**A Project Report**

*on*

**“GO AGRO”**

*carried out as part of the* ***Minor Project IT3270*** *Submitted*

by

**Abhinav Jindal**

**209303239**

*in partial fulfilment for the award of the degree* *of*

**Bachelor of Technology**

in

**Information Technology**

Under the Guidance of

**Guide Name**

Logo

Description automatically generated with medium confidence

**School of Information Technology**

**Department of Information Technology**

**MANIPAL UNIVERSITY JAIPUR**

**RAJASTHAN, INDIA**

**April 2023**

**CERTIFICATE**

Date:20/04/23

This is to certify that the minor project titled ‘GO AGRO’ is a record of the Bonafede work done by **ABHINAV JINDAL** (209303239) submitted in partial fulfilment of the requirements for the award of the Degree of Bachelor of Technology in Information Technologyof Manipal University Jaipur, during the academic year 2022-23.

**Dr. Avani SHARMA**

**Assistant professor**

*Project Guide, Department of Information Technology*

*Manipal University Jaipur*

**Dr. Pankaj Vyas**

*HoD, Department of Information Technology*

*Manipal University Jaipur*

**ABSTRACT**

India's population and unpredictable climate changes pose a threat to global food resources, and drought conditions often challenge farmers. Soil quality significantly affects crop yield, and recommendations for fertilizer use can aid farmers in making informed decisions. Data mining through information and communication technology has enabled studies on predicting crop yield. By analyzing past data, farmers can receive valuable suggestions for selecting crops that yield higher results. This approach helps improve crop yields, making farming more sustainable and contributing to global food security.

Agriculture plays a critical role in India's economy, and farmers have traditionally followed specific farming practices and guidelines. However, it is unrealistic to expect farmers to control all factors that affect crop growth, and impulsive decisions can have adverse effects. To ensure India's economic growth, an advanced agricultural system is essential, considering recent advances in various fields.

To aid farmers in making informed decisions about farm management and agribusiness activities, a web application will be developed that employs a combination of recommender systems and machine learning. This research involves utilizing machine learning algorithms to identify crop conditions, detect diseases, make crop predictions, and provide recommendations. The study aims to demonstrate how a recommender system can be utilized in agriculture to identify and predict crop diseases.

By leveraging cutting-edge technology, such as machine learning, farmers can receive tailored recommendations based on their specific circumstances and environmental conditions. Such recommendations can help farmers improve crop yields, reduce costs, and ultimately increase their incomes. Additionally, this approach can help identify diseases early and prevent them from spreading, minimizing crop losses, and improving overall farm productivity. In sum, the development of an advanced agricultural system using machine learning and recommender systems could be a significant boon to India's agricultural sector and help ensure continued economic growth.

**Table of Contents**

1. Cover Page
2. Certificate
3. Abstract
4. List of Tables
5. List of Figures
6. Introduction
7. Background Details
8. System Design & Methodology
9. Implementation and Results
10. Conclusion
11. References

**LIST OF TABLES**

|  |  |  |
| --- | --- | --- |
| **Table No** | **Table Title** | **Page No** |
|  |  |  |
| 1. | Classification Report |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **Figure No** | **Figure Title** | **Page No** |
|  |  |  |
| 1. | Application Outlook |  |
| 2. | Background Detail |  |
| 3. | System Architecture |  |
| 4. | Application Interface |  |
| 5. | Outputs |  |
| 6. | WebApp User Interface (Frontend) |  |

**Introduction**

Agriculture is a significant contributor to India's economy, employing a large portion of the population. However, farmers' average monthly income in 17 states is only Rs. 1700, leading to farmer suicides and land conversion for non-agricultural activities. Additionally, many farmers do not want their children to continue farming due to incorrect crop selection decisions resulting in lower yields. Planting in the wrong season or choosing a crop that doesn't yield much for the soil are common mistakes. Addressing these issues can help make farming more profitable and sustainable, benefiting both farmers and the economy.

For farmers in India, relying solely on agricultural revenue can be challenging due to factors such as meteorological, geographic, organic, and economic considerations affecting crop yield. Selecting a suitable crop is also difficult due to shifting market prices. Unfortunately, this uncertainty can have dire consequences, as evidenced by India's suicide rate, which has varied between 1.4 and 1.8 percent per 100,000 people in the last ten years. With climate unpredictability, farmers may struggle to determine which crops to cultivate in each season, exacerbating these challenges.

