Eq. 4.9) is used after the features are extracted. ReLU activation function (Fig. 4.33) solves the problem of vanishing gradients which is computationally less expensive as compared to other activation functions like 'tanh'.

$$ReLU = f\left(x\right) = \{x, if\ x > 0 0, if\ x < 0 \tag{4.9}$$

Later in Sect. 4.5.3, we have discussed how a regularization technique can render a better accuracy with increasing performance for our base model. In [39], a comparative analysis of various regularization techniques used in ANN is provided.

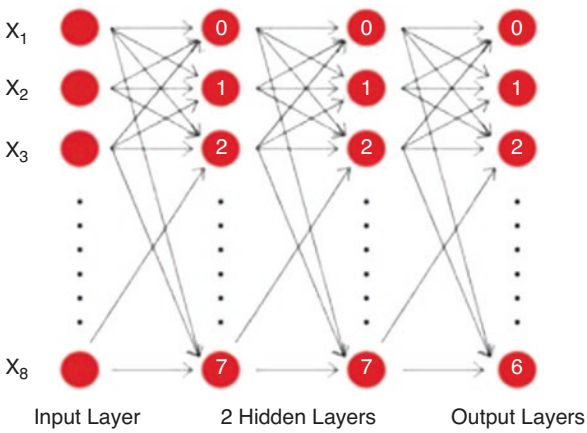


Fig. 4.32 Neural connections in our ANN model for fertilizer prediction

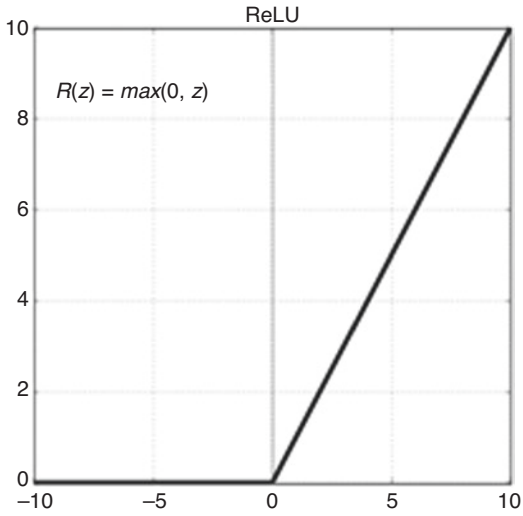


Fig. 4.33 ReLU activation function

4.4.3.5 Output Layer and Prediction

The output layer is also referred to as the classification layer. The sigmoid function is being used for predicting the final output. The number of units used in the output layer is 7 (Fig. 4.34).

Then we call the compile module to combine all the layers to start its training phase and finally predict the favoring class. The optimizer used in the compilation is Adam as in Fig. 4.35. Among the various varieties of SGD such as RMSProp, Adagrad and Adam optimizer gives the best outcome. A more detailed analysis of Adam and a variation on it is in [41].

Many optimizers can be used such as SGD, and RMSProp [41]. But Adam optimizer is considered to be the best one due to its learning rate, reliability, efficiency

```
ann.add(tf.keras.layers.Dense(units=7, activation='sigmoid'))
```

```
ann.compile(optimizer = 'adam', loss = 'sparse_categorical_crossentropy',
            metrics = ['sparse_categorical_accuracy'])
```

Fig. 4.34 Output Layer configuration and compilation step

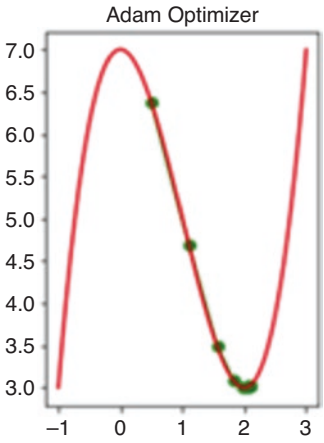


Fig. 4.35 Curve of Adam optimizer

Table 4.7 Learning rate of various Optimizers

Optimizer	Learning rate
Adam	0.001
RMSProp	0.001
SGD	0.01

and cost-effectiveness as compared to others. Table 4.7 shows the learning rate of different algorithms.

Here, Adam and RMSProp show a good learning rate of 0.001. In Sect. 4.5.2, we will further discuss the results obtained from using these various optimizers.

After training the model with Adam optimizer, using metrics like precision, recall, and F1-score, the model was evaluated. The average values of these stated metrics were found close to 1 giving an insight that our model has nice accuracy and fitting.

The ANN model for fertilizer prediction was found to be 96% with our proposed network model with 2 hidden layers of 8 neurons each and the output layer of 7 neurons due to 7 classified fertilizers in our dataset.

4.5 Result Analysis

4.5.1 Crop Prediction

We have extended some scope of hyper-parameter tuning in our proposed random forest algorithm by scaling parameters like n-estimators and max-depth. N-estimators are the number of trees internally built by the algorithm before the averages of the predictions are made. Hence, the higher the number of trees, the algorithm will be able to choose from more options, which will help the classifier learn better. From Table 4.8, it is evident how an increase in the number of estimators yields better accuracy in the random forest classifier.

But n-estimators also depend on the shape of the original dataset or inputs. If n-estimators exceed the number of input rows, the model can lead to fainting situations. So, utmost care must be taken while tuning the parameters. However, reduced n-estimators can also raise problems in the designed model.

As the max_depth increases, the accuracy decreases due to over-fitting. Clearly, from Table 4.9, we can see how the accuracy decreases up to a certain level by increasing the max-depth and then reaches a threshold. The model learns patterns and various correlations to adopt predicting new data correctly. Hence, an optimal value for max-depth must be chosen based on the number of features present in the dataset.

