PAPER • OPEN ACCESS

An IoT Based Gradient Descent Approach for Precision Crop Suggestion using MLP

View the article online for updates and enhancements.

An IoT Based Gradient Descent Approach for Precision Crop Suggestion using MLP

P. Kanaga Priya[1], Dr. N. Yuvaraj[2]

[1]PG Student, Department of Computer Science and Engineering, KPR Institute of Engineering and Technology, Coimbatore, India

[2]Associate Professor, Department of Computer Science and Engineering, KPR Institute of Engineering and Technology, Coimbatore, India

[1]kanagapriyakpr@gmail.com , [2]drnyuvaraj@gmail.com

Abstract—Agriculture plays a prime role in providing food to a massive population. It has a predominant role in improving the economic development of our nation. In recent days due to the changes in weather pattern, the crop cultivation has become the greatest challenge. This directly or indirectly affects the productivity of the crops. Hence to overcome from this and to the improve productivity of the crops new technologies can be brought up for usage. The usage of these new technologies will convert traditional farming practices into precision farming. The new technology specified here includes the data analysis and Internet of Things (IoT). As it is revealed that it improves the productivity of the crops by maintaining the finest crop health. Although changes had been adopted in the recent agricultural practices the major issue yet not gets resolved. One such main issue yet to be resolved is cultivating precise crop at the precise time. This can be done with the help of deep learning algorithm such as Artificial Neural Network (ANN), which is found to be an effective one for predicting the precise crop. Hence the proposed system aims at helping the farmers by providing valuable insights to them. These insights can be driven with the help of parameters such as soil moisture level, humidity, temperature and pH collected from the sensors using IoT. In addition to these other parameters include soil type, land type, and the area sown. The crop prediction Graphical User Interface (GUI) system will take these valuable inputs and give suggestions using the crop suggestion user interface. The prediction is made using the Deep Neural Network (DNN) which predicts about the crop to be get cultivated, fertilizer to be used, production range etc. The crop suggestion system greatly helps the farmers to take a valuable decision.

Keywords—Agriculture, Artificial Neural Network (ANN), Deep Neural Networks (DNN), Crop Suggestion GUI, Crop Suggestion System, Deep Learning, Precision Farming, Prediction, IoT, Suggestion.

1. INTRODUCTION

In the agriculture sector, the main challenges found are the lack of knowledge about the changing variations in climate. Since each crop has its own suitable climatic features, the challenges faced in agriculture can be handled with the help of precise farming techniques. The precision farming helps us to meet the food demand and maintain sustaining productivity of the crops thereby increasing the yield rate of production. In India, the growing needs and population want sustainable agriculture. Although many steps have been carried out to minimize crop loss, there are some disadvantages found in the traditional method. To overcome the disadvantages in traditional farming practices an alternative method such as precision farming can be used for crop selection. The precision farming involves the use of IoT and prediction system which plays an important aspect of decision making. In recent days the agriculture sector aims at bringing smart agriculture practices through the Internet of Things to reduce the crop loss. The IoT system with the help of sensors collects data from the field. The collected data from sensors can then be given to the prediction framework for obtaining suggestion [1]. The two major problems in agriculture found among farmers are the crop selection and the changing climatic condition. These problems can be addressed with the help of existing prediction and monitoring methods, but there is no optimum solution for the crop suggestion. Some of the drawbacks found in the existing system are lack of adequate nutrient supply, the improper analysis, choosing effective algorithms, and efficient selection of attributes all these parameters may affect the yield of the crops.

The drawbacks found in the existing system are aimed to be overcome in the proposed system which includes maximizing the yield of crops, real-time analysis of crops using IoT, the selection of efficient attributes, making smarter decisions and getting a better yield. There is a need to use a more efficient algorithm for early prediction of crops [2]. The Deep Neural Network (DNN) has been found to be more effective for crop prediction and crop suggestion model. In technology point of view by choosing correct factors most useful data can be generated using DNN by intake of various parameters and this generated data helps the farmers by suggesting a right crop to be sown [33]. The principal aim of the crop suggestion model is to suggest precise crop for getting high yield. The goal is to minimize crop loss by choosing the right crop, although earlier crop detection methodologies suggest crops the suggestion varies based on the type of regions, soil characteristics, and other factors. Hence accuracy level of suggesting crops may also vary based on the type of algorithm chosen, so it is necessary to choose the right algorithm with selective features which then maximize the accuracy level of crop suggestion based on favorable condition. Over these days it has been seen that Artificial Neural Network is found to be more effective one for predicting crop and its yield.

2. LITERATURE SURVEY

Niketa Gandhi, Owaiz Petkar, Leisa J. Armstrong used Neural Network approach for predicting the yield of rice. For predicting the yield the algorithm takes parameters such as Maximum temperature, average temperature, minimum temperature, area, production, crop, etc. The collected data is then validated and processed using WEKA tool. But this work suggests that the prediction capabilities of an ANN-based model can be enhanced by using additional crop related parameters [27]. Vijo T Varghese, Kalyan Sasidhar, Rekha aims at helping farmers by the idea of providing Wireless sensor network for the crop yield estimation, so that right crop can be chosen at right time. It consists of sensors like temperature, rainfall, and moisture sensor to monitor the plants. These sensors will sense the data and will intimate the changes in the environment. It will also send information about water required for crops in concern with the changing environmental parameters [24]. Aruul Mozhi Varman S, Arvind Ram Baskaran, Aravindh S, Prabhu E proposed an IoT and deep learning based smart agriculture systems. This system monitors and collects the soil parameters from the field with the help of a wireless sensor network. The collected data is then uploaded in the cloud. Finally, the systems suggest best irrigation practices to the farmers by predicting the crop to be sown for next crop rotation. This information will be sent as an SMS to the farmers. The parameters include soil temperature, atmospheric temperature, and humidity. This system suggests further improving the effectiveness by predicting the suitable time for applying pesticides, fertilizer, and manures [9].

