CROP SELECTION AND POSSIBLE DISEASES PREDICTION SYSTEM USING DEEP LEARNING

A PROJECT REPORT

Submitted by

MERLIN JONE J (810019205062)

YAHAVARSHINI E (810019205110)

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

INFORMATION TECHNOLOGY

UNIVERSITY COLLEGE OF ENGINEERING, BIT CAMPUS,

TIRUCHIRAPPALLI - 620024

ANNA UNIVERSITY :: CHENNAI - 600 025

MAY 2023

CROP SELECTION AND POSSIBLE DISEASES PREDICTION SYSTEM USING DEEP LEARNING

A PROJECT REPORT

Submitted by

MERLIN JONE J (810019205062)

YAHAVARSHINI E (810019205110)

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

INFORMATION TECHNOLOGY

UNIVERSITY COLLEGE OF ENGINEERING, BIT CAMPUS,

TIRUCHIRAPPALLI - 620024

ANNA UNIVERSITY :: CHENNAI - 600 025

MAY 2023

ANNA UNIVERSITY: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report titled " **CROP SELECTION AND POSSIBLE DISEASES PREDICTION SYSTEM USING DEEP LEARNING**" is a bonafide work of " **MERLIN JONE J (810019205062), YAHAVARSHINI E (810019205110)"** who carried out the project work under my supervision, for the partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Information Technology certified further that to the best of my knowledge and belief, the work reported here in does not form part of any other thesis or dissertation on the basis of which a degree or an award was conferred on an earlier occasion.

SIGNATURE	SIGNATURE
Dr. G. ANNAPOORANI,	**Mr. K. SARAVANA KUMAR,**
Assistant Professor (Sl. Gr),	**SUPERVISOR**
HEAD OF THE DEPARTMENT	**Teaching Fellow**
Department of CSE/IT,	Department of IT,
University College of Engineering,	University College of Engineering,
BIT Campus,	BIT Campus,
Tiruchirappalli – 620024.	Tiruchirappalli – 620024.

Submitted for "IT8811 – Project Work" in B.Tech. Information Technology Degree Jan – May 2023 Examination held on …………….

Internal Examiner **External Examiner**

DECLARATION

We hereby declare that the work entitled **"CROP SELECTION AND POSSIBLE DISEASES PREDICTION SYSTEM USING DEEP LEARNING"** submitted in partial fulfillment of the requirement for the award of the degree in B.Tech., University College of Engineering, BIT Campus, Anna University, Tiruchirappalli. is a record of my work carried out by me during the academic year 2022- 2023 under the supervision of **Mr. K. SARAVANA KUMAR, M.E.,** Teaching Fellow, Department of Information Technology, University College of Engineering, BIT Campus, Anna University, Tiruchirappalli. The extent and source of information are derived from the existing literature and have been indicated through the dissertation at the appropriate places. The matter embodied in this work is original and has not been submitted for the award of any other degree or diploma, either in this or any other university.

(Signature of the Candidate)	(Signature of the Candidate)
MERLIN JONE J (810019205059)	YAHAVARSHINI E (810019205110)

I certify that the declaration made by the above candidate is true.

(Signature of the Guide)
Mr. K. SARAVANA KUMAR, M.E.,
Teaching Fellow , Department of Information Technology,
University College of Engineering, BIT Campus,
Anna University, Tiruchirappalli.

ACKNOWLEDGEMENT

A truthful heartfelt and deserved acknowledgement comes from one's heart to convey the real influence others have on one's work.

We express our gratitude to our honorable Dean**, Dr.T. SENTHILKUMAR, M.E., PhD.,** for giving us chance to complete our education in one of the reputed government institutions running under his leadership.

We express our sincere gratitude to our head of the department **Dr. G. ANNAPOORANI, M.Tech., Ph.D.,** for giving us the provision to do the project.

We are much obliged to our project coordinators **Dr. K. UMA MAHESHWARI, M.Tech., Ph.D., Mr. K. SARAVANA KUMAR**, **M.E.,** and our class coordinator **Dr. V. M. PRIYADHARSHINI, M.Tech., Ph.D.,** for giving us the opportunity to do the project, hearty thanks for them.

We stand even more thankful to our project supervisor **Mr.K. SARAVANA KUMAR, M.E.,** for guiding us throughout and giving us the opportunity to present the main project.

We also express our sincere thanks to all other staff members, friends, and our parents for their help and encouragement.

ABSTRACT

India is an agricultural nation. A significant portion of India's economy is based on agriculture. In India, agriculture takes up as much as 60% of the total land area. This provides food for 1.2 billion people worldwide. The Indian farmer is at the centre of the agricultural industry, but the majority of Indian farmers continue to be at the very bottom of society. In addition, despite the limited technical options available today, farmers still struggle to choose the crop that is most profitable for their land due to the wide variety of soil types, unpredictable climatic conditions found throughout the world. In order to increase agricultural output, which also leads to economic growth, it will be very important to predict the correct crop at the right area at the right time and also predicting possible diseases for that crop, can alert the farmers to check and use the fertilizers periodically. This proposed model suggests a crop and possible diseases that makes use of Deep learning Model - Multi-Layer Perception (MLP) algorithm to analyse a variety of criteria, such as the temperature, soil nutrition, weather, rainfall in order to forecast the ideal crop to be cultivated and possible diseases that can affect that crop during cultivation period. It collects data from various sources such as humidity, temperature, rainfall from weather stations annual report, soil nutrition essential parameters like Nitrogen (N), Potassium (K), Phosphorus (P) and pH value from soil test labs report. This model is beneficial for new comers for farming which may helpful in proper crop selection and knowing possible diseases may alter to check periodically on that crop and take care of them properly which result in high yield and profit. Also, this technique would lessen the financial loss that farmers suffer as a result of planting the unsuitable crop.

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

LIST OF ACRONYMS

1.	UML	–	Unified Modelling Language
2.	MLP	–	Multi-Layer Perception
3.	OS	–	Operating System
4.	ML	–	Machine Learning
5.	AI	–	Artificial Intelligence
6.	UI	–	User Interface
7.	DFD	–	Data Flow Diagram
8.	ANN	–	Artificial Neural Network
9.	ReLU	–	Rectified Linear Unit
10.	TP	–	True Positive
11.	TN	–	True Negative
12.	FP	–	False Positive
13.	FN	–	False Negative
14.	IDE	–	Integrated Development Environment
15.	ER	–	Entity-Relationship
16.	IoT	–	Internet of Things

CHAPTER 1
INTRODUCTION

1.1 Introduction

Before technical and application details, the background understating of problem source is very important. If source the background information is understood well, then easier to follow project description. In this chapter, existing state of society, their statistical information, how that issue will affect our future generation were described.

1.2 Background Understanding

The basic understanding of crops and its essentials parameters are very important to have better solution, let us discuss in detail. Traditional crop selection by farmers involves a combination of observation, trial and error, and local knowledge passed down from generation to generation. Farmers typically select crop varieties that have performed well in their particular climate and soil conditions, and that have desirable traits such as yield, taste, and resistance to pests and diseases. Farmers often identify crop diseases through visual symptoms, such as wilting, yellowing, or lesions on leaves and stems. They may also observe changes in plant growth and fruit quality. Traditional knowledge about crop diseases is often based on observation and experience, as well as on knowledge passed down from other farmers or community members. Farmers may use various methods to control crop diseases, including crop rotation, intercropping, and the use of natural remedies and pesticides. Traditional crop selection and disease management practices can be effective, but they are also limited by the availability of resources and information. In many cases, farmers may not have access to the latest scientific research and technologies, which can lead to suboptimal yields and increased vulnerability to crop diseases. However, by combining traditional knowledge with modern research and technology, farmers can improve their crop selection and disease management practices, leading to more sustainable and resilient agricultural systems.

In recent years, many articles and newspapers publish that crop production was daunting because of crop diseases such as “Sugarcane cultivation has turned bitter for hundreds of farmers in Papanasam in Thanjavur due to Pokkaboing disease”, “Mar infection has been seen in pigeon pea crop in some areas in Hingoli” , “Green gram crop hit by yellow patch disease in Belagavi dist”.There are many news which makes crop production and economy down. But with normal and experienced knowledge farmers couldn’t detect the diseases properly and use fertilizers in right time. Its also difficult foe new famers who has less experience and knowledge. New farmers may not have access to the same level of information and resources as established farmers, which can make it more difficult to stay up-to-date on the latest crop selection and disease detection techniques. They may have limited resources, such as land, equipment, and capital, which can make it more difficult to experiment with different crops and invest in disease detection technologies. Climate change and other environmental factors can make it more challenging to select crops and detect disease. By having a good knowledge over the seeding crops and its common diseases may help them to check periodically.

Crop selection and disease prediction are critical aspects of agriculture. The right crop selection can ensure maximum yield, while early detection and prevention of diseases can save crops from damage and enhance production. These tasks require a comprehensive understanding of various factors, such as soil quality, climate conditions, and disease prevalence, which can affect crop growth and health. Traditionally, farmers rely on their experience and knowledge to make crop selection and disease prevention decisions. However, this approach can be time-consuming and prone to errors, leading to reduced crop yield and losses. With the advancements in technology and data analytics, it is now possible to develop automated systems that can assist farmers in these tasks. One popular approach to crop selection and disease prediction is the use of machine learning algorithms. Machine learning algorithms can analyze large datasets and extract insights to make accurate predictions. One popular machine learning algorithm used for these tasks is Multilayer Perceptron (MLP). MLP is a type of artificial neural network that is commonly used for classification and prediction tasks. It is a supervised learning algorithm that requires a labeled dataset for training. MLP consists of multiple layers of nodes connected by weighted edges. The input layer receives the input data, which is then passed through hidden layers that transform the data to extract relevant features. Finally, the output layer produces the predictions. To develop a crop selection and disease prediction system using MLP, historical data on crop yield, soil quality, climate conditions, and disease prevalence are used as input. The MLP algorithm analyzes the data to determine the optimal crop selection and predict the likelihood of disease outbreaks. The system provides farmers with timely alerts and recommendations to take preventive measures to protect their crops.

1.3 Overview of chapters

Chapter 2: This chapter gives detailed view of existing research works, systems, their methodology, algorithms, source of dataset and result.

Chapter 3: The solution provided for this problem by Proposed system, goal and objective of the Famer's friend System, it's features and algorithms used for development and analysis detailed with their definition.

Chapter 4: Overview of the system configuration needed for execution of Proposed as a web application, software requirements, and frame works, libraries used for application development are explained.

Chapter 5: This chapter gives the detailed view of planning for Proposed System development, module split, overview of the working of Proposed System using System Architecture and technical work flow using Technical Architecture, data flow of the system and use case are described.

Chapter 6: In this chapter, Overview of deployed web application with the advantages, disadvantages and complete detail about results and discussions were described.

Chapter 7: In this chapter, conclusion and future work of Proposed System were described.