For our final model, we have trained the Random forest classifier using 1000 n-estimators and max-depth as 10. Using these hyper-parameters, the evaluating metrics such as precision, recall, support and F1-score are close to 1.00.

To visualize the deviation of predicted output from the actual output, a graph was plotted for 100 observations. In the above graph (Fig. 4.36), the blue line indicates the predicted data and the red line indicates the test data. The coinciding lines depict that most of our predictions are in synchronization with the actual output which accounts for the accuracy of 95.22%.

Heatmap is a visualization technique to show the multicollinearity amongst the attributes present in the dataset. Values closer to 1 depict a positive correlation,

Table 4.8 Tabulation of accuracy achieved using different values of n-estimators

N-estimators used in the algorithm	Accuracy (in %)
1000	95.22
500	94.96
100	94.45

Table 4.9 Comparison of accuracy achieved using different max-depth values

Max-depth parameter value used	Accuracy (in %)
10	95.22
50	95.09
100	95.09

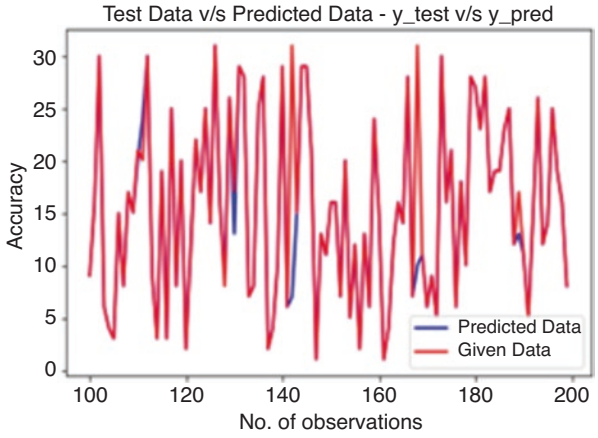


Fig. 4.36 Deviation of predictions from the actual result

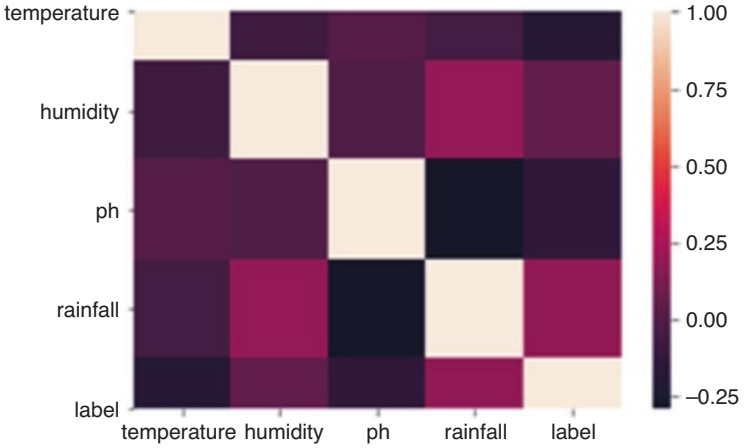


Fig. 4.37 Heatmap showing a correlation between different attributes in crop prediction

whereas being closer to 0 means there is no linear trend. The monochromatic scale parallel to the heatmap represents colour association to correlation. As shown in Fig. 4.37, highly correlated are defined using a lighter shade while the least ones are shown using a darker shade. And the correlation among the same variables is depicted by the white color, which is 1, as shown in the corresponding colour bar of the heatmap since the same attributes will always be completely correlated. The heatmap in Fig. 4.37 also shows how the characteristics like temperature, humidity, pH and rainfall are associated with our target variable. Some negative correlations are between pH and precipitation, temperature and the final label output. At the same time, some positive correlations are humidity and rain.

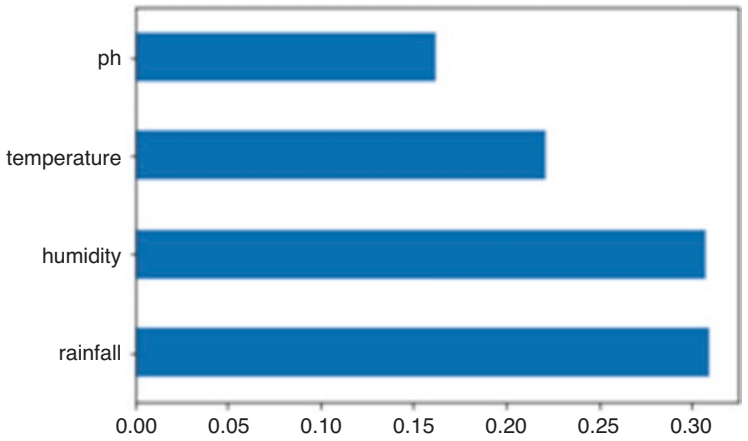


Fig. 4.38 Importance of different features in crop production

Figure 4.38 depicts the importance of the attributes used to predict the crop. India has a monsoon type of climate directly influenced by the water bodies surrounding the sub-continent. So as per the graph, the amount of rainfall is the most critical factor in determining the crops that can be cultivated in a particular region of India. Ensemble learning is used to determine the importance of features with the ExtraTreeClassifier. For better visualizations, the output of the feature_importances_ class is plotted as a "barh" graph.

Finally, the accuracy we have achieved using the Random Forest classifier is 95.22%, which is better than other proposed models [17].

4.5.2 Disease Classification

We will discuss the diseases that affect three crucial crops grown in India for our proposed CNN architecture. Better accuracy has been achieved through regularization techniques and varying the number of epochs. By training our model more times, the model will better understand patterns and produce more correct predictions. Hence, accuracy is a measure of how well our model will reduce mispredictions. Better the accuracy, the better the model we have developed. The relationship of accuracy with varying epochs for all the three crops is graphically depicted in Fig. 4.39.