R. Ramya, C. Sandhya, R. Shwetha proposed sensing technology based smart farming system. Their system basically uses sensors such as pH, temperature, moisture, and sensor to measure the intensity of light. The sensors are positioned in such a way in which it effectively senses about the environmental conditions. This system simply collects the data from the environment without any processing. Hence it is suggested to apply the collected data into the algorithm for better crop production [2]. Giritharan Ravichandran, Koteeshwari R S proposed a predictor for the crop which gives advice to the farmers with the help of smartphones using Artificial Neural Network. In this, a system is proposed to prevent crop loss. The proposed system helps the farmers by choosing them the crop that could benefit the farmers to get higher crop productivity. A system is developed using an android application that uses ANN. The algorithm takes parameters such as pH, nitrogen, phosphate, potassium, temperature, depth and rainfall. The advisor will suggest the crop and the predicted crop productivity. The advantage of this system is it is easily accessible by the farmers. This system would be better if the advisor parameters can be driven from real-time [3].

Suhas Athani, CH Tejeshwar, Mayur M Patil, Priyadarshini Patil, Rahul Kulkarni proposed IoT enabled soil moisture monitoring for predicting the future harvest. This system detects soil moisture level for obtaining good plant growth. Initially, soil samples are given as an input to the sensors then the respective pH content, salinity content and moisture content are measured using Wi-Fi shield. From the database, the moisture reading can be fetched for Neural Network processing. In the final module, the generated correction factor is given as an input and the result produced by the neural

network will get displayed in the form of mobile application. This work suggests neural network can be used for predictions based on seasonal variations [30]. Umair Ayub, Syed Atif Moqurrab proposed a method for crop disease prediction using data mining. This work used various data mining approaches such as Neural Network, Random Forest, Support Vector Machines, Decision Tree, and K-Nearest Neighbor. These classifiers are used for calculating the damage caused by grass grub disease. The result produced by the classifiers is then evaluated using different evaluation criteria such as accuracy, precision, recall, and F1-score. Finally, the evaluation result suggests that the Neural Network and random Forest produces better results compared to the other classifiers. Hence the system suggests improving the results by applying some advanced techniques [32].

Rekha P, Maneesha V. Ramesh, Venkata Prasanna Rangan, Nibi K V in their study proposed a prediction framework for farming which includes sensors that sense the agricultural characteristics with the help of IoT framework hence the farmers will be given advice. These sensors such as soil moisture temperature and PH are deployed in the agricultural field and it uses Radio Frequency communication to transfer the data. The transferred data are given as an input to the Decision Support System (DSS). The DSS will receive the input and give suggestion to the farmers about the detail of crop such as fertilizer usage, time for usage and time for irrigation etc. This information reaches the farmers with the help of an android application [31]. Thus it is found that the farming practices can be enhanced by bringing technological revolution. There is a need for proper nutrients supplement to be given to the soil to overcome the problems faced by changing climatic factors. Thus the developed suggestion model will provide the needed information to cultivate the right crop based on the parameters given by the farmers.

3. METHODOLOGY

The advancement in recent technological development with the use of the Internet of Things and Deep Learning has made everything possible. The IoT system is greatly helpful in collecting real-time data using the sensors. The valuable data can be utilized in a way by which it can be fed to the trained deep learning algorithm such as Artificial Neural Network for making the prediction. The outcome greatly helps in finding out a suitable crop to be sown in the respective field area. This section describes the dataset, features, preprocessing phase, IoT design, and Deep Neural Network.

A. Dataset

The dataset used for the crop suggestion model is collected from the open source websites which contain various parameters related to the cultivation of crops. The dataset is prepared in a way in which it consists of about thirteen attributes like soil type, land type, soil moisture, temperature, humidity, pH, the area sown, soil N, soil P, soil K, rainfall, production, class label. Among these attributes, the parameters like soil type, land type, temperature humidity, the area sown, soil moisture, and pH play an important role in crop suggestion. The crops that are taken for consideration include rice and maize.

B. Feature Description

The soil is the birthplace of nutrients which acts as a medium for plant growth. It also helps in retaining water thereby helps in proper nutrient supply. The most common soil types taken are red soil and black soil. The feature land type is differentiated based on seasonal variations such as wetland, dry land, and garden land. The feature soil moisture level is measured from the respective soil. It indicates the needed moisture level for plant growth. The temperature represents the optimal temperature in which the crop grows. Similarly, the humidity has the greatest effect on the surrounding environment which measures moisture level present in the air. The soil moisture sensor, temperature and humidity sensor readings can be measured and used for processing. The soil pH affects plant growth in several ways. The optimal pH range is 5.5-7.0; if the soil is not in the prescribed pH range then the soil needs to be treated. The features soil Nitrogen (N), Phosphorous

(P), and Potassium (P) simply called NPK is helpful for boosting up plant growth by improving the soil fertility.

C. Data Preprocessing

The dataset of the crop suggestion model needs to be preprocessed since the deep learning algorithm cannot handle noisy, incompatible and incomplete data. The noisy data means that it contains outliers and errors. The incompatible or inconsistent data in the sense it represents the difference in features. The incomplete data in the sense it represents a lack of features or attributes values in a dataset. The basic steps involved in any data preprocessing phase include data cleaning, data integration, data transformation, and data reduction.

1) Encoding Categorical Data using Label Encoding

The dataset of the crop suggestion model can be simply brought into two categories numerical and categorical data. The numerical data is one in which the data will be in a numeric format and categorical data will be in the form of strings. The python does not deal with categorical data it should be converted into numerical values. This can be done with the help of LabelEncoder( ) class. For example, the attribute land type can be encoded in a way that label '0' for the dry land, '1' for the garden land, and '2' for the wetland. This can be used where priority is needed in the features.

