VISVESVARAYA TECHNOLOGICAL UNIVERSITY

Jnana Sangama, Belagavi - 590018

Project Report

on

"MACHINE LEARNING BASED EARLY DETECTION OF YELLOW LEAF DISEASE IN ARECANUT PLANT USING SOIL SAMPLES"

Submitted in partial fulfillment of the requirements for the award of the degree of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE & ENGINEERING

By

PRAKHYATH DEVADIGA	**4MT19CS109**
PRATHIKSHA G K	**4MT19CS114**
PRATHIKSHA S	**4MT19CS115**
SHERVEGAR SHRINIDHI NAGARAJA	**4MT19CS146**

Under the Guidance of

Ms. SHWETHA KAMATH

Assistant Professor

MITE

Invent Solutions

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

(*Accredited by NBA*)

MANGALORE INSTITUTE OF TECHNOLOGY & ENGINEERING

Accredited by NAAC with A+ Grade, An ISO 9001: 2015 Certified Institution

(A Unit of Rajalaxmi Education Trust®, Mangalore - 575001)

Affiliated to VTU, Belagavi, Approved by AICTE, New Delhi

Badaga Mijar, Moodabidri-574225, Karnataka

2022-23

MANGALORE INSTITUTE OF TECHNOLOGY & ENGINEERING

Accredited by NAAC with A+ Grade, An ISO 9001: 2015 Certified Institution

(A Unit of Rajalaxmi Education Trust®, Mangalore - 575001)

Affiliated to VTU, Belagavi, Approved by AICTE, New Delhi

Badaga Mijar, Moodabidri-574225, Karnataka

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

(*Accredited by NBA*)

CERTIFICATE

This is to certify that the project work entitled **"MACHINE LEARNING BASED EARLY DETECTION OF YELLOW LEAF DISEASE IN ARECANUT PLANT USING SOIL SAMPLES"** is a bonafide work carried out by **Prakhyath Devadiga (4MT19CS109), Prathiksha G K (4MT19CS114), Prathiksha S (4MT19CS115), Shervegar Shrinidhi Nagaraja (4MT19CS146)** in partial fulfillment for the award of degree of **Bachelor of Engineering in Computer Science & Engineering of the Visvesvaraya Technological University, Belagavi during the year 2022–23**. It is certified that all corrections and suggestions indicated for Internal Assessment have been incorporated in the report deposited in the departmental library. The project has been approved as it satisfies the academic requirements in respect of project work prescribed for the Bachelor of Engineering degree.

Ms. Shwetha Kamath	**Dr. Ravinarayana B.**	**Dr. Prashanth C M**
Assistant Professor	**Head of the Department**	**Principal**

External Viva

Name of the Examiners	**Signature with Date**
1)	
2)	

ABSTRACT

Arecanut is an important cash crop grown extensively in India, especially in the southern states. It is also known as betel nut and is a major source of income for many farmers in the region. Arecanut is consumed widely in the form of chewing and is an integral part of social and religious practices in India. However, one of the major threats to arecanut cultivation is the Yellow Leaf Disease (YLD). The disease is characterized by yellowing of leaves, stunted growth, and reduced yield, leading to significant economic losses for farmers. The disease is also highly contagious and can spread rapidly, making early detection crucial for effective mitigation measures. Traditionally, the detection of YLD has relied on visual symptoms, which can be challenging to identify in the early stages. This delay in detection can result in significant crop losses. Therefore, there is a need for an early detection system that can identify the disease at an early stage and help farmers take necessary measures to control its spread. In this context, machine learning-based solutions using soil samples have emerged as a promising approach for early detection of YLD in arecanut plants. By analyzing various soil parameters, ID3 algorithm can detect the presence of the disease at an early stage. This can help farmers take timely action to prevent the spread of the disease and improve crop yields.

ACKNOWLEDGEMENT

The satisfaction and the successful completion of this project would be incomplete without the mention of the people who made it possible, whose constant guidance encouragement crowned our efforts with success.

This project is made under the guidance of **Ms. Shwetha Kamath**, **Assistant Professor**, Department of Computer Science and Engineering. We would like to express our sincere gratitude to our guide for all the helping hand and guidance in this project.

We would like to thank our project coordinator **Mr. Shreejith K B, Assistant Professor,** Department of Computer Science and Engineering, for his cordial support, valuable information and guidance, which helped us in completing this project through the various stages.

We would like to express appreciation to **Dr. Ravinarayana B , Associate Professor and Head**, Department of Computer Science and Engineering, for his support and guidance.

We would like to thank our Principal **Dr. Prashanth C M**, for encouraging us and giving us an opportunity to accomplish the project.

We also thank our management who helped us directly or indirectly in the completion of this project.

Our special thanks to faculty members and others for their constant help and support.

Above all, we extend our sincere gratitude to our parents and friends for their constant encouragement with moral support.

PRAKHYATH DEVADIGA	4MT19CS109
PRATHIKSHA G K	4MT19CS114
PRATHIKSHA S	4MT19CS115
SHERVEGAR SHRINIDHI NAGARAJA	4MT19CS146

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter–1

INTRODUCTION

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

Arecanut is a significant cash crop that is extensively cultivated in several parts of the world, especially in India, Indonesia, and the Philippines. However, Arecanut plantations are threatened by several fungal diseases, including the Yellow Leaf Disease (YLD). Yellow Leaf Disease is a devastating fungal disease that affects the Arecanut plant's growth and productivity, leading to significant economic losses for the farmers.

The early detection of Yellow Leaf Disease is crucial to control the disease's spread and prevent economic losses. Traditional methods of Yellow Leaf Disease detection rely on visual inspection, which is time-consuming and not always reliable. Therefore, there is a need for a more reliable and cost-effective method for the early detection of Yellow Leaf Disease.

In this project, we propose a machine learning-based approach for the early detection of Yellow Leaf Disease in Arecanut plants using soil samples. The proposed method will use soil samples collected from the Arecanut plantations to identify the presence of the fungal pathogen responsible for Yellow Leaf Disease. Machine learning algorithms and statistical techniques will be used to analyze the soil samples and predict the probability of Yellow Leaf Disease infection in the Arecanut plants.

The project's main objective is to provide a reliable and cost-effective early detection system for Yellow Leaf Disease in Arecanut plants, which can be used by farmers to prevent the spread of the disease and improve their yields. The proposed system will be based on the analysis of soil samples, which can provide early indications of the presence of Yellow Leaf Disease before any visible symptoms appear in the Arecanut plants. Overall, the project aims to contribute to the development of sustainable and efficient agricultural practices by utilizing the power of machine learning and data analysis.

1.2 PROBLEM STATEMENT

Yellow leaf disease is a significant threat to the arecanut crop, causing significant losses in yield and revenue for farmers. The disease is caused by a fungal pathogen that affects the leaves of the plant, leading to the development of yellow spots and ultimately, the death of the affected leaves.

Early detection of the disease is critical to prevent its spread and minimize its impact on crop yield. However, the current methods of detection are primarily based on visual observation, which may not be effective in identifying the disease at an early stage. Moreover, there is a lack of reliable and efficient systems that can accurately detect yellow leaf disease in arecanut plants at an early stage, particularly in large-scale plantations.

Therefore, this project is to develop a machine learning-based system that can effectively detect yellow leaf disease in arecanut plants at an early stage using soil samples. The system should be reliable, accurate, and user-friendly, enabling farmers and agricultural experts to detect the disease and take necessary precautions to prevent its spread and protecttheir crops. By addressing this problem, the project aims to contribute to the development of a sustainable and profitable arecanut farming industry.

1.3 OBJECTIVES

Our aim is to develop a system that can accurately detect the yellow leaf disease in arecanut plants at an early stage, using machine learning algorithms. Collect and preprocess soil sample data from arecanut plantations affected by yellow leaf disease to train and test the machine learning model. Identify the most relevant features from the soil sample data that can be used to detect yellow leaf disease. Help farmers take necessary precautions to prevent further spread of the disease and protect their crops from potential losses and to reduce the economic impact of yellow leaf disease on the agricultural industry by enabling early detection and timely intervention. Integrate the developed machine learning model into a user-friendly software application that can be used by farmers and agricultural experts for early detection of yellow leaf disease.

1.4 SCOPE OF THE PROJECT

Yellow Leaf Disease (YLD) is a common and devastating fungal disease affecting the Arecanut plant, which is a major cash crop in many parts of the world. The disease causes yellowing of leaves, stunting of growth, and a decrease in the yield of the plant. Early detection of the disease is crucial to prevent the spread of the disease and minimize the economic losses incurred by the farmers. The project aims to develop a machine learning-based model for the early detection of Yellow Leaf Disease in Arecanut plants using soil samples. The proposed model will use soil samples collected from the affected plants to identify the presence of the fungal pathogen responsible for the disease. The model will use various machine learning algorithms and statistical techniques to analyze the soil samples and predict the probability of Yellow Leaf Disease infection in the Arecanut plants. The ultimate goal of the project is to provide a reliable and cost-effective early detection system for Yellow Leaf Disease in Arecanut plants that can be used by farmers to prevent the spread of the disease and improve their yields.

1.5 ORGANIZATION OF THE REPORT

Chapter-1: Introduction

This chapter describes the organization of the report, objectives of the proposed model and brief introduction of our proposed model.

Chapter-2: Literature Survey

This chapter mainly deals with all the observation, which is conducted as initial study before the actual development of the project. It also describes the details regarding existing system and its disadvantages.

Chapter-3: System Requirement Specification

This chapter speaks about the product perspective, user characteristics, its assumptions and dependencies, specific requirements, functionality along with resource requirements.

Chapter-4: Gantt chart

This chapter deals with the Gantt chart. It is a type of bar chart, developed by Henry Gantt that illustrates a project schedule.

Chapter-5: System Design

This chapter deals with the advance software engineering where the entire flow of the project is represented by data flow diagram and the architecture of the project.

Chapter-6: Implementation

This chapter deals with the steps involved in the creation of the project work. It is defined with the help of code snippets for the ease of reader.