Along with the increase in accuracy, another measure that provides insight into the model's robustness is the value of the loss function. Loss is a measure of the distance between the true values of the problem with the values which our model is predicting. Greater is the loss; more are the errors made on the dataset. In our model prediction, we have used the sparse categorical cross-entropy loss function as each dataset sample belongs to only one particular class. In the case of the loss function,

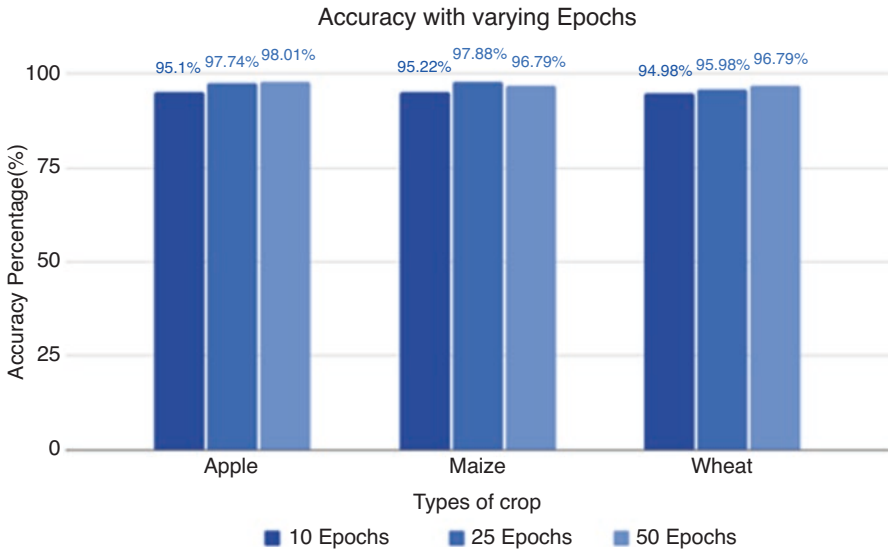


Fig. 4.39 Bar Graph depicting the varying values of accuracy with the number of epochs

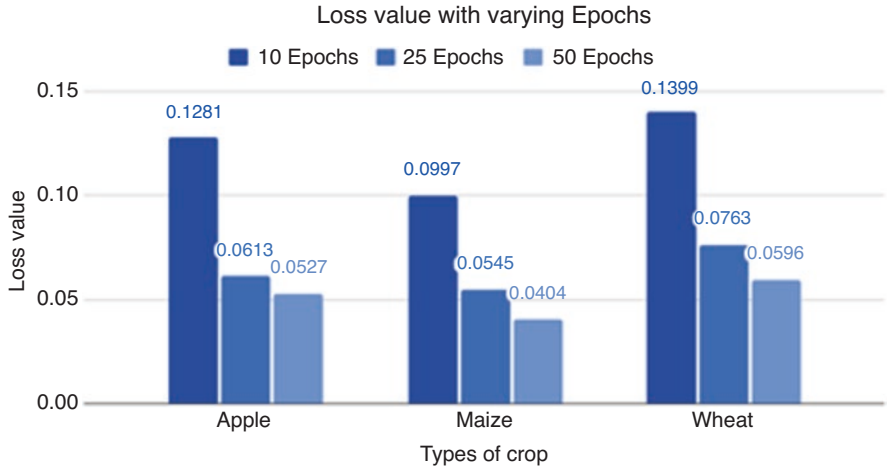


Fig. 4.40 Bar graph depicting the variation of loss values with the number of epochs

the lesser the value of the loss function, the better the model's training. The correlation between the loss values to the number of epochs is represented in Fig. 4.40.

It is evident from Figs. 4.39 and 4.40 that among various epochs, 50 epochs led to the best accuracy and least loss which allowed our model to be trained efficiently. Hence, we will be considering 50 epochs as a suitable benchmark for our further training process to improve model accuracy to a better pedestal.

While modeling any architecture, one must take utmost caution while selecting the optimal number of epochs for model training based on the size of the dataset and the quality of images. For instance, if the size of the dataset is small, then a large number of epochs will cause our model to overfit. Similarly, if a small number of epochs are employed for a large dataset, the model will be under-fitted. Both of these extreme conditions can lead to an increase in the loss function, degrading the model's efficiency.

Now the performance of our model has been amplified by using a regularization technique adding a dropout layer as mentioned earlier in Sect. 4.4.2. The dropout layer shuts off some neurons so our model does not overfit.

In Fig. 4.41, we can infer that with a decrease in dropout percentage, model accuracy increases to some extent which makes the learning better and even reduces the probability of overfitting. In Fig. 4.42, it is shown that with a decrease in dropout percentage, the loss function decreases, building the model less erroneous.

The percentage value of the dropout layer must be chosen carefully because if a very huge dropout percentage is taken, the maximum number of neurons will be switched off and our model will be unable to learn properly. If a low dropout percentage is used, it may lead to overfitting. While modeling our algorithm we observed that at 15% dropout we are obtaining somewhat good accuracy for some crops, but the graphs obtained showed that the learning process was not very efficient due to some inconsistency of validation curves. From Figs. 4.41 and 4.42, we got an optimum dropout percentage range. The best results with high accuracy and lower loss values were generated keeping the dropout percentage between 25% and 50%. Any value below that can lead to a scope of overfitting and any value above 50% may lead to a poor learning curve. Moreover, Alvin and Dae-Ki [42] have explained different dropout regularization techniques used in Neural Networks.

Here sparse categorical accuracy and sparse categorical cross-entropy are used to evaluate the model performance and efficiency in predicting new data (Figs. 4.43, 4.44 and 4.45).

The graphs compare the learning curves based on validation and training data where the gap area between them is significantly less and somewhat superimposing, which aligns with the general trend of CNN architecture. Table 4.10 contains the final accuracy of our described architecture.