2) Encoding Categorical Data using One Hot Encoding

During label encoding, the encoder will assume that 1>0 and 2>1. This implies that label 1 has a higher priority than label 0, and label 2 has a higher priority than label 1, and label 0. More priority will be given to wetland, the next to garden land and then to the dry land. But this type of encoding is not applicable for all types of features. This problem can be solved with the help of OneHotEncoder( ) class which encodes the column to be encoded by splitting across multiple columns and thereby replacing it with 0 and 1. For example, the attribute soil type contains features like red soil, and black soil that can be encoded as '0' for red soil, '1' for black soil along with these two dummy features will be added for both red and black soil containing the values of 0 and 1.

3) Numerical and Categorical Data Preprocessing

To preprocess these types of data set basic preprocessing step applied here is data cleaning. The main step involved in data cleaning is dealing with the missing values. If the missing values in the dataset are not treated properly then the result drawn using that dataset will get affected. There are three methods for handling the missing values that include deleting the entire row, calculating mean, median or mode, and filling missing value by zero. The best method found for the crop suggestion is filling the missing values of the null value feature by zero. This is an effective method since the dataset was taken into account contains a minimal number of null value entry. The filling of missing value is also found to be an efficient method for preventing data loss. This method will also produce a better prediction outcome when compared to the other two methods.

4) Feature Scaling

The feature scaling is used to standardize the varying features in a dataset which is to limit the feature range. In data preprocessing the feature scaling, is also called normalization which comes under data transformation. One of the methods involved in feature scaling is called feature rescaling otherwise called min-max scaling or min-max normalization. This feature rescaling method is the simplest method in feature scaling for handling numerical features. By which the features can be scaled in the range of [0, 1] or [-1, 1]. The target range can be selected depending on the nature of

the data set. The variable with the huge range will have an effect on the outcome of the result and it will make less accurate predictions. The rescaling can be used to overcome this problem by scaling all the features in the same range. This can be done with the help of MinMaxScaler( ) provided by scikit-learn library. By using this all the features in the dataset can be scaled in the range of 0 and 1. The formula for MinMaxScaler is mentioned in (1),

$$X = x - min(x) / max(x) - min(x) \qquad (1)$$

Where X is the normalized data value, x is the original value to be normalized, min(x) is the minimal value of the feature to be normalized, and max(x) is the maximum value of the feature. For the feature scaled value binarization can be applied to convert numerical feature value into a boolean value.

D. IoT System Design

The main role of IoT sensors is extracting real-time data from the field area to be predicted. As known that IoT is a part of recent technological development from which valuable information can be driven. Hence the cost-effective sensors such as soil moisture sensor, humidity sensor, temperature sensor, and pH meter are used for measuring parameters such as soil moisture level, surrounding temperature, humidity, and soil pH. These sensors are greatly helpful for sensing the weather pattern and nutrient found in the soil. The sensed information with the help of communication module reaches the database with the help of a microcontroller. The data in the database can be used for making future climatic pattern analysis These sensors can also be used for identifying the characteristics of the cultivation land at an earlier stage. Monitoring these characteristics seems to be a simple one. But this information can be used to make great changes in the crop cultivation practices. The design of the IoT module from the field to database storage is shown in Fig. 1.
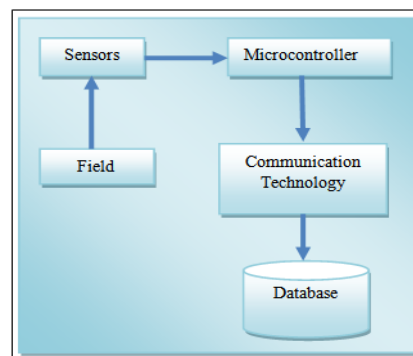

Fig. 1: IoT System Design

E. *Artificial Neural Network for Building Prediction Model*

The deep learning algorithm is one in which the training or testing of data takes over a number of layers during these phases the features are learned by the algorithm itself without the need for manual feature extractor. The Deep Neural Network or Artificial Neural Network (ANN) is based on Neural Network which is a supervised learning algorithm. The ANN contains a system of neuron which is similar to the human brain. Each neuron accepts an input with weight and processes them and it responds with an output [9]. The Artificial Neural Network Algorithm works in a way in which it will be trained with the respective data and if an instance is given it can make a prediction for the future. The Deep Learning Neural Network contains more hidden layers. Each hidden layer identifies more complex features. These hidden layers are used for extracting the features, thus in Deep ANN, there is no need to extract the features manually. The flow of deep learning based Artificial Neural Network is shown below in Fig. 2.
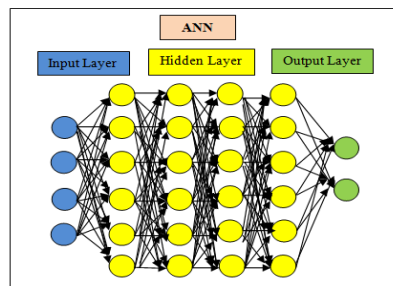
Fig. 2: Deep Neural Network

4. IMPLEMENTATION

The implementation phase includes various phases like collecting data from IoT, designing the ANN model, architecture of the implemented model, training the preprocessed data, testing the trained model, minimizing the error rate by learning rate, choosing the best activation functions, and improving the accuracy with the help of gradient descent and backpropagation.

