Soil Sensors Based Prediction System for Plant Diseases using Exploratory Data Analysis and Machine Learning

Manish Kumar, Ahlad Kumar and Vinay S. Palaparthy

***Abstract*— Plant diseases cause losses to agricultural production and hence, the economy. This necessitates a need to develop prediction models for the plant disease detection and assessment. Fungal infection, the most dominant disease, can be controlled by taking appropriate measures if detected at an early stage. The paper aims to develop an expert system for the prediction of various fungal diseases (powdery mildew, anthracnose, rust, and root rot/leaf blight). A multi-layered perceptron model is used for the classification of the diseases which not only detects the plant diseases effectively but can also increase the production drastically. The proposed technique incorporates three significant steps of dataset pre-processing, exploratory data analysis, and detection module. Firstly, the real-time data is captured by the soil sensors system installed at agriculture field at Sardarkrushinagar Dantiwada Agricultural University, Gujarat, India, along with the satellite data for other micro-meteorological factors. Next, an extensive exploratory data analysis has been performed to get insights into the collected data. Finally, the proposed machine learning model has been employed to predict plant diseases. The experimental results indicate that the model outperforms several existing methods in terms of accuracy. Average accuracy in predicting each disease has been found more than 98%. This work also proves the feasibility of using this technique for faster plant disease detection at an affordable cost.**

***Index Terms*— Artificial Neural Network, Multi-label Classification, Plant Diseases, Soil based sensors**

I. Introduction

IN 2019, the United Nations estimated 2 billion increase in the worlds' population by next 30 years, a significant increase of nearly 25% [1]. According to the report of the Food and Agricultural Organisation(FAO), to feed this population, about 70-90% more food will be required. Of the total agricultural crop production worldwide, damage of nearly 16% has been caused by the microbial diseases. In order to minimize the occurrence of diseases as well as maximizing the productivity and ensuring agricultural sustainability, there is a need for advanced disease detection in preventing damages to crops. Hence, predetermining plant diseases and their prevention have raised a great interest in researchers.

Diseases prediction in crops depends on various environmental and weather conditions, under which a pathogen can survive. When pathogen comes in contact with a susceptible host, it can infect and can cause severe losses to the agriculture production. The diseases in plants causes a drop in the quality and quantity of the agricultural output [2]. One of the most common diseases is fungi, present in the plant leaves. Fungi is the most diverse group of plant pathogens, accounting for over 70-80% of plant diseases [3]. There are over 20,000 species of fungi that are parasitic and responsible for infections in crops and plants, thereby the quality of leaves, fruits, stem, vegetables, and their products gets suffered. There are two key factors 'Disease' and 'Disorder' that affect the crops and their products. Disease, the biotic factors, are caused either by fungi or by bacteria or algae, and the disorder are the abiotic factors caused by the atmoshperic conditions (temperature, rainfall, moisture etc.) [4]. These infectious crop diseases, if not treated timely, can significantly reduce the yield, thus endangering global food security.

Early disease diagnosis and providing the control measures can help the farmers to save the crops. These measures include direct or indirect disease identification methods. Direct detection methods mainly include laboratory-based techniques, while indirect methods use optical sensors for thermography, fluorescence imaging, and hyperspectral techniques [5] [6]. The limitation of various optical sensing techniques is the large amount of data acquired and the complexity of the data collected. In order to effectively utilize these techniques, it requires high setup and computational costs along with the knowledge of data analytics and statistical methods.

Although providing accurate data, direct methods cannot be used for on-field detection and of limited use. Indirect methods are employed directly for on-field disease detection. Based on plant stress and level of plant volatility, various sensors are used to identify the weather conditions, various biotic and abiotic stresses as well as pathogenic diseases in crops. Many other advances in fungal pathogen detection emphasize on biosensors; however, the drawback lies in the fact that these biosensors are expensive [7]. Various prediction approaches,

M. Kumar has completed his M.Tech. in ICT from DA-IICT, Gandhinagar, Gujarat (e-mail: manish.k3209@gmail.com). URL: https://manishk32.ml/

Dr. Ahlad Kumar is working as an Assistant Professor at DA-IICT. (e-mail: ahlad_kumar@daiict.ac.in). URL: https://www.daiict.ac.in/profile/ahlad-kumar/

Dr. Vinay S. Palaparthy is working as an Assistant Professor at DA-IICT. (e-mail: vinay_shrinivas@daiict.ac.in). URL: https://www.daiict.ac.in/profile/palaparthy-vinay/

based on conventional multiple regression (REG), generalized regression neural network (GRNN), support vector machine (SVM) have been used in building prediction models for plant diseases [8] [9]. Plant disease identification, based on GRNNs and probabilistic neural networks (PNNs) have been used as the classifiers to identify wheat and grape diseases [10]. Based on high-resolution multispectral stereo images, K-nearest neighbor classifiers (KNN) have been employed for pixel-wise classification for automatic classification of leaf diseases [11]. Models, like a multi-layered perceptron model (MLP), are also used in plant disease recognition using various illness symptoms, for real-time visual diagnosis [12]. Other techniques involve multivariate linear regression (MLR) and partial least square regression (PLSR), which use hyperspectral data to estimate the severity of plant disease [13]. Deep neural networks are now, being applied as image classifiers in plant pathology to detect the diseases [14]. Various deep learning architectures such as AlexNet and SqueezeNet are being used to detect the diseases on the leaves of tomato plants. Tomato leaf images from the PlantVillage dataset had been used for the training with ten different disease classes [15]. Deep convolutional neural network (DCNN) is being used to conduct symptom-wise recognition of four cucumber diseases where each symptom images were segmented from cucumber leaf images captured under field conditions [16]. Neural nets such as Region Proposal Network (RPN) identify the best search space in the leaf image, which reduces the search space for the classifier and provides better initial information during detection [17]. Advancement in the exploratory data analysis has opened up the avenues for various sensing applications in areas such as agriculture [18]–[21], defence [22], [23], biomedical [24], environmental [25], [26] etc.

Expert systems have been developed, aiming to improve the decision-making process for the detection of diseases. Many of them are knowledge-based and follows a series of if-then-else rules like the one developed in Spain [27]. These systems identify harmful pathogens at an early stage which would otherwise harm the crops, so that suitable actions can be taken [28]. Other systems with automatic knowledge acquisition have been developed to improve the accuracy and is directly dependent on the expert's advice [29].

Here, the proposed model aims to attain a real-time robust predictive model for plant disease detection using soil condition and various other environmental factors. On the contrary, various other traditional models assume these variables to be homogeneous, and advise farmers to use pesticides. It has been observed that the occurrence of the plant disease depends on specific environmental factors, and thus diseases exhibit a heterogeneous distribution in the field. In this work, the use of data analysis for the agricultural applications have been explored. Several data mining techniques that are continually being introduced in the area of plant science and are becoming a key technology in this field are the areas of this study. Moreover, the disease detection should be cheap, reliable, sensitive, cost-effective. Here, to predict plant diseases based on various environmental factors, sensors have been employed and the data from micro-meteorological factors like temperature, relative humidity, wind speed, solar radiation etc., measured by satellite or local weather station have been used. The sensors record temperature, soil moisture, and humidity.

Soil water content measurement is one of the important parameter for plant disease management. The soil moisture content is measured either in volumetric or gravimetric forms [30]–[33]. Gravimetric method is time consuming and limited to the lab measurements, whereas, volumetric content is used for both lab and *in-situ* measurements. Time domain reflectometry (TDR) [34], frequency domain reflectometry, (FDR) [35] and capacitance technique [36], [37], thermal dissipation block technique [38], and micro-electro-mechanical systems [39]–[41] are some of the *in-situ* measurement techniques.

The paper has been organized as follows. Section II highlights the information about the sensors used in this work. Section III explains the four most dominant fungal diseases with their symptoms, severeness, and how it spreads. Section IV details the method used, dataset description, pre-processing steps used, exploratory data analysis, and the architecture model used. Section V mentions the experimental setup along with evaluation measures. Furthermore, the results are discussed in section VI. whereas section VII concludes the article.

II. Sensors

Monitoring soil parameters; such as soil moisture, soil temperature, humidity and environmental parameters; such as wind speed, wind direction, UV index, and rain can be of high significance in plant disease monitoring. A traditional approach takes all the measurements manually, and checks them. Mostly, this is being done by farmers manually, but

Fig. 1. Soil monitoring system deployed in the agriculture field

remote monitoring, is an effective way to avoid interference with the environment and improve efficiency and has been used an indigenous low-cost remote soil monitoring system commercialized by Proximal Soilsens Tech Pvt. Ltd. [42]. The system, equipped with a soil moisture sensor, humidity sensor, and temperature sensor, has been used to detect plant diseases in the present study.

This system can find applications in open farms, greenhouses, gardening, and research/agricultural labs. The wireless monitoring of the agricultural field not only allows farmers to reduce human power, but it also allows them to work and make an accurate decision. The system is solar powered and can operate for three to four days without sun. The entire system is modular in design, where each subsystem (solar panel, signal processing unit, sensors) is easily replaceable. The system has been designed to suit the Indian farmer's condition. The height of the system is adjustable between 1m to 3m. [42].

There are two nodes of these systems deployed in agriculture field Sardarkrushinagar Dantiwada Agricultural University (SDAU) [43], Gujarat, namely *node*1 and *node*2. These systems send data wirelessly to a central server. Fig. 1 shows the soil monitoring systems deployed in agriculture field SDAU, Gujarat [43]. Table I shows the technical specification of the sensors. Following are the details of the each sensors:

- Soil moisture sensor: Developed soil moisture sensor is a capacitive, where the two probes of the sensor act as a capacitor and soil between the probes act as a dielectric medium. As the soil moisture increases, the dielectric constant of the soil matrix increases, which results in the change in the sensor capacitance [36].
- Temperature sensor: Temperature sensors comprises commercially available MCP9701A temperature sensor [44].
- Relative humidity sensor: For relative humidity sensor, the commercially available HIH-5030-001 sensor has been used [45].

