CROP RECOMMENDATION SYSTEM USING SUPPORT VECTOR CLASSSIFICATION

A PROJECT REPORT

Submitted by

MANO S (312418205046)

KARTHIKEYAN S (312418205037)

in partial fulfillment for the award of the degree

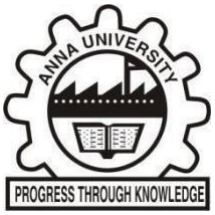
of

BACHELOR OF TECHNOLOGY

in

INFORMATION TECHNOLOGY

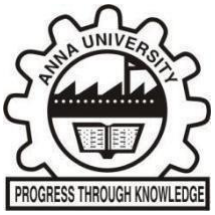
ST. JOSEPH'S INSTITTUE OF TECHNOLOGY



ANNA UNIVERSITY: CHENNAI 600 025

MAY 2022

ANNA UNIVERSITY: CHENNAI 600 025



BONAFIDE CERTIFICATE

Certified that this project report **"CROP RECOMMENDATION SYSTEM USING SUPPORT VECTOR CLASSIFICATION"** is the bonafide work of **MANO SRITHAR (312417205046) and KARTHIKEYAN S (312418205046)** who carried out the project work under my supervision, for the partial fulfilment of the requirements for the award of the degree of Bachelor of Technology in Information Technology. Submitted for the Viva-Voce held on _____.

SIGNATURE

Dr. S.KALARANI M.E.,Ph.D.,

HEAD OF THE DEPARTMENT

Department of Information Technology

St. Joseph's Institute of Technology

Old Mamallapuram Road

Chennai- 600119

SIGNATURE

Dr.M.K.KIRUBAKARAN M.E.,Ph.d.,

ASSISTANT PROFESSOR

Department of Information Technology

St. Joseph's Institute of Technology

Old Mamallapuram Road

Chennai- 600119

CERTIFICATE OF EVALUATION

College Name **:** St. Joseph's Institute of Technology

Branch & Semester **:** Information Technology (VIII)

S.NO	NAME OF STUDENT	TITLE OF THE PROJECT	NAME OF THE SUPERVISOR WITH DESIGNATION
1	**Karthikeyan**	Crop Recommendation System using Support Vector Classifier	Mr.Kirubakaran M.E.,Ph.D., Assistant Professor
2	**Mano Srithar**	Crop Recommendation System using Support Vector Classifier	Mr.Kirubakaran M.E.,Ph.D., Assistant Professor

The report of the project work submitted by the above students in partial fulfilment for the award of Bachelor of Technology degree in Information Technology of Anna University were evaluated and confirmed to be reports of the work done by the above students and then evaluated.

(HOD/PROJECTCOORIDINATOR) **(INTERNALEXAMINER)**

ACKNOWLEDGEMENT

The contentment and elation that accompany the successful completion any work would be incomplete without mentioning the people who made it possible.

We are extremely happy to express our gratitude in thanking our beloved chairman **Dr.B.Babu Manoharan M.A.,M.BA.,Ph.D.,** who has been a pillar of strength to this college.

Words are inadequate in offering my sincere thanks and gratitude to our respected Managing Director **Mrs.B.Jessie Priya M.Com.,** heartfelt gratitude to our respected Director **Mr.B.SashiSekar M.Sc.,** and our beloved Principal **Dr.P.Ravichandran M.Tech.,Ph.d.,** for having encouraged us to do our under graduation in Information Technology in this esteemed college.

We also express my sincere thanks and most heartfelt sense of gratitude to our eminent Head of the Department **Dr.S.Kalarani M.E.,Ph.D.,** for having extended her helping hand at all times.

It is with deep sense for gratitude that we acknowledge our indebtedness to our beloved supervisor **Mr.Kirubakaran M.E.,Ph.D.,** a perfectionist for her expert guidance and connoisseur suggestion.

Last but not the least, we thank our family members and friends who have been the greatest source of support to me.