The use of fertilizers in agriculture can be challenging due to seasonal climate fluctuations and essential components such as soil, water, and air. This uncertainty, coupled with dropping crop yields, makes it imperative to offer farmers a user-friendly recommender system. Our proposed model addresses these concerns by using economic and environmental factors to recommend crops that boost yields, meeting the country's growing food demand. The model predicts crop yield by analyzing rainfall, temperature, humidity, soil nutrients, and the pH value of the soil. Additionally, the webapp will assist farmers in maintaining soil nutrient levels and identifying plant diseases. This approach helps farmers make informed decisions and increases sustainability in agriculture.

The objectives of my project are:

**Crop Recommendation -** Our ML model predicts the most suitable crop for farmers by analyzing various parameters such as soil quality, climatic conditions, and historical crop yield data.

**Fertilizer Recommendation** - Our ML model uses various parameters to provide farmers with personalized suggestions to manage nutrient levels in the soil. By analyzing factors such as soil quality, crop type, and historical data, the model offers recommendations for fertilizer usage and soil management practices.

**Crop Disease Classification** - Our ML model utilizes image recognition technology to classify plant diseases based on input images provided by farmers. By analyzing the image, the model identifies the disease and explains why it has occurred in the plant. Additionally, the model provides personalized remedies to cure the plant, based on the disease identified and other environmental factors.

Detecting diseases early in crop plants is crucial for farmers to prevent crop damage and maximize production. By utilizing deep learning technology, we can identify plant diseases through the analysis of their leaves. Our website includes a fertilizer store that recommends specific fertilizers to farmers after detecting the disease through an API. This can help farmers make informed decisions about the treatment of their crops, leading to higher yields and profits.

Diagram

Description automatically generated

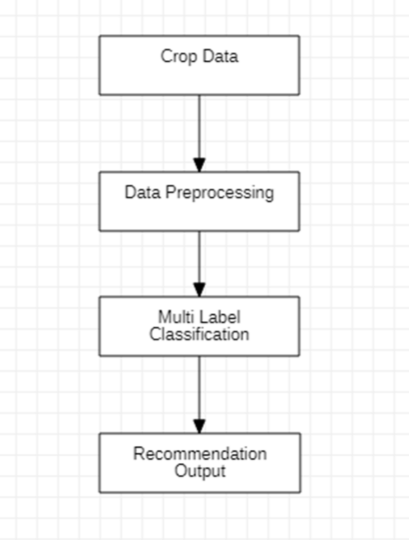
The web application comprises three modules, namely crop recommendation, fertilizer recommendation, and plant disease classification. The crop recommendation module utilizes a Random Forest machine learning algorithm to suggest suitable crops based on various parameters such as soil NPK levels, pH value, and rainfall in the area. The fertilizer recommendation module provides suggestions on improving soil fertility, while the plant disease classification module aids farmers in identifying diseases by analyzing input images of plant leaves.

Our website features a crop recommendation system based on the Random Forest machine learning algorithm. This system utilizes various parameters such as N, P, K, Ph levels of soil and rainfall values to provide farmers with recommendations on which crop is most suitable for their field.

**Background Details**

The study utilized the random forest algorithm to predict agricultural yield and determine product cost based on temporal data. The proposed system was evaluated using a static dataset and applied to a small area. By analyzing weather data, the system suggested appropriate crops for farmers to grow in specific soil and climate conditions.

Our web application offers a crop recommendation system that utilizes algorithms such as Random Forest, Naive Bayes, and K nearest neighbor to suggest suitable crops for the available soil. Precision farming, also known as digital farming, is becoming increasingly important in agriculture as it involves the use of hi-tech computer systems to calculate various parameters such as weed detection, crop prediction, yield detection, crop quality, and other machine learning farming techniques.



Crop data including parameters such as temperatures, humidity, and moisture content were collected for various crops. The system utilizes a Random Forest Classifier algorithm to recommend the appropriate crop based on the input parameters such as temperature, humidity, and location. The model achieved an accuracy of 90.5% in its predictions.

To ensure the highest accuracy of the application, we compared the performance of all the four algorithms that were implemented individually. After evaluating the results, we selected the algorithm that demonstrated the highest accuracy to be used in the model.