1.4 Conclusion

Overall, the use of machine learning algorithms such as MLP can significantly enhance the efficiency and accuracy of crop selection and disease prediction in agriculture. These technologies can help farmers make informed decisions and prevent crop losses, leading to increased yield and profitability.

CHAPTER 2
LITERATURE SURVEY

2.1 Introduction

A literature survey is an essential component of any research project. It involves reviewing and analyzing the existing literature in a particular field of study to identify gaps in knowledge, areas of agreement and disagreement, and potential avenues for further research. The literature survey serves several purposes, including establishing the significance of the research question or problem, providing a foundation of knowledge and understanding of the field, and identifying the research methods and tools used by previous researchers. In following of this chapter, research works which are related to crop selection and possible diseases system using deep learning were discussed. The excellent research works are discussed in depth more than others.

2.2 Related Works

Omkar Kulkarni et al.,[1] crop recommendation system using machine learning algorithm. The data sensed by these sensors is stored on the microcontroller and analyzed using machine learning algorithms like random forest based on which suggestions for the growth of the suitable crop are made. This system would help the farmers in making an informed decision about which crop to grow depending on some parameters like Nitrogen, Phosphorous, Pottasium, PH Value, Humidity, Temperature, and Rainfall. The Analysis has been performed on these six types of machine learning algorithms and out of these six algorithms XGBoost achieved best accuracy result.

Dhanush Vishwakarma et al.,[2] 2020, Crop Prediction using Machine Learning Approaches. This system will suggest the best suitable crop for particular land based on content and weather parameters. And also, the system provides information about the required content and quantity of fertilizers, required seeds for cultivation. This system will also provide information about required nutrients to add up, required seeds for cultivation, expected yield and market price.

Motwani, Aditya et al.,[3] "Soil Analysis and Crop Recommendation using Machine Learning." This paper proposes a crop recommendation system that uses a Convolutional Neural Network (CNN) and a Random Forest Model to predict the optimal crop to be grown by analyzing various parameters including the region, soil type, yield, selling price, etc. The CNN architecture gave an accuracy of 95.21%, and the Random Forest Algorithm had an accuracy of 75%. The system can be further improved by expanding the crop production dataset to get more accurate results for yield prediction.

Archana Gupta,Dharmil et al., [4] ”Smart Crop Prediction using IoT and Machine Learning.” Machine learning in agriculture is used to improve the productivity and quality of the crops in the agriculture sector. Use of appropriate algorithms on the sensed data can help in recommendation of suitable crop. Thus system will be used to reduce the difficulties faced by the farmers and will increase the quantity and quality of work done by them.

Ommane, Y et al.,[5]. Machine Learning based Recommender Systems for Crop Selection: A Systematic Literature Review. This paper draws a systematic literature review about the use of Machine learning based recommender systems for crop selection, with respect to the PRISMA protocol for systematic reviews. The second section, describes an overview of existing recommender systems in literature. The outline of this study is explained, as well as the method of content analysis used in this article to sort out the papers is introduced in the third section. In the fourth section, the selection process and the literature review matrix are provided. The SLR was conducted with the aim of providing insights of the kind of solutions that were proposed in the recent years for the CS task. Such insights are valuable in suggesting new directions for research studies and in providing a good understanding of the recent research trends.

K.D.Yesugade et al., [6] Crop Suggesting System Using Unsupervised Machine Learning Algorithm. The yield of farm extremely depends on the crop selected for cultivation and environmental parameters therefore correct selection of crop before cultivation is vital in farming. This system will facilitate farmers in deciding the proper crop as per the given climatic conditions which will help to maximize yield rate.

Sundari V et al., [7] Crop recommendation and yield prediction using machine learning algorithms, A comparison of machine learning algorithms was conducted in order to identify which algorithm was more accurate in predicting the best harvest. The results show that the proposed machine learning algorithm technique has the best accuracy when comparing entropy calculation, precision, Recall, F1 Score, Sensitivity, Specificity, and Entropy.

Parikh et al., [8] Machine Learning Based Crop Recommendation System. This system focuses on building a predictive model to recommend the most suitable crops to grow in a particular farm based on various parameters. This can be helpful for the farmers to be more productive and competent without wasting any resources by farming the most competent crops. This will help them in improving their crop production both qualitatively and quantitatively. This will also help them to maintain the quality and nutrition contents of the soil.

Tiago Domingues et al [9] Machine Learning for Detection and Prediction of Crop Diseases and Pests: A Comprehensive Survey, This survey aims to contribute to the development of smart farming and precision agriculture by promoting the development of techniques that will allow farmers to decrease the use of pesticides and chemicals while preserving and improving their crop quality and production. This survey presented an insight into existing research addressing the application of ML-based techniques for forecasting, detection, and classification of diseases and pests.

Bandara et al., [10]. Crop Recommendation System. This solution of crop recommendation system predicts the user, what crop type would be the most suitable for the selected area by collecting the environmental factors for plant growth and processing them with the trained sub-models of the main of the system. Before selecting any plant to grow it is important to have the knowledge and an understanding of the factors that affect the cultivation and how to maintain or control them. From this system, these above-mentioned factors are automatically processed and select the crop type to be cultivated.

The authors of [11] .The Times of India -Newspaper headlines, “Sugarcane cultivation has turned bitter for hundreds of farmers in Papanasam in Thanjavur due to Pokkaboing disease”, “Mar infection has been seen in pigeon pea crop in some areas in Hingoli” , “Green gram crop hit by yellow patch disease in Belagavi dist”. ‘’Dwarf disease hits 5% paddy, early sown Punjab crop more prone: Govt”. These headlines indicates that diseases are also plays a major role in crop culitivation. Its important for farmers to have a knowledge on crop diseases in prior of seedling of that particular crop.

S. Mamatha et al.,[20] Crop Recommendation Using Machine Learning Techniques. In this paper, the authors present a crop recommendation system that employs machine learning techniques to assist farmers in selecting appropriate crops. They utilize historical data on crops and environmental factors such as temperature, rainfall, and soil type to train the machine learning models. Several algorithms, including Decision Tree, Naive Bayes, Random Forest, and Support Vector Machine, are evaluated and compared for their performance in crop recommendation. The experimental results demonstrate that the Random Forest algorithm outperforms the other models in terms of accuracy and efficiency. The system successfully recommends crops based on the input environmental factors, providing valuable insights to farmers for decision-making

M.V.R. Vivek et al.,[21]. A Survey on Crop Recommendation Using Machine Learning. In this paper, we have proposed an examination of the soil information utilizing distinctive calculations and forecast strategy. From the investigation in this paper, we presumed that there is as yet a need of research in the Agricultural field to improve precision. Utilizing group techniques is a decent method to guarantee better precision of the framework. Additionally, on the off chance that we need to think about just a single calculation for the proposal framework, we can utilize SVM because of its basic computational necessities.

Sapna Jaiswal et al [22], Collaborative Recommendation System For Agriculture Sector. In this paper, the Query Analysis has been done using Collaborative filtering using KNN Algorithm with Gradient Ascent Singular Valued Decomposition and Support Vector Machine. After comparing the results obtained from KNN and SVM implementation, it is observed that KNN performs intuitively better. The accuracy obtained nearly stands at 87percent.

Table 2.2.1 Different strategies used in Literature on Crop recommendation system

SI . No	Title	Authors & year	Concepts/ Techniques	Disadvantages
1.	Crop Recommendation System Using Machine Learning Algorithm.	Omkar Kulkarni et al., 2022	Machine learning Random Forest Algorithm	It is Simple model representation
2.	Crop Prediction using Machine Learning Approaches .	Nischitha K et al.,2020	Machine Learning, Decision tree, SVM	Decision trees may not perform well. Sensitivity to feature selection
3.	Soil Analysis and Crop Recommendation using Machine Learning.	AdityaMotwani et al.,2021	Random Forest, Deep learning, Convolutional Neural Network	Recommend crop only on basis of the soil type which can minimise yield production
4.	Smart Crop Prediction using IoT and Machine Learning	Archana Gupta et al.,2021	IoT and Decision Tree, KNN and Support Vector Machine (SVM)	It is Lack of transparency and Data quality
5.	Machine Learning based Recommender Systems for Crop Selection: A Systematic Literature Review	Younes Ommane et al.,2022	Systematic Literature Review	The accuracy of the recommendation s can be limited if there is not enough historical data available to train the machine learning algorithms
6.	Crop Suggesting System Using Unsupervised Machine Learning Algorithm	K.D.Yesugade et al., 2019	K-means algorithms	Its only taken two case and its may be complex for many cases(crops)

7.	Survey on Machine Learning for Crop Prediction Using Soil Nutrients Data	R.Kesavamoorthy et al.,2021	Support Vector Machine, Bagged Tree, Adaboost, Naive Bayes, and Artificial Neural Network.	Only intake of soil nutrients parameters, so data insufficitent to give high accuracy
8.	Machine Learning Based Crop Recommendation System	Dhruv Piyush Parikh et al.,2021	Machine Learning, Random Forest Classifier, Tkinter	Crop is recommended but yield is not predicted sometimes it may be affect profit
9.	Machine Learning for Detection and Prediction of Crop Diseases and Pests: A Comprehensive Survey	João C. Ferreira et al., 2022	Random Forests, ANN,SVM Classfier Algorithm	Survey not deeply discussed about many research papers
10.	Crop Recommendation System	Pradeepa Bandara et al., 2020	Naïve Bayes, Support Vector Machine, K-Means Clustering and also Natural Language Processing (Sentiment Analysis)	Date is more accurate but lack of output accuracy compared to other models

2.3 Conclusion

In above, various research works were discussed. Among them Omkar Kulkarni, Sushil Buragute, Ashwin Khode, Melvin Fernando, Nilesh Korade,[1] crop recommendation system using machine learning algorithm is base paper. Dataset methodology were learned from youtube and google. From newspapers headlines, idealogy of possible diseases is obtained [11].

CHAPTER 3
PROPOSED SYSTEM

3.1 Introduction

1. The proposed system is aimed at developing a web-based application for crop selection and possible diseases prediction system using MLP algorithm.
2. The system uses a dataset that contains information about various crops and their characteristics, as well as the common diseases that affect them.
3. The system uses MLP algorithm to train a predictive model that can recommend suitable crops based on the user's input parameters like soil type, climate, temperature, rainfall, etc.
4. Additionally, the system also predicts possible diseases that may affect the selected crop and provides recommendations for prevention and treatment.
5. The web application interface allows users to input their parameters and receive crop recommendations and disease predictions in real-time.
6. The system is designed to be user-friendly and accessible, providing farmers and agricultural experts with an efficient tool to optimize crop yield and minimize disease outbreaks.
7. The proposed system has the potential to contribute to sustainable agriculture and improve food security by guiding farmers towards optimal crop selection and disease management.

3.2 Goal

The main goal of the Crop selection and possible diseases prediction system web application is to provide a user-friendly and efficient platform for farmers and agriculture enthusiasts to identify the most suitable crop for a given set of environmental conditions and predict the possible diseases that might affect the crop.