LIST OF FIGURES

LIST OF TABLES

TABLE OF CONTENT

ABSTRACT

Agriculture and its allied sectors are undoubtedly the largest providers of livelihoods providers of livelihoods in rural India. Agriculture plays a vital role in the socioeconomic fabric of India. The agriculture sector is also a significant contributor factor to the country's Gross Domestic Product (GDP). Blessing to the country is the overwhelming size of the agricultural sector. However, regrettable is the yield per hectare of crops in comparison to international standards. This is one of the possible causes for a higher suicide rate among marginal farmers in India.

Failure of farmers to decide on the most-suited crop for the land using traditional and non-scientific methods is a serious issue for a country where approximately 58 percent of the population is involved in farming. Sometimes farmers were failed to choose the right crops based on the soil conditions, and season. This results in suicide, quitting the agriculture field, moving towards urban areas for livelihood. To overcome this issue, this research work has proposed a considering all the factors like N, P, K, and rainfall.

Furthermore, precision agriculture is being implemented with a modern agricultural technology and it is evolving in developing countries that concentrates on site-specific crop management. The proposed system provides a connectivity to farmers via a web application. The user provides the ph, rainfall, N, P, and K. Machine Learning algorithms allow choosing the most profitable crop for user provided input. To recommend the crop for cultivation system uses machine learning algorithms like random forest, decision tree, support vector classifier.

CHAPTER 1

INTRODUCTION

1.1 SYSTEM OVERVIEW

1.1.1 DATA ANALYSIS

Data analysis is a process of inspecting, cleansing, transforming, and modelling data, with the goal of discovering useful information, informing conclusions, and supporting decision-making. Data analysis has multiple facets and approaches, encompassing divers' techniques under a variety of names, and is used in different business, science, and social science domains. In today's business world, data analysis plays a role in making decisions more scientific and helping businesses operate more effectively. Data mining is a particular data analysis technique that focuses on statistical modelling and knowledge discovery for predictive rather than purely descriptive purposes, while business intelligence covers data analysis that relies heavily on aggregation, focusing mainly on businesses information. In statistical applications, data analysis can be divided into descriptive statistics, exploratory data analysis, and confirmatory data analysis. EDA (Exploratory Data Analysis) focuses on discovering new features in the data while CDA(Confirmatory Data Analysis) focuses on confirming or falsifying existing hypotheses. Predictive analytics focuses on the application of statisticalmodels for predictive forecasting or classification, while text analytics applies statistical, linguistic, and structural techniques to extract and classify information from textual sources, a species of unstructured data. All above are varieties of data analysis. Data analysis is a process for obtaining raw data, and subsequently converting it into information useful for decision-making by users.

1.1.2 SCIKIT-LEARN

Scikit-learn is a free software machine learning library for the python programming language. It features various classification regression and clustering algorithms

including support-vector machines; random forests, gradient boosting, k-vector machines, random forests, gradient boosting, k-means and clustering algorithms including support-vector machines, random forests, gradient boosting, k-means and is designed to interoperate with the python numerical and scientific libraries Numpy and Scipy.

1.1.3 CLASSIFICATION ANALYSIS

Classification analysis is a data analysis task within data mining that identifies and assigns categories to a collection of data to allow for more accurate analysis. The classification method makes use of mathematical techniques such as decision trees, linear programming, neural network, and statistics. Classification is the process of predicting the class of given data pints. Classes are sometimes calledas targets/labels or categories. Classification predictive modeling is the task of approximating a mapping function from input variables to discrete output variables.

Classification analysis can be used to question, decide, or predict behavior through the use of an algorithm. It works by developing a set of training data which contains a certain set of attributes as well as the likely outcome. The jobof the classification algorithm is to discover how that set of attributes reaches its conclusion. There are two steps in the construction of a classification model. Learning Step and Classification Step. Learning step is where different algorithms are used to build a classifier by making the model learn using the training set available. The model has to be trained for the prediction of accurate results.