The measurement ranges are defined in the table, along with accuracy and response time of the sensors [42].

TABLE I
TECHNICAL SPECIFICATION OF SENSORS

Sensors	Measurement Range	Accuracy	Response Time
Soil moisture	0 – 100% GWC^*	$\pm 3\%$ GWC^*	10 seconds
Ambient humidity	0 – 100% $RH^{\#}$	$\pm 3\%$ $RH^{\#}$	5 seconds
Ambient temperature	5 - 80°C	$\pm 2\%$ °C	5 seconds

Note: GWC^*: Gravimetric water content (Accuracy of the sensors is benchmarked with the standard gravimetric soil moisture measurement method), $RH^{\#}$: Relative humidity

III. Diseases in Green Gram

In this section, various fungal diseases in green gram crops have been studied and used in our work.

A. Powdery mildew (D1)

Sometimes, patches of white powdery substance appear on leaves and other green parts, which gradually increase in size to cover the lower leaf surface if not handled. In extreme cases of infections, both sides of the leaf are completely covered by whitish powdery growth, resulting in infected plant and thereby the major yield loss. Such pathogens have a wide host range and live in a conidial form on various off-season hosts which spread through seasonally developed airborne conidia [46].

B. Anthracnose (D2)

Symptoms of the disease occur mostly on leaves and pods having circular, black, sunken spots with a dark center and bright red-orange margins. In serious infections infected sections wither off. It has an detrimental impact on seedlings, as it gets blighted shortly after seed germination due to infection. The pathogen mainly lives on seed and plant waste, spreading by airborne conidia [46].

C. Rust (D3)

Here, the symptoms are described as circular reddish-brown pustules usually seen on the underside of the leaves. In severe conditions, the Rust pustules fully cover both sides of the surfaces. Defoliation is followed by shriveling, resulting in loss of yield. The primary infection comes from the sporidia that develops from teliospores. The secondary distribution is by wind-borne uredospores. The fungus also lives on other host legumes [46].

D. Root rot and leaf blight (D4)

The fungus in initial phases causes symptoms of seed rot, seedling blight, and root rot. The affected leaves turn yellow in colour, and irregular brown lesions appear on them, which further form large blotches, resulting in leaves dying prematurely. Roots and the stem's basal part turn black in colour, and the bark quickly peels off. In green gram, the pathogens cause seed decline, root rot, damping-off, seedling blight, stem canker and leaf blight. The primary cause of infection is the saprotrophic species of pathogens found in the soil, while secondary infection occurs by asexual spores [46].

IV. Materials and Methods

A. Proposed Method

A neural network, implemented for the classification of diseases caused by various fungal diseases has been discussed (Section III). In order to perform this classification task, first the acquired data from sensors and also from the satellite has been discussed in IV-B. The acquired data, has been pre-processed through transformation methods discussed in IV-C [47]. Once the data has been cleaned and labelled properly, an exploratory data analysis needs to be carried out (IV-D) to gain more insight into the dataset acquired through soil sensors. Then, the splitting of the dataset into training and testing datasets has been performed. Based on these training and test datasets, proposed neural network model needs to be trained and tested, and is discussed in IV-E. The flowchart of our proposed method has been shown in Fig.2.

B. Sensor Dataset Acquisition

The data repositories have been used at two different sites viz., *node*1 and *node*2. Both the sites are equipped with multiple sensors that record time-series data and send it to the central database. Both the sites are at a distance of 200 meters apart. These sensors record 5-8 data points per day. During the

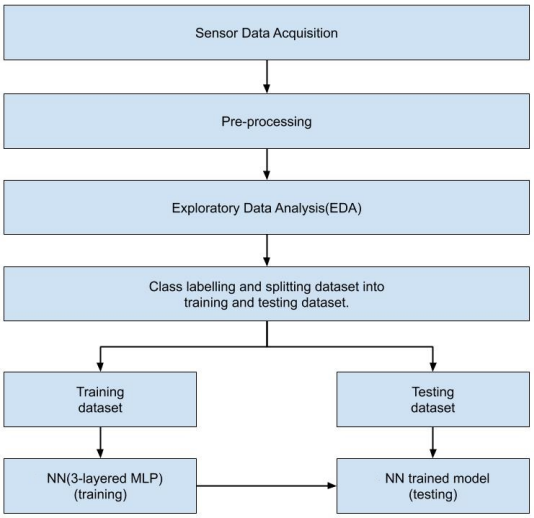


Fig. 2. Flowchart of the proposed work

day, at each timestamp, multiple sensors record features such as temperature($^\circ$C), humidity(%RH), soil moisture(%) denoted by $\mathbf{F_1}$, $\mathbf{F_2}$, $\mathbf{F_3}$, respectively at both nodes. Other than sensors data, satellite data has been also acquired for each *node*1 and *node*2, respectively. Satellite data contains features like rainfall ($\mathbf{F_4}$), pressure ($\mathbf{F_5}$), wind speed ($\mathbf{F_6}$), clouds ($\mathbf{F_7}$), solar/infrared radiation ($\mathbf{F_8}$), ultra-violet radiation ($\mathbf{F_9}$) and month ($\mathbf{F_{10}}$). The summary of these feature vectors is given in Table II

TABLE II
LIST OF FEATURES OBTAINED THROUGH SENSOR AND SATELLITE DATA

Feature Name	Feature Description
$\mathbf{F_1}$	Ambient Temperature
$\mathbf{F_2}$	Ambient Humidity
$\mathbf{F_3}$	Soil Moisture
$\mathbf{F_4}$	Rainfall
$\mathbf{F_5}$	Pressure
$\mathbf{F_6}$	Wind-Speed
$\mathbf{F_7}$	Clouds
$\mathbf{F_8}$	Solar/Infrared Radiation
$\mathbf{F_9}$	Ultra-violet Radiation
$\mathbf{F_{10}}$	Month

A total of 4139 data points have been recorded in this work, out of which 1916 are from *node*1, and 2223 from *node*2. The data is collected from Jan. to Sep. 2019. Each data point is categorized among four disease labels viz., **D1** (powdery mildew), **D2** (anthracnose), **D3** (rust), and **D4** (root rot/leaf blight). Table III provides the numerical ranges of various features $\mathbf{F_1}$, $\mathbf{F_2}$, $\mathbf{F_3}$, and $\mathbf{F_4}$, which are specific to the occurrence of a particular disease. For instance, the occurrence of disease **D1** is more prevalent if the numerical range of $\mathbf{F_1}$, $\mathbf{F_2}$, $\mathbf{F_3}$ and $\mathbf{F_4}$ lies in between $10-20^\circ$C, $90-100\%$, $10-14\%$ and $< 1\%$, respectively

C. Preprocessing of Data

Data pre-processing is an essential step, as the quality of data and useful information that can be derived from it directly affects the ability of machine learning model used. At first, the data collection was made using sensors, containing three features $\mathbf{F_1}$, $\mathbf{F_2}$, and $\mathbf{F_3}$, in which all the null or not a number (NaN) values are identified. Once identified, instead of removing these data point, *SimpleImputer*, a class for imputation provided by $scikit$ [48] has been used.

After this, the fourth feature $\mathbf{F_4}$ has been taken from the satellite data, and mapped based on timestamp value in *node*1 and *node*2, respectively. Once all four features $\mathbf{F_1}$, $\mathbf{F_2}$, $\mathbf{F_3}$, $\mathbf{F_4}$ are ready; the dataset is then labeled based on the various constraints as defined in Table III. The label for each data point is a 4-bit binary sequence where '1' denotes that disease can occur and '0', otherwise. While working on the feature set, other features such as $\mathbf{F_5}$, $\mathbf{F_6}$, $\mathbf{F_7}$, $\mathbf{F_8}$, $\mathbf{F_9}$ and $\mathbf{F_{10}}$ have been incorporated. The standardization technique, an essential requirement for machine learning estimators has been used on all the ten features $[\mathbf{F_1}, \mathbf{F_2}, ..., \mathbf{F_{10}}]$. In this technique, the shape of the distribution has been ignored and then transform the data to its center mean value for each feature vector. After this the data is scaled by dividing non-constant features by their standard deviation. Here, pre-processing of the class *StandardScaler* has been taken from $sklearn$ library. The final standardized dataset with all ten features acts as input to the MLP model discussed in Section IV-D.

TABLE III
NUMERICAL RANGE OF FEATURES FOR A PARTICULAR DISEASE [46]

Disease D	$\mathbf{F_1}$ Ambient Temp.	$\mathbf{F_2}$ Ambient Humidity	$\mathbf{F_3}$ Soil Moisture	$\mathbf{F_4}$ Rainfall
D1	10-20°C	90-100%	10-14%	<1%
D2	10-15°C	80-100%	10-14%	<1%
D3	21-26°C	75-100%	10-14%	0.5-1%
D4	>30°C	>80%	10-14%	>0.5%

D. Exploratory Data Analysis

To identify the significant features of the dataset and generated insights for further investigation. Here, the data has been analyzed using statistics, data visualization using Matplotlib library [49], and other techniques to get meaningful insights into the dataset before applying machine learning models for classification. The dominant features present in the dataset have been measured by using the feature importance property of the model (Fig.3). Feature importance has given a score for each feature of the data; higher the score more critical/relevant is the feature. The plot suggests that the disease depends primarily on $\mathbf{F_1}$, $\mathbf{F_2}$, $\mathbf{F_3}$, and $\mathbf{F_{10}}$, whereas $\mathbf{F_5}$, $\mathbf{F_6}$, $\mathbf{F_7}$, $\mathbf{F_8}$, and $\mathbf{F_9}$, though are equally essential but have less score.