Using our CNN architecture we obtained an accuracy of 98.70% for wheat prediction, which is 0.1% more than the best model used in [23].

The overall classifier accuracy for our model is 98.82% for the common diseases affecting the maize crop, which is higher than the one achieved by the model used in [21, 22]. In addition, we have also achieved a mean accuracy of 98.93% for classifications of apple infections, which is greater than the mean accuracy obtained using the model defined in [22].

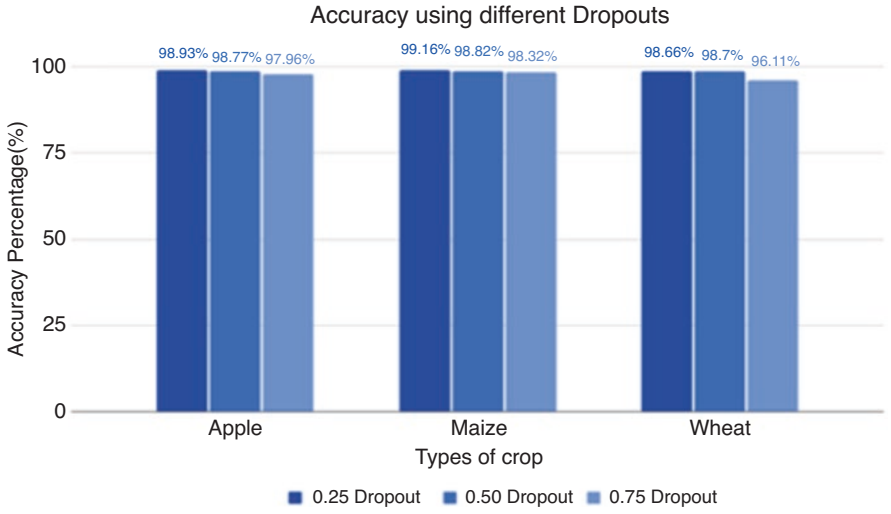


Fig. 4.41 Bar graph representing the varying accuracy values with different dropout values

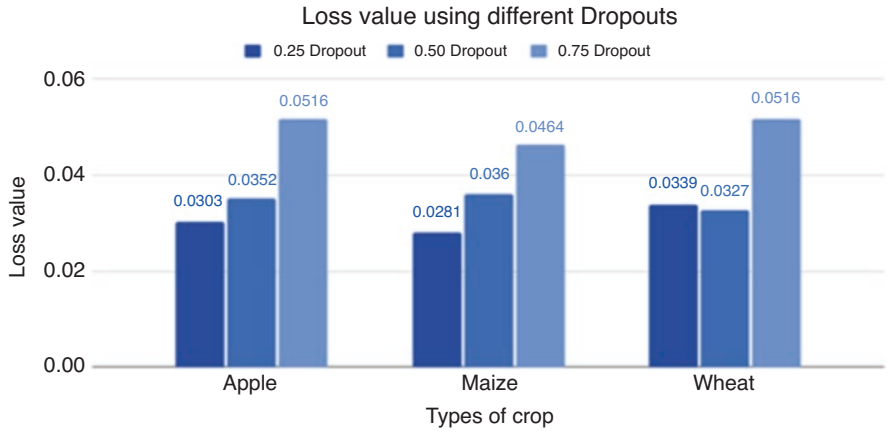


Fig. 4.42 Bar graph representing the varying loss values with different dropout values

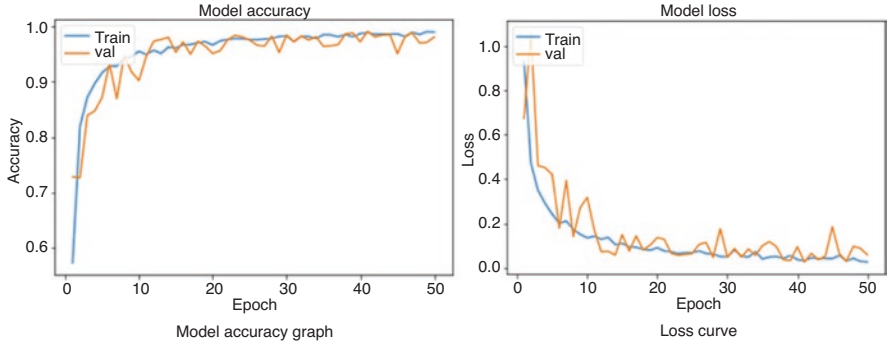


Fig. 4.43 Sparse validation performance curve for crop 'Apple' with dropout = 0.25

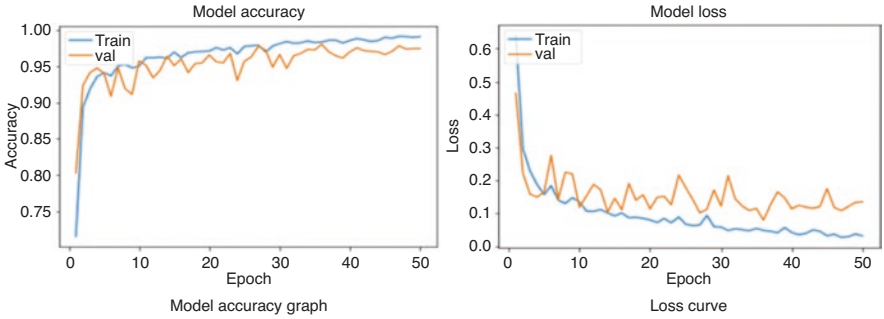


Fig. 4.44 Sparse validation performance curve for crop 'Maize' with dropout = 0.25