Chapter-7: Testing

This chapter deals with various test cases and test levels such as unit testing, system testing, integration testing and acceptance testing.

Chapter-8: Results and discussion

This chapter mainly deals with the graphical user interface of the project to show the output of the application.

Chapter-9: Conclusion and Future work

These sections provide the conclusion for the project and also suggest some of the enhancement idea which couldn't be covered up due to constraint of time and resources.

Chapter-2

LITERATURE SURVEY

CHAPTER 2

LITERATURE SURVEY

2.1 EXISTING SYSTEM

Shuhan Lei., et al. [1] proposed a UAV structure from motion technology is used to measure the LVV of a single areca tree and establish a model of the correlation between the LVV and the severity of the yellow leaf disease of arecanut. This study presents the novel quantitative expression of the severity of the yellow leaf disease of arecanut, along with the correlation between the LVV of areca and the severity of the yellow leaf disease of arecanut. Therefore, it is necessary to quantitatively analyze each areca tree and effectively improve the monitoring accuracy of the yellow leaf disease of arecanut. This process can help in increasing the assessment accuracy of the disease severity and yield of areca, and can also play an important role in the future control of the yellow leaf disease of arecanut.

Jiawei Guo., et al. [2] proposed a model which uses Random Forest (RF), Backward Propagation Neural Network (BPNN) and AdaBoost algorithms based on feature space optimization to construct double classification (healthy, diseased) monitoring models for the areca yellow leaf disease. Here results exhibit the feasible application of Planet Scope imagery for the regional large-scale monitoring of areca yellow leaf disease, with the RF method identified as the most suitable for this task. Areca yellow leaf disease is the most important factor harming arecanut production and planting. The continuous occurrence of areca yellow leaf disease and the obvious decline of the industrial benefits of arecanut have highlighted the urgent need of a method that timely and accurately detects the areca yellow leaf disease. In this model they employ images collected from Planet Scope to perform the large-scale monitoring of areca yellow leaf disease. The results provide a reference for the regional large-scale monitoring and prevention of yellow leaf disease.

Gittaly Sandhu., et al [3] proposed a system which uses a mobile client-server architecture for leaf disease detection and diagnosis using a novel combination of Gabor wavelet transform (GWT) and gray level co-occurrence matrix (GLCM). The mobile client captures and pre-processes the leaf image, segments diseased patches in it, and transmits it to the Pathology Server. The Server performs computational tasks such as GWT–GLCM feature extraction and k-Nearest Neighbor classification, and the result is sent back to the user's screen via an SMS. This model also focuses on design of a Human-mobile interface (HMI) which is useful even for the illiterate farmers. Android is currently used to run this system, which can be easily extended to other mobile operating systems.

Muhammad E. H. Chowdhury., et al [4] Early detection of plant diseases using computer vision and artificial intelligence (AI) has been proposed as a technique for mitigating disease effects and overcoming the shortcomings of continuous human monitoring. To classify tomato diseases, they proposed using a deep learning architecture based on a recent convolutional neural network called EfficientNet on 18,161 plain and segmented tomato leaf images. For the segmentation of leaves, the performance of two segmentation models, U-net and Modified U-net, is reported. The models' comparative performance for binary classification (healthy and unhealthy leaves), six-class classification (healthy and various groups of diseased leaves), and ten-class classification (healthy and various types of unhealthy leaves) is also discussed.

R Meena Prakash., et al [5] Detection of leaf diseases and classification using digital image processing in which image processing techniques are used to detect the plant leaf diseases. The objective of this work is to implement image analysis & classification techniques for detection of leaf diseases and classification. The proposed framework consists of four parts. They are Image preprocessing, Segmentation of the leaf using K-means clustering to determine the diseased areas, feature extraction & Classification of diseases. Texture features are extracted using statistical Gray-Level Co-Occurrence Matrix (GLCM) features and classification is done using Support Vector Machine (SVM).

2.2 LIMITATIONS OF EXISTING SYSTEMS

- It is not possible to accurately identify the symptoms of the disease, especially at an early stage when the symptoms are subtle using remote sensing technology.
- Using UAV technology is not accessible or affordable for small scale farmers or in areas with limited infrastructure.
- The above systems do not take into account other factors that may affect the development or spread of the disease, such as environmental conditions and soil characteristics.
- The cost and accessibility of diagnostic tools or methods may limit the ability of small-scale farmers or those in remote areas to detect the disease.

2.3 PROPOSED SYSTEM

We proposed a machine learning based approach for early detection of yellow leaf disease in which we classified a given input instance based on the decision tree built using the training dataset. We collected soil samples from different areas where the arecanut plants are grown and analyzed the soil samples for pH, Organic Carbon, Micronutrients, N, P, and K. Trained a machine learning model using the extracted features to build the decision tree. Decision Tree is built based on the calculation of information gain of each of the attributes selected. Once the decision tree is built, the new input instance is provided to classify. The suggestions can be provided based on the values of the Nitrogen(N), Phosphorous(P) and Potassium(K) to improve the soil quality.

Chapter-3

SYSTEM REQUIREMENTS SPECIFICATION

CHAPTER 3

SYSTEM REQUIREMENTS SPECIFICATION

A software requirements specification is a detailed description of a software system that will be created. The software requirements specification specifies functional and non-functional requirements, and it may include a set of use cases that describe user interactions that the software must provide. Use cases are also known as functional requirements. In addition to use cases the SRS also contains non-functional requirements. Non-functional requirements are requirements which impose constraints on the design or implementation. For the hardware requirements the SRS specifies the logical characteristics of each interface between the software product and the hardware components. It specifies the hardware requirements like memory restrictions, cache size, the processor, RAM size etc. those are required for the software to run. Software requirements specification is a rigorous assessment of requirements before the more specific system design stages, and its goal is to reduce later redesign.

3.1 OVERALL DESCRIPTION

The System Requirement Specification is a document, which describes completely the external behavior of the software. This section describes the general factors that affect the product and its requirements. The system will be explained in its context to show how the system interacts with other systems and introduce the basic functionalities of it.

3.1.1 PRODUCT PERSPECTIVE

- User can select the different values for the attributes available.
- User can view the result for the provided information.
- User can enter the NPK amount available in the report.
- User can get the suggestions to improve the soil quality.

3.1.2 PRODUCT FUNCTIONS

Our proposed project is machine learning based early detection of yellow leaf disease that intends to provide the prediction output. The user can provide the attribute values and also get the suggestions related to the improvement of soil quality. The system uses optimized algorithm to provide the prediction

3.1.3 ASSUMPTIONS AND DEPENDENCIES

Assumptions

- The obtained soil report has accurate values for each attribute.
- The classification tree obtained using the algorithm provide the accurate results.

Dependencies

- Active internet connection.
- Dependent on ML Model.
- User must have the soil report with the attributes provided in the site.

3.2 SPECIFIC REQUIREMENTS

This section includes the detailed description about the hardware requirements, software requirements, functional requirements and non-functional requirements.

3.2.1 HARDWARE REQUIREMENTS

Hardware requirements refer to the physical parts of a computer and related devices. Internal hardware devices include motherboards, hard drives and RAM External hardware devices include monitors, keyboards, mice, printers, and scanner.

- Standard PC (Min. 500 GB HDD, 8GB RAM, 5 CPU Core)
- Operating System: Microsoft Windows 10 or above

3.2.2 SOFTWARE REQUIREMENTS

Software requirement is a field within Software Engineering that deals with establishing the project. The software requirements of our project are given below:

Jupyter Notebook

Jupyter Notebook is a free and open-source online tool that lets users create and share interactive documents with live code, equations, visualizations, and explanatory prose. It is commonly used for data analysis, prototyping, and data visualization in data science and scientific computing. It is compatible with a number of programming languages, including Python, R, Julia, and others. Cells in the Notebook interface can include code, text, equations, or visualizations. It is easily shared and accessible online using sites like as GitHub, Binder, and Google Colab, allowing users to collaborate on projects and share their discoveries with others. Overall, Jupyter Notebook is a robust and versatile tool that can be used for a wide range of activities, including data analysis and visualization as well as scientific research.

Flask Web Framework

Flask is a popular micro web framework for building web applications in Python. It is lightweight and flexible, with a small footprint and minimal dependencies. It is not tied to any particular database or ORM, and has a built-in development server and a powerful debugger. Flask provides support for a variety of extensions and libraries, such as Flask-SQL Alchemy and Flask-Login, which can be used to add functionality to web applications. Overall, Flask is a popular choice for building web applications in Python due to its simplicity, flexibility, and ease of use.

Visual Studio

Visual Studio is a powerful integrated development environment (IDE) developed by Microsoft that is used to develop a wide range of applications. It provides a rich set of tools and features for developing, debugging, and deploying applications in a variety of programming languages, including C#, C++, Visual Basic, JavaScript, and Python. It also provides support for a wide range of frameworks, libraries, and platforms, as well as productivity features such as IntelliSense and Live Share. Visual Studio is widely used by developers around the world and has a large and active community of users and contributors.

Python

Python is a well-known high-level programming language that is utilized in a variety of applications. Python's syntax is intended to be simple and clear, making it easy to understand and write. It is an interpreted language, which means that the interpreter executes the code directly without the need for compilation. Python has a huge and active developer community that contributes to its libraries and modules, allowing it to be used for a wide range of applications such as web development, scientific computing, data analysis, artificial intelligence, machine learning, and more. Python is well-known for its readability, simplicity, and versatility, making it a popular choice among both novice and professional programmers.

3.2.3 FUNCTIONAL REQUIREMENTS

Functional requirements describe the system function in detail, its input and output Functional requirements may explicitly state what the system should do. These requirements depend on the type of software being developed. It gives the overall details of how the proposed system works.

- Generate the decision tree using the training dataset created using the obtained soil reports.
- Classify the new input data using the created machine learning model.
- Provide suggestions to the user for improving the soil quality.