**System Design & Methodology**

**Crop Recommendation**: This module considers the soil composition and predicts the most suitable crop to be grown.

**Fertilizers Recommendation**: This module analyzes the soil content and suggests the type of fertilizers that are required to improve the fertility of the soil in your farm.

**Crop Disease Detection**: This module requires an image of an affected leaf and processes it to provide information about the type of crop disease and possible solutions to it.

Systems Architecture

**Diagram, schematic

Description automatically generated**

For Machine Learning and deep learning, we use algorithm as follow:

1) **KNN - (K Nearest Neighbors):** The K-NN algorithm is a non-parametric method used for predicting class membership. First, the algorithm identifies the k nearest neighbor for each incoming instance. Then, the instances are classified through a majority vote of the k neighbors. In the second step, the k neighbors are used to predict the label for the new instance based on the labeled sets. This approach is known for its ability to handle complex relationships between variables and produce accurate results.

2) **Random – Forest**: The Random-Forest method is an ensemble learning technique used for both classification and regression tasks. To utilize this algorithm for prediction, the test features must be passed through the rules of each randomly created tree. Each tree predicts a distinct target for the same test feature, and votes are computed based on each predicted target. The final prediction of the algorithm is then based on the highest voted target. One of the key advantages of using the Random-Forest algorithm is that it can handle missing values and prevent over-fitting of the model.

3) **Decision – Tree:** A decision tree is a type of machine learning algorithm that falls under the category of supervised learning. In this algorithm, the attributes and class labels are represented using a tree structure. The root attribute is compared with the attributes of the record and based on the comparison; a new node is created. This process of comparison is repeated until a leaf node with a predicted class value is reached. Decision trees are known for their efficiency in predicting outcomes and are commonly used in many fields such as finance, healthcare, and marketing.

4) **Neural – Network** Neural network systems use connected nodes called neurons to improve their performance by learning from examples. Signals are transmitted between neurons via connections, each of which has an associated weight that is updated and adjusted during the learning process. This allows neural networks to continually improve their performance as they process more data.

5) **Conventional Neural Network (CNN):** A Convolutional Neural Network, also known as a ConvNet or CNN, is a type of Deep Learning algorithm that can analyze input images by assigning significance to different objects or features within the image through learnable weights and biases. This process allows the network to distinguish and classify various objects within the image with high accuracy.

Objectives:

1. We study and understand Machine learning techniques.
2. We analyze and design Deep Learning Model
3. We implemented on Jupiter Notebook
4. We evaluate the performance by Appling various test cases.

Software Requirements:

1) Operating System - Windows 7 and above

3) Front End - HTML, CSS, JavaScript

4) Database - CSV File

5) IDE - Jupiter, Google colab, VS Code

6) Python Libraries - NumPy, pandas, matplotlib, seaborn

**Dataset**

In any Machine Learning System, data plays a crucial role. To achieve a good harvest, certain parameters such as temperature, humidity, soil pH, sunlight, and soil moisture must be met for healthy plant growth. However, these conditions may vary depending on the plant variety. To predict the crops, datasets from various sources such as government websites and Kaggle were utilized. The dataset used for crop recommendation includes 22 crops grown in India, and the parameters considered are nitrogen, phosphorus, potassium, rainfall, temperature, humidity, and pH. For crop disease classification, the dataset includes images of leaves from 14 plants with 26 types of images displaying a particular disease in a plant, excluding healthy leaves. Each plant disease type consists of 1800 images.

**Data Exploration**

Data exploration involves analyzing and understanding the relationships and patterns within a dataset. One common technique for identifying correlations between columns in a dataset is by using a heatmap to visualize the correlation matrix. The correlation matrix is calculated using statistical measures like Pearson Correlation and is then represented visually as a heatmap. This helps in identifying any significant correlations or patterns within the data and can be used to gain insights into the underlying factors influencing the dataset.

**Interface:**

Diagram

Description automatically generated

That sounds like a well-designed system architecture for a comprehensive agriculture-related application. The three modules cover important aspects of farming, such as selecting the right crop, optimizing soil fertility, and identifying plant diseases. This can greatly benefit farmers in making informed decisions and improving their yield.