Fig.4 shows the correlation between various features, emphasizing how the features are related to each other. The correlation can be positive (increase in one value of feature increases the value of the target variable) or negative (increase in one value of feature decreases the value of the target variable). A heatmap of correlated features has been plotted using the $seaborn$ library [50] (Fig.4), which makes it easy to identify which features are strongly related to the target

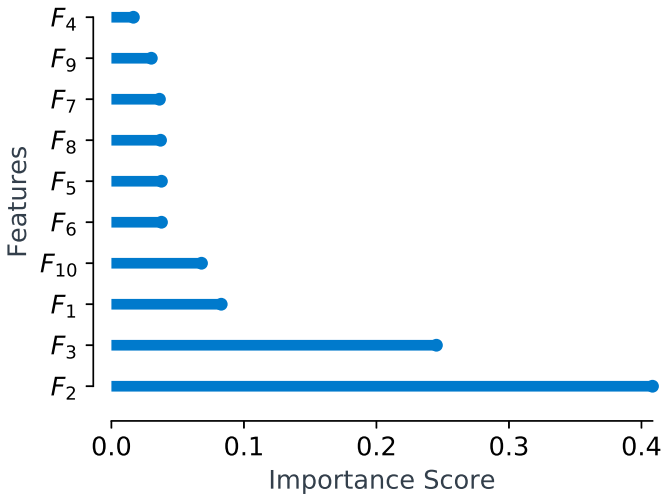


Fig. 3. Dominant features based on importance score.

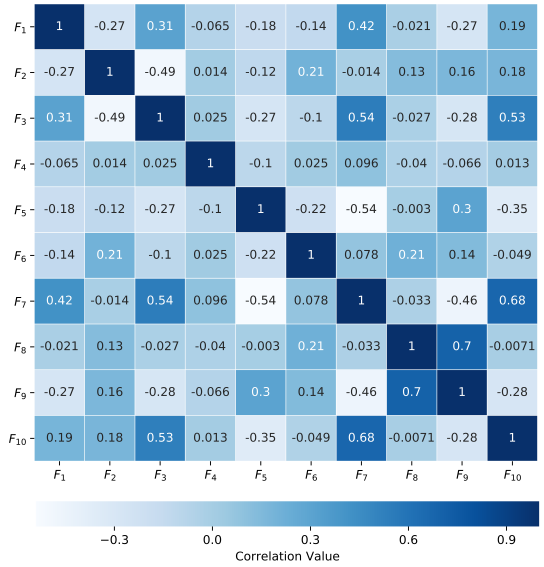


Fig. 4. Correlation plot among the components of feature vector $\mathbf{F}$

variable. On analysis of the heatmap, it has been found that the feature pairs ($\mathbf{F_8}$, $\mathbf{F_9}$), ($\mathbf{F_7}$, $\mathbf{F_{10}}$), ($\mathbf{F_3}$, $\mathbf{F_7}$), and ($\mathbf{F_3}$, $\mathbf{F_{10}}$) are strongly correlated having correlation value greater than 0.5.

Fig.5 shows the disease prevalence percentage in 2019 for data collected from *node*1 and *node*2, respectively. It may be noted that the infections due to **D1** and **D2** are most frequent ($\geq$ 90%) through out the year, whereas **D3** ranges from 25% to 30%, while **D4** ranges from 45% to 50%. The inference from these visualizations is that the disease **D1** and **D2** are most frequent out of all four diseases throughout the year.

Further, the dataset has also been grouped into three categories based on the month, and has been divided into three categories as of January to March, April to June, and July to September represented as **Q1**, **Q2**, and **Q3**, respectively. Fig. 6 shows the trend of disease prevalence in all of the three categories. **Q1** is mostly cold season followed by **Q2**, a summer season and **Q3**, the monsoon/rainy season in most parts of India. I has been observed that **D1** and **D2** are standard in all the seasons, **D3** generally occurs in the rainy season, i.e. **Q2** when soil moisture and rainfall is high, and **D4** requires high temperature to sustain, hence prevails in **Q2**.

In order to observe the probabilistic dependency of the diseases among themselves, Fig.7 provides the conditional probability values $P(X|Y)$ [51], defined as

$$P(X|Y) = \frac{P(X \cap Y)}{P(Y)} \tag{1}$$

where, X belongs to the columns and Y belongs to the rows of Fig 7. Here, for each disease **D**, represented by the

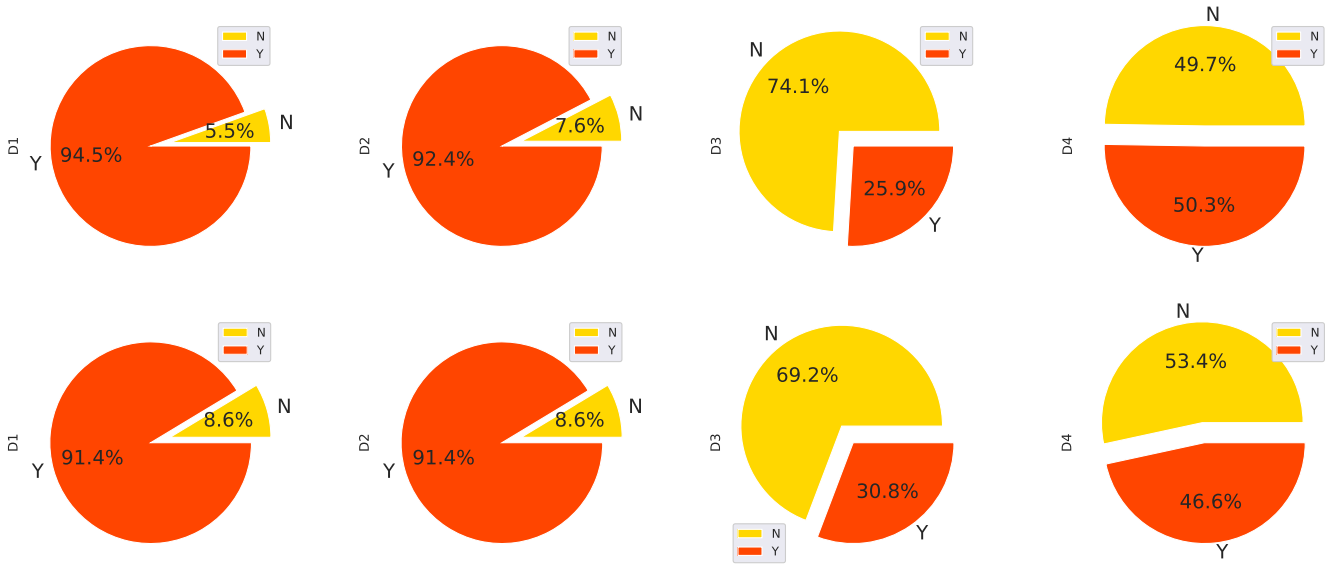


Fig. 5. Disease occurrence percentage for *node*1(first row) and *node*2(second row) in 2019.

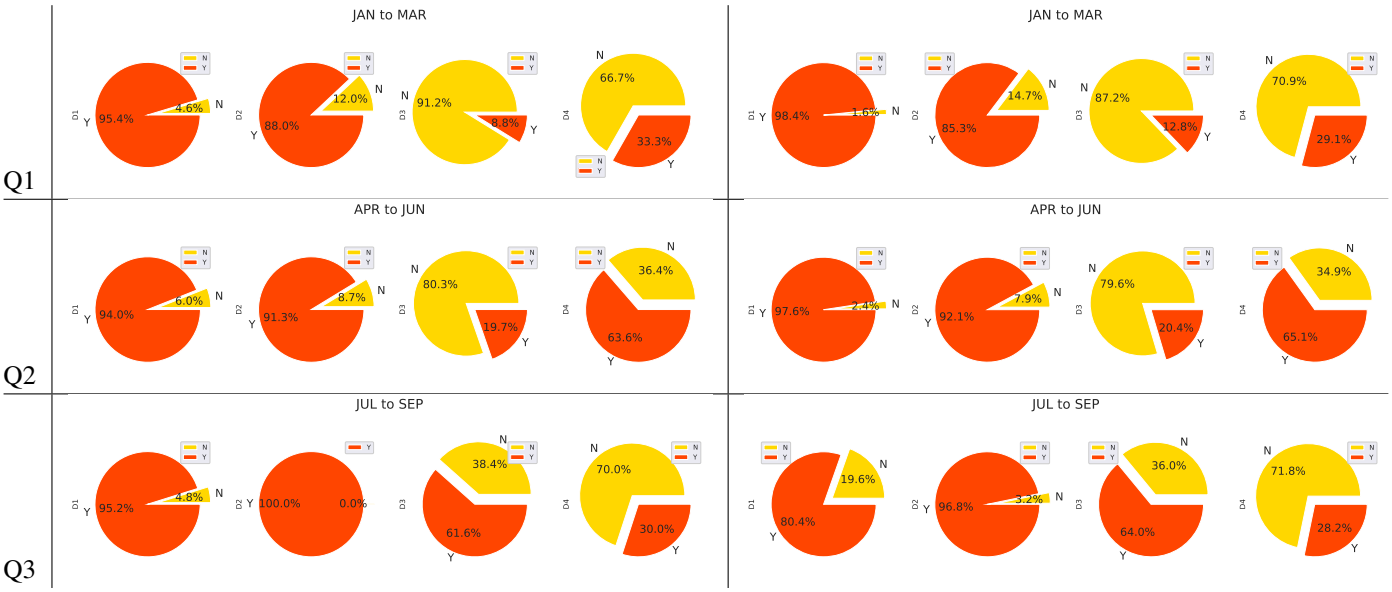


Fig. 6. Disease prevalence in *node*1(left) and *node*2(right) based on different quarters Q1(Jan. to Mar.), Q2(Apr. to May), Q3(Jul. to Sep.)