3.2.4 NON-FUNCTIONAL REQUIREMENTS

Non-functional requirements are the requirements which are not directly concerned with the specific function delivered by the system. They specify the criteria that can be used to judge the operation of a system rather than specific behavior. They may relate to Emergent system properties such as reliability, response time and store occupancy, non-functional requirements anise through the user needs, because of budget constraints, organizational policies and the need for interoperability with other software.

- Graphical User Interface: The system should have an easy-to-use interface that farmers with little technological skills may operate.
- Reliability: The system should be dependable and always available for usage, with little downtime or system faults.
- Accuracy: The system should be highly accurate in detecting the presence of yellow leaf disease following arecanut plant planting, with few false positives and false negatives.
- Ethical Considerations: When using machine learning models and when collecting, using, and sharing data, the system should take ethical issues like bias, fairness, and accountability into account.

Chapter-4

GANTT CHART

CHAPTER 4

GANTT CHART

A Gantt chart is a type of bar chart, developed by Henry Gantt that illustrates a project schedule. Gantt charts illustrate the start and finish of the terminal elements and summary elements of the project. Terminal elements and summary elements comprise the work breakdown structure of the project. The following is the Gantt chart of the project **"MACHINE LEARNING BASED EARLY DETECTION OF YELLOW DISEASE IN ARECANUT TREE USING SOIL SAMPLES"**

Table 4.1 Gantt chart of planning and scheduling of project

Number	Task	Start	End	Duration(days)
1	Synopsis	27-Oct-2022	04-Nov-2022	14
2	Presentation on idea	07-Nov-2022	07-Nov-2022	1
3	Software Requirement Specification	08-Nov-2022	21-Nov-2022	13
4	System Design	22-Nov-2022	11-Dec-2022	20
5	Implementation	12-Dec-2022	10-Apr-2023	86
6	Presentation on work progress	14-Apr-2023	14-Apr-2023	1
7	Testing	15-Apr-2023	24-Apr-2023	10
8	Result and Report	25-Apr-2023	09-May-2023	15

ACTIVITY/ MONTH	OCT	NOV	DEC	JAN	FEB	MAR	APR	MAY
SYNOPSIS								
PRESENTATION ON IDEA								
SRS								
DESIGN								
IMPLEMENTATION								
TESTING								
REPORT								

Chapter-5

SYSTEM DESIGN

CHAPTER 5

SYSTEM DESIGN

5.1 ARCHITECTURAL DIAGRAM

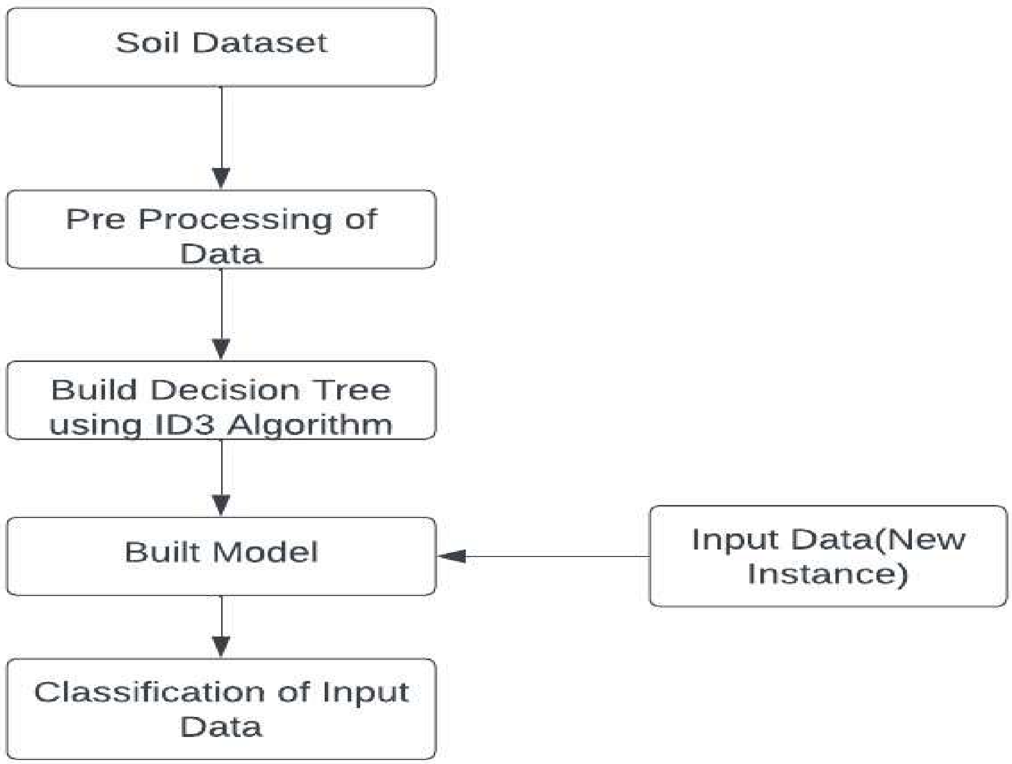

Figure 5.1 Architectural Diagram

The above figure shows the architectural diagram of the project which includes pre-processing of data, building the decision tree using the ID3 algorithm for the classification new instance.

5.2 ACTIVITY DIAGRAM

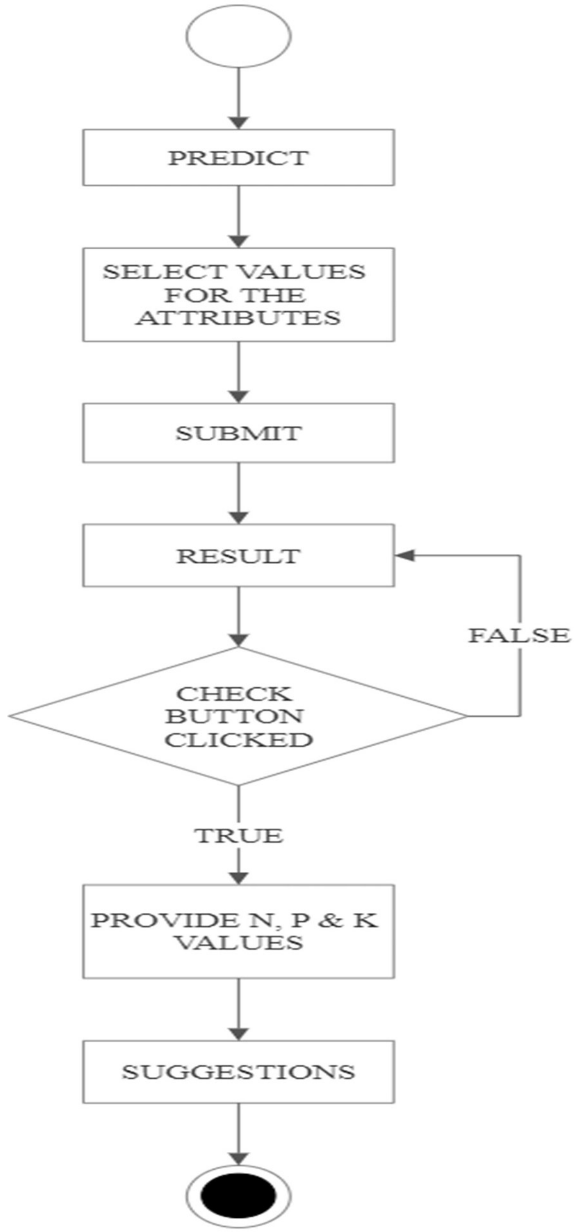

Figure 5.2 Activity Diagram

The activity starts with the Predict where the user selects the values for the attributes that are needed to be analyzed by the model. After selecting the values, the model then generates a result activity information about the occurrence of the disease. The user can also obtain the suggestions by providing the available N, P and K values. Overall, the activity diagram provides a clear visualization of the workflow of the system, which helps the stakeholders to understand and optimize the performance of the system.

5.3 DATA FLOW DIAGRAM

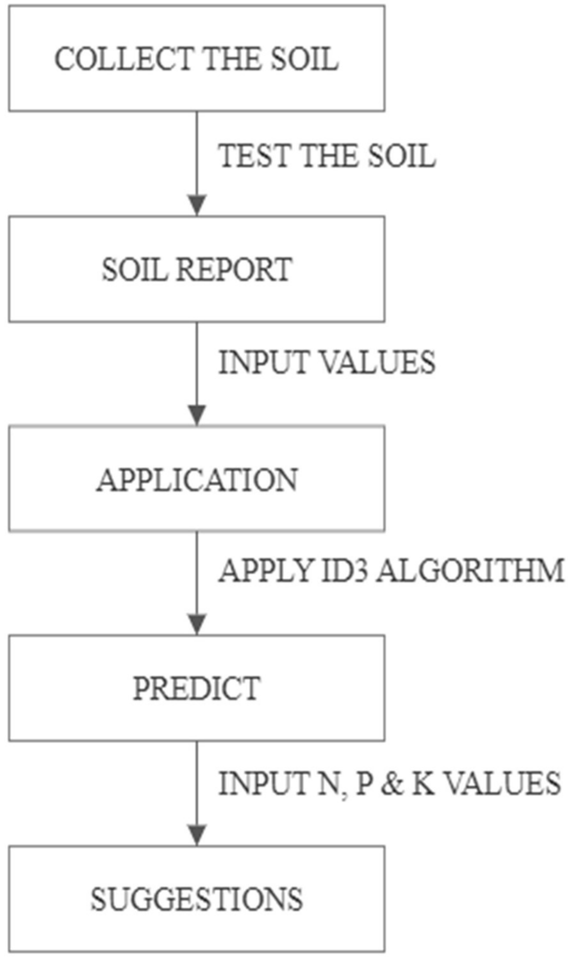

Figure 5.3 Data Flow Diagram

Data-Flow functions or processes that capture, manipulate, store, and distribute data between a system and its environment as well as between system components are represented. The visual representation makes it an effective tool for communication between the user and the system designer.