3.3 Objectives

1. Accurately recommend crops based on environmental factors.
2. Early detection and prevention of crop diseases.
3. Improve crop yields and reduce costs.
4. Enhance decision-making for farmers.
5. Design for adaptability to changing environmental conditions.

3.4 Multi - Layer Perceptron (MLP)

A multi-layer perceptron (MLP) is a type of artificial neural network (ANN) that consists of multiple layers of interconnected nodes, called neurons. It is a feedforward neural network, meaning that information flows through the network in one direction, from the input layer to the output layer.

Here's a step-by-step explanation of how an MLP works:

Architecture: An MLP typically consists of three types of layers: an input layer, one or more hidden layers, and an output layer. Each layer is composed of multiple neurons, and each neuron is connected to neurons in the adjacent layers. The input layer receives the input data, the hidden layers process the information, and the output layer produces the final predictions or outputs.

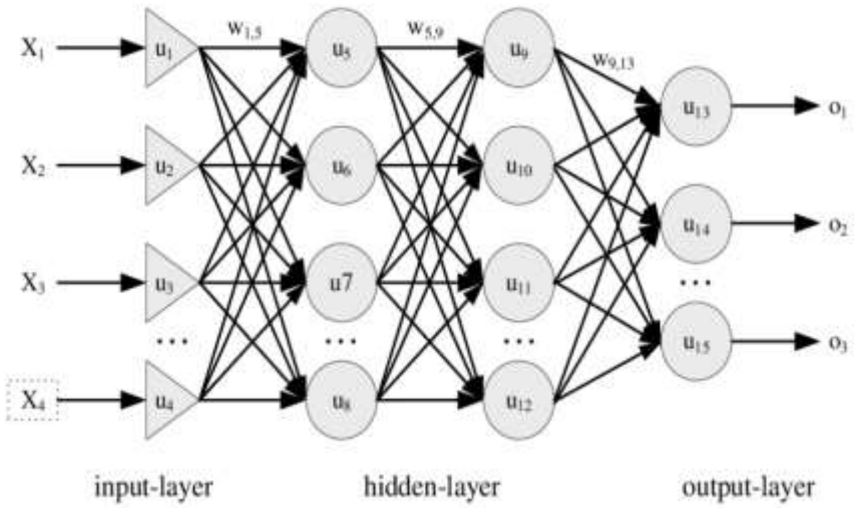

Figure 3.4.1 MLP Architecture

Neurons and Activation Functions: Each neuron in the MLP performs a weighted sum of the inputs it receives, applies an activation function to the sum, and produces an output. The activation function introduces non-linearity into the network, enabling it to model complex relationships in the data. Common activation functions include the sigmoid, tanh, and ReLU functions.

ReLu : The Rectified Linear Unit (ReLU) activation function is a widely used activation function in neural networks, including multi-layer perceptrons (MLPs). It is a simple yet effective non-linear function that introduces non-linearity to the network and helps it learn complex patterns in the data.

Mathematically it is represented as:

$$\mathbf{f(x)=max(0,x)}$$

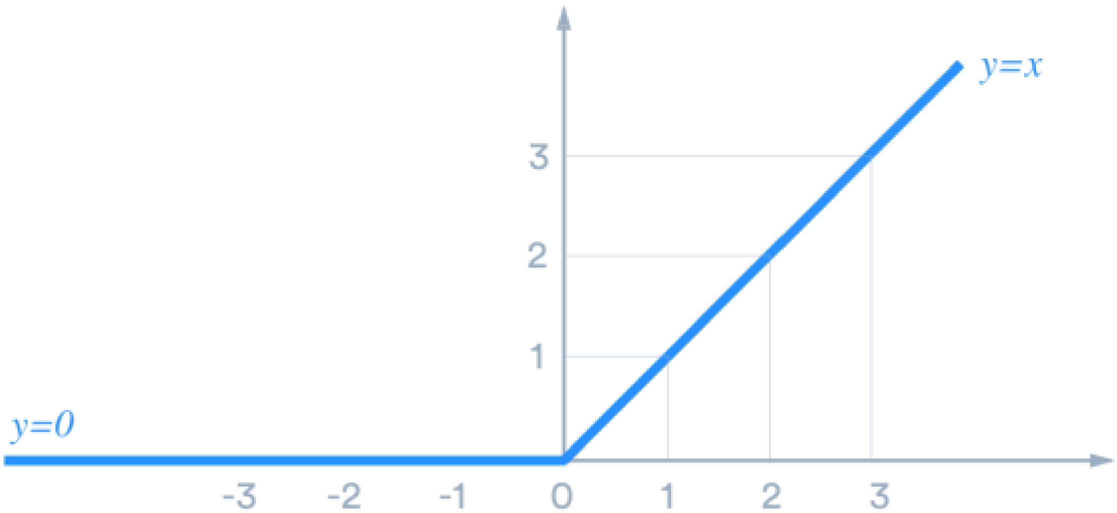

Figure 3.4.2 Mathematical Representation of MLP

Feedforward Propagation: During the feedforward process, the input data is passed through the network layer by layer. The output of each neuron in a layer serves as the input to the neurons in the next layer. The weighted sum of inputs is calculated at each neuron, and the activation function is applied to produce the output. This process continues until the output layer is reached, and the final predictions are obtained.

Training: The training of an MLP involves adjusting the weights of the connections between neurons to minimize the difference between the predicted outputs and the true outputs. This is done using a technique called backpropagation, which uses the gradient descent optimization algorithm. During backpropagation, the error between the predicted outputs and the true outputs is propagated backward through the network, and the weights are updated in the direction that minimizes the error.

Backpropagation: In the backpropagation phase, the error is first calculated at the output layer and then propagated backward through the hidden layers. The gradient of the error with respect to the weights is computed, and the weights are updated accordingly using the gradient descent algorithm. This process iteratively repeats for a number of epochs or until a convergence criterion is met.

Model Evaluation: Once the MLP is trained, it can be used to make predictions on new, unseen data. The input data is fed through the network, and the output layer produces the predicted outputs. The performance of the model can be evaluated using various evaluation metrics, such as accuracy, precision, recall, or mean squared error, depending on the type of problem.

MLPs are capable of learning complex non-linear relationships in the data and are widely used for tasks such as classification, regression, and pattern recognition. However, they can be prone to overfitting if the model complexity or the number of hidden layers and neurons is too high. Regularization techniques, such as L1 and L2 regularization, are often used to prevent overfitting in MLPs.

3.5 EVALUATION METRICS

3.5.1 Accuracy

Machine Learning model accuracy is the measurement used to determine which model is best at identifying relationships and patterns between variables in a dataset based on the input or training data.

Accuracy = (Number of correctly classified instances) /

(Total number of instances)

3.5.2 Classification Report

The classification report visualizer displays the precise, recall, F1 and support scores for the model. In order to support easier implementation and problem detection, the report integrates numerical scores with a color-coded heatmap. All heatmaps are in the range(0,0,1,0) to facilitate easy comparison of classification models across different classification reports. The classification report always a representation of the main classification metrics on a per-class basis. This gives a deeper intuition of the classifier behaviour over global accuracy which can mask functional weakness in one class of a multiclass problem.

i. Precision
   - Precision is the ability of a classifier not to label an instance positive that is negative. For each class, it is defined as the ratio of true positives to the sum of a True Positive and False Positive.
   - Precision = TP/ (TP + FP)

ii. Recall

- Recall is the ability of a classifier to find all positive instances. For each class, it is defined as the ratio of true positives to the sum of true positives and false negatives.
- Recall = TP/(TP+FN)

iii. F-measure

- The F1 score is a weighted harmonic mean of precision and recall such that the best score is 1.0 and the worst is 0.0. F1 scores are lower than accuracy measures as they embed precision and recall into their computation. As a rule of thumb, the weighted average of F1 should be used to compare classifier models, not global accuracy.
- F1 Score = 2*(Recall * Precision) / (Recall + Precision)

3.5.3 Confusion matrix

- A confusion matrix is a table that is often used to **describe the performance of a classification model** (or "classifier") on a set of test data for which the true values are known. The confusion matrix itself is relatively simple to understand, but the related terminology can be confusing.

		Actual values	
		Positive (1)	Negative(0)
Predicted values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Table 3.5.3 Confusion matrix

3.6 Features of Crop selection and possible diseases prediction system

1. User-friendly web interface.
2. Predicts the best crop for cultivation based on input parameters.
3. Provides information on possible diseases that may occur in selected crop
4. Uses machine learning algorithm (MLP) to make accurate predictions.
5. Accepts input parameters such as N, P, K, temperature, humidity, pH, and rainfall.
6. Provides real-time predictions to help farmers make informed decisions.
7. Provides detailed information about the recommended crop and its cultivation practices.
8. Can be used by farmers with little to no technical knowledge.
9. Can be easily accessed from any device with an internet connection.
10. Helps farmers maximize crop yields and minimize losses due to diseases.

3.7 Conclusion

In this chapter, objectives and goals of Crop selection and possible diseases prediction System using deep learning, various algorithms and evaluation metrics which are used for measure the performance of models are learned.

CHAPTER 4

REQUIREMENT ANALYSIS

4.1 Introduction

For development of system, knowledge about hardware and software configurations must be known. Following of this chapter, hardware requirements such as CPU, RAM, GPU, Storage, Software requirements such as OS, Programming language, IDE, frameworks such as machine learning and deep learning frameworks, web hosting framework and libraries used for system development will discuss.

4.2 Hardware requirements

- CPU: Laptop or PC with Intel Core i5 6^{th} generation processor or higher with clock speed 2.5 GHz or above. Equivalent processors in AMD will also be optimal.
- RAM: Minimum 8 GB of RAM is required; 16 GB is recommended.
- GPU: NVIDIA GeForce GTX 960 or higher.
- Storage: SSD is recommended for faster pre-processing of data than HDD.

4.3 Software requirements

- OS – Windows 7 or higher version but Windows 10 is recommended / Minimum Ubuntu 16.04 is required.
- Python (version: 3.7.9) – Programming Language used for Machine Learning and Deep Learning.
- Jupyter Notebook – Development environment.
- Spyder – Both deployment and development environment.

4.4 Frameworks

- Sci-kit Learn – open-source machine learning library developed by the Python community.
- Flask – Python framework used for web hosting.

4.5 Libraries:

Sci-kit: Scikit-learn, also known as sklearn, is a popular open-source machine learning library for Python. It is built on top of other scientific computing packages like NumPy, SciPy, and matplotlib. Scikit-learn provides a wide range of machine learning algorithms and tools for tasks such as classification, regression, clustering, dimensionality reduction, model selection, and pre-processing of data.

Matplotlib: This is a popular plotting library in python that provides a wide range of functionalities to create high-quality visualizations. It is widely used for data visualization.

Seaborn: It also used for high level interfacing for creating attractive and informative statistical graphics.

NumPy: NumPy is a popular open-source Python library used for scientific computing and data analysis. It provides efficient array and matrix operations, mathematical functions, random number generators, and more.