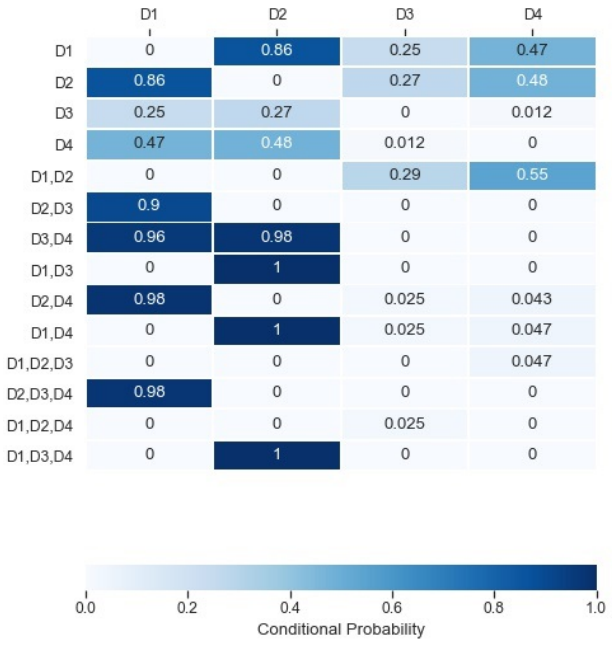


Fig. 7. Conditional probability table for **D1**, **D2**, **D3**, **D4** with respect to other combination of diseases.

column label, conditional probability with respect to each set of disease represented by row-labels has been calculated. For instance, $P(\mathbf{D1}|\mathbf{D2},\mathbf{D3},\mathbf{D4})$ and $P(\mathbf{D1}|\mathbf{D2},\mathbf{D4})$ have the highest conditional probability of 98% compared to all others when cases of **D1** have been considered. In other words, the occurrence of **D1** disease is highest when **D2**, **D3**, **D4** occur together and also when **D2**, **D4** occur together. Similarly, for the disease **D2**, the probabilities $P(\mathbf{D2}|\mathbf{D1},\mathbf{D3},\mathbf{D4})$, $P(\mathbf{D2}|\mathbf{D1},\mathbf{D4})$ and $P(\mathbf{D2}|\mathbf{D1},\mathbf{D3})$ are prevalent. Disease **D3** and **D4** commonly happen only when the plant is affected by both **D1** and **D2**, indicated by the corresponding high probabilities of occurrence $P(\mathbf{D3}|\mathbf{D1},\mathbf{D2})$ and $P(\mathbf{D4}|\mathbf{D1},\mathbf{D2})$, respectively. The $P(\mathbf{D1}|\mathbf{D2})$ has a value of 86% which indicates that occurrence of disease **D1** and **D2** together is high, since both diseases mostly affects the leaves of plant; also the responsible factors for both diseases are nearly same. Another observation is that the diseases **D3** and **D4** are not related to each other as the $P(\mathbf{D3}|\mathbf{D4})$ and $P(\mathbf{D4}|\mathbf{D3})$ are just around 1.2%. These observations lead to a great insight about the disease correlation.

Further, the dataset has been divided into four, categories based on different time slots (Fig.8). The timestamp 24 hours has been divided into four slots, named as **S1**(4 am to 10 am), **S2**(10 am to 4 pm), **S3**(4 pm to 10 pm), and **S4**(10 pm to 4 am). On plotting the data based on the time slots, **D1** and **D2** are standard throughout, **D4** is prevalent mostly during the day time, i.e. in **S2** and **S3**, but **D4** increases during night time (**S4** and **S1**) when the temperature is much lower.

E. Artifical Neural Network Model

Artificial neural networks (ANNs) are suitable for classification and prediction problems where inputs are assigned a class(label). Firstly, the basics of ANN have been discussed and then the proposed architecture has been introduced. ANN has a parallel distributed computational model consisting of structures and functions of biological neural networks that help in recognizing the relationships in a dataset that mimics the way the human brain works. ANN are excellent tools for finding out patterns that are far too complicated for a human programmer to extract and make the machine learn. Given enough data with proper initialization, ANNs can provide optimal solutions by adjusting their inner structure. ANNs contains an input layer, multiple hidden layers and a single output layers. Each layer consists of nodes or units that transform the input layer in a way such that the output layer can use it. There are three types of nodes in ANN: the input

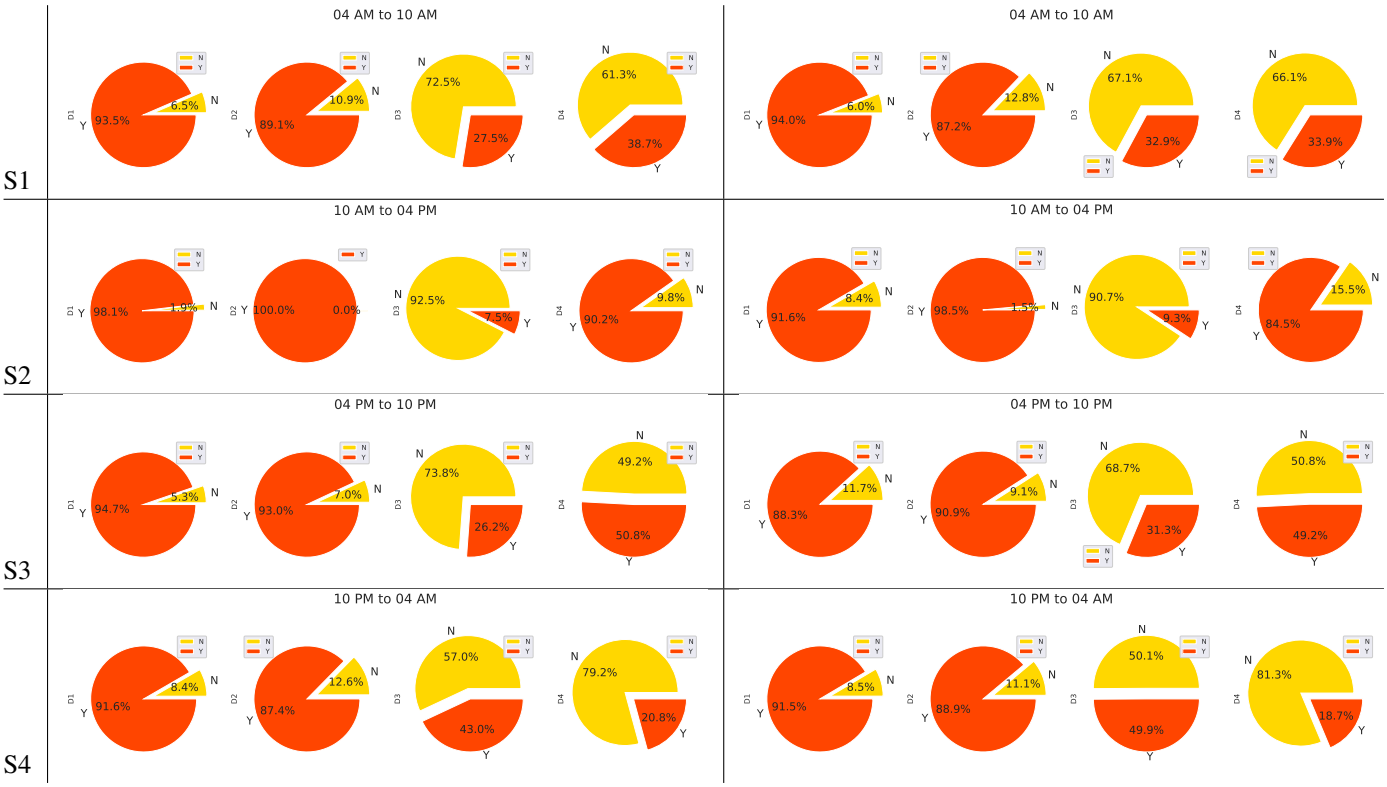


Fig. 8. Disease prevalence in *node*1(left) and *node*2(right) based on different time slots S1(4 am to 10 am), S2(10 am to 4 pm), S3(4 pm to 10 pm), and S4(10 pm to 4 am).

nodes receive the external data, while the output nodes send data outside of the network, and the hidden nodes have their input and output remain within the system. These hidden nodes provide the non-linearity and perform complex internal computations, which makes ANN powerful. Each node in a neural network can have multiple inputs but only one output. An input to a node is either the external data or the output of another node. The nodes in the hidden layer and output layer linearly combines all the values that are fed as input, which is then transformed by an activation function such as sigmoid or softmax. The output generated is then compared with the target values to compute the value of the loss function. There exist a variety of optimization techniques to minimize the loss function.

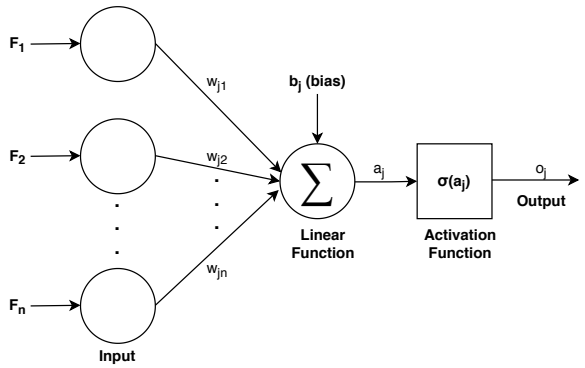


Fig. 9. Details of the neural network architecture used for our experiment.