**Implementation and Result**

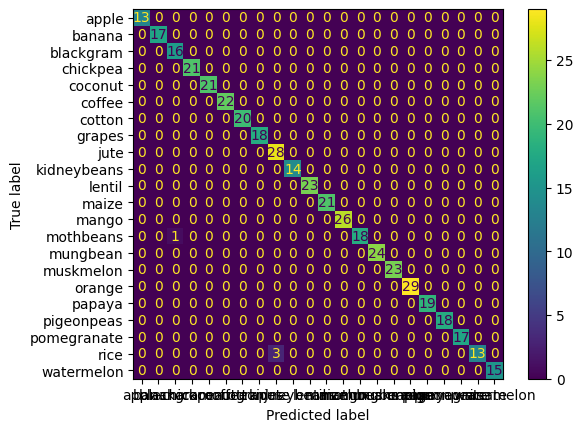
**Crop Recommendation System:**

Accuracy and Classification report

A picture containing text

Description automatically generated

Confusion Matrix:



Model comparison:

**Chart, bar chart

Description automatically generated**

**Fertilizer Recommendation System:**

Confusion matrix:

On test data:

A picture containing graphical user interface

Description automatically generated

Model Comparison:

Chart, bar chart

Description automatically generated

**Disease Detection:**

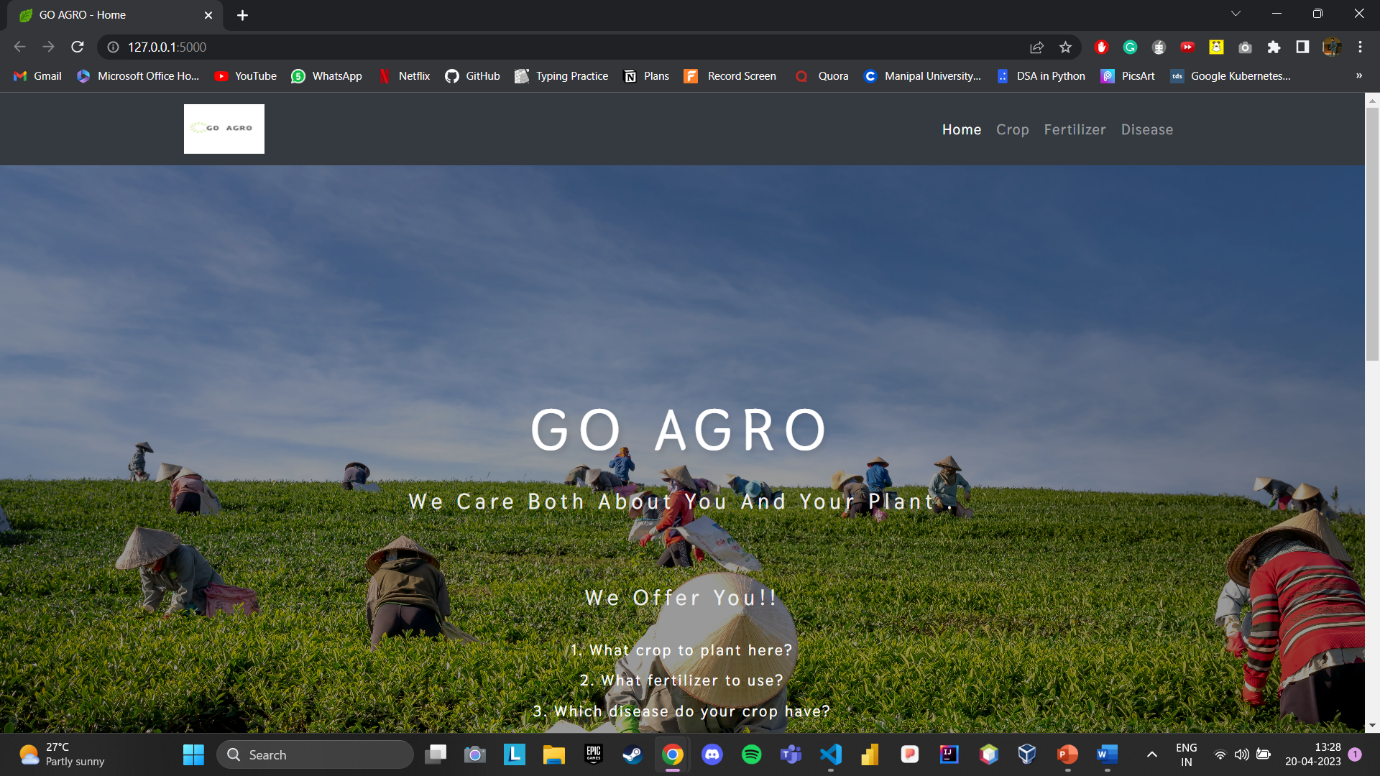
The accuracy of the model is 90.62%

Validation Accuracy and Loss:

Chart, line chart

Description automatically generated

APPLICATION UI:



Graphical user interface, website

Description automatically generated

**Conclusion**

That sounds like a great initiative to help the farmers in India. A proper recommendation system for crops and fertilizers can improve the productivity of the farmland and can contribute to the overall economy of the nation. In addition, the timely detection and treatment of plant diseases can prevent the spread of the disease to other plants and can save the crops from damage. With the use of advanced technologies like machine learning and data analysis, such systems can provide accurate recommendations and predictions. Further research and improvement of the dataset can help to enhance the yield prediction system.

This approach can help to create a comprehensive system for the farmers, which will not only help them with crop selection but also provide them with a complete solution for their agricultural needs. With the integration of yield prediction, the farmers can also predict the expected yield and plan their resources accordingly. Additionally, by including more data and pictures of infected plants, the accuracy of the disease detection system can also be improved, which will help the farmers to take timely action and prevent crop losses. This system can also be extended to include weather predictions and pest control recommendations to provide a complete agricultural solution to the farmers. Overall, the use of technology and analytics can revolutionize the agricultural sector and help the farmers to increase their income and contribute to the growth of the economy.

Our project offers a smart crop recommendation system and fertilizer recommendation system that is designed to be accessible and useful for farmers throughout India. With this system, farmers can make informed decisions about which crops to grow and how to best use fertilizers to maximize their yields. Additionally, we have developed a separate module for crop disease detection, which uses textural analysis of leaves to predict the type of disease affecting the crops. This approach provides farmers with a valuable tool for identifying and addressing potential crop diseases before they can cause significant damage.

We have also incorporated a supplementary system, referred to as Crop Disease Detection, which predicts the type of crop disease by analyzing the textural similarity of leaves.

The interface of the application is designed to be user-friendly, making it accessible to anyone.

**References**

The resources used for this project include:

•**Ensemble Methods** - a collection of machine learning techniques that use multiple models to improve performance. Link: <https://scikit-learn.org/stable/modules/ensemble>

•**ResNet-9** - a deep convolutional neural network architecture with nine layers and skip connections. Link: <https://www.researchgate.net/figure/ResNet-9-architecture-A-convolutional-neural-net-with-9-layers-and-skip-connections_fig1_363585139>

•**Recommendation System** - a system that provides personalized suggestions based on user preferences and behaviour. Link: <https://towardsdatascience.com/recommendation-systems-a-review-d4592b6caf4b>

•**Deployment** - the process of making a software application available to users. Link: <https://www.geeksforgeeks.org/deploy-python-flask-app-on-heroku/>

•**JetIR** - a journal of engineering and technology research. Link: <https://www.jetir.org/papers/JETIRAU06049>

•**Stackoverflow** - a community-driven question and answer website for programming-related topics. Link: <https://stackoverflow.com/>

•**Detection** - the process of identifying objects or patterns within an image or video. Link: <https://towardsdatascience.com/implementing-real-time-object-detection-system-using-pytorch-and-opencv-70bac41148f7>

•**Research Paper** - a scholarly article that presents original research or analysis. Link: <https://paperswithcode.com/task/real-time-object-detection>

•**Computer Vision** - a field of study focused on enabling machines to interpret and understand visual data. Link: <https://docs.opencv.org/4.x/>

•**Globalaveragepooling2d** - a layer in Keras that computes the average value of each feature map in a convolutional neural network. Link: <https://www.tensorflow.org/api_docs/python/tf/keras/layers/GlobalAveragePooling2D>

•**Pytorch** - an open-source machine learning library for Python. Link: <https://pytorch.org/>

•**Tensorflow & Keras** - machine learning libraries for Python that enable the creation and training of deep learning models. Link: <https://www.tensorflow.org/>