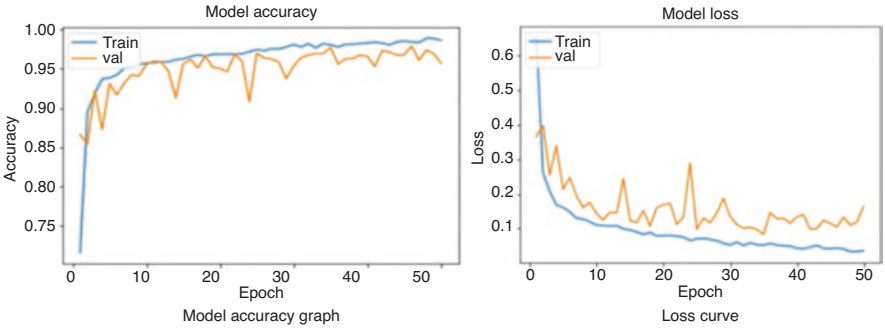


Fig. 4.45 Sparse validation performance curve for crop 'Wheat' with dropout = 0.5

Table 4.10 Accuracy of defined crops

Crop Type	Accuracy (in %)
Apple	98.93
Wheat	98.70
Maize	98.82

4.5.3 Fertilizer Prediction

As mentioned earlier in Sects. 4.4.2 and 4.4.3, Adam is the best-known optimizer used in CNN architecture. Adam is an acronym for the Adaptive Moment Estimation Algorithm, which estimates moments and utilizes them to optimize a function. It combines the gradient descent with the momentum algorithm and RMSProp algorithm.

Figures 4.46 and 4.47 show the validation accuracy score and validation loss parameter among various used optimizers like RMS Prop and SGD, which is why Adam is considered the best optimizer.

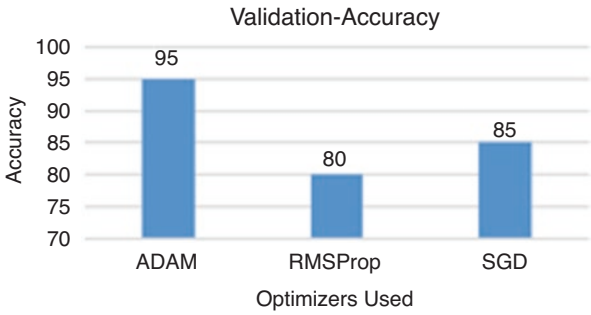


Fig. 4.46 Validation accuracy of various optimizers

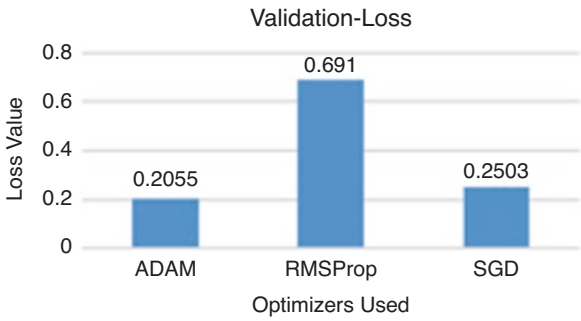


Fig. 4.47 Validation loss values of various optimizers

As is depicted in Fig. 4.46, the Adam optimizer's accuracy is the best optimizer. The accuracy used here is sparse categorical accuracy. Similarly in Fig. 4.47, Adam optimizer shows the lowest loss value among others. Less is the loss function, and more efficient is our model. So from Fig. 4.47, it is evident that RMSProp gives the maximum loss, hence the predictions made by using that optimizer are too far from the ground truths. The optimizer Adam gives the least amount of loss and greater accuracy. Hence, Adam optimizer is used.

Fewer epochs are used so that the neural network generalizes better on unseen data. On the other hand, multiple epochs help the neural network see the previous data and readjust the various parameters of the model so that our model is not biased towards the last few data points while being trained.

An optimal value must be chosen as epochs can affect the distance between predicted and actual values. As evident from the graphs in Fig. 4.48, the model with 500 epochs has reached the global minimum resulting in the lowest value of the loss. As the epochs are decreased to 250 and beyond, the loss increases. Similarly, when the epochs are increased to 1000, the loss increases significantly. Both the extreme cases are detrimental to the prediction model. Hence, for our model, the minimum loss is obtained at 500 epochs.

Our dataset has 99 rows of data for which the optimal range of epochs for our model was 350 to 700 beyond which the model can be erroneous. The best accuracy was obtained at around 500 epochs. If epochs are less than 350, there are chances of

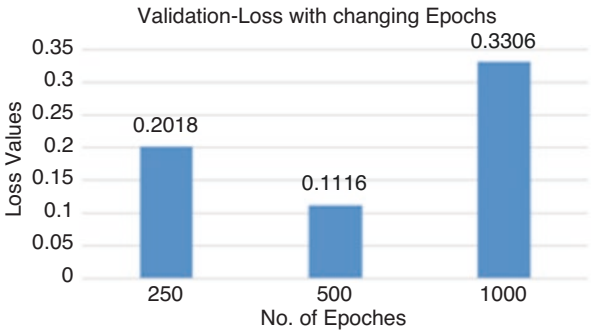


Fig. 4.48 Validation loss values with changing Epochs

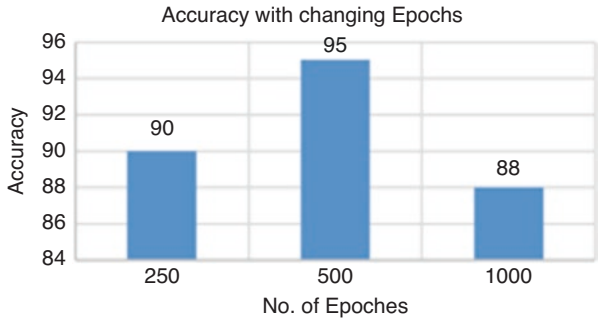


Fig. 4.49 Accuracy with changing Epochs

underfitting. Consequently, if epochs are taken more than 1000, there are chances of overfitting. So an optimal value of 500 epochs gave the best accuracy as depicted by the graphs in Fig. 4.49.