As shown in Fig.9, the input data vector $\mathbf{F} = [\mathbf{F_1}, \mathbf{F_2}, ..., \mathbf{F_n}]$ has been applied at the input layer of the neural network with assigned relative weight $W = [w_{j1}, w_{j2}, ..., w_{jn}]$. The weighted input feature vector a_j is then passed through the activation function $\sigma(a_j)$ resulting in the output feature vector o_j. The vector , o_j will be decided on which specific component of the input vector $\mathbf{F}$ is dominant. This is analogous to the varying synaptic strengths of the biological neurons, where weights are the adaptive coefficients within the neural network, determining the importance of the input data.

In this study, a fully connected feed-forward network (FCFN), also known as a multi-layered perceptron (MLP) model with back-propagation learning algorithms has been employed. MLPs are based on a supervised procedure, i.e., the network builds a model based on examples in input data with known output labels. The number of layers, neurons in each layer, and the type of activation function for the neurons are chosen a priori. The MLP extracts this relation solely from the presented cases, which together are assumed to contain the necessary information for this relation implicitly. Neurons in each layer computes a linear function [52] based on its inputs and then applies an activation function to produce an output as follows:

$$o_j = \sigma\left(\sum_{i=1}^{n} w_{ji}\mathbf{F_i} + b_j\right) \tag{2}$$

where, o_j is the output neuron, n is the number of inputs, w_{ji} is the synaptic weight which connects input F_i to the output neuron, and f is the activation function. MLP uses back-

propagation as the learning algorithm, with gradient descent to minimize the loss function. Since minimizing binary cross-entropy function (L) [52] is superior for handling multi-labels, it has been used in the proposed model, and is given by

$$L = -\sum_{k=1}^{p}(t_k \log(o_k) + (1 - t_k) \log(1 - o_k)) \quad (3)$$

where, target output is represented as $t = [0, 1, 1, 0]$ for p output neurons. As there are 4 classes of diseases in the discussion, p has been taken as 4.

The proposed MLP architecture comprises five sequential layers, i.e, one input layer consisting of 10 neurons; next are three hidden layers having rectified linear unit (Relu) as an activation function, each with 36, 30, and 24 neurons, respectively. In the end, there is an output layer with 4 neurons and sigmoid as an activation function. Since output classes are not mutually exclusive, and a classifier that can identify more than one class of disease, needs to be built, hence the sigmoid function has been used on the network's raw output. Each layer is a fully connected layer, meaning that all the neurons in a layer are connected to those in the next layer. There are a total of 2350 trainable parameters in our proposed network. The optimization algorithm used in the proposed architecture is Adam. The selection of activation function "Relu" and optimization algorithm "Adam" is decided based on the experimental study carried out in the appendix (Section IX-A.1 and IX-A.2) [53]. Predictive modeling is applied by building a classification model based on the training dataset and then evaluated the model by predicting the classes for the testing dataset.

V. Experiment and Evaluation

In this section, the experimental setup has been discussed along with the evaluation measures used in the experiment. The $python$ libraries used in this experiments has been given in Section V-A,. Further, in Section V-B, various evaluation measures such as precision, recall, $F\text{-}score$, Hamming loss, and subset accuracy that are used in the paper has been presented. Moreover, the receiver operating characteristic (ROC) curve, which depicts the performance of the proposed classification model at all classification thresholds, has been presented in brief.

A. Experimental Setup

All the experiments based on the $Scikit\text{-}learn$, $Matplotlib$ [49], $Seaborn$, and $TensorFlow$ library [53] have been carried out. $Scikit\text{-}learn$ library has been employed to implement several machine learning models. $Matplotlib$ and $Seaborn$ are the most widely used data visualization libraries which have been used to get insights and help in exploratory data analysis. $TensorFlow$ has also been used to implement deep learning models, such as GoogleNet [54], VGGNet [55], AlexNet [56], Long Short-Term Memory (LSTM), and Gated Recurrent Unit (GRU) [57].

The entire dataset has been divided using $train\_test\_split$ function of the $sklearn$ library into random training and the testing subsets before feeding into MLP. The training dataset comprises 70% of the data, whereas the validation and testing dataset constitutes about 15% each.

B. Evaluation Measures

Four standard evaluation measures, namely, $precision$, $recall$ and $F\text{-}score$ have been used to evaluate the performance of proposed MLP for disease classification . The accuracy is a measure to ensure that the ratio of the prediction of true labels is correct. $Precision$ is a measure of how many predictions are correct. $Recall$ is a fraction of true labels that were predicted correctly. $F\text{-}score$ is the harmonic mean of precision and recall, which is often used evaluation scheme in the area of machine learning research. The following equations show how to calculate these values [51].

$$Precision = \frac{TP}{TP + FP} \quad (4)$$

$$Recall = \frac{TP}{TP + FN} \quad (5)$$

$$F_{score} = \frac{2 * (Precision * Recall)}{Precision + Recall} \quad (6)$$

where, true positive, true negative, false positive, and false negative are denoted as TP, TN, FP, and FN, respectively. The ROC curve, which is a probability curve, is used to illustrate how good the classifier classifies the classes based on TP and FP rates. The area under curve (AUC) is a metric that ranges from 0 to 1. The more the area under the curve, the better is the model in distinguishing between classes.

Another metric used for evaluating the performance of a classifier is Hamming Loss (HL). It reports how many times on an average, the relevance of an example to a class label is incorrectly, predicted. Instead of counting the number of correctly classified data instances, HL calculates the loss generated in the bit sequence of class labels during prediction. It does XOR operation between the target binary sequence of class labels and predicted class labels for a data instance and calculates the average across the dataset. The equation [48] is given as follows

$$HL = \frac{1}{|N|.|C|}\sum_{i=1}^{|N|}\sum_{j=1}^{|C|} XOR(y_{ij}, \hat{y}_{ij}) \quad (7)$$

where, $|N|$ is the number of data points in testing set, $|C|$ is number of classes, y_{ij} is target bit of class label j in data point i and $\hat{y}_{ij}$ is predicted bit of class label j in data point i. HL value ranges from 0 to 1. As it is a loss metric, its interpretation is reverse in nature unlike normal accuracy ratio which means lesser value of HL indicates a better classifier.

Another metric called as subset accuracy (SA) [48] has also been computed, indicating the percentage of tested samples that have all their labels classified correctly.

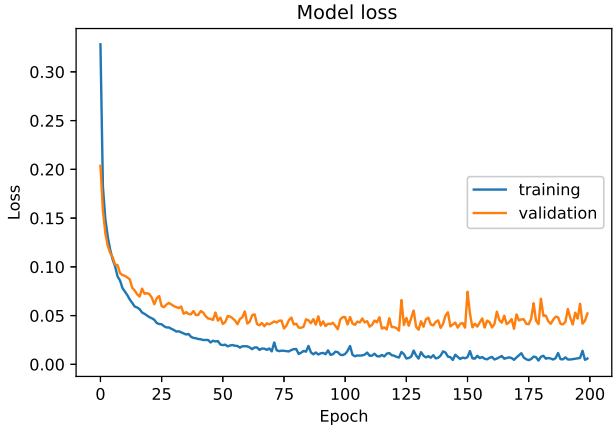
$$SA = \frac{1}{|N|}\sum_{i=1}^{|N|} I(y_i, \hat{y}_i) \quad (8)$$

where $|N|$ is the number of the data points in testing set. The y_i is target vector of class labels for i^{th} data point and $\hat{y}_i$ is

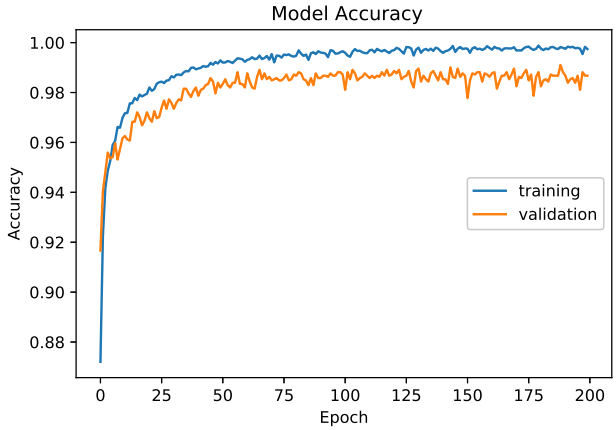
predicted vector of class labels for i^{th} data point. Function I compares each bit of target vector y_i with predicted vector $\hat{y}_i$. I returns 1 if all bits are equal, otherwise, 0.

VI. Results and Discussion

The proposed model has been trained using $Keras$ [58] and the callbacks have been used so as to ensure that the model doesn't overfits the training data. Early stopping method available in $TensorFlow$ has been used to monitor the difference between training loss and validation loss and save the model for each epoch. After 95th epoch, model loss has been found in which both training loss and validation loss have started diverging as shown in Fig.10(a). The weights of the proposed neural network at 95th epoch has been used to calculate evaluation measures given in Section V-B. Fig.10(b) shows the training accuracy vs. validation accuracy plot of the proposed model. The model has achieved more than 98%, training and validation accuracy at 95th epoch. The plot, also depicts the difference between training and validation accuracy, which is very less at 95th epoch.



(a) Training loss and validation loss



(b) Training accuracy and validation accuracy

Fig. 10. Model accuracy and loss plot for training and validation dataset.

The individual disease classification accuracy has been tabulated in Table IV. The average accuracy of our model in predicting each disease individually is nearly 98.99%. Another Table V shows the values of other evaluation measures such as $precision$, $recall$ and $F\text{-}score$ for each plant disease. It can be observed that the recall is high for each disease, which infers that the proposed model is able to correctly classify the relevant disease with more than 98% accuracy.