A. IoT for Data Collection

The IoT implementation part contains collecting data from the respective field area. In a particular field area, the parameters such as moisture, temperature, humidity, and pH are monitored. The monitored parameters are then stored and given as an input to the GUI. The data collection from IoT in the field area is illustrated in Fig. 3.
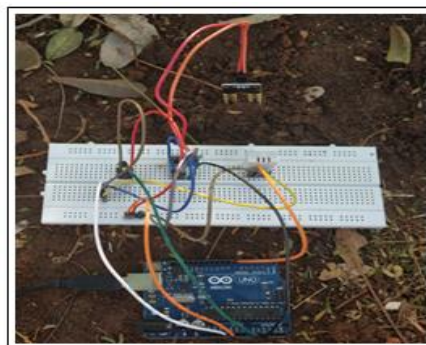

Fig. 3: IoT for Data Collection

The role of Arduino soil moisture sensor is to measures the water content since it is the basis for providing nutrients to the soil. The LM35 temperature sensor is used to measure the temperature present around the surrounding field area. The DHT22 humidity sensor measures the surrounding air temperature and moisture. In a similar way, the pH of the soil should be constantly maintained because the availability of soil nutrients and the growth rate of crops depends on the pH. The pH level can be measured using the pH meter. These sensors collect the information using an Arduino microcontroller which is responsible for controlling the sensors. The sensed information with the help of Wi-Fi will get stored in an excel file. For storing in an excel file Arduino software is used with the help of Parallax data acquisition tool named PLX-DAQ. The collected data from IoT sensors using PLX-DAQ is shown in Table I.

TABLE 1:   Collected Data using PLX-DAQ

S. No	TIME (AM)	Temperature (C)	Humidity (%)	Moisture (%)
1	6:33:34	31.25	42.4	76.54
2	6:33:36	31.25	42.3	83.19
3	6:33:38	31.25	42.4	82.6
4	6:33:40	31.25	42.6	82.6
5	6:33:42	31.25	43.5	82.21
6	6:33:44	31.25	43.6	82.21
7	6:33:46	31.25	44	81.82
8	6:33:48	30.76	44	81.43
9	6:33:50	30.76	43.8	80.16
10	6:33:52	31.25	43.5	80.45
11	6:33:54	31.25	43.2	80.65
12	6:33:56	31.25	43	80.45

B. Model Design by ANN

The Anaconda is an open source python distribution which is currently used for various data science projects. It has many deep learning packages like tensorflow, scikit-learn, and theano. These packages can be installed to train a deep learning model. There are many applications available on the navigator among those spyder and jupyter is chosen for predictive analytics. The jupyter can be used to run a whole code in a partitioned manner. The spyder provides the most powerful editing, visualization, debugging, and analysis option.

1) Package Description

The packages can be installed using the anaconda command prompt. For installing the packages initially, a separate tensorflow virtual environment is created for implementing the ANN algorithm. The created environment is used for installing the needed packages. The packages to be installed include matplotlib, tensorflow, numpy, pandas, and scikit-learn. The matplotlib is used for plotting various types of graphs such as scatterplot, histogram, and two-dimensional plots etc. The tensorflow is a deep learning library mainly used to build neural network models. The numpy package is used to

perform different operations on a multidimensional array and to store the resultant value of the same data type in each column. The pandas package is used for working with labeled data and to perform necessary data manipulation. The sklearn-preprocessing is a function based on scikit-learn which supports splitting a dataset for training and testing, and preprocessing a data using label encoder, MinMaxScaler.

C. Architecture

The architecture of the crop suggestion system contains collecting real-time data for prediction and building the prediction model for Artificial Neural Network algorithm, a user interface for giving inputs and obtaining suggestions. The flow of the architecture starts from preprocessing. Once the preprocessing is made the dataset is undergone training using the ANN algorithm for generating the crop suggestion prediction model. To the generated model test data is given for predicting. During testing the trained model is tested against random input values based on the accuracy rate and error value produced by the prediction model the weights are adjusted using backpropagation by gradient descent. This process is repeated until the error value is reduced and the accuracy rate is improved. The GUI is used for gathering input from IoT sensors and the user, which is then given to the crop suggestion model for giving precision crop suggestion. The architecture of precision crop suggestion system is shown below in Fig. 4. involving training, testing, and prediction.
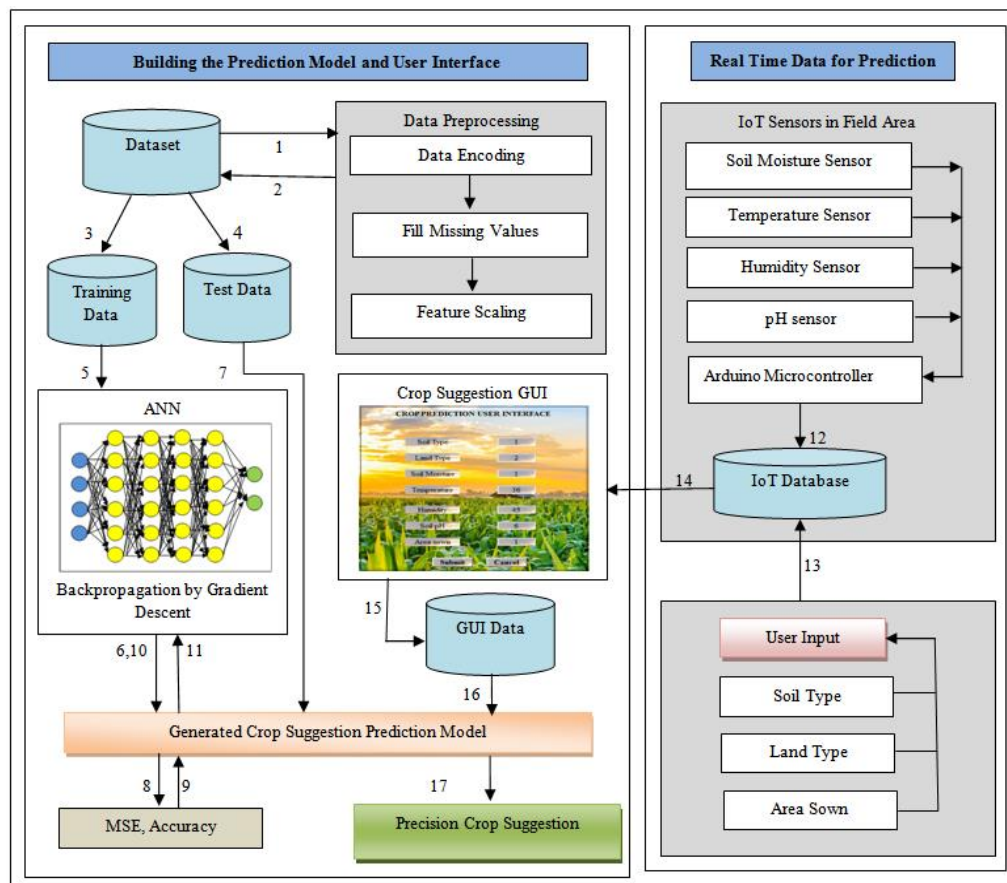