Pandas: It is a popular open-source Python library used for data analysis and manipulation. It provides fast, flexible, and easy-to-use data structures for working with structured data. Overall, Pandas is a versatile library that can be used for a variety of data analysis tasks. Its popularity is largely due to its ease of use and ability to handle large datasets efficiently.

Pickle: It is a Python library that is used for serializing and de-serializing Python objects. Pickle is a powerful library that can be used for a variety of purposes. Its popularity is largely due to its simplicity and ease of use, as well as its ability to serialize and de-serialize Python objects quickly and efficiently.it is also used for inter-process communication.

4.6 Conclusion

The knowledge about system requirements, software, libraries, frameworks are understood.

CHAPTER 5

SYSTEM DESIGN

5.1 Introduction

The detailed explanation of each module such as General Analysis, Learning Required Technology, Dataset Collection and Semantic Analysis, Dataset Pre-processing, Model Development and Implementation, WebApp implementation, Testing Overall system and timeline for each module between 04.02.2023 and 21.04.2023, architectures such as system architecture, technical architecture and other technical details of the system will discussed in following topics.

5.2 Module Split

The complete system development process is split into 7 modules such as

- General Analysis
- Learning Required Technology
- Dataset Collection and Semantic Analysis
- Dataset Pre-processing
- Model Development and Implementation
- WebApp implementation
- Testing Overall system

5.2.1 Modules

- General Analysis

  The General Analysis module consists of a Literature Survey (study of existing systems, research works, relevant and similar systems with their approaches and used algorithms, gaining experience and knowledge from them), Requirement Analysis (defining the environments and identifying system needs, finding the required tools, technology, hardware and software with their availability, capabilities, consistency, features, supported platforms and languages then make sure the right fit of them for further development process)

- Learning Required Technology

After finding the right tools and technology, studying and learning the unknown tools and technologies in-depth which are needed.

Machine Learning Fundamentals: Start by understanding the basics of machine learning, including supervised learning, classification, and regression. Familiarize yourself with the concept of neural networks and the Multilayer Perceptron (MLP) architecture

Programming Language: Choose a programming language suitable for implementing your system. Popular choices for machine learning include Python, flask. Python, with libraries like TensorFlow or PyTorch, is commonly used for neural network implementations

- Data Collection and Dataset Pre-processing

Collect date of soil nutrition, rainfall, weather, rainfall and crop diseases from agricultural resources. Create a new dataset using pandas library. Check the null values using df.isnull().sum(). Normalize the data using mean, median, mode method to replace a null value. Plot the data using correction matrix and Scatterplot for data visualization.

Gather disease occurrence data: Collect information on the occurrence of diseases affecting different crops in the region. Local agricultural agencies, research institutions, or farmers' organizations may have relevant data on disease outbreaks.

Pre-process the collected data: Once you have gathered the required data, perform preprocessing steps to prepare it for MLP training:

a. Handle missing values: Check for missing values in the collected data and decide on an appropriate strategy to handle them. You can either remove the samples with missing values or use techniques like imputation to fill in the missing data.

b. Normalize/standardize numerical data: Scale numerical features to a similar range to prevent one feature from dominating others. Common techniques include normalization (scaling values between 0 and 1) or standardization (transforming data to have zero mean and unit variance).

c. Encode categorical variables: If your data includes categorical variables like crop types or disease categories, encode them into numerical values. One-hot encoding or label encoding can be used depending on the nature of the data.

- Model Development and Implementation

  Exact the input features as X and output features as Y fit to multiple output values using MultiLabelBinarizer Split the datasets into training and testing sets. Create the MLP Classifier model using the MLPClassifier class from Scikit-learn library. Three hidden layers is created with neurons of each [Hidden_layer_sizes=(100,50,10)]. Backpropagation techniques is used to train the datasets into MLP Classifier model. The weights of connections are adjusted in each iteration until the error is minimized(max_iter=1000). fit() method is used to Fit the MLP model on the training data.

- WebApp implementation

  Create a simple form using HTM, CSS, JS to get input parameters from users. MLP classifier model and the Webpage is integrated using Flask framework. Predicted output is displayed in output page.

- Testing Overall system

Predict the output targets for the test data using Predict() method. The accuracy of the MLP classifier model is calculated using accuracy_score method from Scikit-learn library. The accuracy score is calculated by comparing the predict output targets with actual output targets in the testing sets.

5.2.2 Feature Extraction

Feature extraction is the process of identifying and selecting relevant information, or features, from raw data. This process is an important step in many machine learning and data analysis tasks, as it can significantly impact the performance of the resulting models. Feature extraction typically involves several steps, including data preprocessing, feature selection, feature extraction, feature encoding, and model training. The goal of feature extraction is to transform the raw data into a more compact and informative representation that can be easily analyzed or used for machine learning. The choice of feature extraction techniques depends on the specific application, as well as the characteristics of the data and the desired outcome. Overall, feature extraction is a critical step in many data-driven applications, and its success often depends on the quality and relevance of the selected features. Data preparation consist of two phases,

i. Data gathering
ii. Feature extraction

i. Data gathering- Dataset is collected from Kaggle website. Diseases data gathered from some web resources.The dataset consists 6600 data which has 22 labels.

ii. Feature extraction - Feature extraction for crop selection and possible diseases prediction system using multilayer perceptron (MLP) involves selecting and transforming relevant features from the collected data, training the MLP model using the extracted features, and evaluating and fine-tuning the model performance. MLP is a type of artificial neural network that consists of multiple layers of neurons, each of which is connected to the previous layer. The neurons in each layer use an activation function to transform the input signals and pass the output to the next layer. During training, the MLP learns the relationship between the input features and the output labels by adjusting the weights and biases of the neurons. To extract features, the selected features are transformed into a suitable representation for the MLP model. This can be done using techniques such as Principal Component Analysis (PCA), Wavelet Transform, or other feature extraction methods. The extracted features should be informative, discriminative, and representative of the data. In MLP, the performance of the prediction system can be improved by fine-tuning the model hyperparameters, which can include the number of layers, neurons per layer, learning rate, activation function, and regularization. Overall, feature extraction for crop selection and possible diseases prediction system using MLP involves selecting the most relevant features, transforming them into a suitable representation, training the MLP model, evaluating its performance, and fine-tuning the model hyperparameters to optimize the performance.

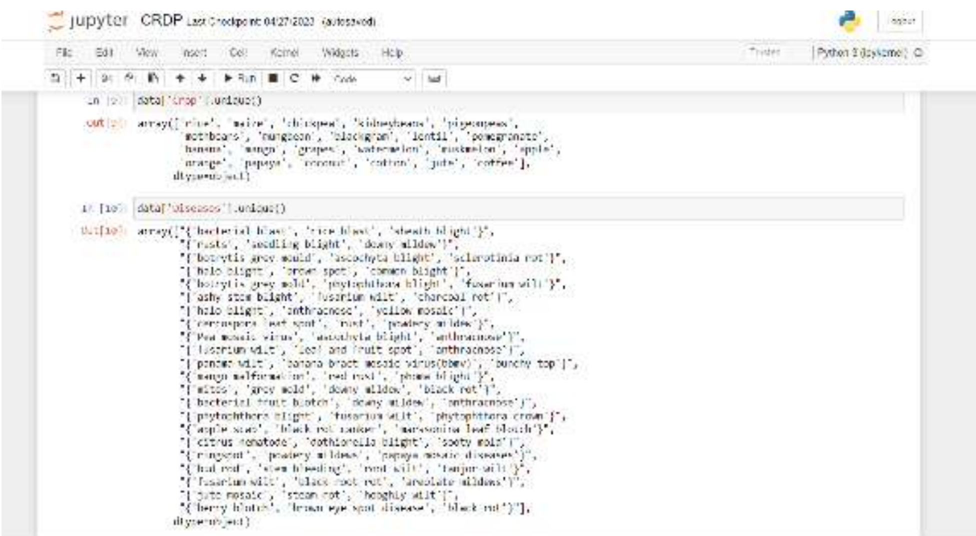

Figure 5.2.2.1 Output labels

The above figure 5.2.2.1 output labels show the individual output labels of the dataset.The input features include N (Nitrogen), P (Phosphorus), K (Potassium), temperature, humidity, pH, and rainfall. The output features include the crop and diseases.To split the dataset into input and output features, you can separate the N, P, K, temperature, humidity, pH, and rainfall columns as input features and the Crop and diseases columns as output features.

# Split the input and output features shown in figure 5.2.2.2
X = crop_data[['N', 'P', 'K', 'temperature', 'humidity', 'pH', 'rainfall']]
y = crop_data[['Crop', 'diseases']]
The X DataFrame will contain the input features, and the y DataFrame will contain the output features. You can then use these input and output features to train a machine learning model to predict the crop and diseases based on the environmental factors.

```
In [22]: # Preprocess the data by separating the input features (X) and output labels (y)
         X = data[['N', 'P', 'K', 'temperature', 'humidity', 'ph', 'rainfall']]
         y = data[['Crop', 'Diseases']]

In [23]: # Create a MultiLabelBinarizer object and fit it to the target data
         mlb = MultiLabelBinarizer()
```

Figure 5.2.2.2 Split the input and output features

5.2.3 Gantt chart

A Gantt chart is a visual representation of a project schedule that uses horizontal bars to illustrate the project's tasks, their start and end dates, and their dependencies.