5.4 SEQUENCE DIAGRAM

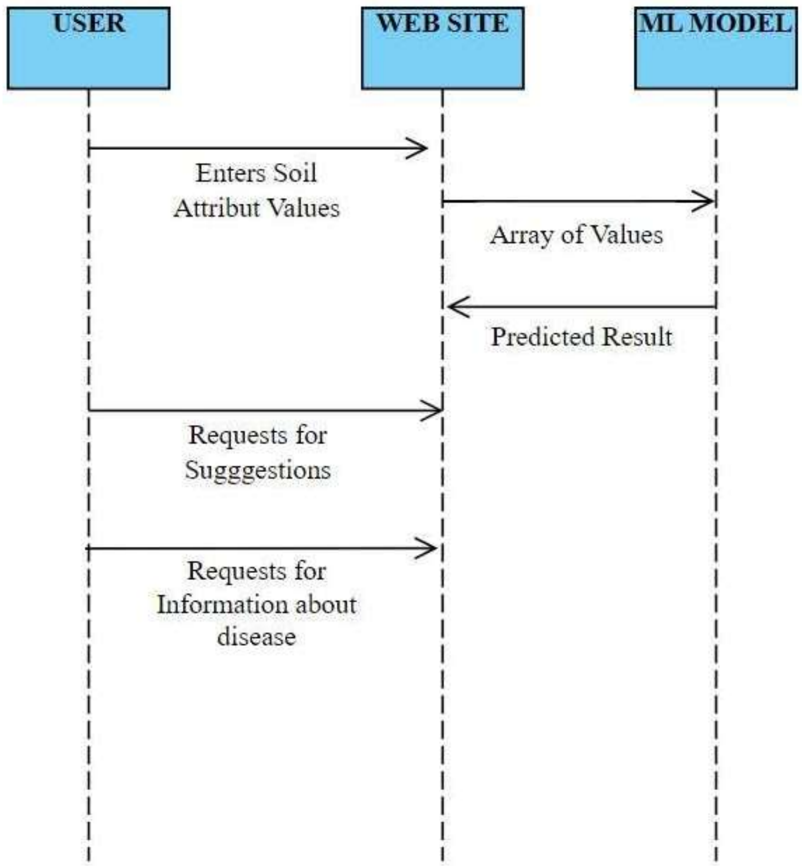

Figure 5.4 Sequence Diagram

A sequence diagram is a type of interaction diagram that visualizes the interactions and messages exchanged between different objects or actors in a system over a specific time period. It illustrates the flow of events and the order in which these events occur, highlighting the communication and behavior between the objects. The user interacts with the website by entering the soil attribute values and the website receives the soil attributes from the user and passes them to the ML model. The user can also request suggestions from the website based on the provided soil attributes. The ML model receives the soil attributes from the website and performs the prediction. The ML model sends the predicted disease result back to the website. The website receives the disease result from the ML model and presents it to the user.

5.5 USE CASE DIAGRAM

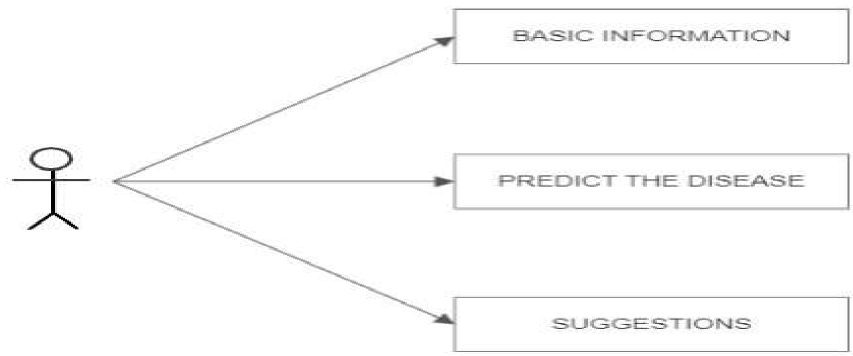

Figure 5.4 Use Case Diagram

First use case involves providing the basic information about the soil sample and the disease to the user. The second use case is predicting the disease, which involves the user submitting the soil sample attributes to the machine learning model for analysis. The model then predicts the occurrence and severity of the yellow leaf disease in the arecanut plant. This use case represents the core functionality of the system, which is to detect the disease at an early stage and provide appropriate suggestions for remedial action. The third use case is suggestions, which involves the system providing suggestions for the improvement of soil quality.

Chapter-6
IMPLEMENTATION

CHAPTER 6

IMPLEMENTATION

6.1 MODULE IMPLEMENTATION

Implementing a machine learning-based system for the early detection of Yellow Leaf Disease in Arecanut plants using soil samples can be done in several steps. Here is an overview of the process:

- Data Collection: Collecting soil samples from Arecanut plantations where yellow leaf disease is present. Also, collecting soil samples from healthy Arecanut plantations for comparison and collecting data from various locations and at different times to ensure that our data set is comprehensive.
- Data Preprocessing: Preprocess the soil sample data, which may involve cleaning and filtering the data, and normalizing the data. This step is crucial in ensuring that the data is in a format that can be easily fed into a machine learning algorithm.
- Feature Extraction: Extracting meaningful features from the soil samples, such as pH level, nitrogen, phosphorus, and potassium levels, micronutrients content, organic matter content, etc. These features should be selected based on their relevance to yellow leaf disease and the availability of the data.
- Model Training: Train a machine learning model on the preprocessed and extracted feature dataset. The model used is a classification algorithm that predicts whether a given soil sample is from a healthy or diseased Arecanut plantation.
- Model Evaluation: Evaluating the performance of the machine learning model using appropriate metrics such as accuracy, precision and F1-score.
- Deployment: Once the model is trained and validated, it is deployed as a software application that takes soil sample data as input and outputs a prediction of whether the sample is from a healthy or diseased Arecanut plantation.
- It is essential to ensure that the data collected is of high quality and that the machine learning model is trained on a representative dataset to ensure that the results are accurate and reliable.

ID3 DECISION TREE BUILDING ALGORITHM

ID3 (Iterative Dichotomiser 3) is a decision tree algorithm used in machine learning for classification problems. The algorithm is used to build a tree-like model that predicts the class of a new instance based on a set of features. The algorithm is based on the concept of entropy and information gain to split the data and construct the decision tree. It is used for both binary and multi-class classification problems. In binary classification, the algorithm predicts the class of a new instance as either 0 or 1. In multi-class classification, the algorithm predicts the class of a new instance as one of several possible values.

To use the ID3 algorithm, the data needs to be in a tabular format where each row represents an instance and each column represents a feature. The target variable or the class variable is also included as a column in the data. The data is split into training and testing sets, and the ID3 algorithm is used to build the decision tree using the training data.

The algorithm continues recursively until all the attributes have been used, or the data has been completely classified. Once the decision tree is constructed, it is used to predict the class of a new instance by traversing the tree from the root node to the leaf node.

In conclusion, the ID3 algorithm is a decision tree algorithm used for classification problems. It works by recursively selecting the attribute that provides the maximum information gain and splitting the data based on the values of that attribute. The resulting decision tree is used to predict the class of a new instance. The ID3 algorithm is widely used in machine learning for classification problems because of its simplicity and speed.

MODEL IMPLEMENTATION CODE-

```
import pandas as pd
from pandas import DataFrame
import csv
from pprint import pprint

def ID (array):
    with open ('data.csv', mode='w', newline='') as file:
        writer = csv.writer(file)
        for row in array:
            writer.writerow(row)
        file.close()
    result=ID_ALGM ()
    return result

def ID_ALGM ():
```

```
df_yld = pd.read_csv("Project_file_train.csv")
att_names = list(df_yld.columns)
att_names.remove('YLD')
def ent_of_list(lst):
    from collections import Counter
    cnt = Counter (x for x in lst)
    num = len(lst)*1
    probs = [x/num for x in cnt.values()]
    return ent(probs)

def ent(probs):
    import math
    return sum ([-prob*math.log (prob, 2) for prob in probs])

def info_gain(df , split , target , trace=0):
    df_split = df.groupby(split)
    nobs = len(df.index)*1
    df_agg_ent = df_split.agg ({target: [ent_of_list, lambda x: len(x)/nobs]})
    df_agg_ent.columns = ["entropy" , "propob"]
    new_ent = sum(df_agg_ent["entropy”] *df_agg_ent["propob"])
    old_ent = ent_of_list(df[target])
    print (split, “IG:” old_ent-new_ent)
    return old_ent-new_ent

def id3(df, target, attr_names , defclass = None):
    from collections import Counter
    cnt = Counter (x for x in df[target])
    if len(cnt)==1:
        return next(iter(cnt))
    elif df.empty or (not attr_names):
        return defclass
    else:
        defclass = max(cnt.keys())
        gain = [info_gain(df , attr , target) for attr in attr_names]
```

```
        index_max = gain.index(max(gain))
        bestat  =  attr_names[index_max]
        tree = {bestat: {}}
        rem = [i for i in attr_names if i! =bestat]

        for atval, subset in df.groupby(bestat):
            subtree = id3(subset, target, rem, defclass)
            tree[bestat][atval] = subtree
        return tree

tree = id3(df_yld, 'YLD', att_names)
pprint(tree)

def predict (tree, instance):
    if not isinstance (tree, dict):
        return tree
    else:
        root_node = next(iter(tree))
        feature_value = instance[root_node]
        if feature_value in tree[root_node]:
            return  predict(tree[root_node]  [feature_value],  instance)
        else:
            return None

def evaluate (tree, test_data_m, label):
    for index, row in test_data_m.iterrows():
        result = predict (tree, test_data_m.iloc[index])
        print ("Predicted value =", result)
        return result
test = pd.read_csv("data.csv")
final_result=evaluate  (tree,  test,'YLD')
return final_result
```

FLASK FRAMEWORK

Flask is based on the WSGI (Web Server Gateway Interface) specification, which defines a standard interface between web servers and web applications. Flask provides a number of features to make it easy to build web applications, including request routing, templates, sessions, and support for various extensions and plugins.

One of the key features of Flask is its routing system, which allows developers to map URLs to Python functions. This makes it easy to create RESTful APIs and to handle HTTP requests in a simple and intuitive way. Flask also provides support for templates, which allows developers to generate HTML pages dynamically based on user input and data from a database or other sources.