Fig. 4: Precision Crop Suggestion System

1) Training the model

Once the packages are imported the dataset is read using pandas and then it is subjected to preprocessing. The preprocessing involves data encoding, filling the missing values and feature

scaling as discussed in the data preprocessing part of the methodology section. After preprocessing is done the dataset is divided for training and testing part. It is divided in a way that 80% of the dataset is utilized for training and around 20% of the dataset is allotted for testing. The Deep Artificial Neural Network contains various classes of algorithms to perform the implementation. One such class used here is MultiLayer Perceptron (MLP).

2) Learning Rate

To train the model it is needed to mention the learning rate of the algorithm. The learning rate should be low enough for a neuron to get trained. If the learning rate has become high then the model will be trained incorrectly. Based on the learning rate the weights are adjusted in a neural network with the help of gradient descent approach for error calculation. For a good learning rate the loss decreases when the number of epochs increases, for very high learning rate the loss increases, for low learning rate the loss will be somewhat low, and for high learning rate, the loss will be high. The loss is measured in terms of Mean Squared Error (MSE). Hence the training will be reliable if the learning rate is low, so the optimizer will take some time to minimize the loss function. If the learning rate is high then the training will become worse and the loss will be high it cannot be minimized. The most wisdom approach to choose with the learning rate is to start with giving random values and check for minimum loss thereby not delaying in training speed. Initially, a large value can be chosen like 0.3, 0.2 and it can be reduced to values like 0.01, 0.001 with a hundred iterations or epochs. The learning rate chosen here is 0.2 since it produces a low error with high training accuracy as shown in Fig. 5.
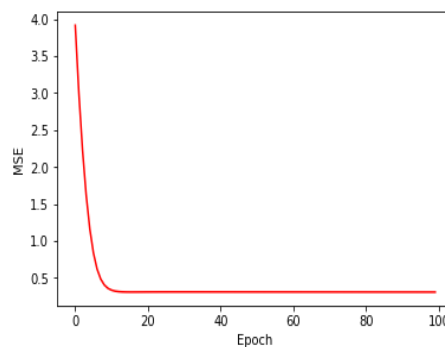

Fig. 5: Effect of Learning Rate on MSE

D. Activation Function

The activation function is one which is used by the artificial neuron. If x1 and x2 are the inputs and w1 and w2 are the weight of the neuron then the output of the typical neuron is mentioned in (2),

$$y = sum\ (weights * inputs) + bias \qquad (2)$$

Where y is the output of weighted input value, and weights denote the value assigned to a neuron while processing, the input denotes the number of inputs fed to the network, and bias is the parameter that is used to adjust the output. The output produced by the neuron may vary from range -infinity to +infinity. In each layer, activation function is used for mapping the input value in terms of (0, 1) and producing the required output. The activation function will decide whether the particular neuron is activated or dead. The three commonly used activation functions are Tanh, ReLU, and Sigmoid. The comparison of different activation functions based on the hidden layer with its test accuracy is shown below in Table II.

TABLE 2:   Comparison of Activation Functions

Hidden	Activation	Hidden	Test Accurac

Layer	Function	Neurons	y
2	Sigmoid	4	91.43%
3	Sigmoid	6	93.75%
4	Sigmoid	9	96.81%
5	Sigmoid	12	94.12%
2	ReLU	4	79.67%
3	ReLU	6	81.56%
4	ReLU	9	84.78%
5	ReLU	12	83.12%
2	Tanh	4	69.34%
3	Tanh	6	73.76%
4	Tanh	9	76.45%
5	Tanh	12	74.89%

E. Activation function over Test Accuracy

In the model design, three activation functions are taken for comparison. The test accuracy of the activation functions like sigmoid, ReLu and Tanh with the respective hidden layers like two, three, four and, five layers are shown in Fig. 6, 7 and 8.
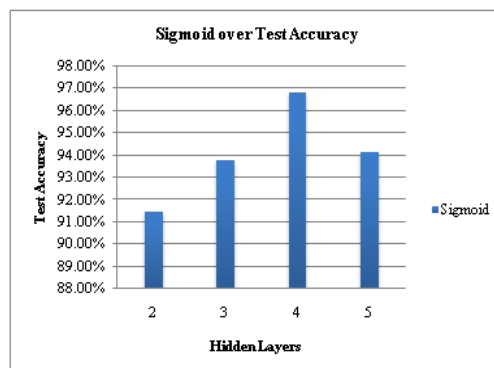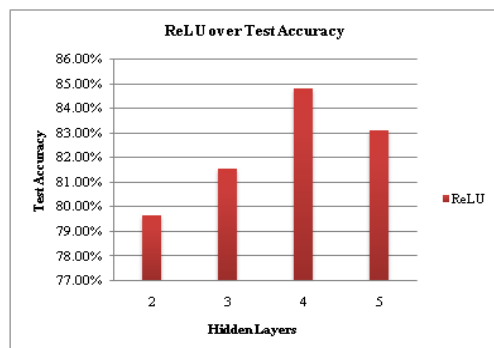

