Indian Institute of Information Technology Vadodara (Gandhinagar Campus)

Design Project Report-2022

on

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

under the supervision of

Dr. Kamal Kishore Jha

submitted by

Gopisetti Giridhar
202051074

Kayala Vishwaksen Reddy
202051101

Kommula Chiranjeevi Sagar
202051103

***Abstract*—Plant diseases reduce crop yields, which affects the economy. As a result, prediction models for plant disease detection and evaluation must be created. If caught early enough, the major disease, fungus infection, may be treated by adopting the proper precautions. Powdery mildew, anthracnose, rust, and root rot/leaf blight are some of the fungal diseases that this project aims to design an expert system for the prediction of. For the categorization of diseases, a multi-layered perceptron model is adopted, which not only successfully identifies plant problems but also could massively improve productivity. The suggested method includes three crucial steps: exploratory data analysis, dataset preparation, and detection module. On an average More than 94% of predictions for each disease turned out to be accurate. This result determines the practicality of implementing this method for faster plant disease detection at an affordable cost.**

Key words - plant diseases, artificial neural networks, machine learning.

I. Introduction

The quality of leaves, fruits, stems, vegetables, and their products suffers as a result of the approximately 20,000 species of parasitic fungus that cause diseases in crops and plants. If these infectious crop diseases are not promptly treated, the production might be drastically reduced, affecting global food security. Farmers can be helped in protecting their crops using early disease identification and adoption of control methods. The world's population is projected to increase by 200 crores, more than a quarter, over the span of the next 30 years, according to the United Nations in 2019 [1]. The Food and Agricultural Organization research estimates that an additional 70–90% more food would be needed to feed this population. Nearly 16% of the world's total agricultural crop yield has been damaged by microbial diseases. There is a need for better disease detection to stop crop damage in order to reduce the growth of diseases, increase production, and ensure agricultural sustainability.

II. LITERATURE SURVEY

The previous on this project has been published in IEEE sensors journal [2]. They have discussed about the diseases in green gram and the problems that are being faced by farmers in the fields. They deployed the sensors in the field and collected data and most of the features are from satellite data. Through the data that has been collected they have developed a machine learning model with 10 features and then an artificial neural network. They have developed this prediction model with the help of libraries scikit-learn, tensor flow, matplotlib.

III. DISEASES IN GREEN GRAM

In this section, various fungal diseases in green gram crops have been studied and used in our work [2].

A. Powdery mildew (D1)

Patches of white powdery substance appear on leaves and other green parts from time to time, gradually expanding to cover the lower leaf surface if not handled. In severe infections, both sides of the leaf are completely covered by whitish powdery growth, resulting in an infected plant and a significant yield loss. These pathogens have a broad host range and live in conidial form on a variety of off-season hosts, where they spread via seasonally developed airborne conidia.

B. Anthracnose (D2)

The disease occurs primarily on leaves and pods as circular, black, sunken spots with a dark center and bright red-orange margins. In severe infections, infected sections die. It has a negative impact on seedlings because it becomes blighted soon after seed germination due to infection. The pathogen feeds primarily on seed and plant waste and spreads via airborne conidia.

C. Rust (D3)

In this case, the symptoms are described as circular reddish-brown pustules on the underside of the leaves. Rust pustules completely cover both sides of the surfaces in severe cases. Defoliation is followed by shriveling, which results in yield loss. The sporidia that develop from teliospores cause the primary infection. Wind-borne uredospores are responsible for secondary distribution. The fungus can also be found on other host legumes.

D. Root rot and leaf blight (D4)

In its early stages, the fungus causes seed rot, seedling blight, and root rot. The affected leaves turn yellow, and irregular brown lesions appear on them, forming large blotches and causing the leaves to die prematurely. The roots and the basal part of the stem turn black, and the bark peels off quickly. Pathogens in green gramme cause seed decline, root rot, damping-off, seedling blight, stem canker, and leaf blight. The primary cause of infection is saprotrophic pathogens found in soil, while asexual spores cause secondary infection.

IV. MATERIALS AND METHODS

A. Proposed Method

A neural network has been discussed for the classification of diseases caused by various fungal diseases (Section III). To perform this classification task, the data was first manually collected. Once the data has been properly cleaned and labeled, an exploratory data analysis (V-B) must be performed to gain more insight into the dataset [4]. Following that, the dataset was divided into training and testing datasets. The proposed neural network model must be trained and tested using these training and test datasets, as discussed in V-B. Table-I provides the numerical ranges of various features F0, F1, F2, and F3, which are specific to the occurrence of a particular disease.