Classification Step is where the model used to predict class labels, tests the constructed model on test data. Which in turn estimates the accuracy of the classification rules. Some scenarios where classification is used includespredicting the weather and analyzing health conditions

1.1.3 FLASK

Flask is a micro web framework written in python it is classified as a microframework because it does not require tools or libraries. It has no database abstraction layer, form validation, or any other components where pre- existing third-party libraries provide common functions. However, Flask supports extensions that can add application feature as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, and upload handling, various open authentication technologies and several common framework related tools summary. In contrast, abstractive methods build an internal semantic representation and then use natural language generation techniques to create a summary that is closer to what a human might express. Such a summary might include verbal innovations.

1.2 SCOPE OF THE PROJECT

In this project system are going to recommend the crop for cultivation based the characteristics of lands like potassium, nitrogen and rainfall. The project as a web application which everyone can access and instantly get their recommendation for their field. As a result, farmers can get the maximum amount yield because of choosing the right crop for production. The model has been deployed as a web application, anyone needs help in the selecting the crop for cultivation can use this application and select the most suitable crop for cultivation based on the input their given and past agricultural changes that has been analyzed by the machine learning model. The web application will produce the result within few seconds. The user gives input into the application the data has been send to machine learning model via the web application framework which connect the pickle file which contain the ml model and sends back the prediction, this prediction has been converted into recommendation based on the condition that has been given in the web application. Web application will reroute to another web page which most suited for the result of the crop recommendation system.

CHAPTER 2

LITERATURE SURVEY

2.1 Crop Prediction Based on Characteristics of the Agricultural Environment Using Various Feature Selection Techniques and Classifiers (S. P. Raja; Barbara Sawicka; Zoran Stankovic; and G.Mariammal, IEEE Access, volume:10,2022)

Agriculture is a growing field of research. Crop prediction in agriculture is critical and is chiefly contingent upon soil and environment conditions, including rainfall, nitrogen, potassium. In the past, farmers were able to decide on the crop to be cultivated, monitor its growth, and determine when it could be harvested. Today, however, rapid changes in environmental conditions have made it difficult for the farming community to continue to do so. Consequently, in recent years, machine learning techniques have taken over the task of prediction, and this work has used several of these to determine crop yield. To ensure that a given machine learning (ML) model works at a high level of precision, it is imperative to employ efficient feature selection methods to preprocess the raw data into an easily computable Machine Learning friendly dataset. To reduce redundancies and make the ML model more accurate, only data features that have a significant degree of relevance in determining the final output of the model must be employed. Thus, optimal feature selection arises to ensure that only the most relevant features are accepted as a part of the model. Conglomerating every single feature from raw data without checking for their role in the process of making the model will unnecessarily complicate the model. Furthermore, additional features which contribute little to the ML model will increase its time and space complexity and affect the accuracy of the model's output. The result depict that an ensemble technique offers better prediction accuracy than the existing classification technique.

2.2 Machine Learning Application for Precision Agriculture (Abhinav Sharma; Arpit Jain; Prateek Gupta; Vinay Chowdary, IEEE Access, volume:9,2021)

Agriculture plays a vital role in the economic growth of any country. With the increase of population, frequent changes in climatic conditions and limited resources, it becomes a challenging task to fulfil the food requirement of the present population. Precision agriculture also known as smart farming have emerged as an innovative tool to address current challenges in agricultural sustainability. The mechanism that drives this cutting-edge technology is machine learning (ML). It gives the machine ability to learn without being explicitly programmed. ML together with IoT (Internet of Things) enabled farm machinery are key components of the next agriculture revolution. In this article, authors present a systematic review of ML applications in the field of agriculture. The areas that are focused are prediction of soil parameters such as organic carbon and moisture content, crop yield prediction, disease and weed detection in crops and species detection. ML with computer vision is reviewed for the classification of a different set of crop images in order to monitor the crop quality and yield assessment. This approach can be integrated for enhanced livestock production by predicting fertility patterns, diagnosing eating disorders, cattle behavior based on ML models using data collected by collar sensors, etc. Intelligent irrigation which includes drip irrigation and intelligent harvesting techniques are also reviewed that reduces human labor to a great extent. This article demonstrates how knowledge-based agriculture can improve the sustainable productivity and quality of the product.