Fig. 6: Sigmoid over Test Accuracy

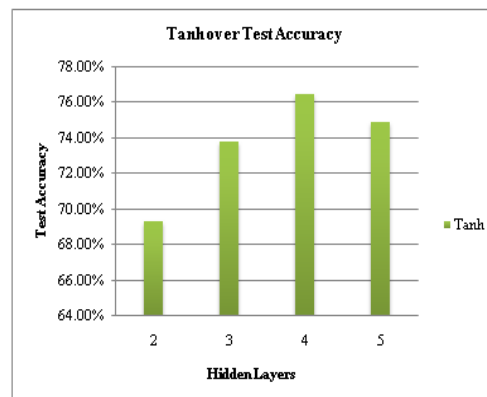**Fig. 7: ReLU over Test Accuracy**

Fig. 8: Tanh over Test Accuracy

It is found that among the three activation functions like sigmoid, ReLU, and tanh the sigmoid produces the highest accuracy of about 96.81% with four hidden layers compared to ReLU and Tanh. The sigmoid function can be used in cases where there is a need to predict the outcome based on some probability measure. The value of the sigmoid curve lies in the range of 0 and 1. The sigmoid activation function will determine the output be zero when the value is less than 0.5 and the outcome is predicted to be one when the value is greater than 0.5. The sigmoid function is mentioned in (3),

$$S(x) = 1/\ 1+e^{-x} \qquad (3)$$

Where $S(x)$ is the sigmoid curve, e is the base of natural logarithms and x is the weighted sum given as an input to the activation function. The consolidated graph of activation functions over test accuracy is shown below in Fig. 9.
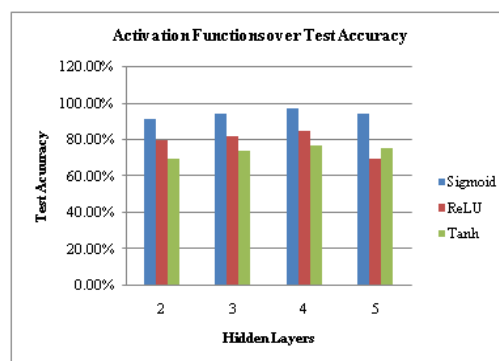

Fig. 9: Activation versus Test Accuracy

1) Role of Hidden Layer

In Deep ANN when the number of data gets increased the accuracy of the neural network will also get increased. For improving the accuracy the Deep Neural Network has more hidden layers in its architecture. As known that in each hidden layer a feature is extracted. Hence, it is predominant one to choose the number of neurons in each of the hidden layers, thus not resulting in overfitting and underfitting. The under fitting occurs in the hidden layer when less number of neurons are used. In a similar way overfitting in the hidden layer occurs with a high number of neurons. The hidden neurons combination set containing the hidden layer neurons with its test accuracy can be used for choosing the respective hidden layer as shown below in Table III.

TABLE 3: Hidden Neurons versus Test Accuracy

Hidden Layer Set	H1	H2	H3	H4	Test Accuracy
S1	2	3	4	5	82.63%
S2	5	6	6	7	84.45%
S3	8	8	8	8	85.87%
S4	4	6	8	10	89.92%
S5	7	8	8	8	90.92%
S6	8	9	10	11	94.56%
S7	9	9	9	9	96.89%
S8	9	10	11	12	94.22%
S9	11	11	11	11	92.54%
S10	10	10	11	13	90.99%
S11	11	11	12	12	87.32%

The hidden layer set S1 to S11 denotes the combination of neurons taken in each hidden layer H1, H2, H3, and H4 for processing. It is found that for S7 set with nine neurons in each layer gives the highest accuracy of about 96.89%. There are about three layers used in the multilayer perceptron model they are one input layer, one or more hidden layer, and an output layer. The number of hidden layers and the number of neurons should be chosen carefully since the hidden layers play a major role in the predicted output. The model contains thirteen neurons in the input layer. The hidden layer taken is about four and the output layer contains two neurons that tell about whether to cultivate rice or maize. It is inferred that in each hidden layer nine neurons can be used for processing. The number of neurons in the hidden layer can be chosen based on its accuracy rate is shown below in Fig. 10.
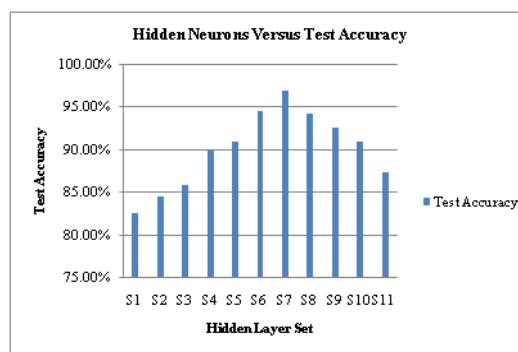
Fig. 10: Hidden Neurons versus Test Accuracy