TABLE IV
TESTING ACCURACY(%) CALCULATED BASED ON THE CONFUSION MATRIX TABLE FOR EACH DISEASE.

Disease	D1	D2	D3	D4
Accuracy	98.87%	98.55%	98.87%	99.68%

TABLE V
CLASS-WISE PRECISION, RECALL AND F1-SCORE.

Disease	Precision	Recall	F1-score
D1	1.00	0.98	0.99
D2	0.99	0.99	0.99
D3	0.95	0.98	0.97
D4	1.00	0.99	1.00

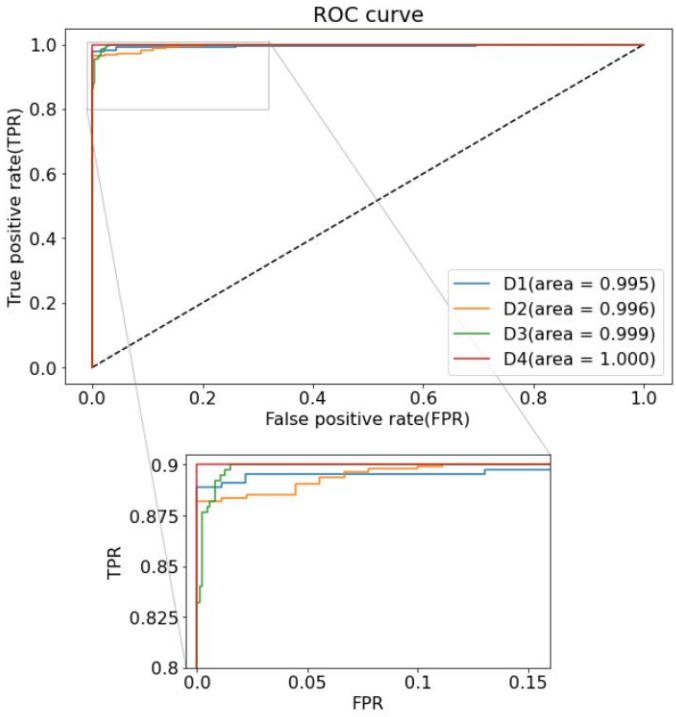


Fig. 11. Receiver operating characteristic (ROC) curve for **D1**, **D2**, **D3**, **D4**. Above figure also denote the area under the curve (AUC) for each curve.

To find the accuracy for the individual label based accuracy, the quality of the classifier will be difficult to judge. Thus, HL presents one clear single performance value for multiple-label case in contrast to the $precision$, $recall$ and $F\text{-}score$ that can be evaluated only for independent binary classifiers for each label. Hence, we used HL as the evaluation metric whose

TABLE VI
COMPARISONS OF VARIOUS PLANT DISEASE DETECTION TECHNIQUES.

Authors	Technique Used	Species	Diseases	Dataset	Data Format	Accuracy
Wang et al. (2010) [10]	BPNN	Wheat	Anthracnose, red rot etc.	Self-collected	Image	89.5%
Bauer et al. (2011) [11]	KNN	Sugar Beet	Leaf Spots, Leaf Blotch	Self-collected	Image	93%
Zhang et al. (2012) [13]	PLSR / MLR	Wheat	Powdery Mildew	Hyper-spectal Measurements	Image	90%
Stegmayer et al. (2013) [12]	MLP	Citrus	Multiple	Self-collected (Symptoms-based)	Numerical Data	84%
Durmus et al. (2017) [15]	AlexNet and SqueezeNet	Tomato	Multiple	PlantVillage (Leaf Images)	Image	95.65% and 94.3%
Ma et al. (2018) [16]	DCNN	Cucumber	Multiple	Self-collected	Image	93.4%
Barbedo et al.(2018) [14]	GoogLeNet	Multiple	Multiple	Digipathos	Image	80.75%
Sun et al. (2020) [17]	RPN	Maize	Leaf Blight	NLB (Northern Leaf Blight)	Image	91.83%
This work	MLP with Soil and Environmental Monitoring Sensor	Greengram	Powdery mildew(**D1**), Anthracnose(**D2**), Rust(**D3**), Root rot and leaf-blight(**D4**)	Self-collected	Numerical Data	**D1** (98.87%), **D2** (98.55%), **D3** (98.87%), **D4** (99.68%)

value is obtained as 0.0144 for this particular experiment. This value is very close to 0 which verifies that the proposed model is a good classifier. Also, the SA of our proposed model is obtained as high as 94.36%.

The ROC curve (Fig.11), suggests that the model is performing very well by balancing between the true-positive rate (TPR) and the false-positive rate (FPR). It shows the trade-off between sensitivity and specificity. The curve demonstrates that closer the curve follows the left-hand border and then the top-border of the ROC curve, more accurate the test classifier. The accuracy of the prediction test depends on how well the test of disease separates the group being tested into those with and without the disease. Accuracy is measured by the area under the ROC curve. This area measures discrimination, that is, the ability of the test to correctly classify data points both with the disease or without. An area of 1 represents a perfect classifier; an area of 0.5 represents a bad classifier. Out of the four curves plotted, one can conclude that the proposed classifier performs very well for **D4**. However, the AUC is more than 90% for other three diseases **D1**, **D2** and **D3** as well. It means that our classifier is performing better for these diseases also.

Table VI shows the comparison of various techniques that are used in plant disease detection. Using the KNN classifier, an accuracy of 93% is achieved based on the dataset of leaf images collected over the sugar beet plant [11]. A disease detection technique applied over citrus plants is achieved by combining a feature selection method and a classifier over quarantine illness symptoms. The study has resulted in 84% accuracy by using the MLP model [12]. Another study conducted over wheat plant uses neural network for multiple disease detection. The dataset used in the study was image-based, and the model had produced an accuracy of 89.5% [10]. Various other models, such as GoogleNet, AlexNet, and SqueezeNet have been used to identify plant diseases [14] [15]. All these methods have been implemented over image-based datasets to attain an accuracy of 80.75%, 95.65%, and 94.3%, respectively. RPN based on convolutional neural network, is proposed to identify leaf blight, a common fungal disease in maize crops. The proposed model detects the diseased leaves with a high mean average precision of 91.83% [17]. DCNN method carried out symptom-wise disease recognition by taking symptom images as input. The method achieved good recognition results with an accuracy of 93.4% on disease images captured in field conditions [16]. It has been observed that the proposed model has outperformed many of the listed techniques in Table VI with subset accuracy of 94.36%. Looking at the results as mentioned in Table VI, one can conclude that sensors can play an essential role in solving plant disease detection problem, and can result in highly accurate prediction of plant diseases along with reducing the overall setup cost that farmers have to incur in their field.

VII. CONCLUSION

By controlling the abiotic factors, one can enhance the productivity and quality of the plants as well as their products, but if not controlled timely, it can result in severe losses. The paper addresses how the sensors can be incorporated to look into the insights over various abiotic factors that can help in

identifying plant diseases. The paper has proposed a real-time detection approach based on MLP model for plant diseases which takes input as ten features and 4139 data points. This neural network based approach, thus detects the four common types of diseases with high accuracy. We achieved a subset accuracy of 94.36% on the test set and Hamming loss of 0.0144, which demonstrates the feasibility of the approach used. Most of the individual testing accuracy for disease over the test set are higher than 98% using proposed multi-label classifier, which demonstrates the efficiency of the proposed work. Incorporating sensors in the study has decreased the overall cost of the expert system that makes it cost-effective, reliable, and robust. This system is easy to deploy and use in the field. Future work of this study is to deploy more sensors and system across various locations and understand the spatial variations. The accuracy achieved in this work can be further improved by incorporating more sensors and copious testing will help us to improve the developed model performance. The predictive model should incorporate the various diseases for different crops and provide the control measures to protect the crop. Thus, it helps in solving the problem of plant disease detection using inexpensive sensors and deep neural networks.

VIII. Acknowledgement

The authors are thankful for the support received from Prof. B. S. Parmar for providing space in the agriculture filed at Sardarkrushinagar Dantiwada Agricultural University (SDAU). The authors are thankful to Department of Science and Technology (DST) for financial assistance. VSP would like to thank DST-SERB for the start-up grant (FILE NO. SRG/2019/000895).

IX. Appendix

A. Neural Network Characterization

1) Selection of Neural Network Architecture: The results of various structures of neural networks that are simulated for the selection of the proposed neural network model are presented here. It can be observed from Table VII, that the chosen neural network architecture (Model-5) gives the lowest hamming loss as 0.0144 with the highest subset accuracy as 94.36% over other architectures used in the experiment. It can also be observed from the table that three hidden layers are used in the proposed neural network model. The table indicates that if we increase the number of layers to four, it impacts various evaluation measures such as subset accuracy and hamming loss due to over-fitting. This can be seen from the Table 1 below where we can clearly observe that each time we increase the size of the neural network, it gets more adaptive, so it learns even the smaller details, hence an increase in accuracy is observed. But for Model-6, the accuracy has dropped, which clearly suggests that the respective model gets more affected by over-fitting and the generalization of the classifier is also decreased.

2) Selection of Activation Function: In this section, experimental results for various activation functions are included. Here, the proposed model has been used with an input layer of 10 neurons, three hidden layers with 36-30-24 neurons respectively, and an output layer with four neurons each representing one of the four diseases (**D1**, **D2**, **D3**, **D4**). Fig. 12, presented results for multiple activation functions in hidden layers such as $linear$, $relu$, $tanh$, and $leaky\text{-}relu$ [53]. It can be observed that the $linear$ activation has been outperformed by every other activation function used in this experimental setup. While training for 200 epochs, we also observed that $leaky\text{-}relu$ and $tanh$ converge faster than $relu$, but at later epochs of training, it becomes clear that $relu$ works best out of all four activation functions. At the end, after analysing the experimental results, it is concluded that $relu$ gives the best results in the training phase of the model in minimizing the binary-cross entropy loss function.