FLASK DEPLOYMENT CODE-

```
from flask import Flask, render_template, request
from algorithm import ID
import numpy as np
import csv
app = Flask(__name__)

@app.route('/')
def main():
    return render_template('homepage.html')

@app.route('/homepage', methods=['GET','POST'])
def input_router():
    return render_template('input.html')

@app.route('/', methods=['GET','POST'])
def about_router():
    return render_template('about.html')

@app.route('/input', methods=['GET','POST'])
def home():
    PH = request.form['pH']
```

```
    EC=request.form['ec']
    ORGANIC_CARBON=request.form['organiccarbon']

    NITROGEN=request.form['nitrogen']
    PHOSPHOROUS=request.form['phosphorous']
    POTASSIUM=request.form['potassium']
    ZINC=request.form['zinc']
    BORON=request.form['boron']
    IRONS=request.form['iron']
    MANGANESE=request.form['manganese']
    COPPER=request.form['copper']
    SULPHUR=request.form['sulphur']

array_file=np.array([['ph','ec','OC','N','P','K','Zn','B','Fe','Mn','Cu','S'],[PH,EC,ORGANIC_CA
RBON,NITROGEN,PHOSPHOROUS,POTASSIUM,ZINC,BORON,IRONS,MANGANESE,
COPPER,SULPHUR]])
    output=ID(array_file)
    print(output)
    return  render_template('output.html',output=output)

@app.route('/output',   methods=['GET','POST'])
def suggestion_router():
        n=0
        p=0
        k=0
        return   render_template('npk_checker.html',nitrogen=n,phosphorous=p,potassium=k)

@app.route('/npk_checker', methods=['GET','POST'])
def suggestion_router_final():
    if request.form['Nitrogen']!='':
        if  request.form['Phosphorus'] !='':
            if request.form['Potassium']  !='':
                n=int(request.form['Nitrogen'])
                p=int(request.form['Phosphorus'])
                k=int(request.form['Potassium'])
```

```
            return render_template('npk_checker.html',nitrogen=n, phosphorous=p,potassium=k)
    return render_template('output.html')

if __name___== "_main_":
    app.run(debug=True)
```

Chapter-7

TESTING

CHAPTER 7

TESTING

Testing is the process of evaluating a software system or application to identify defects, errors, or other issues that may affect its quality or performance. Testing is an essential part of the software development lifecycle and helps ensure that the software meets the specified requirements, is reliable, and functions as expected.

There are several types of testing that can be performed during the software development process, including: Unit Testing, Integration Testing, System Testing and Acceptance Testing.

Testing can be performed manually or with the help of automated testing tools and frameworks. The choice of testing methods and tools depends on the specific requirements of the project and the resources available. Overall, testing is an essential part of software development, helping to ensure that software systems are of high quality and meet the needs of their users.

7.1 TESTING LEVELS

Testing is part of Verification and Validation. Testing plays a very critical role for quality assurance and for ensuring the reliability of the software.

The objective of testing can be stated in the following ways:

- A successful test is one that uncovers as-yet-undiscovered bugs.
- A better test case has a high probability of finding unnoticed bugs.
- A pessimistic approach of running the software with the intent of finding errors.

Testing can be performed in various levels like unit test, integration test and system test.

7.1.1 UNIT TESTING

Unit testing is a type of software testing that involves testing individual units or components of a system in isolation from the rest of the system. The purpose of unit testing is to verify that each unit of code performs as expected and meets the specified requirements. Unit testing provides several benefits, including catching bugs early in the development cycle, improving code quality, facilitating code refactoring, and reducing the overall cost of testing. Initially the user provides with the attributes value to the system. If the user fails to provide with the proper input value, the prediction goes wrong which provides with the wrong output. Performance of the system can be checked by providing with different input values and the time taken to predict the output.

7.1.2 INTEGRATION TESTING

Integration testing is a type of software testing that focuses on testing the integration between different components or modules of a software system. The purpose of integration testing is to ensure that the various components of the system work together as intended and that any issues related to the interactions between components are identified and addressed.

Integration testing in our proposed system involves testing the integration between the user interface and the other components of the system to ensure that data is correctly displayed and user interactions are correctly processed.

7.1.3 SYSTEM TESTING

System testing is a type of software testing that is performed on a complete and integrated system to ensure that it meets the specified requirements and works as expected. This type of testing is typically performed after the completion of the development and integration phases, and before the system is released to production. The purpose of system testing is to identify defects, bugs, and other issues that may affect the performance and functionality of the system. It is an essential part of the software development life cycle and helps ensure that the system is of high quality and ready for deployment.

In this project it is ensured that all the integrated workflows are running without any errors.

7.1.4 ACCEPTANCE TESTING

Acceptance Testing is a method of software testing where a system is tested for acceptability. The major aim of this test is to evaluate the compliance of the system with the business requirements and assess whether it is acceptable for delivery or not. After testing the system at different levels, how the complete system is accepted which meets all the mentioned functional and non-functional requirements.

In the case of this system, acceptance testing involves testing the system with real-world soil sample data and gathering feedback from end-users, such as farmers or agricultural experts, to ensure that the system accurately detects yellow leaf disease and provides useful information for disease management.

7.2 TEST CASES

Test cases are specific scenarios or conditions that are designed to test a particular aspect of a software application or system. Test cases are used in software testing to verify that the system or application works as intended and to identify any defects or issues.

To test the system's ability to accurately detect yellow leaf disease using machine learning, a set of soil sample data with and without yellow leaf disease is inputted so that the system could accurately detect the presence or absence of yellow leaf disease. All the tests need to get a Yes or No result which is specified in the test data sample sets taken from the soil testing centers which was given for testing.

Chapter-8

RESULTS AND SNAPSHOTS

CHAPTER 8

RESULTS AND SNAPSHOTS

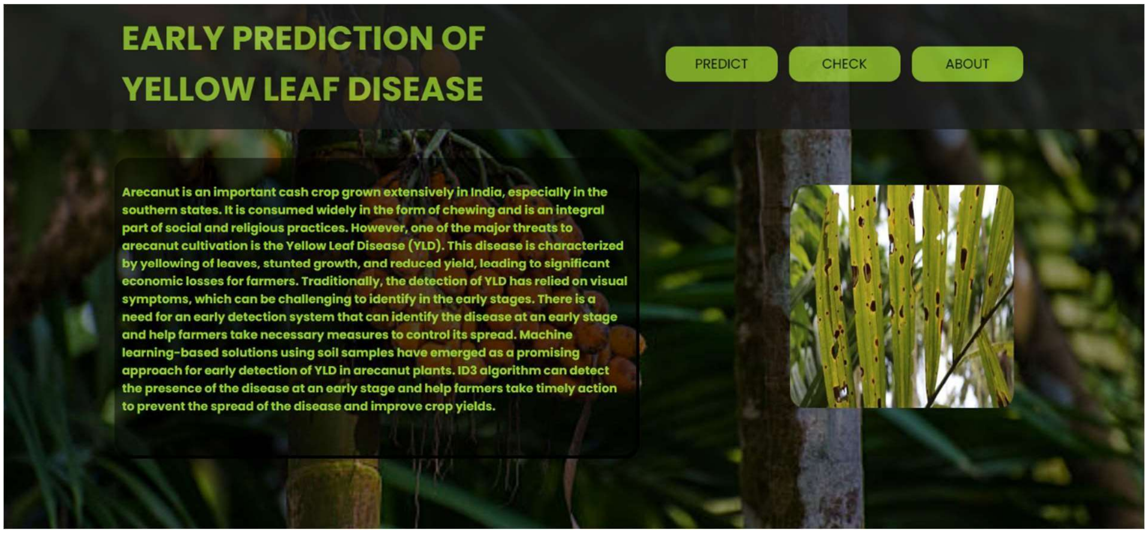

Fig 8.1 Home Page

Home Page consists of details of the system along with the options which includes Predict, Check, About.

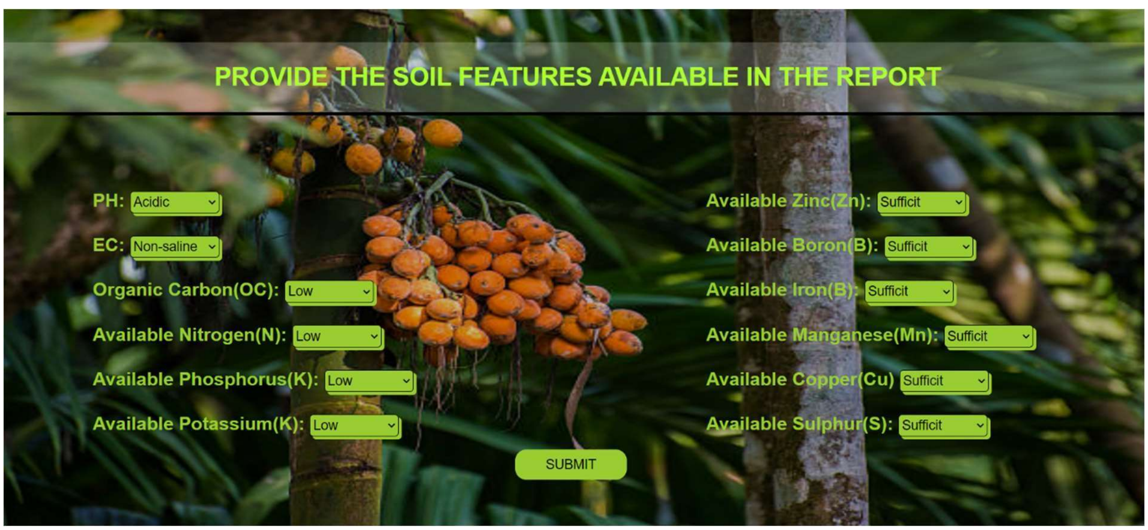

Fig 8.2 Input Page

Once the Predict option is selected, the Input page asks user to select the soil attributes values. Each soil attributes values needs to be selected using drop down option.