F. Backpropagation by Gradient Descent

For training the network the MLP uses backpropagation and gradient descent technique. The backpropagation is the training algorithm that has the ability to change the weights automatically and produce an output for a given input. A gradient descent tries to minimize the loss function by

improving the prediction accuracy and it can be accessed as a part of training. During the design of a neural network, the weights are initialized using an approach named random value weight assignment. The backpropagation method is used by gradient descent optimization to adjust the weight of the neurons. This is done by calculating the gradient of the loss function. The gradient descent optimization also helps to reduce the cost of the prediction model. The gradient is a measure that tells about the change in error when there is a change in weight. Hence based on the error produced the weight can be adjusted, this step is needed to check whether the value selected will fit our model. While training it is found that there is a difference in the actual and predicted output this refers there is an increase in error rate. In such cases, backpropagation with the help of gradient descent finds out the minimum error by propagating forward and backward. The weights are adjusted in such a way in which the cost function and loss in error value are to be minimized. Hence at final, the accuracy is improved to be around 97%. The improvement in accuracy rate during the iteration is shown below in Fig. 11.
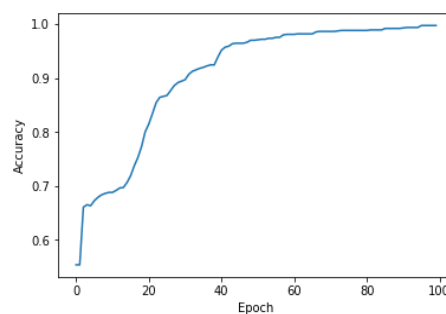

Fig. 11: Effect of MSE on Accuracy

G. Crop Suggestion User Interface

Beyond the scope, the spyder can also be used with the third party software such as Qt Designer. This software is a tool for building Graphical User Interface (GUI). The PyQt5 is a package available in anaconda for creating interactive GUI. This package can be used to run the GUI generated by the Qt Designer by converting the .ui user interface file to the .py python file with the help of pyuic5 (python user interface) utility in the anaconda command line. The crop suggestion Graphical User Interface (GUI) is created using the Qt designer which fetches the value such as soil type, land type, area sown from the user view and remaining data such as soil moisture, temperature, humidity, and pH are taken from the IoT database. The IoT database contains data collected from the respective field area. The crop prediction input GUI is shown below in Fig. 12.

Fig. 12: Crop Prediction GUI

The whole features entered in the GUI are then stored in a GUI database and given for the crop prediction model for prediction. The trained crop prediction model then performs the necessary operations and gives it to the crop suggestion GUI. The suggestion GUI suggests about the crop that can be chosen for cultivation. Hence precision decision is made by the farmers with the help of the predicted result. The crop suggestion GUI is shown below in Fig. 13.

Fig. 13: Crop Suggestion GUI

H. Result and Discussion

The collected dataset is initially divided into training and testing dataset. The training dataset is then given to the ANN model for generating the crop suggestion prediction model. During model generation, the activation function and the hidden layers are carefully chosen to acquire the best results. Once the model is generated test data is given to the model for calculating the error and accuracy. Based on the calculated error value the backpropagation by gradient descent adjusts the error value by assigning weights to the neurons. To the generated model when the inputs are fed using the user interface the model will predict and suggest the crops to be sown with an accuracy of about 96.89%.

5. CONCLUSION AND FUTURE WORK

In the future, there is a need for giving importance to the agriculture sector. Although several works have been carried out using Deep ANN still its performance is increasing day by day. Generally, the Deep Neural Network has high processing capability and its performance cannot be improved with the limited dataset. Hence for an ANN when Deep Learning is applied its performance is improved by considering additional dataset. The current agricultural practices try to overcome the issues by using an efficient prediction method. The precision crop suggestion model is designed in such a way that it can handle the hurdles faced by the farmers. Thus irrespective of the seasonal variations the crop prediction model efficiently suggests the right crop for cultivation. The suggestion given by the GUI greatly helps the farmers to know the crop that suits their field area location. In the future, the system can be enhanced by using hybrid approaches for suggesting fertilizers to be added in a timely manner for achieving high profit and yield.

6. REFERENCES

[1] Hemavathi B. Biradar, Laxmi Shabadi, "Review on IOT based multidisciplinary models for smart farming", 2nd IEEE International Conference On Recent Trends in Electronics Information & Communication Technology (RTEICT), 2017.

[2] R. Ramya, C.Sandhya, R. Shwetha, "Smart farming system using sensors", IEEE International Conference on Technological Innovations in ICT For Agriculture and Rural Development, 2017.

[3] Giritharan Ravichandran, Koteeshwari R S, "Agricultural crop predictor and advisor using ANN for smartphones", IEEE International Conference on Emerging Trends in Engineering, Technology and Science (ICETETS), 2016.

[4] Sjaak Wolfert, Lan Ge, Cor Verdouw, Marc-Jeroen Bogaardt, "Big Data in smart farming – A review", Published in Elsevier at Wageningen University and Research, The Netherlands and Information Technology Group, Wageningen University, The Netherlands, 2017.

[5] Siwakorn Jindarat, Pongpisitt Wuttidittachotti, "Smart farm monitoring using Raspberry Pi and Arduino", IEEE International Conference on Computer, Communication, and Control Technology, 2015.

[6] Rekha M.E1, S. Muthu Selvi, "Android Arduino interface with smart farming system", International Journal Of Engineering And Computer Science, 2017.

[7] Ankita Patil, Mayur Beldar, Akshay Naik, Sachin Deshpande, "Smart farming using Arduino and Data Mining", IEEE International Conference on Computing for Sustainable Global Development, 2016.

[8] Khaja Moinuddin, Nalavadi Srikantha, Lokesh KS, Aswatha Narayana, "A survey on secure communication protocols for IoT systems", International Journal Of Engineering And Computer Science, 2017.

[9] Aruul Mozhi Varman S, Arvind Ram Baskaran, Aravindh S, Prabhu, "Deep Learning and IoT for Smart Agriculture Using WSN", IEEE International Conference on Computational Intelligence and Computing Research, 2017.

[10] S.R. Juhi Reshma and Anitha S. Pillai, "Impact of Machine Learning and Internet of Things in agriculture: state of the art", Springer International Publishing in Proceedings of the Eighth International Conference on Soft Computing and Pattern Recognition, Advances in Intelligent Systems and Computing, 2018.