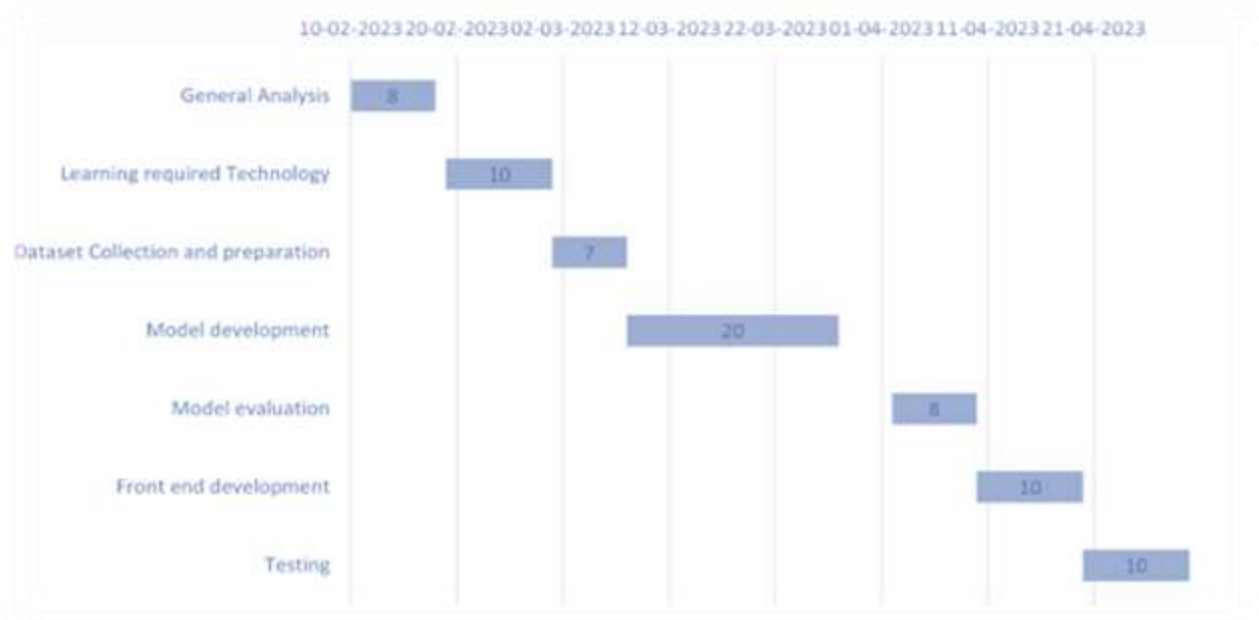

Figure 5.2.3.1 Gantt Chart

In the above figure 5.2.3.1 The Gantt chart depicts the schedule for a project spanning from February 10, 2023, to April 24, 2023. It provides a visual representation of the project tasks, their durations, and dependencies during this timeframe.The chart begins with an overview of the project initiation phase, where the project goals and objectives are defined, stakeholders are identified, and necessary resources are allocated. This initial phase sets the foundation for the project's success.Following the project initiation, the data collection phase takes place. During this period, the team identifies relevant data sources and gathers information on soil type, weather conditions, crop types, and diseases. It is crucial to ensure the quality and completeness of the collected data to obtain accurate results.The subsequent phase involves data preprocessing. This step focuses on preparing the data for further analysis. Tasks such as handling missing values, removing duplicates, normalizing or standardizing the data, and performing feature selection or dimensionality reduction are carried out. These preprocessing steps aim to improve the quality and efficiency of the data for subsequent modeling.The core of the project lies in the model development phase. Here, the team chooses the Multilayer Perceptron (MLP) as the predictive model. The data is split into training and testing sets, and the MLP model is trained on the training data. This phase involves configuring the MLP architecture, setting hyperparameters, and optimizing the model to achieve accurate predictions.Throughout the project timeline, regular progress monitoring and updates are crucial. This allows the team to track the completion of tasks, identify any delays or bottlenecks, and make necessary adjustments to keep the project on schedule.The Gantt chart visually represents the project tasks, their durations, and their dependencies, enabling project managers and team members to have a clear understanding of the project timeline. It facilitates effective project planning, resource allocation, and progress tracking, ultimately ensuring the successful completion of the project within the specified timeframe from February 10, 2023, to April 24, 2

5.3 SYSTEM ARCHITECTURE

A system architecture diagram is a visual representation of the overall structure and components of a system, including hardware, software, and any communication or data flows between them. It can be used to communicate the high-level design of a system to stakeholders such as developers, architects, project managers, and other interested parties. Typically, a system architecture diagram will include blocks or components representing different parts of the Lsystem, such as servers, databases, application layers, user interfaces, and external interfaces. The connections between these components are also usually shown, representing the flow of data or communication between them. System architecture diagrams can be used in a variety of contexts, such as in software development projects, network infrastructure planning, or system integration projects. They can help stakeholders to better understand the design and structure of a system, identify potential issues or challenges, and make informed decisions about how to proceed with the project. A system architecture is the conceptual design of a system that defines its structure, behaviour, and components. It provides a high-level view of the system, outlining the key features and functions, and how they interact with each other. The architecture serves as a blueprint for building the system and guides the development process.

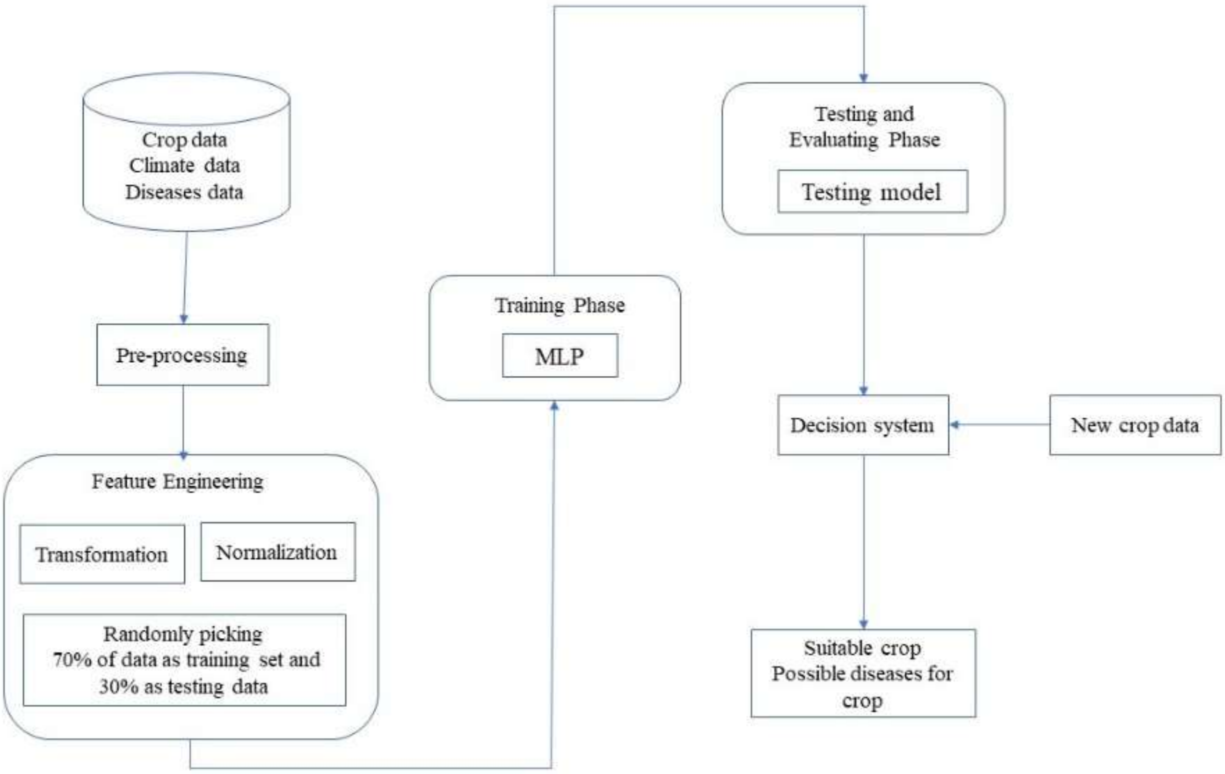

Figure 5.3.1 Block Diagram of System Architecture

In the above figure 5.3.1 shows the system architecture of the Proposed -Crop selection and possible diseases prediction system some of the steps involved in this system are listed below,

- Pre-processing step
- Feature engineering step
- Training phase
- Testing and Evaluation phase
- User Interface

5.4 DATA FLOW DIAGRAMS

Data Flow Diagram (DFD) is the representation of information flows in the system. It shows how data enters, what process takes place and where is data stored in that system. It is also known as data flow graph. It is classified into three different levels based on increasing information and functionality of the system by,

- DFD Level 0
- DFD Level 1
- DFD Level 2

5.4.1 LEVEL 0

The 0th level DFD provides a high-level view of the system, showing the main inputs and outputs of the system and the processes involved in transforming those inputs into outputs. It is useful for understanding the overall structure of the system and identifying areas for further analysis and design.

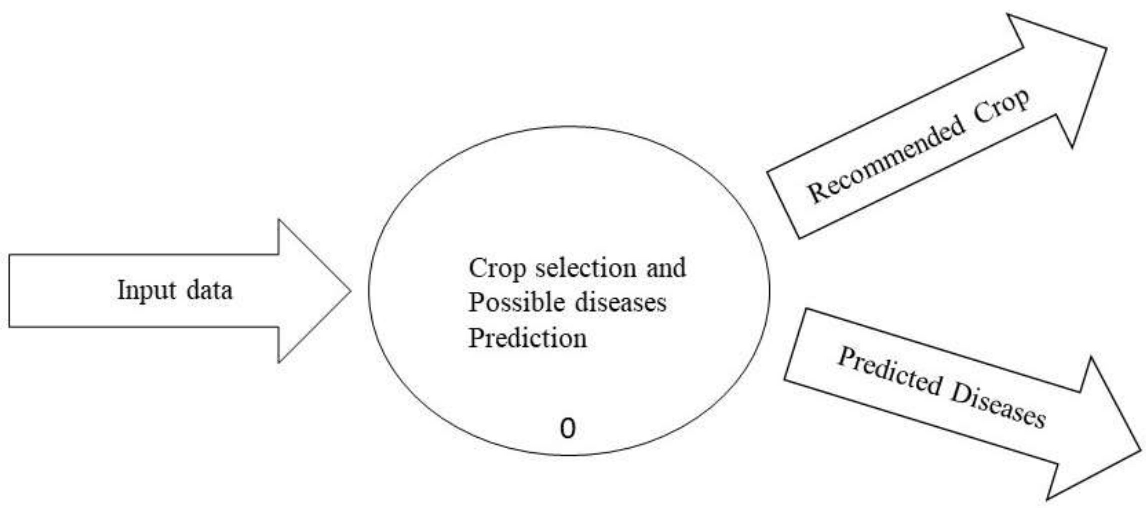

Figure 5.4.1.1 DFD - Level 0

5.4.2 LEVEL 1

A first level Data Flow Diagram (DFD) provides a more detailed view of the system than the 0th level DFD. It shows the major processes within the system and how data flows between them. The 1st level DFD decomposes the main process of the 0th level DFD into sub processes and provides a more detailed view of the inputs, outputs, and data stores used by each process.

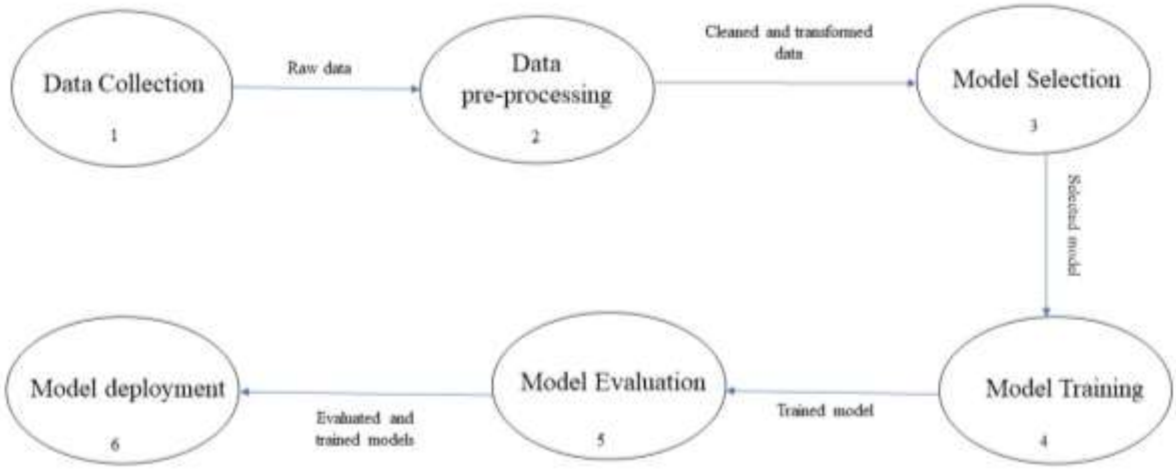

Figure 5.4.2.1 DFD - Level 1

In the above figure 5.4.2.1 DFD: level 1 shows major processes within the system and how data flows between them.