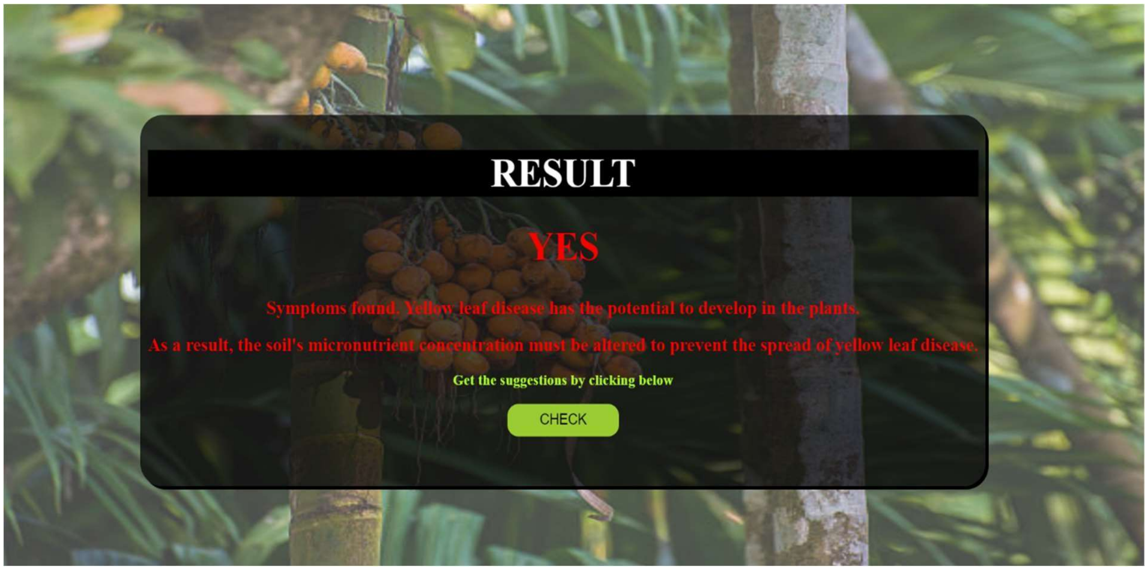

Fig 8.3 Output page

Output page consists of the obtained result along with the check option to get the suggestion from the system.

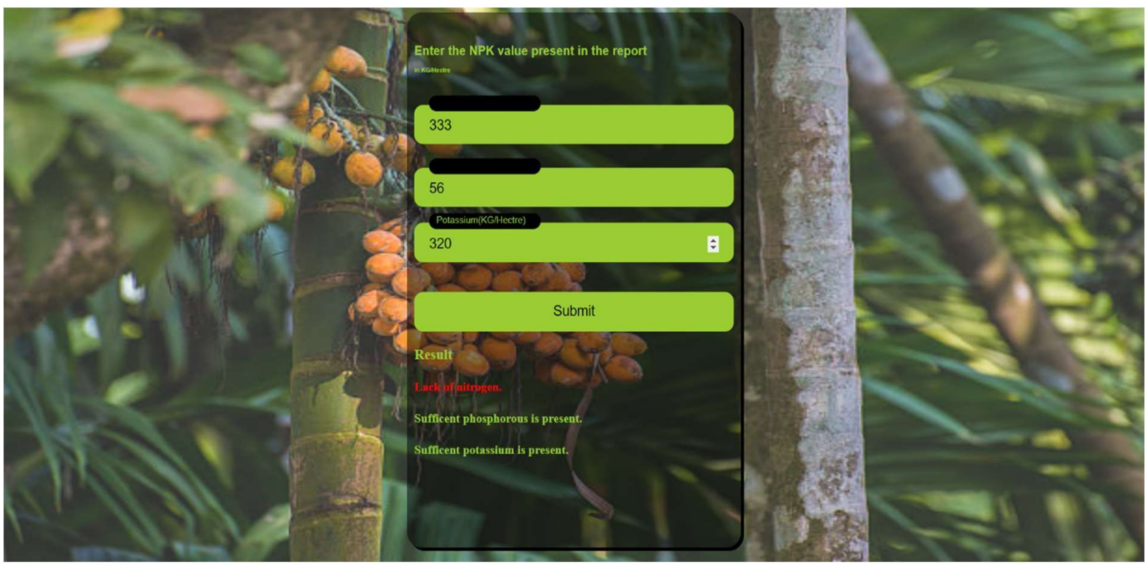

Fig 8.4 Suggestion page

Suggestion page takes the values of N, P and P in Kg/Hectre to provide with the suggestion regarding each of the input.

```
{'Fe': {'deficient': 'no',
        'sufficient': {'B': {'deficient': {'OC': {'high': 'yes',
                                                  'low': 'yes',
                                                  'medium': 'no'}},
                             'sufficient': 'yes'}}}}
Predicted value = yes
yes
```

Fig 8.5 Decision Tree

Visualization of the Decision tree generated using the ID3 Algorithm.

Chapter-9

CONCLUSION AND FUTURE WORK

CHAPTER 9

CONCLUSION AND FUTURE WORK

9.1 CONCLUSION

The project highlights the potential in providing a solution for the early detection of Yellow Leaf Disease in Arecanut plants. The project involves several steps, including the collection of soil samples from various sources, categorizing the samples based on several factors such as pH, moisture, and nutrients, and preprocessing the data. The ID3 algorithm is then used to train a model to predict the occurrence of Yellow Leaf Disease in Arecanut plants using soil samples. The algorithm's advantage is that it can handle both categorical and continuous data and is a powerful tool for decision making problems. The model’s performance is evaluated using a test dataset, and the parameters are optimized to achieve the best possible accuracy. ID3 algorithm has a fast training and prediction time, making it suitable for large datasets and real-time applications such as detection of yellow leaf disease in arecanut plant. It uses a top-down approach to decision-making, which is based on selecting the best attribute to split the data. Overall, the ID3 algorithm is a useful tool for decision-making problems, as it allows us to sort data into categories, making it easier to make predictions and decisions based on the available data. The project’s potential impact is significant, as early detection and prevention of Yellow Leaf Disease can result in better crop yields and improved economic outcomes for farmers. The project can be deployed as an early warning system for the disease, allowing farmers to take action to mitigate its effects. In summary, the project has the potential to reduce the impact of Yellow Leaf Disease on Arecanut plants, improve productivity, and provide economic benefits to farmers.

9.2 FUTURE WORK

We have currently included the suggestions for the values of N, P and K. This can be extended further to provide with the quantity of urea that needs to added and also include suggestions on improving the micronutrients contents. The values of each micronutrient are taken as input from the user and can be provided with suggestions to increase each of the micronutrient’s content in the soil.

The project could be integrated with precision agriculture techniques to optimize resource utilization, increase yield, and reduce costs. For instance, the model's predictions could be used to determine where to apply fertilizers, pesticides, or other inputs, leading to more efficient and sustainable farming practices. The model could be adapted to detect diseases in other crops, such as coffee, tea, or cocoa, where early detection could have significant economic and social impact.

REFERENCES

REFERENCES

[1] Kanan, L. V et al., (2021). Arecanut Yield Disease Forecast using IoT and Machine Learning. International Journal of Scientific Research in Engineering & Technology, 2(2), 11-15

[2] Anilkumar M G., Karibasaveshwara TG., Pavan HK., Sainath Urankar., Dr. AbhayDeshpande. (2021). Detection of Diseases in Arecanut Using Convolutional Neural Networks.

[3] Huang, K. Y. (2012). Detection and classification of areca nuts diseases with machine vision.Computers & Mathematics with Applications, 64(5), 739-746.

[4] L.Sathish Kumar, Mrs.A.Padmapriya.M.C ID3 Algorithm Performance of Diagnosis for Common Disease.

[5] Mallikarjuna, S B, Palaiahnakote Shivakumara, Vijeta Khare, Vinay Kumar N, Basavanna M, Umapada Pal, and Poornima B. (2021). CNN based method for multi-type diseased arecanut image classification. Malaysian Journal of Computer Science, 34(3), 255-265.

[6] Jiawei Guo., Yu Jin., Huichun Ye., Wenjiang Huang., Jinling Zhao., Bei Cui., Fucheng Liu., Jiajian Deng. Recognition of Areca Leaf Yellow Disease Based on Planet Scope Satellite Imagery.

[7] Sunil S. Harakannanavar, Jayashri M. Rudagi, Veena I Puranikmath, Ayesha Siddiqua, R Pramodhini. Plant leaf disease detection using computer vision and machine learning algorithms.

PUBLISHED PAPER AND CERTIFICATES

IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

MACHINE LEARNING BASED EARLY DETECTION OF YELLOW LEAF DISEASE IN ARECANUT PLANT USING SOIL SAMPLES

[1]Ms. Shwetha Kamath, [2]Prakhyath Devadiga, [3]Prathiksha G K, [4]Prathiksha S, [5]Shrinidhi Shervegar

[1]Assitant Professor, Mangalore Institute of Technology and Engineering, Moodabidre, Karnataka, India
[2] Student, Mangalore Institute of Technology and Engineering, Moodabidre, Karnataka, India
[3]Student, Mangalore Institute of Technology and Engineering, Moodabidre, Karnataka, India
[4]Student, Mangalore Institute of Technology and Engineering, Moodabidre, Karnataka, India
[5]Student, Mangalore Institute of Technology and Engineering, Moodabidre, Karnataka, India

Abstract: Arecanut is an important cash crop grown extensively in India, especially in the southern states. It is also known as betel nut and is a major source of income for many farmers in the region. Arecanut is consumed widely in the form of chewing and is an integral part of social and religious practices in India. However, one of the major threats to arecanut cultivation is the Yellow Leaf Disease (YLD). The disease is characterized by yellowing of leaves, stunted growth, and reduced yield, leading to significant economic losses for farmers. The disease is also highly contagious and can spread rapidly, making early detection crucial for effective mitigation measures. Traditionally, the detection of YLD has relied on visual symptoms, which can be challenging to identify in the early stages. This delay in detection can result in significant crop losses. Therefore, there is a need for an early detection system that can identify the disease at an early stage and help farmers take necessary measures to control its spread. In this context, machine learning-based solutions using soil samples have emerged as a promising approach for early detection of YLD in arecanut plants. By analyzing various soil parameters, ID3 algorithm can detect the presence of the disease at an early stage. This can help farmers take timely action to prevent the spread of the disease and improve crop yields.