Concerning the detailed analysis of the heatmap in Sect. 4.5.1, we observe that in Fig. 4.50 the attributes are phosphorus, potassium, nitrogen, moisture, humidity and temperature. Among these, temperature and humidity are positively correlated while phosphorus and nitrogen are negatively correlated.

As discussed earlier we have set the optimal number of epochs used for training as 500. Along with only considering the number of epochs, we observed that although the accuracy was good still the validation curve showed a slight tendency of overfitting which was regularized using a dropout layer. After iterating our ANN model through different values of dropout percentages, we derived that with a dropout percentage of 25 we achieved a good accuracy and eliminated overfitting (Fig. 4.51).

Figure 4.52 shows the change in accuracy and loss function values obtained at various dropout percentages at an optimal value of 500 epochs. Maximum accuracy and minimum loss were obtained with the dropout parameter value of 0.25, hence we used this proposed technique.

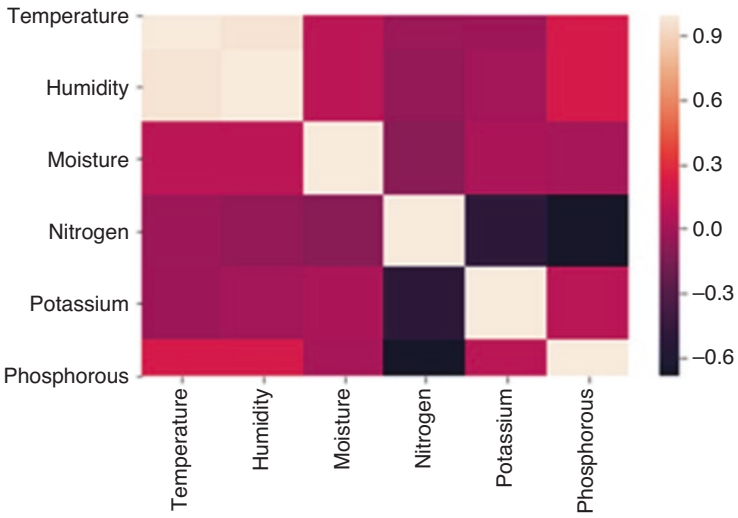


Fig. 4.50 Heatmap showing the correlation between attributes in fertilizer prediction

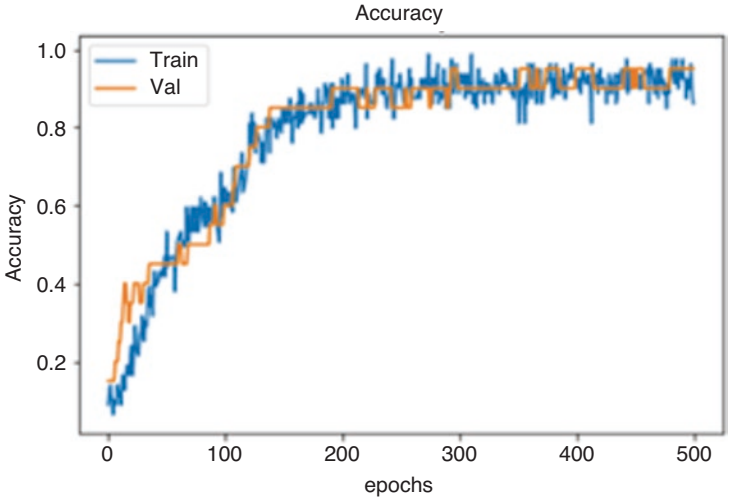


Fig. 4.51 Graph between epochs and accuracy for train and validation data

The optimal value of the loss is obtained within a range of 200–500 epochs. The values within this range returned a minimum loss. As in Fig. 4.53, the graphs show a decreasing trend which concludes our model to be well trained. Furthermore, the gap between training and validation loss curves is less or somewhat superimposing, proving that our proposed model is in good agreement with the general trend of perfect CNN architecture.