5.4.3 LEVEL 2

A second level Data Flow Diagram (DFD) provides an even more detailed view of the system than the 1st level DFD. It shows the subprocesses identified in the 1st level DFD broken down into even more detailed subprocesses, and it provides a more detailed view of the inputs, outputs, and data stores used by each subprocess.

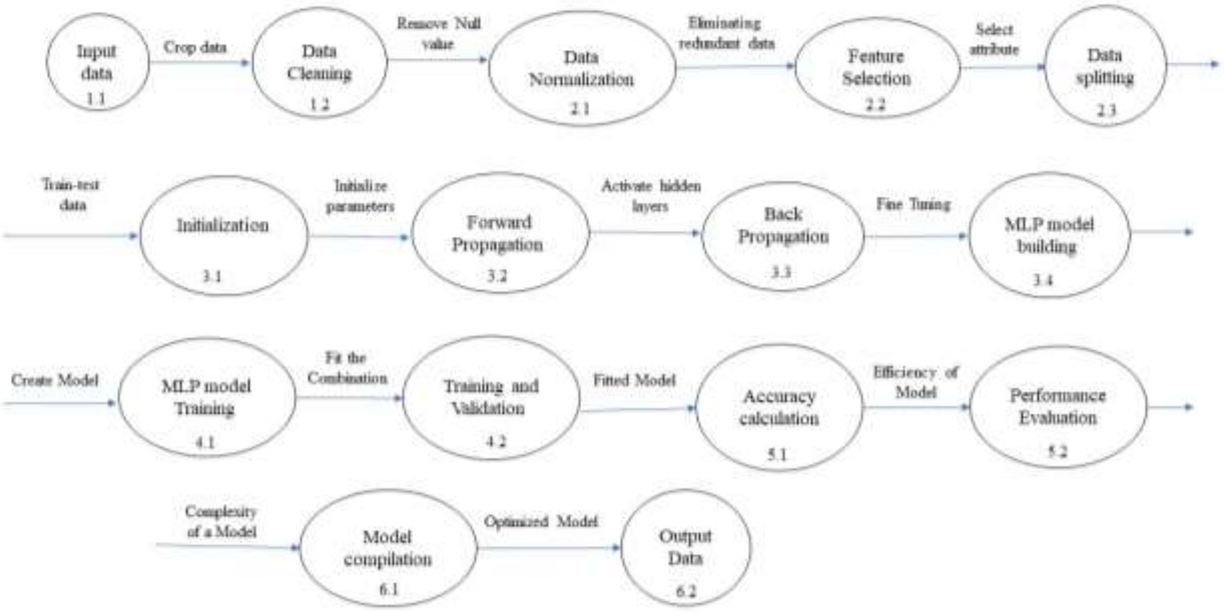

Figure 5.4.3.1 DFD - Level 2

In the above figure 5.4.3.1 DFD-Level 2 the subprocesses identified in the 1st level DFD broken down into even more detailed subprocesses, and it provides a more detailed view of the inputs, outputs, and data stores used by each subprocess.

5.5 UML DIAGRAMS

UML diagrams are visual representations used in software engineering to illustrate different aspects of a system's design. They provide a standardized language for communicating and documenting software systems, capturing their structure, behavior, and interactions. UML diagrams such as class diagrams, use case diagrams, sequence diagrams, activity diagrams, state machine diagrams, component diagrams, and deployment diagrams allow developers, designers, and stakeholders to better understand and analyze the system's architecture, relationships between components, user interactions, and system behavior. By using UML diagrams, software professionals can effectively communicate, plan, and design software systems, facilitating collaboration and improving the overall quality of the development process.

5.5.1 USECASE DIAGRAM

A use case diagram represents the behaviour of the system. It expresses how the user can interact with the system, the services provided by the system, and relationship between them. The components are actors represented by stick figures and use cases represented by ellipse, system and lines. Ellipse represents the role of the actors and whole system functions bounded by rectangle bound which portrays complete system functionality

Following use case diagram shows the actors of PROPOSED System and their functions. The Proposed system has two actors

i. **User**
ii. **Developer**

User can give input and saw the output System accepts the user input then pre-process, convert and detect the result which is provided as output for user. Developer collects new dataset then pre-process, train and test the model, deploy the trained model and update system if required.

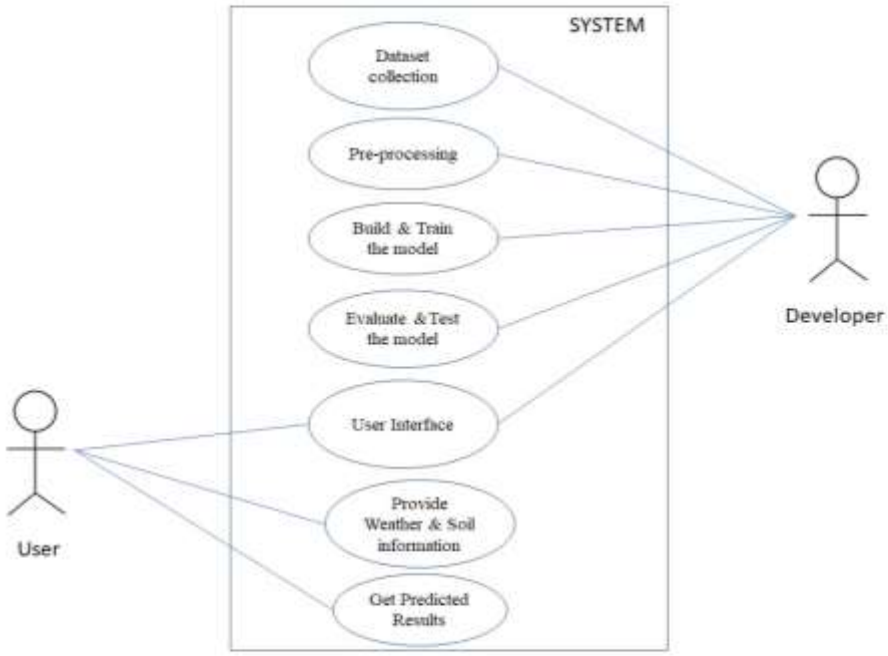

Figure 5.5.1.1 Use case Diagram of proposed system

In the above figure 5.5.1.1 provides an overview of the main interactions and functionalities of the crop selection and possible diseases prediction system. It helps identify the different tasks performed by actors and how they relate to the system's features.

5.5.2 ER DIAGRAM

An ER (Entity-Relationship) diagram is a visual representation of the relationships between different entities or objects within a system. It is commonly used in software engineering and database design to help developers and designers understand the structure of data in a system and how different pieces of data relate to each other. In an ER diagram, entities are represented by rectangles, and relationships between entities are represented by lines connecting them. The lines can indicate different types of relationships, such as one-to-one, one-to-many, or many-to-many relationships. Attributes of each entity are represented as ovals within the entity rectangle. Attributes describe the properties or characteristics of the entity, such as its name, age, or address. The ER diagram is an important tool for database design as it helps to ensure that the data structure is well-organized, optimized for efficient access and retrieval, and can accommodate the system's business requirements**.**

In the below figure 5.5.2.1 , the ER diagram captures the entities, attributes, and relationships in the crop selection and possible diseases prediction system, providing a visual representation of the system's data structure. In the crop selection and possible diseases prediction system, the ER diagram represents the main entities involved in the system, their attributes, and the relationships between them.

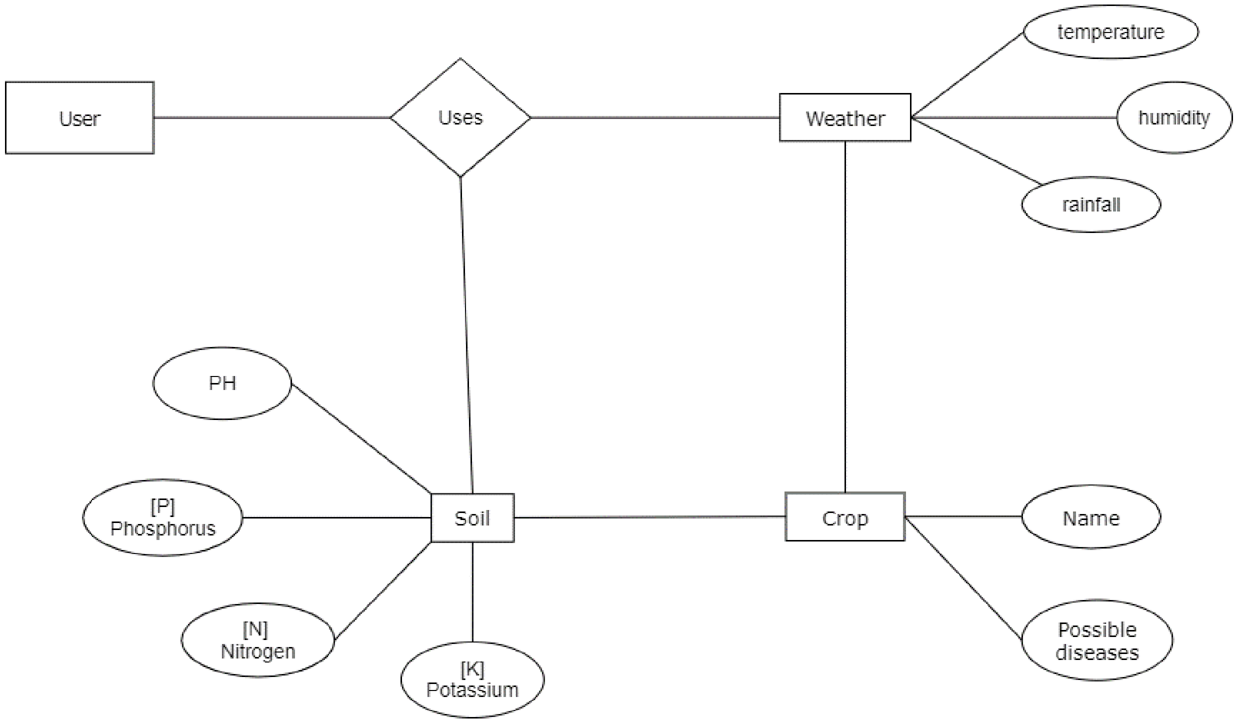

Figure 5.5.2.1 ER Diagram of proposed system

5.6 Conclusion

Working of the system, their data flow and overview of development process are seen in this chapter.