Keywords - Yellow Leaf Disease, Arecanut, Machine Learning, Soil parameters, ID3 Algorithm.

INTRODUCTION

Arecanut is a significant cash crop that is extensively cultivated in several parts of the world, especially in India, Indonesia, and the Philippines. However, Arecanut plantations are threatened by several fungal diseases, including the Yellow Leaf Disease (YLD). YLD is a devastating fungal disease that affects the Arecanut plant's growth and productivity, leading to significant economic losses for the farmers.

The early detection of YLD is crucial to control the disease's spread and prevent economic losses. Traditional methods of YLD detection rely on visual inspection, which is time-consuming and not always reliable. Therefore, there is a need for a more reliable and cost-effective method for the early detection of YLD.

In this project, we propose a machine learning-based approach for the early detection of YLD in Arecanut plants using soil samples. The proposed method will use soil samples collected from the Arecanut plantations to identify the presence of the fungal pathogen responsible for YLD. Machine learning algorithms and statistical techniques will be used to analyze the soil samples and predict the probability of YLD infection in the Arecanut plants.

The project's main objective is to provide a reliable and cost-effective early detection system for YLD in Arecanut plants, which can be used by farmers to prevent the spread of the disease and improve their yields. The proposed system will be based on the analysis of soil samples, which can provide early indications of the presence of YLD before any visible symptoms appear in the Arecanut plants. Overall, the project aims to contribute to the development of sustainable and efficient agricultural practices by utilizing the power of machine learning and data analysis.

NEED OF THE STUDY

Yellow Leaf Disease (YLD) is a common and devastating fungal disease affecting the Arecanut plant, which is a major cash crop in many parts of the world. The disease causes yellowing of leaves, stunting of growth, and a decrease in the yield of the plant. Early detection of the disease is crucial to prevent the spread of the disease and minimize the economic losses incurred by the farmers.

The project aims to develop a machine learning-based model for the early detection of Yellow Leaf Disease in Arecanut plants using soil samples. The proposed model will use soil samples collected from the affected plants to identify the presence of the fungal pathogen responsible for the disease. The model will use various machine learning algorithms and statistical techniques to analyze the soil samples and predict the probability of YLD infection in the Arecanut plants.

The project will involve the collection of soil samples from Arecanut plantations and the extraction of relevant features from the soil samples. These features will include various physical and chemical parameters of the soil, such as pH, moisture content, and nutrient levels, as well as the presence of specific fungal pathogens. The extracted features will be used to train the machine learning model, which will then be used to predict the likelihood of YLD infection in the Arecanut plants based on the soil samples.

The ultimate goal of the project is to provide a reliable and cost-effective early detection system for YLD in Arecanut plants that can be used by farmers to prevent the spread of the disease and improve their yields.

LITERATURE REVIEW

1.Kanan, L. V et al., (2021). Arecanut Yield Disease Forecast using IoT and Machine Learning. International Journal of Scientific Research in Engineering & Technology, 2(2), 11-15.

This paper presents a system to predict if an arecanut plant is susceptible to disease infection by comparing data values from IoT sensors and historical research. A range is determined for scoring results and a value greater than 8 indicates a high possibility of disease infection. The output is manually confirmed with experience, helping to evaluate the model efficiently. The farmers are warned about the possibility of disease and can take necessary precautions. The model can be extended to multiple crops and diseases based on thorough research.

2.Anilkumar M G., Karibasaveshwara TG., Pavan HK., Sainath Urankar., Dr. Abhay Deshpande. (2021). Detection of Diseases in Arecanut Using Convolutional Neural Networks. International Research Journal of Engineering and Technology (IRJET), 8(5), 4282-4286.
They used Convolutional Neural Network (CNN) a Deep Learning used to diagnose arecanut illnesses. It takes an image as input, assigns learnable weights and biases to distinct items in the image, and then learns to distinguish one from the other based on the results. To train and evaluate the CNN model, they built a dataset of 620 photos of healthy and ill arecanuts. The train and test data are split in an 80:20 split. For model compilation, categoricalcross-entropy is utilized as the loss function, with Adam as the optimizer function and accuracy as the metrics. A total of 50 Epochs are used to train the model to achieve high validation and test accuracy with minimum loss.

3.Huang, K. Y. (2012). Detection and classification of areca nuts with machine vision. Computers & Mathematics with Applications, 64(5), 739-746.

In this initially the grade of arecanuts was found and determined. To categorize the arecanut diseases, a detection line (DL) approach was employed. Six geometrical features, three-color features, and the fault area were used in the categorization technique.

4. L.Sathish Kumar., Mrs.A.Padmapriya.M.C. ID3 Algorithm Performance of Diagnosis for Common Disease.

This paper discusses the problem of constraining and summarizing two algorithms of data mining used in medical prediction. It focuses on using Decision Tree and Bayesian classification for intelligent and effective common disease diagnosis. The outcome of diagnosis data mining technique on the same dataset reveals that Decision Tree outperforms and Bayesian classification is having similar accuracy as of ID3. This paper proposes a procedure for retrieval of dataset with relevant fields using ID3 algorithm, based on individual diagnosis for specific symptoms of the disease.

5. Mallikarjuna, S B, Palaiahnakote Shivakumara, Vijeta Khare, Vinay Kumar N, Basavanna M, Umapada Pal, and Poornima B. (2021). CNN based method for multi-type diseased arecanut image classification. Malaysian Journal of Computer Science, 34(3), 255-265.

This work proposes a new method for classification of arecanut images of different diseases. It exploits multi-gradient concept to generate four gradient images, which enhance fine details irrespective of disease effect. The proposed method outperforms existing methods in terms of classification rate, recall, precision and F-measure. The future target is to extend the idea for more classes and identify disease by analyzing the content of the images.

6.Jiawei Guo., Yu Jin., Huichun Ye., Wenjiang Huang., Jinling Zhao., Bei Cui., Fucheng Liu., Jiajian Deng. Recognition of Areca Leaf Yellow Disease Based on Planet Scope Satellite Imagery

This article compares field crops like wheat to areca yellow leaf disease remote sensing surveillance. The RF model, which outperformed the BPNN and AdaBoost models by 2.95% and 20.59%, had the greatest overall recognition accuracy for areca yellow leaf disease, according to the results (88.24%). The geographic distribution map's improved recognition accuracy made it more useful for areca yellow leaf disease detection and monitoring.

METHODOLOGY

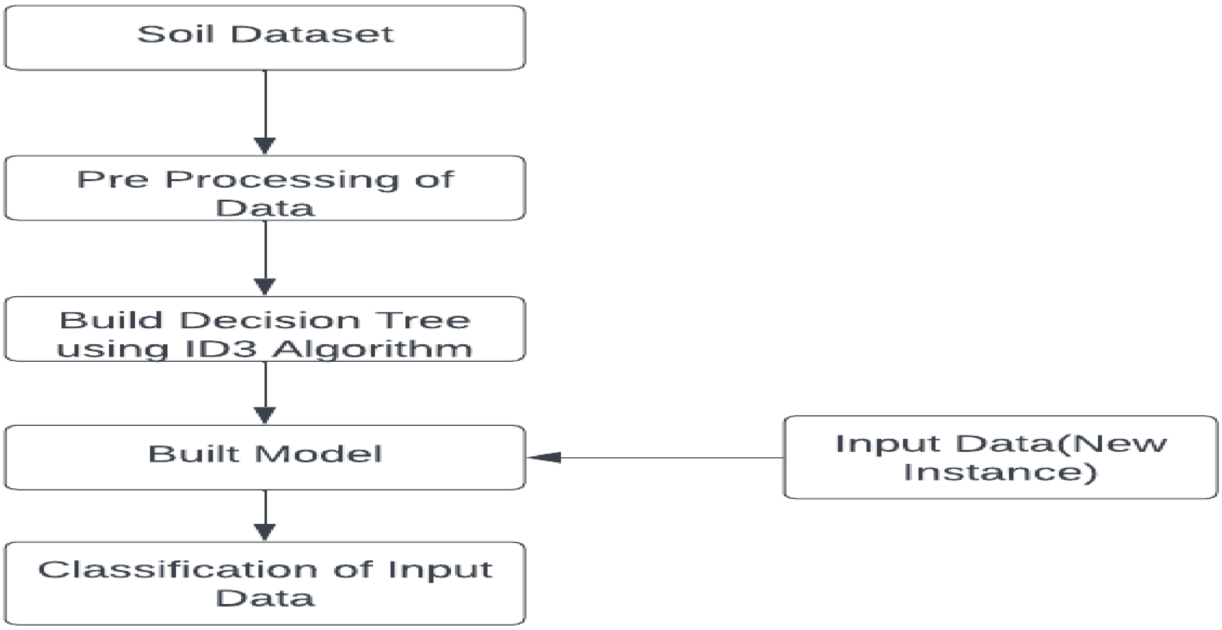

Fig 1. Methodology of early prediction of yellow leaf disease

The soil dataset would typically consist of various attributes or features related to soil characteristics. These features can be used as input variables to train a machine learning model to predict the occurrence of YLD in Arecanut plants.

Soil dataset features that may be relevant for detecting YLD:

- pH level: Soil pH is a measure of its acidity or alkalinity, and it can influence the availability of nutrients in the soil. Some pathogens, including the one that causes YLD, may prefer certain pH levels for optimal growth. Therefore, pH level could be an important feature in the dataset.

- Micronutrients content: Micronutrients are essential elements required by plants in small quantities for their growth and development. Imbalanced or deficient micronutrient levels in the soil can affect plant health and make them more susceptible to diseases, including YLD.