Disease D	F0 Rainfall	F1 Ambient Temperature	F2 Ambient Humidity	F3 Soil Moisture
D1	<1%	10-20°C	90-100%	10-14%
D2	<1%	10-15°C	80-100%	10-14%
D3	0.5-1%	21-26°C	75-100%	10-14%
D4	>0.5%	>30°C	>80%	10-14%

Fig. 1. Table-1:Numerical Range of features for a particular disease

B. Exploratory Data Analysis

To identify the significant features of the dataset and generate insights for further investigation. Here, the data has been analyzed using statistics, data visualization using Matplotlib library, and other techniques to get meaningful insights into the dataset before applying machine learning models for classification [4]. The dominant features present in the dataset have been measured by using the feature importance property of the model(fig.2).

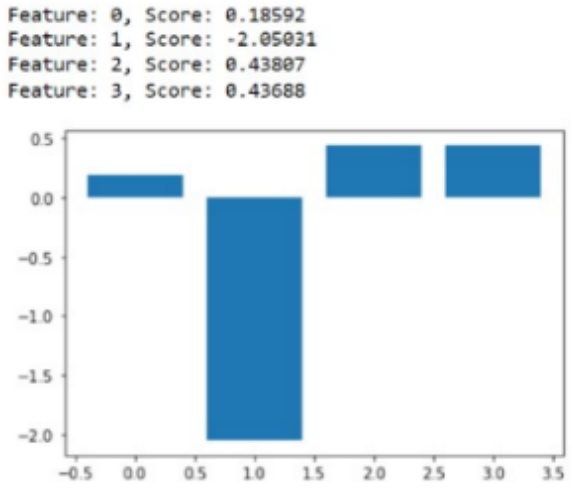

Fig. 2. Dominant features based on important score

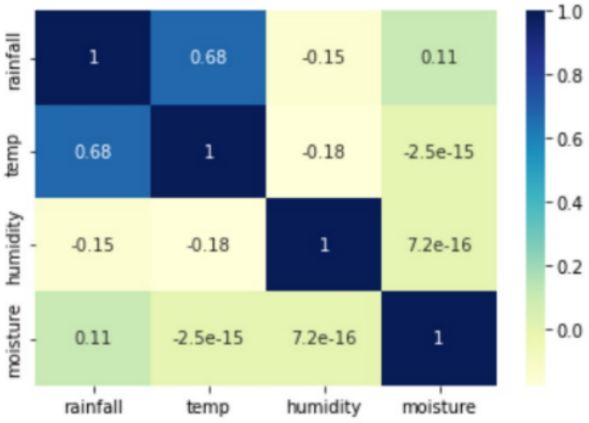

Fig. 3. Correlation plot among the features

Fig.3 depicts the correlation between various features, emphasizing how they are related to one another. The correlation can be positive (an increase in one feature's value raises the value of the target variable) or negative (an increase in one feature's value lowers the value of the target variable). Using the seaborn library, a heatmap of correlated features has been plotted (Fig.2), making it easy to identify which features are strongly related to the target variable. variable. On analysis of the heatmap, it has been found that the feature pair (F0, F1) is strongly correlated having correlation values greater than 0.5.

C. Artificial Neural Network Model

Artificial neural networks (ANNs) are well-suited for classification and prediction problems in which inputs are classified (label). First, the fundamentals of ANN were discussed, followed by an introduction to the proposed architecture. ANN has a parallel distributed computational model made up of structures and functions of biological neural networks that aid in the recognition of relationships in a dataset that mimics how the human brain works. ANN are excellent tools for discovering patterns that are far too complex for a human programmer to extract and teach to the machine. With sufficient data and proper initialization, ANNs can provide optimal solutions by adjusting their inner structure. In the artificial neural network model we have developed, we use relu as activation function for input and hidden layers and softmax as activation function for output layer. We used 'adam' as optimizer with categorical crossentropy loss function. We implemented the model with a batch size of 20 and have done 100 epochs to our model. I have checked the model with many hidden layers and activation functions and i found that three hidden layers with relu activation function gives best performance.

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 experiment have been given in Section VI-A,. Further, in Section VI-B, various evaluation measures such as precision, recall and F-score that are used in the paper have been presented.

A. Experimental Setup

All the experiments based on the Scikit-learn, Matplotlib, Seaborn, and TensorFlow library have been carried out. Scikit-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 is a python-friendly open source library for numerical computation that makes machine learning and developing neural networks faster and easier [5]. 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 80% of the data, whereas the testing dataset constitutes about 20%.