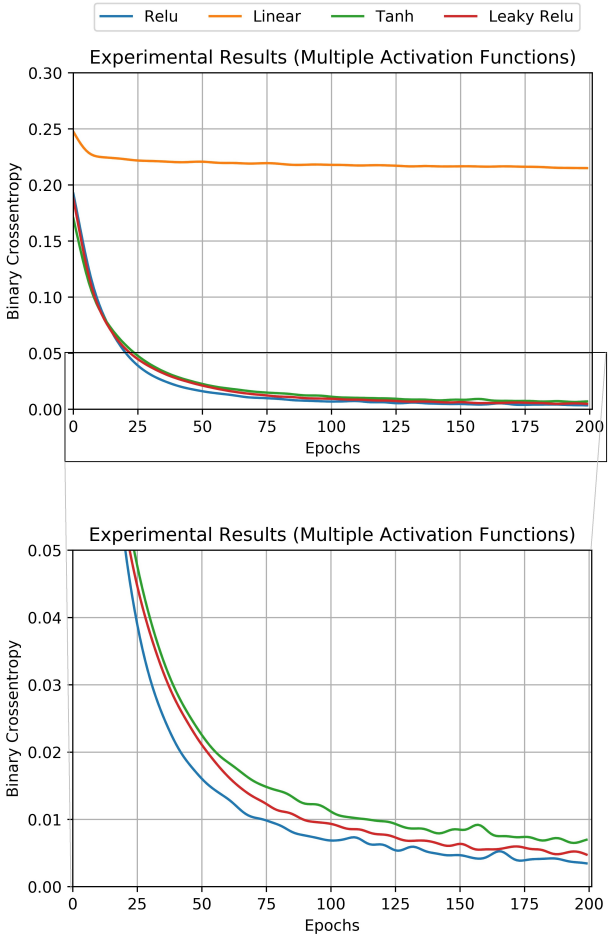


Fig. 12. Binary cross-entropy loss plot using different activation functions

3) Selection of Optimizer: In the proposed neural network, the optimizer used is $Adam$ optimizer. The optimizer is chosen based on the experimental work performed on the neural network. The experimental setup is based on MLP model with distinct optimizers being used in training. In this experiment, six distinct optimizers, such as SGD (stochastic gradient descent), $RMS\text{-}prop$, $FTRL$ (Follow The Regularized Leader), $Ada\text{-}grad$, $Ada\text{-}delta$, and $Adam$ [53] have been used. The main problem of using SGD is the global learning rate associated with the same. Hence, it does not work well when parameters are on different scales since a low learning rate will

TABLE VII
COMPARISONS OF VARIOUS NEURAL NETWORK ARCHITECTURE.

Model	Input Layer (No. of neurons)	Hidden Layer		Output Layer (No. of neurons)	Total params.	Accuracy (Subset Accuracy)	Hamming Loss
		Layers	Neurons				
1.	10	1	16	4	244	88.88	0.0305
2.	10	2	16-12	4	432	90.71	0.0213
3.	10	3	20-16-12	4	812	91.33	0.0186
4.	10	3	30-24-16	4	1542	93.68	0.0152
5.	**10**	**3**	**36-30-24**	**4**	**2350**	**94.36**	**0.0144**
6.	10	4	36-32-32-24	4	3528	93.81	0.0165

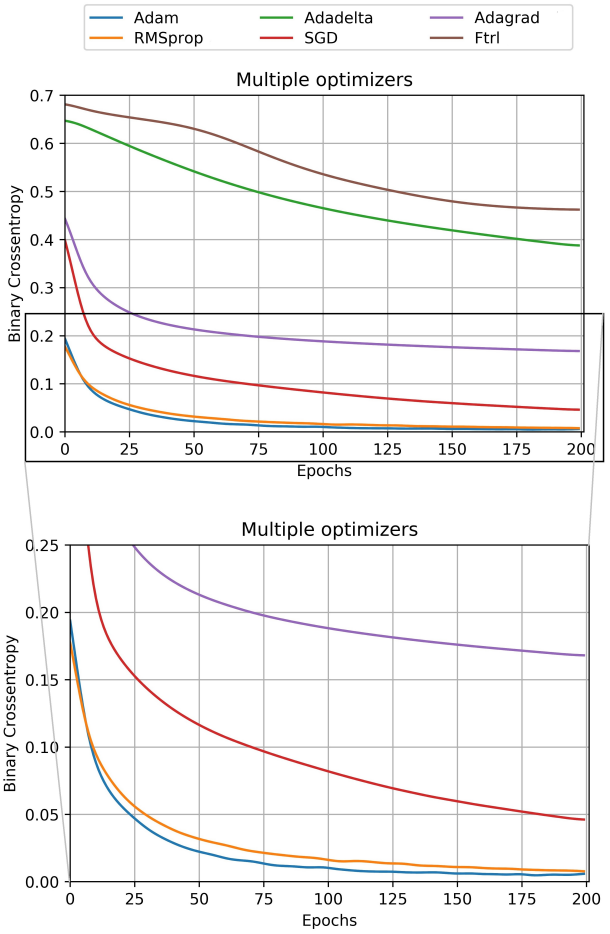


Fig. 13. Binary cross-entropy loss plot using different optimizers.

make the learning slow while a higher value of learning rate might lead to oscillations. It generally has a hard time escaping the saddle points. *Ada-grad* penalizes the learning rate too harshly for parameters that are frequently updated and assigns more learning rates to sparse parameters that are not updated frequently. *RMS-Prop* is similar to *Ada-delta* with the only difference that in *Ada-delta*, one cannot define the initial learning rate constant. Another method, *Adam* computes adaptive learning rates for each parameter. It combines the useful properties of *Ada-delta* and *RMS-prop* and hence, tends to perform better for most of the problems.

From Fig. 13, one can observe that *Adam* has outperformed every other optimizer that was taken into consideration. *FTRL* and *Ada-delta* have performed the worst out of six optimizers. *Adam* optimizer converged faster than *RMS-prop* but at higher epochs minimizes the binary cross-entropy loss function by 0.15 more than *RMS-prop* optimizer. So it can be concluded that *Adam* has performed best of all six optimizers.

REFERENCES

[1] United Nations, Department of Economic and Social Affairs, Population Division (2019), "World Population Prospects 2019: Highlights (ST/ESA/SER.A/423)." 2019.

[2] I. Z. et al, "An automated detection and classification of citrus plant diseases using image processing techniques: A review," *Computers and electronics in agriculture*, vol. 153, pp. 12–32, 2018.

[3] A. Jain, S. Sarsaiya, Q. Wu, Y. Lu, and J. Shi, "A review of plant leaf fungal diseases and its environment speciation." *Bioengineered*, pp. 409–424, 2019.

[4] S. S. Chouhan, A. Kaul, U. P. Singh, and S. Jain, "Bacterial foraging optimization based radial basis function neural network (brbfnn) for identification and classification of plant leaf diseases: An automatic approach towards plant pathology," *IEEE Access*, vol. 6, p. 8852–8863, 2018.

[5] Y. Fang and R. P. Ramasamy, "Current and prospective methods for plant disease detection," *Biosensors, www.mdpi.com/journal/biosensors/*, vol. 4, pp. 537–561, 2015.

[6] K. Golhani, S. K. Balasundram, G. Vadamalai, and B. Pradhan, "A review of neural networks in plant disease detection using hyperspectral data," *INFORMATION PROCESSING IN AGRICULTURE*, vol. 5, pp. 354–371, 2018.

[7] M. Ray, A. Ray, S. Dash, A. Mishra, K. G. Achary, S. Nayak, and S. Singh, "Current and prospective methods for plant disease detection," *Biosensors and Bioelectronics*, vol. 87, pp. 708–723, 2017.

[8] R. Kaundal, A. S. Kapoor, and G. P. Raghava, "Machine learning techniques in disease forecasting: a case study on rice blast prediction," *BMC Bioinformatics*, p. 47–58, 2006.

[9] Y. Chtioui, S. Panigrahi, and L. Francl, "A generalized regression neural network and its application for leaf wetness prediction to forecast plant disease," *Elsevier, Chemometrics and Intelligent Laboratory Systems*, vol. 48, p. 47–58, 1999.

[10] H. Wang, G. Li, Z. Ma, and X. Li, "Image recognition of plant diseases based on principal component analysis and neural networks," *2012 8th International Conference on Natural Computation*, pp. 246–251, 2012.

[11] S. Bauer, F. Korc, and W. Förstner, "The potential of automatic methods of classification to identify leaf diseases from multispectral images," *Precision Agriculture*, vol. 12, pp. 361–377, 2011.

[12] G. Stegmayer, D. H. Milone, S. Garran, and L. Burdyn, "Automatic recognition of quarantine citrus diseases," *Expert Systems With Applications*, vol. 40, pp. 3512–3517, 2013, lugar: Amsterdam. [Online]. Available: http://sinc.unl.edu.ar/sinc-publications/2013/SMGB13

[13] J.-C. Zhang, R. liang Pu, J. hua Wang, W. jiang Huang, L. Yuan, and J. hua Luo, "Detecting powdery mildew of winter wheat using leaf level hyperspectral measurements," *Computers and Electronics in Agriculture*, vol. 85, pp. 13–23, 2012.

[14] J. G. Barbedo, "Factors influencing the use of deep learning for plant disease recognition," *Biosystems Engineering*, vol. 172, pp. 84 – 91, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1537511018303027

[15] H. Durmuş, E. Gunes, and M. Kirci, "Disease detection on the leaves of the tomato plants by using deep learning," 08 2017, pp. 1–5.