- Nutrient levels: Soil nutrient levels, such as nitrogen, phosphorus, and potassium, play a critical role in plant growth and development. Imbalanced nutrient levels can weaken plant resistance to diseases, including YLD.

- Organic matter content: The organic matter content of the soil is an important indicator of soil health. Soil with higher organic matter content is generally more fertile and supports healthier plant growth. Organic matter content can influence the availability of nutrients and microbial activity in the soil, which can impact the occurrence of YLD.

The specific features and their importance may vary depending on the context, location, and factors affecting the occurrence of YLD in the target area.

Preprocessing of data involves converting numerical values to the discrete values to feed to csv file. Discrete feature values play a crucial role in feature selection, tree construction, tree pruning, and classification in the ID3 algorithm, which is a widely used decision tree learning algorithm for machine learning tasks.

The ID3 (Iterative Dichotomiser 3) algorithm is a popular decision tree algorithm used for building decision trees for classification tasks. Here's how it works:

- Selecting the Root Node: The ID3 algorithm starts by selecting the best feature to use as the root node of the decision tree. The feature selection is based on the concept of information gain, which measures the reduction in entropy (a measure of impurity) of the dataset after splitting on a particular feature. The feature with the highest information gain is selected as the root node.

- Splitting the Dataset: Once the root node is selected, the dataset is split into subsets based on the values of the selected feature. Each subset contains samples with the same value for the selected feature.

- Repeating the Process: The ID3 algorithm recursively repeats the above steps for each subset created in the previous step. The algorithm continues to split the dataset based on the remaining features with the highest information gain until one of the following conditions is met:

1.All samples in the subset have the same class label, resulting in a leaf node with that label.
2.There are no more features left to split on, resulting in a leaf node with the majority class label of the remaining samples.

- Pruning: Once the decision tree is fully constructed, it may be pruned to prevent overfitting. This can be done using various techniques such as pre-pruning (limiting the depth of the tree) or post-pruning (pruning unnecessary branches after tree construction).

- Prediction: To make predictions with the decision tree, new samples are recursively traversed down the tree based on the feature values, and the majority class label of the samples in the leaf node reached is used as the predicted class label for the input sample.

Once the decision tree is built the input instances can be provided to model which is being built to predict the presence of yellow leaf disease in arecanut.

RESULTS AND DISCUSSION

Fig 2. Home Page

The input page contains the input instances that needs to be considered to predict the yellow leaf disease. The attributes include Micronutrients along with pH, N, P, K values. Once the user input is given the result page provides with the result of the prediction. The result may be “Yes” or “No” based on the ID3 algorithm which builds decision tree and the input provided.

Fig 3. Input Page

Fig 4. Result Page

Fig 5. Result Page

CONCLUSION

The project highlights the potential in providing a solution for the early detection of Yellow Leaf Disease in Arecanut plants. The project involves several steps, including the collection of soil samples from various sources, categorizing the samples based on several factors such as pH, moisture, and nutrients, and preprocessing the data.

The ID3 algorithm is then used to train a model to predict the occurrence of Yellow Leaf Disease in Arecanut plants using soil samples. The algorithm's advantage is that it can handle both categorical and continuous data and is a powerful tool for decision-making problems. The model's performance is evaluated using a test dataset, and the parameters are optimized to achieve the best possible accuracy. ID3 algorithm has a fast training and prediction time, making it suitable for large datasets and real-time applications such as detection of yellow leaf disease in arecanut plant. It uses a top-down approach to decision-making, which is based on selecting the best attribute to split the data.

The project's potential impact is significant, as early detection and prevention of Yellow Leaf Disease can result in better crop yields and improved economic outcomes for farmers. The project can be deployed as an early warning system for the disease, allowing farmers to take action to mitigate its effects.

In summary, the project has the potential to reduce the impact of Yellow Leaf Disease on Arecanut plants, improve productivity, and provide economic benefits to farmers.

REFERENCES

[1] Kanan, L. V et al., (2021). Arecanut Yield Disease Forecast using IoT and Machine Learning. International Journal of Scientific Research in Engineering & Technology, 2(2), 11-15

[2] Anilkumar M G., Karibasaveshwara TG., Pavan HK., Sainath Urankar., Dr. AbhayDeshpande. (2021). Detection of Diseases in Arecanut Using Convolutional Neural Networks.

[3] Huang, K. Y. (2012). Detection and classification of areca nuts diseases with machine vision.Computers & Mathematics with Applications, 64(5), 739-746.

[4] L.Sathish Kumar, Mrs.A.Padmapriya.M.C ID3 Algorithm Performance of Diagnosis for Common Disease.

[5] Mallikarjuna, S B, Palaiahnakote Shivakumara, Vijeta Khare, Vinay Kumar N, Basavanna M, Umapada Pal, and Poornima B. (2021). CNN based method for multi-type diseased arecanut image classification. Malaysian Journal of Computer Science, 34(3), 255-265.

[6] Jiawei Guo., Yu Jin., Huichun Ye., Wenjiang Huang., Jinling Zhao., Bei Cui., Fucheng Liu., Jiajian Deng. Recognition of Areca Leaf Yellow Disease Based on Planet Scope Satellite Imagery

INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG

An International Open Access, Peer-reviewed, Refereed Journal

ISSN : 2456-4184

Certificate of Publication

The Board of

International Journal of Novel Research and Development

Is hereby awarding this certificate to

Prakhyath Devadiga

In recognition of the publication of the paper entitled

Machine Learning based early detection of Yellow leaf disease in arecanut plant using soil samples

Published In IJNRD (www.ijnrd.org) ISSN Approved & 8.76 Impact Factor

Published in Volume 8 Issue 4, April-2023 | Date of Publication: 2023-04-25

Co-Authors - Shwetha Kamath,Prathiksha G K,Prathiksha S,Shrinidhi Shervegar

Registration ID : 192227 **Paper ID - IJNRD2304477** **Editor-In Chief**

INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT | IJNRD

An International Scholarly, Open Access, Multi-disciplinary, Indexed Journal

Website: www.ijnrd.org | Email: editor@ijnrd.org | ESTD: 2016

An International Scholarly, Open Access, Multi-disciplinary, Monthly, Indexing in all Major Database & Metadata, Citation Generator

Manage By: IJPUBLICATION Website: www.ijnrd.org | Email ID: editor@ijnrd.org

IJNRD | ISSN : 2456-4184

Certificate of Publication

IJNRD | ISSN : 2456-4184

International Research Journal
IJNRD
Research Through Innovation

INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal
ISSN : 2456-4184

The Board of

International Journal of Novel Research and Development

Is hereby awarding this certificate to

Prathiksha G K

In recognition of the publication of the paper entitled

Machine Learning based early detection of Yellow leaf disease in arecanut plant using soil samples

Published In IJNRD (www.ijnrd.org) ISSN Approved & 8.76 Impact Factor

Published in Volume 8 Issue 4, April-2023 | Date of Publication: 2023-04-25

International Journal of Novel Research and Development
ISSN 2456-4184
IJNRD

Co-Authors - Shwetha Kamath,Prakhyath Devadiga,Prathiksha S,Shrinidhi Shervegar

Registration ID : 192227 **Paper ID - IJNRD2304477** **Editor-In Chief**

INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT | IJNRD
An International Scholarly, Open Access, Multi-disciplinary, Indexed Journal
Website: www.ijnrd.org | Email: editor@ijnrd.org | ESTD: 2016

An International Scholarly, Open Access, Multi-disciplinary, Monthly, Indexing in all Major Database & Metadata, Citation Generator
Manage By: IJPUBLICATION Website: www.ijnrd.org | Email ID: editor@ijnrd.org

INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal
ISSN : 2456-4184

International Research Journal
IJNRD
Research Through Innovation

Certificate of Publication

The Board of

International Journal of Novel Research and Development

Is hereby awarding this certificate to

Prathiksha S

In recognition of the publication of the paper entitled

Machine Learning based early detection of Yellow leaf disease in arecanut plant using soil samples

Published In IJNRD (www.ijnrd.org) ISSN Approved & 8.76 Impact Factor

Published in Volume 8 Issue 4, April-2023 | Date of Publication: 2023-04-25

International Journal of Novel Research and Development
ISSN 2456-4184
IJNRD

Co-Authors - Shwetha Kamath,Prakhyath Devadiga,Prathiksha G K,Shrinidhi Shervegar

Registration ID : 192227 **Paper ID - IJNRD2304477** **Editor-In Chief**

IJNRD | ISSN : 2456-4184

INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT | IJNRD
An International Scholarly, Open Access, Multi-disciplinary, Indexed Journal
Website: www.ijnrd.org | Email: editor@ijnrd.org | ESTD: 2016

An International Scholarly, Open Access, Multi-disciplinary, Monthly, Indexing in all Major Database & Metadata, Citation Generator
Manage By: IJPUBLICATION Website: www.ijnrd.org | Email ID: editor@ijnrd.org

INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal
ISSN : 2456-4184

The Board of

International Journal of Novel Research and Development

Is hereby awarding this certificate to

Shrinidhi Shervegar

In recognition of the publication of the paper entitled

Machine Learning based early detection of Yellow leaf disease in arecanut plant using soil samples

Published In IJNRD (www.ijnrd.org) ISSN Approved & 8.76 Impact Factor

Published in Volume 8 Issue 4, April-2023 | Date of Publication: 2023-04-25

Co-Authors - Shwetha Kamath,Prakhyath Devadiga,Prathiksha G K,Prathiksha S

Registration ID : 192227 **Paper ID - IJNRD2304477** **Editor-In Chief**

INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT | IJNRD
An International Scholarly, Open Access, Multi-disciplinary, Indexed Journal
Website: www.ijnrd.org | Email: editor@ijnrd.org | ESTD: 2016

An International Scholarly, Open Access, Multi-disciplinary, Monthly, Indexing in all Major Database & Metadata, Citation Generator
Manage By: IJPUBLICATION Website: www.ijnrd.org | Email ID: editor@ijnrd.org

Certificate of Publication

IJNRD | ISSN : 2456-4184