[11] Tanmay Banavlikar, Aqsa Mahir, Mayuresh Budukh, Soham Dhodapkar, "Crop recommendation system using Neural Networks", International Research Journal of Engineering and Technology (IRJET), 2018.

[12] S.Pudumalar, E.Ramanujam, "Crop Recommendation System for Precision Agriculture", IEEE Eighth International Conference on Advanced Computing, 2016.

[13] Niketa Gandhi, Leisa J. Armstrong, "A review of the application of data mining techniques for decision making in agriculture", IEEE 2nd International Conference on Contemporary Computing and Informatics, 2016.

[14]   Yuvaraj N & Sripreethaa, K R December 2017, "Diabetes prediction in healthcare systems using Machine Learning algorithms on Hadoop cluster", Cluster Computing – Springer (SCI), 2017.

[15]   Xiang Gao, Tancheng Lu, Peng Liu, Qiyong Lu, "A soil moisture classification model based on SVM used in agricultural WSN", IEEE The National High-tech R&D Program of China, 2014.

[16]   Yuvaraj N, Sabari A, "Twitter sentiment classification using binary shuffled frog algorithm", Auto Soft - Intelligent Automation And Soft Computing Published By Taylor And Francis (SCI), 2016.

[17]   https://suyati.com/blog/the-role-of-internet-of-things-in-agriculture/

[18]   https://en.wikipedia.org/wiki/Soil_moisture_ sensor.

[19]   https://www.finoit.com/blog/top-15-sensortypes-used-iot/

[20]   https://en.wikipedia.org/wiki/PH_meter

[21]   https://www.elprocus.com/different-types-of-microcontroller-boards/

[22]   https://www.rs-online.com/designspark/eleven-internet-of-things-iot-protocols-you-need-to-know-about

[23]   P. Kanaga Priya, Dr. N. Yuvaraj, "A survey on sensors & communication technology in IoT for smart crop cultivation using Machine Learning algorithms", IJSRD - International Journal for Scientific Research & Development, 2018.

[24]   Vijo T Varghese, Kalyan Sasidhar, Rekha P, "A Status Quo of WSN Systems for Agriculture", International Conference on Advances in Computing, Communications and Informatics (ICACCI), IEEE, 2015.

[25]   P.Bhargavi, Dr.S.Jyothi, "Applying Naive Bayes data mining technique for classification of agricultural land soils", IJCSNS International Journal of Computer Science and Network Security, 2009.

[26]   Niketa Gandhi, Leisa J. Armstrong, Owaiz Petkar, Amiya Kumar Tripathy, "Rice crop yield prediction in India using Support Vector Machines", International Joint Conference on Computer Science and Software Engineering (JCSSE), IEEE, 2016.

[27]   Niketa Gandhi, Owaiz Petkar, Leisa J. Armstrong, "Rice crop yield prediction using Artificial Neural Networks", International Conference on Technological Innovations in ICT For Agriculture and Rural Development (TIAR 2016), IEEE, 2016.

[28]   Amir Haghverdia, Robert A. Washington-Allenb, Brian G. Leibc, "Prediction of cotton lint yield from phenology of crop indices using Artificial Neural Networks", Computers and Electronics in Agriculture, ELSEVIER, 2018.

[29]   K.R. Sri Preetha, P. Kanaga Priya, K.DivyaPrabha, S. Dharanipriya, "Crop rotation and yield analysis using Naive Ratio Classification", 4th National Conference on Recent Advances in Computer Science (NCRACS' 17), 2017.

[30]   Suhas Athani, CH Tejeshwar, Mayur M Patil, Priyadarshini Patil, Rahul Kulkarni, "Soil monitoring using IoT enabled Arduino sensors with Neural Networks for improving soil management for farmers and predict seasonal rainfall for planning future harvest in North Karnataka - India", International conference on I-SMAC(IoT in Social, Mobile, Analytics and Cloud I-SMAC 2017), IEEE, 2017.

[31]   Rekha P, Maneesha V. Ramesh, Venkata Prasanna Rangan, Nibi K V, "High yield groundnut agronomy: An IoT based precision farming framework", IEEE, 2017.

[32]   Umair Ayub, Syed Atif Moqurrab, "Predicting crop diseases using data mining approaches: classification", IEEE 1st International Conference on Power, Energy and Smart Grid (ICPESG), 2018.

[33]   Hulya Yalcin, "Phenology recognition using Deep Learning", IEEE, 2018.

[34]   http://marubon-ds.blogspot.com/2017/09/simple-tutorial-to-write-deep-neural.html

[35]   https://towardsdatascience.com/metrics-to-evaluate-your-machine-learning-algorithm-f10ba6e38234

[36]   https://medium.com/greyatom/performance-metrics-for-classification-problems-in-machine-learning-part-i-b085d432082b

[37]   https://data.world/chandrasekar/semistructured-data

[38]   https://www.kaggle.com/vijayaghanapathy/tamil-nadu-agriculture-data-set

[39]   https://www.deeplearningbook.org/
[40]   https://www.manning.com/books/deep-learning-with-python
[41]   https://www.tutorialspoint.com/python_deep_learning/index.html
[42]   https://uberpython.wordpress.com/2012/01/01/precision-recall-sensitivity-and-specificity/
[43]   http://corysimon.github.io/articles/classification-metrics/
[44]   https://towardsdatascience.com/activation-functions-and-its-types-which-is-better-a9a5310cc8f
[45]   https://towardsdatascience.com/understanding-learning-rates-and-how-it-improves-performance-in-deep-learning-d0d4059c1c10
[46]   https://medium.com/datathings/neural-networks-and-backpropagation-explained-in-a-simple-way-f540a3611f5e