2.3 Cascading Feature Filtering and Boosting Algorithm for Plant Type Classification Based on Image (Adel Bakshipour; IEEE Access, Volume:9,2021)

Crop and weeds identification is of important steps towards the development of efficient automotive weed control systems. The higher the accuracy of plant detection and classification, the higher the performance of the weeding machine. In this study, the

capability of two popular boosting methods including Adaboost.M1 and LogitBoost algorithms was evaluated to enhance the plant classification performance of four classifiers, namely Multi-Layer Perceptron (MLP), k-Nearest Neighbors (kNN), Random Forest (RF), and Support Vector Machine (SVM). Four feature filtering techniques including Correlation-based Feature Selection (CFS), Information Gain (IG), Gain Ratio (GR), and OneR were applied to the image-extracted features and 10 of the most significant features were selected and fed into single and boosted classifiers. The RF model trained by IG selected features (IG-RF) was the most appropriate classifier among the evaluated models whether in single or boosted modes. It was also found that boosting of IG-RF by using Adaboost.M1 and LogitBoost algorithms improved the classification accuracy. Regarding the performance values, the LogitBoost-IG-RF structure, which provided a classification accuracy of 99.58%, a kappa (k) of 0.9948, and a Root Mean Squared Error (RMSE) of 0.0688 on training dataset, was selected as the most appropriate classifier for plant discrimination in peanut fields. The accuracy, k, and RMSE criteria of this combination on test dataset were 95.00%, 0.9375, and 0.1591, respectively. It was concluded that combination of boosting algorithms and feature selection methods can promote plant type discrimination accuracy, which is a crucial factor in the development of precision weed control systems.

2.4 Soil Sensors-Based Prediction System for Plant Diseases Using Exploratory Data Analysis and Machine Learning (Manish Kumar; Ahlad Kumar; Vin S.Palaparthy, IEEE Sensors Journal, volume 21, Issue:16,2021)

In this paper various prediction approaches, based on conventional multiple regression (REG), generalized regression neural network (GRNN), support vector machine (SVM) has been used in building prediction models for plant diseases. Plant disease identification, based on GRNNs and probabilistic neural networks have been

used as the classifiers to identify wheat and grape diseases. Based on high-resolution multispectral stereo images, K-nearest neighbor classifiers (KNN) have been employed

for pixel-wise classification for automatic classification of leaf diseases. Models, like a multi-layered perceptron model (MLP), are also used in plant disease recognition using various techniques involve multivariate liner regression (MLR) and partial least square regression (PLSR), which use hyperspectral data to estimate the severity of plant disease.

2.5 Ensemble Classification and IoT-Based Pattern Recognition for Crop Disease Monitoring System (Gayathri Nagasubramanian; Rakesh Kumar Sakthivel; Rizwan Patan; Muthuramalingam Sankaya; Mahnoud Daneshmand; Amir H.Gandomi, IEEE Internet of Things Journal, Volume:8, Issue:16,2021)

Internet of Things in the agriculture field provides crops-oriented data sharing and automatic farming solutions under single network coverage. The components of IoT collect the observable data from different plants at different points. The data gathered through IoT components such as sensors and cameras can be used to be manipulated for a better farming oriented decision-making process. This work proposes a system that observes the crops growth and leaf diseases continuously for advising farmers in need. To provide analytical statistical on plan growth and disease patterns, the proposed framework uses machine learning techniques such as support vector machine and convolutional neural network. This framework produces efficient crop condition notifications to terminal IoT components which are assisting in irrigation, nutrition planning and environmental compliance related to the farming lands, in this regard this work proposes Ensemble Classification and Pattern Recognition for Crop Monitoring System (ECPRC) to identify plant diseases at the early stages. The Proposed ECPRC uses ensemble Nonlinear Support Vector Machine (ENSVM) for detecting leaf and crop diseases.