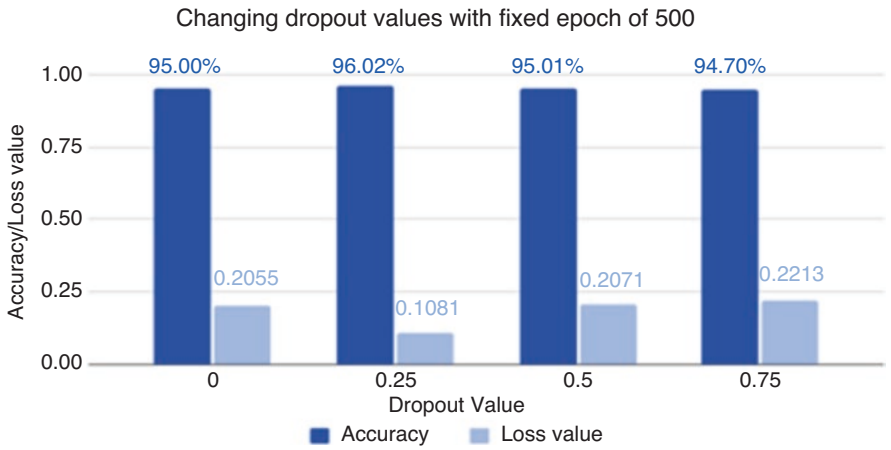


Fig. 4.52 Comparing accuracy and loss function for varying dropout values

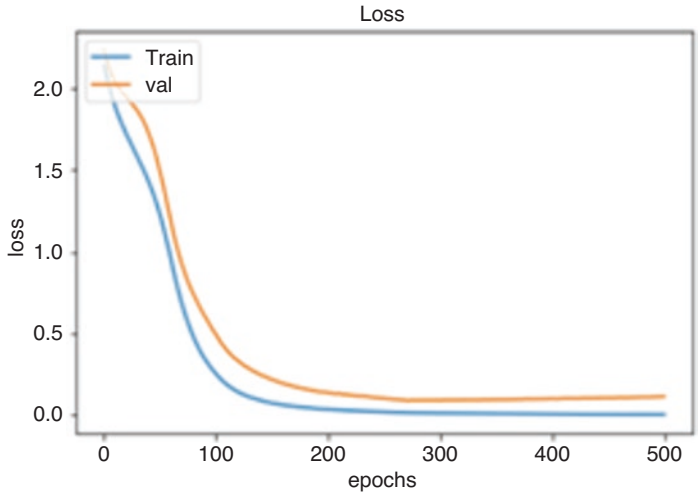


Fig. 4.53 Variation curve between epochs and loss function

4.6 Conclusions

Agriculture contributes to a substantial percentage of our Indian economy. But due to a lack of updated technology, the agricultural sector still suffers from various problems. This chapter aims to propose a machine learning kit that contains different data mining techniques for crop prediction and farm management. Automating such processes will increase the productivity of a farm to a large extent. Several techniques like CNN, ANN and random forest classifier will aid the farmers to

understand what is best for their growth, owing to which their profits can increase. The various alterations made in the hyper-parameters of these techniques, helped us pick the most optimum parameters to build our model. Accuracy close to 95% was achieved in the first stage of our model of crop prediction using Random Forest Classifiers. Also, among the given input parameters, principal component analysis elicited criteria like rainfall and humidity to be most imperative. Secondly, for disease detection in crops, our designed CNN architectural model with 4 hidden layers and softmax activation function gave an accuracy of 98%. Lastly, our ANN model to predict fertilizers suitable for any infected soil gave us an accuracy of 95%, using adam optimizer and regularization dropout value set to 0.25. Also, using heatmaps, parameters like temperature and humidity were found to be positively correlated.

Machine learning and deep learning have been applied in various industries and if the agriculture sector adopts them quickly, we can see a boom in the agro-economic conditions. Precise and intelligent agriculture will be a research hotspot soon. With different advancements in AI and machine learning, the long wait for an intelligent farming solution will soon be over.

Further enhancement of this model is to use different IoT devices to directly gain on-field values of soil moisture, and NPK values and serve it on cloud technologies where the ML models can be hosted. This will help in understanding high dimensionality patterns between seasonal climatic changes, which can help predict the effects of drought and severe climatic repercussions.

References

1. Jambekar, S., Nema, S., & Saquib, Z. (2018). Prediction of Crop production in India Using data mining techniques. In *Fourth International Conference on Computing Communication Control and Automation (ICCUBEA).*
2. Madhukar, A., Kumar, V., & Dashora, K. (2020). Spatial and temporal trends in the yields of three major crops: Wheat, rice and maize in India. *International Journal of Plant Production, 14*, 187–207. Springer.
3. Singh, J., Singhal, N., Singhal, S., Sharma, M., Agarwal, S., & Arora S. (2018). Environmental implications of rice and wheat stubble burning in North-Western states of India. In *Advances in health and environment safety* (Transactions in civil and environmental engineering) (pp. 47–55). Springer.
4. Adebiyi, M. O., Ogundokun, R. O., & Abokhai, A. A. (2020). *Machine learning-based predictive farmland optimization and crop monitoring system* (pp. 1–12). Scientifica.
5. Pathak, S., Jain, N., & Bhatia, A. (2012). *Crop residues management with conservation agriculture: Potential, constraints and policy needs.* Published by Indian Agriculture Research Institute. https://www.researchgate.net/publication/256378461
6. Talaviya, T., Shah, D., Patel, N., Yagnik, H., & Shah, M. (2020). Implementation of artificial intelligence in agriculture for optimization of irrigation and application of pesticides and herbicides. *Artificial Intelligence in Agriculture, 4*, 58–73. Elsevier.
7. Kansal, N., Bhushan, B., & Sharma, S. (2021). Architecture, security vulnerabilities, and the proposed countermeasures in Agriculture-Internet-of-Things (AIoT) Systems. *Internet of Things and Analytics for Agriculture, 3*, 329–353. Springer.