[16] J. Ma, K. Du, F. Zheng, L. Zhang, Z. Gong, and Z. Sun, "A recognition method for cucumber diseases using leaf symptom images based on deep convolutional neural network," *Computers and Electronics in Agriculture*, vol. 154, pp. 18–24, 11 2018.

[17] J. Sun, Y. Yang, X. He, and X. Wu, "Northern maize leaf blight detection under complex field environment based on deep learning," *IEEE Access*, vol. 8, pp. 33 679–33 688, 2020.

[18] O. Elijah, T. A. Rahman, I. Orikumhi, C. Y. Leow, and M. Hindia, "An overview of internet of things (iot) and data analytics in agriculture: Benefits and challenges," *IEEE Internet of Things Journal*, vol. 5, no. 5, p. 3758–3773, 2018.

[19] H. Ochiai, H. Ishizuka, Y. Kawakami, and H. Esaki, "A dtn-based sensor data gathering for agricultural applications," *IEEE Sensors Journal*, vol. 11, no. 11, p. 2861–2868, 2011.

[20] F.-H. Tseng, H.-H. Cho, and H.-T. Wu, "Applying big data for intelligent agriculture-based crop selection analysis," *IEEE Access*, vol. 7, p. 116965–116974, 2019.

[21] S. Ivanov, K. Bhargava, and W. Donnelly, "Precision farming: Sensor analytics," *IEEE Intelligent systems*, vol. 30, no. 4, p. 76–80, 2015.

[22] V. S. Palaparthy, G. G. Shambhulingayya.N.D, P. Das, S. Chandorkar, S. Mukherji, M. S. Baghini, and V. Rao, "E-nose: Multichannel analog signal conditioning circuit with pattern recognition for explosive sensing," *IEEE Sensors Journal*, vol. 20, no. 3, p. 1373–1382, 2020.

[23] S. G. Surya, R. S. Dudhe, D. Saluru, B. K. Koora, D. K. Sharma, and V. Rao, "Comparison among different algorithms in classifying explosives using ofets," *Sensors Actuators B: Chemical (Elsevier)*, vol. 176, p. 46–51, 2013.

[24] H. Harb, A. Mansour, A. Nasser, E. M. Cruz, and I. Torre Díez, "A sensor-based data analytics for patient monitoring in connected healthcare applications," *IEEE Sensors Journal*, 2020.

[25] E. Wilhelm, S. Siby, Y. Zhou, X. J. S. Ashok, M. Jayasuriya, S. Foong, J. Kee, K. L. Wood, and N. O. Tippenhauer, "Wearable environmental sensors and infrastructure for mobile large-scale urban deployment," *IEEE Sensors Journal*, vol. 16, no. 22, p. 8111–8123, 2016.

[26] A. Mamun, M. Abdulla, and M. R. Yuce, "Sensors and systems for wearable environmental monitoring toward iot-enabled applications: A review," *IEEE Sensors Journal*, vol. 19, no. 18, p. 7771–7788, 2019.

[27] J. Gonzalez-Andujar, "Expert system for pests, diseases and weeds identification in olive crops," *Expert Systems with Applications*, vol. 36, pp. 3278–3283, 2009.

[28] L. Gonzalez-Diaz, P. Martínez-Jimenez, F. Bastida, and J. Gonzalez-Andujar, "Expert system for integrated plant protection in pepper (capsicum annuun l.)," *Expert Systems with Applications*, vol. 36, p. 8975–8979, 2009.

[29] P. AK, B. RR, K. L, and N. RM, "Perspectives and challenges for sustainable management of fungal diseases of mungbean [vigna radiata (l.) r. wilczek var. radiata]: A review." *Frontiers in Environment Science*, vol. 6, 2018.

[30] D. Robinson, C. Campbell, J. Hopmans, B. K. Hornbuckle, S. B. Jones, R. Knight, F. Ogden, J. Selker, and O. Wendroth, "Soil moisture measurement for ecological and hydrological watershed-scale observatories," *A review." Vadose Zone Journal 7*, no. 1, p. 358–389, 2008.

[31] *ASTM D 2216, Standard Test Methods for Laboratory Determination of Moisture (Moisture) Content of Soil*. West Conshohocken, PA: ASTM International, 2008.

[32] D. ASTM, *Standard Test Method for Field Determination of Moisture (Moisture) Content of Soil by the Calcium Carbide Gas Pressure Tester*. West Conshohocken, PA: ASTM International, 2008.

[33] V. Palaparthy, M. Baghini, and D. Singh, "Review of polymer-based sensors for agriculture-related applications," *Emerging Materials Research*, vol. 2, p. 166–80, 2013.

[34] G. Topp, J. Davis, and A. Annan, "Electromagnetic determination of soil water content using tdr: I: Applications to wetting fronts and steep gradients ii: Evaluation of installation and configuration of parallel transmission lines." journal of the american society of soil science 49," p. 672–678, 1982.

[35] J. Whalley, W. R., T. Dean, and P. Izzard, "Evaluation of the capacitance technique as a method for dynamically measuring soil water content." journal of agricultural engineering research 52," p. 147–155, 1992.

[36] P. Aravind, "A wireless multi-sensor system for soil moisture measurement," in *2015 IEEE SENSORS, IEEE2015*, p. 1–4.

[37] J. John, "A multi-hop wireless sensor network for in-situ agricultural applications," in *2019 URSI Asia-Pacific Radio Science Conference (AP-RASC*. IEEE, 2019.

[38] N. Jorapur, "A low-power, low-cost soil-moisture sensor using dual-probe heat-pulse technique". sensors and actuators a," p. 108–117, 2015.

[39] H. Kalita, V. Palaparthy, M. Baghini, and M. Aslam, "Graphene quantum dot soil moisture sensor," *Sensors and Actuators B: Chemical*, vol. 233, p. 582–90, 2016.

[40] V. Palaparthy, H. Kalita, S. Surya, M. Baghini, and M. Aslam, "Graphene oxide based soil moisture microsensor for in situ agriculture applications," *Sensors and Actuators B: Chemical*, vol. 273, p. 1660–9, 2018.

[41] T. Jackson, K. Mansfield, M. Saafi, T. Colman, and P. Romine, "Measuring soil temperature and moisture using wireless mems," p. 381–390, 2008.

[42] *Soilsens Tech Pvt. Ltd.*, 2019. [Online]. Available: https://soilsens.com/

[43] *Sardarkrushinagar Dantiwada Agricultural University, Satsan, Gujarat, India*, 2019. [Online]. Available: http://www.sdau.edu.in/

[44] *MCP9701A*, 2019. [Online]. Available: http://ww1.microchip.com/downloads/en/DeviceDoc/20001942G.pdf

[45] *HIH-5030-001*, 2019. [Online]. Available: https://sensing.honeywell.com/honeywell-sensing-hih5030-5031-series-product-sheet-009050-2-en.pdf

[46] "AESA based IPM - Blackgram and Greengram, National Institute of Plant Health Management, 2014." *Department of Agriculture and Cooperation, Ministry of Agriculture, Government of India*, 2014.

[47] P. Virtanen et al., "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," *Nature Methods*, vol. 17, pp. 261–272, 2020.

[48] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[49] J. D. Hunter, "Matplotlib: A 2d graphics environment," *Computing in Science & Engineering*, vol. 9, no. 3, pp. 90–95, 2007.

[50] M. Waskom, O. Botvinnik, D. O'Kane, P. Hobson, S. Lukauskas, D. C. Gemperline, and A. Qalieh, "Seaborn," 2017.

[51] F. Provost and T. Fawcett, *Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking*, 1st ed. O'Reilly Media, Inc., 2013.

[52] S. Haykin, *Neural Networks: A Comprehensive Foundation*, 2nd ed. USA: Prentice Hall PTR, 1998.

[53] M. Abadi et al., "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: https://www.tensorflow.org/

[54] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," 2014.

[55] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014.

[56] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks."

[57] M. H. Saleem, J. Potgieter, and K. M. Arif, "Plant disease detection and classification by deep learning," *MDPI - Multidisciplinary Digital Publishing Institute*, vol. 8, 2019.

[58] F. Chollet *et al.*, "Keras," 2015.

Manish Kumar has completed M.Tech. in ICT (Information and Communication Technology) at DA-IICT in 2020, with a specialization in Machine Intelligence. He has received his B.Tech. degree in Computer Science and Engineering from Dr. A.P.J. Abdul Kalam University, Lucknow in 2015. He also has an experience of 3 years in IT industry and worked on various cutting-edge technologies. His current research interests include developing machine learning algorithms, data science and solving real-time problems.

Dr. Ahlad Kumar has joined DA-IICT in July 2019. He has served as the postdoctoral fellow in Concordia University and won the prestigious Horizon Postdoctoral Fellowship at Concordia University, Montreal, Canada in 2017. Dr. Kumar holds a Ph.D. from the University of Malaya, Malaysia in 2016. He has received the gold medal for his performance during M.Tech. (VLSI) from ABV-Indian Institute of Information Technology and Management, Gwalior in 2007. He received the B.Tech. degree in Electronics and Communication Engineering from Jamia Millia Islamia in 2005.

Dr. Vinay Palaparthy is working as the assistant professor in DA-IICT. He has received Ph.D degree from Indian Institute of Technology, Bombay (IIT Bombay). He is awarded with DST INSPIRE Fellowship for pursing Ph.D degree from Department of Science and Technology (DST). He has around 2 years of research experience in the field of MEMS and System design. He is a co-recipient of Millennium Alliance award for the start-up name Proximal Soilsens Technologies Pvt. Ltd, where he is a co-founder and director.