CHAPTER 6

RESULT AND DISCUSSIONS

6.1 Introduction

Crop selection and possible diseases prediction system model deployed as a web application using Flask web hosting framework, overview of the UI features and conclusion with advantages, disadvantages of Proposed System will be described.

6.2 Web Application

For users to interact with the model, Crop selection and possible diseases prediction system was developed as a web application.

Flask is used to deploy the Crop selection and possible diseases prediction system. It is a popular Python framework used for deploying and hosting web applications using Python language. A web application is a software application that runs on a web server and is accessed by users through a web browser or a web-based interface.

Following figures show the user interface of proposed with its interactive components,

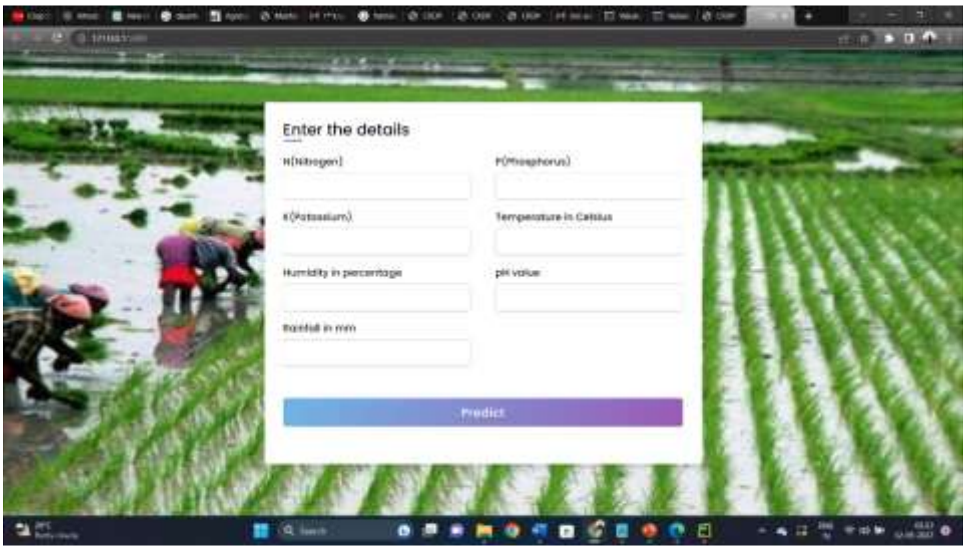

Figure 6.2.1 Frontend of Proposed system

In the above figure 6.2.1 is a frontend of proposed system.It is a simple web application of crop selection and possible diseases system.

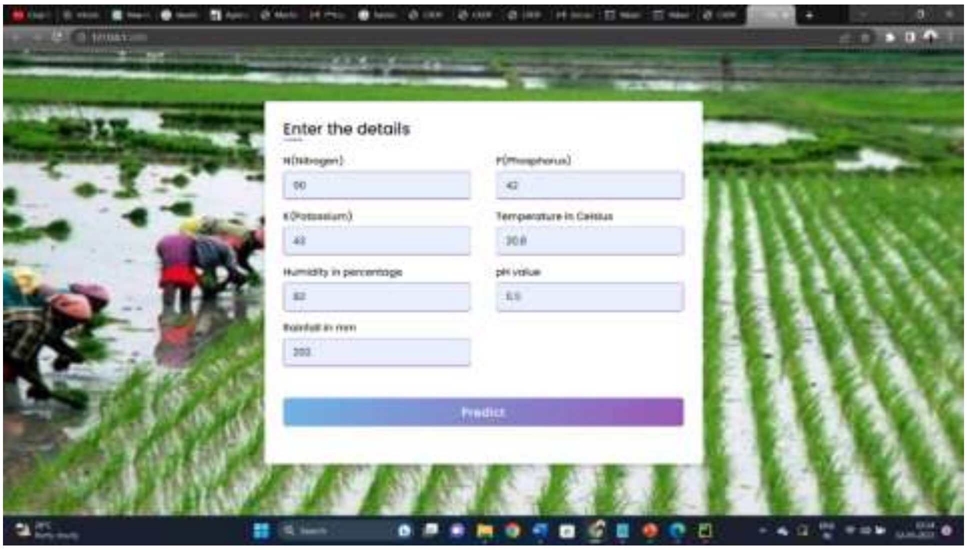

Figure 6.2.2 Input form of proposed system

In the above figure 6.2.2 is a Input form of proposed system. It is simple input form with seven input parameters and predict button were included. It user friendly for users. They can simply enter the details from soil test report and weather report.

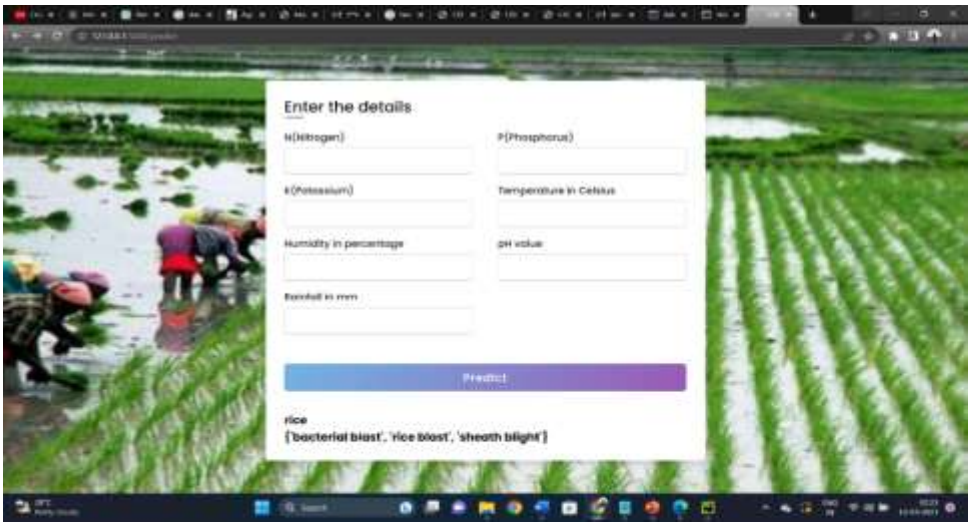

Figure 6.2.3 Output of proposed system

In the above figure 6.2.3 is displaying output of proposed system. In the below of predict button, the predicted output of proposed system is displayed.

6.3 Advantages and Disadvantages of Proposed System

6.3.1 Advantages

1. Accurate prediction: The MLP algorithm can provide accurate predictions for crop selection and possible diseases, based on historical data and other relevant factors.
2. Increased crop yield: By selecting the best crop varieties and optimizing resources, the system can help farmers increase crop yields, resulting in more food production and greater profits.
3. Reduced risk of crop failure: By predicting possible diseases and other risks, farmers can take proactive measures to reduce the risk of crop failure, such as using the appropriate pesticides or crop varieties.
4. Efficient use of resources: By optimizing resource use such as water and fertilizers, the system can help farmers reduce waste and environmental impact, while also saving on costs.
5. Improved decision-making: The system can provide farmers with real-time information to make informed decisions about crop selection and management, resulting in better outcomes for their farms.
6. Cost savings: By reducing the risk of crop failure and optimizing resource use, farmers can save on costs associated with failed crops and excessive use of resources.
7. Increased food security: By increasing crop yields and reducing the risk of crop failure, the system can contribute to increased food security for communities and regions.
8. Ease of use: The system is user-friendly and easy to use for farmers, with a simple web interface that can be accessed from anywhere with an internet connection.

6.3.2 Disadvantages

1. Need for accurate data: The system relies heavily on accurate data inputs, which may not always be available. Poor quality data can lead to inaccurate predictions and recommendations.
2. Limited scope: The system may only be effective for certain types of crops and environmental conditions. It may not be able to provide accurate predictions for all crops and regions.

6.4 Conclusion

The obtained results, advantages and disadvantages of proposed system are discussed in this chapter

CHAPTER 7

CONCLUSION AND FUTURE WORK

7.1 Conclusion

Crop selection and disease prediction are critical aspects of agriculture that can have a significant impact on crop yields, food security, and sustainability. Farmers face numerous challenges, such as weather conditions, soil quality, pest infestations, and diseases, that can adversely affect their crops. Therefore, it is essential to develop tools and systems that can support farmers in making informed decisions about crop selection and management. A crop selection and possible diseases prediction system using MLP (Multilayer Perceptron) algorithm can provide valuable insights to farmers to optimize their crop yields and reduce waste. MLP is a popular machine learning algorithm that is commonly used for classification and prediction tasks. It is a feedforward neural network that consists of multiple layers of interconnected neurons, making it capable of learning complex patterns and relationships in data. FARMER’S FRIEND-The crop selection and possible diseases prediction system using MLP algorithm can use various data sources, such as soil quality, weather patterns, and crop history, to predict the most suitable crop options for a particular farm. Additionally, by integrating disease prediction models, the system can help farmers anticipate and manage potential crop diseases, leading to improved crop yields and better financial outcomesThe benefits of a crop selection and possible diseases prediction system using MLP algorithm include optimized crop yields, reduced waste, and increased profitability for farmers. Furthermore, such a system can contribute to sustainable farming practices by helping farmers make more informed decisions about crop selection and management.

In conclusion, a crop selection and possible diseases prediction system using MLP algorithm can be a valuable tool for farmers to make informed decisions and optimize their crop yields. It has the potential to significantly improve the sustainability and profitability of agriculture while reducing the negative impacts of crop diseases on the environment and food security.

7.2 Future Work

Incorporating sensors and other IoT devices into the system can provide real-time data on various environmental factors, improving the accuracy of the prediction. Expanding the dataset to include a wider range of crops, environmental factors, and diseases can improve the system's accuracy and make it more versatile.