8. Mehta, S., Bhushan, B., & Kumar, R. (2022). Machine learning approaches for smart city applications: Emergence, challenges and opportunities. *Recent Advances in Internet of Things and machine Learning*, 147–163. Springer.
9. Verma, B., Sharma, N., Kaushik, I., & Bhushan, B. (2021). Applicability of machine learning algorithms for intelligent farming. In *Advanced soft computing techniques in data science, IoT and cloud computing* (pp. 121–147). Springer.
10. Environment: No smoke without fire. http://www.businessworld.in/article/Environment-No-Smoke-Without-Fire/14-11-2018-164129/
11. Pathak, H., Aggarwal, P. K., Roetter, R., Kalra, N., Bandyopadhaya, S. K., Prasad, S., & Van Keulen, H. (2003). Modelling the quantitative evaluation of soil nutrient supply, nutrient use efficiency, and fertilizer requirements of wheat in India. *Nutrient Cycling in Agroecosystems, 65*(2), 105–113. Springer.
12. Kumar, L., & Indira, M. (2017). Trends in fertilizer consumption and Foodgrain production in India: A co-integration analysis. *SDMIMD Journal of Management, 8*(2), 45–50.
13. Tripathi, M. K., & Maktedar, D. D. (2016). Recent machine learning based approaches for disease detection and classification of agricultural products. In *International Conference on Computing Communication Control and Automation (ICCUBEA)*.
14. Turkoglu, M., Hanbay, D., & Sengur, A. (2022). Multi-model LSTM-based convolutional neural networks for detection of apple diseases and pests. *Journal of Ambient Intelligence and Humanized Computing, 13*, 3335–3345. Springer.
15. Liakos, K., Busato, P., Moshou, D., Pearson, S., & Bochtis, D. (2018). Machine learning in agriculture: A review. *Sensors, 18*(8), 2674. MDPI.
16. Tidake, A. H., Sharma, Y. K., & Deshpande, V. S. (2019). Design efficient model to increase crop yield using deep learning. In *International conference on innovative trends and advances in engineering and technology* (pp. 221–226). IEEE.
17. Veenadhari, V., Misra, B., & Singh, C. (2014). Machine learning approach for forecasting crop yield based on climatic parameters. In *International conference on computer communication and informatics* (pp. 1–5).
18. Lobell, D. B. (2007). Changes in diurnal temperature range and national cereal yields. *Agricultural and Forest Meteorology, 145*(3), 229–238.
19. Haynes, R. J., & Naidu, R. (1998). Influence of lime, fertilizer and manure applications on soil organic matter content and soil physical conditions: A review. *Nutrient Cycling in Agroecosystems, 51*, 123–137.
20. Ashraf, M. A. (2012). Waterlogging stress in plants: A review. *African Journal of Agriculture Research, 7*(13), 1976–1981.
21. Sibiya, M., & Sumbwanyambe, M. (2019). A computational procedure for the recognition and classification of maize leaf diseases out of healthy leaves using convolutional neural networks. *AgriEngineering, 1*(1), 119–131. MDPI.
22. Geetharamani, G., & Pandian, A. J. (2019). Identification of plant leaf diseases using a nine-layer deep convolutional neural network. *Computers & Electrical Engineering, 76*, 323–338. Elsevier.
23. Singh, A., & Arora, M. (2020). CNN based detection of healthy and unhealthy wheat crop. In *International Conference on Smart Electronics and Communication (ICOSEC)*. IEEE.
24. Subramanian, M., Narasimha, L.V.P., Janakiramaiah, B., Mohan, B.A., & Ve, S.K. (2022). Hyperparameter optimization for transfer learning of VGG16 for disease identification in corn leaves using Bayesian optimization. *Big Data*.
25. Alharbi, A.G., & Arif, M. (2021). Detection and classification of apple diseases using convolutional neural networks. In *International Conference on Computer and Information Sciences (ICCIS)* (pp. 1–6).
26. Bhaya, W. (2017). Review of data Preprocessing techniques in data mining. *Journal of Engineering and Applied Sciences, 12*, 4102–4107.

27. Raju, V. N. G., Lakshmi, K. P., Jain, V. M., Kalidindi, A., & Padma, V. (2020). Study the influence of normalization/transformation process on the accuracy of supervised classification. In *International Conference on Smart Systems and Inventive Technology (ICSSIT)* (pp. 729–735).
28. Gulati, P., Sharma, A., & Gupta, M. (2016). Theoretical study of decision tree algorithms to identify pivotal factors for performance improvement: A review. *International Journal of Computer Applications, 141*(14), 19–25.
29. Raileanu, L., & Stoffel, K. (2004). Theoretical comparison between the Gini Index and Information Gain Criteria. *Annals of Mathematics and Artificial Intelligence, 41*, 77–93.
30. Ali, J, Khan, R., Ahmad, N., & Maqsood, I. (2012). Random forests and decision trees. *International Journal of Computer Science Issues (IJCSI)*, *9*.
31. Davis, J., & Goadrich, M. (2006). The relationship between precision-recall and ROC curves. In *International conference on machine learning* (pp. 233–240). ACM.
32. https://i.pinimg.com/564x/fc/b8/35/fcb8358bbc2fd692e9ce9d85e0c2ebbf.jpg
33. Kawahara, M., Inoue, T., & Nishio, Y. (2010). Image processing application using CNN with dynamic template. In *International workshop on cellular nanoscale networks and their applications (CNNA 2010)*.
34. O'Shea, K., & Nash, R. (2015). An introduction to convolutional neural networks. *ArXiv*, abs/1511.08458.
35. Gholamalinejad, H., & Khosravi, H. (2020). Pooling methods in deep neural networks, a review. *arXiv*.
36. Jadhav, S. B., Udupi, V. R., & Patil, S. B. (2020). Identification of plant diseases using convolutional neural networks. *International Journal of Information Technology, 13*, 1–10.
37. Mumtaz, D., Jakhetiya, V., Nathwani, K., & Subudhi, B. N. (2021). Non-intrusive perceptual audio quality assessment for user-generated content using deep learning. *IEEE Transactions on Industrial Informatics*.
38. Sharma, S., Sharma, S., & Athaiya, A. (2020). Activation functions in neural networks. *International Journal of Engineering Applied Sciences and Technology, 4*(12), 310–316.
39. Gupta, S., Gupta, R., Ojha, M., & Singh, K. P. (2018). A comparative analysis of various regularization techniques to solve overfitting problem in artificial neural network. In *Communications in Computer and Information Science* (pp. 363–371).
40. Mazumdar, A., & Rawat, A. S. (2019). Learning and recovery in the ReLU model. In *Annual Allerton conference on communication, control, and computing (Allerton)* (pp. 108–115).
41. Zhang, Z. (2018). Improved Adam optimizer for deep neural networks. In *International Symposium on Quality of Service (IWQoS)* (pp. 1–2). IEEE.
42. Poernomo, A., & Kang, D.-K. (2018). Biased dropout and crossmap dropout: Learning towards effective dropout regularization in convolutional neural network. *Neural Networks, 104*, 60–67.