B. Evaluation Measures

To assess the performance of the proposed MLP for disease classification, four standard evaluation measures, namely precision, recall, F-score and accuracy were used. The accuracy is a metric used to ensure that the ratio of true label predictions is correct. Precision is a measure of how many correct predictions there are. Recall is a percentage of true labels that were correctly predicted. The F-score is the harmonic mean of precision and recall, and it is a popular evaluation scheme in machine learning research. The Fig.4 demonstrates equations on how to compute these values. True positive, true negative, false positive, and false negative are denoted as TP, TN, FP, and FN, respectively.

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

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

Fig. 4. Equations

VI. RESULTS AND DISCUSSION

```
In [15]: confusion_matrix(b_test,a_test_prediction)
Out[15]: array([[201,  38,   3,   0],
                [ 85,  41,   0,   0],
                [  6,   0, 142,   8],
                [  0,   0,  10, 368]], dtype=int64)

In [16]: matrix = classification_report(b_test,a_test_prediction)
         print(matrix)

              precision    recall  f1-score   support

           D1       0.69      0.83      0.75       242
           D2       0.52      0.33      0.40       126
           D3       0.92      0.91      0.91       156
           D4       0.98      0.97      0.98       378

    accuracy                           0.83       902
   macro avg       0.78      0.76      0.76       902
weighted avg       0.83      0.83      0.82       902
```

Fig. 5. The values of the precision, recall and f-score

The Fig.5 gives the calculated values of the above measures of our model. First we have trained our model with a machine learning model, logistic regression [5]. From this model we got the accuracy of around 83.3 percent.

```
In [7]: a_train_prediction = model.predict(a_train)
        training_data_accuracy = accuracy_score(a_train_prediction,b_train)
        print(training_data_accuracy)

        0.8508869179600886

In [8]: a_test_prediction = model.predict(a_test)
        testing_data_accuracy = accuracy_score(a_test_prediction,b_test)
        print(testing_data_accuracy)

        0.8337028824833703
```

Fig. 6. Accuracy of machine learning model

Fig.6 and Fig.7 gives the values of the accuracy of the two

models. Machine Learning model has given us an accuracy of 83.3 percent.So,we have developed artificial neural network model which gave us an accuracy of around 92.5 percent.

```
In [23]: m = tf.keras.metrics.Accuracy()
         m.update_state(np.argmax(b_test_cat, axis=1), np.argmax(b_pred,axis=1))
         m.result().numpy()

Out[23]: 0.924612
```

Fig. 7. Accuracy of ANN model

Finally, we created a user interface that forecasts the results and indicates what diseases might manifest for a given set of data is shown in Fig.8.

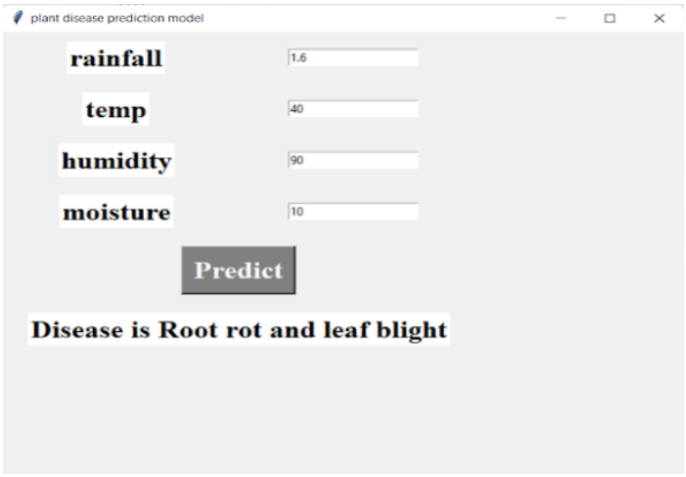

Fig. 8. Interface of the model

VII. CONCLUSION AND FUTURE WORK

From the work we have made, we brought the accuracy of around 93% , which is a good prediction accuracy for the given constraints and we have not taken the data from either satellite nor the sensor's. Further, we are trying to take the dataset from the sensors placed in the crop field and test the neural network model for more accuracy. Further more work can also be done on this project. In the future, there is a scope for development of IoT robots which will directly give pesticide to plants based on the disease it predicts [3].

VIII. ACKNOWLEDGMENT

From this project we have developed our fundamentals on ML and neural networks and we are thankful to the IIIT vadodara and our mentor Dr.Kamal Kishore Jha sir for this opportunity and encouraging us for the development of this project.

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] M. Kumar, A. Kumar and V. S. Palaparthy, "Soil Sensors-Based Prediction System for Plant Diseases Using Exploratory Data Analysis and Machine Learning," in IEEE Sensors Journal, vol. 21, no. 16, pp. 17455-17468, 15 Aug.15, 2021, doi: 10.1109/JSEN.2020.3046295.

[3] 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

[4] 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.

[5] 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.