APPENDIES

A. Source Code

A.1 Index.html

```
<!DOCTYPE html>
<html lang="en" dir="ltr" xmlns="http://www.w3.org/1999/html">
  <head>
    <meta charset="UTF-8">
    <title> CRDP </title>
    <link rel="stylesheet" href="{{url_for('static',filename='style.css')}}">
     <meta name="viewport" content="width=device-width, initial-scale=1.0">
   </head>
<body>
  <div class="container">
    <div class="title">Enter the details</div>
    <div class="content">
      <form action="{{ url_for('predict')}}" name="myForm" onsubmit="return validateForm()" method="post" required>
        <div class="user-details">
          <div class="input-box">
            <span class="details">N(Nitrogen)</span>
            <input type="text" name="N"  placeholder=" " required>
          </div>
          <div class="input-box">
            <span class="details">P(Phosphorus)</span>
            <input type="text" name="P"   placeholder="" required>
          </div>
          <div class="input-box">
            <span class="details">K(Potassium)</span>
```

```html
          <input type="text" name="K"  placeholder="" required>
        </div>
        <div class="input-box">
          <span class="details">Temperature in Celsius</span>
          <input type="text" name="temp" placeholder="" required>
        </div>
        <div class="input-box">
          <span class="details">Humidity in percentage</span>
          <input type="text" name="humi"  placeholder="" required>
        </div>
        <div class="input-box">
          <span class="details">pH value</span>
          <input type="text" name="pH"  placeholder="" required>
        </div>
        <div class="input-box">
          <span class="details">Rainfall in mm</span>
          <input type="text" name="rf"  placeholder="" required>
        </div>
      </div>
      <div class="button">
        <input type="submit" value="Predict">
      </div>
    </form>
    <h3>{{crop}}</h3>
    <h3>{{disease}}</h3>
  </div>
</div>
```

```
</body>
</html>
```

A.2 Style.css

```
@import
url('https://fonts.googleapis.com/css2?family=Poppins:wght@200;300;400;500;
600;700&display=swap');
*{
  margin: 0;
  padding: 0;
  box-sizing: border-box;
  font-family: 'Poppins',sans-serif;

}
body{
  height: 100vh;
  background: url('bg1.jpg');
  background-repeat: no-repeat;
  background-size: cover;
  display: flex;
  justify-content: center;
  align-items: center;
  /*padding: 10px;
  background: linear-gradient(135deg, #71b7e6, #9b59b6);*/
}
.container{
  max-width: 700px;
  width: 100%;
  background-color: #fff;
```

```css
  padding: 25px 30px;
  border-radius: 5px;
  box-shadow: 0 5px 10px rgba(0,0,0,0.15);
}
.container .title{
  font-size: 25px;
  font-weight: 500;
  position: relative;
}
.container .title::before{
  content: "";
  position: absolute;
  left: 0;
  bottom: 0;
  height: 3px;
  width: 30px;
  border-radius: 5px;
  background: linear-gradient(135deg, #71b7e6, #9b59b6);
}
.content form .user-details{
  display: flex;
  flex-wrap: wrap;
  justify-content: space-between;
  margin: 20px 0 12px 0;
}
form .user-details .input-box{
  margin-bottom: 15px;
```

```
  width: calc(100% / 2 - 20px);
}
form .input-box span.details{
  display: block;
  font-weight: 500;
  margin-bottom: 5px;
}
.user-details .input-box input{
  height: 45px;
  width: 100%;
  outline: none;
  font-size: 16px;
  border-radius: 5px;
  padding-left: 15px;
  border: 1px solid #ccc;
  border-bottom-width: 2px;
  transition: all 0.3s ease;
}
.user-details .input-box input:focus,
.user-details .input-box input:valid{
  border-color: #9b59b6;
}
 form .gender-details .gender-title{
  font-size: 20px;
  font-weight: 500;
 }
 form .category{
```

```
  display: flex;
  width: 80%;
  margin: 14px 0 ;
  justify-content: space-evenly;
 }
 form .category label{
  display: flex;
  align-items: center;
  cursor: pointer;
 }
 form .category label .dot{
  height: 18px;
  width: 18px;
  border-radius: 50%;
  margin-right: 10px;
  background: transparent;
  border: 5px transparent;
  transition: all 0.3s ease;
}
 #dot-1:checked ~ .category label .one,
 #dot-2:checked ~ .category label .two,
 #dot-3:checked ~ .category label .three{
  background: #9b59b6;
  border-color: #d9d9d9;
 }
 form input[type="radio"]{
  display: none;
```

```
}
form .button{
  height: 45px;
  margin: 35px 0
}
form .button input{
  height: 100%;
  width: 100%;
  border-radius: 5px;
  border: none;
  color: #fff;
  font-size: 18px;
  font-weight: 500;
  letter-spacing: 1px;
  cursor: pointer;
  transition: all 0.3s ease;
  background: linear-gradient(135deg, #71b7e6, #9b59b6);
}
form .button input:hover{
 /* transform: scale(0.99); */
 background: linear-gradient(-135deg, #71b7e6, #9b59b6);
 }
@media(max-width: 584px){
.container{
 max-width: 100%;
}
form .user-details .input-box{
```

```
    margin-bottom: 15px;
    width: 100%;
  }
  form .category{
    width: 100%;
  }
  .content form .user-details{
    max-height: 300px;
    overflow-y: scroll;
  }
  .user-details::-webkit-scrollbar{
    width: 5px;
  }
  }
  @media(max-width: 459px){
  .container .content .category{
    flex-direction: column;
  }
}
```

A.3 App.py

```
from flask import Flask, request, render_template
import pickle
import pandas as pd
from sklearn.preprocessing import MultiLabelBinarizer
app = Flask(__name__)
@app.route('/')
def index():
```

```
    return render_template('index.html')
@app.route("/predict", methods=['POST','GET'])
def predict():
    data = pd.read_csv('CR_DataN.csv')
    y = data[['Crop', 'Diseases']]
    mlb = MultiLabelBinarizer()
    y = mlb.fit_transform(y.values)
    features=[int(x) for x in request.form.values()]
    model=pickle.load(open('model.pkl','rb'))
    output= model.predict([features])
    output_data = mlb.inverse_transform(output)
    return render_template('index.html', crop=output_data[0][0],disease=output_data[0][1])


if __name__ == '__main__':
    app.run(debug=True)
```

REFERENCES

[1] Omkar Kulkarni, Sushil Buragute, Ashwin Khode, Melvin Fernando, Nilesh Korade, CROP RECOMMENDATION SYSTEM USING MACHINE LEARNING ALGORITHM , International Journal of Creative Research Thoughts (IJCRT), Volume 10, Issue 5 May 2022.

[2] Mahendra N , Dhanush Vishwakarma , Nischitha K , Ashwini, Manjuraju M. R, 2020, Crop Prediction using Machine Learning Approaches, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) Volume 09, Issue 08 (August 2020).

[3] Motwani, Aditya, Param Patil, Vatsa Nagaria, Shobhit Verma, and Sunil Ghane. "Soil Analysis and Crop Recommendation using Machine Learning." In *2022 International Conference for Advancement in Technology (ICONAT)*, pp. 1-7. IEEE, 2022.

[4] Archana Gupta,Dharmil Nagda,Pratiksha Nikhare,Atharva Sandbhor, ”Smart Crop Prediction using IoT and Machine Learning.” International Journal of Engineering Research & Technology (IJERT)-2021

[5] OMMANE, Y., RHANBOURI, M. A., CHOUIKH, H., JBENE, M., CHAIRI, I., LACHGAR, M., & BENJELLOUN, S. (2022). Machine Learning based Recommender Systems for Crop Selection: A Systematic Literature Review.

[6] K.D.Yesugade, Hetanshi,Chudasam,Aditi, Kharde, Ketki Mirashi Kajal Muley. Crop Suggesting System Using Unsupervised Machine Learning Algorithm.Research Gate. Vol.-7, Issue-3, March 2019

[7] Sundari V, Anusree M, Swetha U and Divya Lakshmi R, Crop recommendation and yield prediction using machine learning algorithms, World Journal of Advanced Research and Reviews, 2022, 14(03), 452–459.

[8] Parikh, Dhruv & Jain, Jugal & Gupta, Tanishq & Dabhade, Rishit. (2021). Machine Learning Based Crop Recommendation System. International Journal of Advanced Research in Science, Communication and Technology. 891-897. 10.48175/IJARSCT-1509.

[9] Tiago Domingues, Tomás Brandão & João C. Ferreira, Machine Learning for Detection and Prediction of Crop Diseases and Pests: A Comprehensive Survey, MDPI Volume: 09 Issue: 01 | Sept 2022.

[10] Bandara, Pradeepa & Weerasooriya, Thilini & T.H., Ruchirawya & Nanayakkara, W.J.M. & M.A.C, Dimantha & M.G.P, Pabasara. (2020). Crop Recommendation System. International Journal of Computer Applications. 175. 22-25. 10.5120/ijca2020920723.

[11] The authors of , “Sugarcane cultivation has turned bitter for hundreds of farmers in Papanasam in Thanjavur due to Pokkaboing disease”, “Mar infection has been seen in pigeon pea crop in some areas

in Hingoli" , "Green gram crop hit by yellow patch disease in Belagavi dist". ''Dwarf disease hits 5% paddy, early sown Punjab crop more prone: Govt".-he Times of India -Newspaper headlines(2021&2022)

[12] S.Pudumalar, E.Ramanujam, R.HarineRajashreeń, C.Kavyań, T.Kiruthikań, J.Nishań, "Crop Recommendation System for Precision Agriculture", 2016 IEEE Eighth International Conference on Advanced Computing (ICoAC).

[13] Miftahul Jannat Mokarrama; Mohammad Shamsul Arefin, "RSF: A Recommendation System for Farmers", 2017 IEEE Region 10 Humanitarian Technology Conference.

[14] Pudumalar, S., E. Ramanujam, R. HarineRajashree, C. Kavya, T. Kiruthika, and J. Nisha. "Crop recommendation system for precision agriculture." In Advanced Computing (ICoAC), 2016 Eighth International Conference on, pp. 32-36. IEEE, 2017

[15] Rakesh Kumar, M.P. Singh, Prabhat Kumar and J.P. Singh , "Crop Selection Method to Maximize Crop Yield Rate using Machine Learning Technique", International Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials (ICSTM).,2015

[16] S. Panchamurthi, M. D. Perarulalan, A. Syed Hameeduddin and P. Yuvaraj, “Soil Analysis and Prediction of Suitable Crop for Agriculture using Machine Learning”, International Journal for Research in Applied Science & Engineering Technology (IJRASET), Volume 7, Issue 3, pp. 2328-2335, 2019.

[17] Thomas, Kevin Thomas Tom, Varsha S, Merin Mary Saji, Lisha Varghese, and Er. Jinu Thomas. " Crop Prediction Using Machine Learning." International Journal of Future Generation Communication and Networking (2020)

[18] R.Kesavamoorthy.,Sameer Paudel.,Riyaz Nakarmi.,Prashant Giri., Shantanu Bahadur Karki, Survey on Machine Learning for Crop Prediction Using Soil Nutrients Data, International Journal of Software & Hardware Research in Engineering (IJSHRE), Volume 9 Issue 1 January 2021

[19] prof. D.S. Zingade ,Omkar Buchade ,Nilesh Mehta ,Shubham Ghodekar ,Chandan Mehta “Crop Prediction System using Machine Learning”.

[20] Jajur, S. Mamatha, Soumya N. G., and G. T. Raju. "Crop Recommendation Using Machine Learning Techniques." International Journal of Innovative Technology and Exploring Engineering (2019).

[21] M.V.R. Vivek, D.V.V.S.S. Sri Harsha, P. Sardar Maran, “A Survey on Crop Recommendation Using Machine Learning” Volume-7, Issue-5C, February 2019.

[22] Sapna Jaiswal, Tejaswi Kharade, Nikita Kotambe, and Dr.Shilpa Shinde, “Collaborative Recommendation System For Agriculture Sector”, ITM Web of Conferences 32, 03034 (2020)