2.6 Weather Based Crop Prediction in India using Big Data Analytics (Rishi Gupta, Akhilesh Kumar Sharma, Krishna mode, Shareen Kasim, Zirawani Baharum, IEEE Access, Volume 9,2021)

This paper aims at collecting and analyzing temperature, rainfall, soil, seed, crop production, humidity and wind speed data (in a few regions), which will help the farmers improve the produce of their crops. Firstly, data pre-process the data in a Python environment and then apply the MapReduce framework, which further analyses and processes the large volume of data. Secondly, k-means clustering is employed on results gained from MapReduce and provides a mean result on the data in terms of accuracy. After that, the use bar graphs and scatter plots to study the relationship between the crop, rainfall, temperature, soil, and seed type of two regions (Ahmednagar, Maharashtra and, Andaman and Nicobar Islands). Further, a self-designed recommender system has been used to predict the crops and display them on a Graphic User Interface designed in a Flask environment. The system design is scalable and can be used to find the recommended crops of other states in a similar manner in the future.

CHAPTER 3

SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

The existing system has uses different machine learning algorithms for the factor they used to get the recommendation, for productivity-based recommendation they are using ensemble technique, they use k-means algorithm and clustering analysis in productivity and season-based recommendation. They only use certain parameters to make a recommendation. In their system they only use parameters to get recommendation. They use decision tree, support vector machine, logistic regression algorithms based on the input dataset. This machine learning algorithms are hyper tuned to the dataset they use for recommendation of crop; this leads to improper hyperparameter turning.

3.1.1 DISADVANTAGES

- Unable to fulfill the factors that provide high yield of crop
- It provides lower accuracy in real time scenario
- This system provides the biased results of the parameters they use
- In larger perspective it also led to rise in food prices, starvation, crippling of economy

3.2 PROPOSED SYSTEM

The proposed system is implemented using machine learning. In the proposed system the environmental parameters such as, temperature, and geographical location in terms of the state along with characteristics such as soil type, pH value, and nutrients concentration all the factors are being considered to recommend a suitable crop to the user. In this system all the agricultural factors are cumulative considered for recommendation, in this system use support vector classifier algorithm. In support vector classifier hyperparameter tunning has been done for algorithms which considers all factors to make recommendation.

3.2.1 ADVANTAGES

- Recommendation of most suitable crop
- Eliminates the biased result for certain factors
- More accurate result

3.3 SYSTEM REQUIREMENTS

3.3.1 SOFTWARE REQUIREMENTS

REQUIREMENT	SPECIFICATION
Operating System	Linux/Mac OS/ Windows7+
System Type	64 Bit OS
Tool	Jupyter, Anaconda, Spyder
Coding Language	Python

TABLE 3.1- SOFTWARE REQUIREMENTS

3.3.2 HARDWARE REQUIREMENTS

REQUIREMENT	SPECIFICATION
Processor	Intel® Core™ i3 or AMD
Clock Speed	3.40 GHz
RAM	4 GB

TABLE 3.2- HARDWARE REQUIREMENT

3.4 LANGUAGE SPECIFICATION

3.4.1 PYTHON

Python is a general-purpose interpreted, interactive, object-oriented scripting and high-level programming language. Python is designed to be highly readable. It uses English keywords frequently whereas other languages use punctuation and it has fewer syntactical constructions than other languages. Its design philosophy emphasizes code readability, and its syntax allows programmers to express concepts in fewer lines of code than would be possible in languages such as c++ or java. The language provides constructs intended to enable clear programs on both a small and large scale.

Python supports multiple programming paradigms, including object-oriented, imperative and functional programming or procedural styles. It features a dynamic type system and automatic memory management and has a large and comprehensive standard library. Python interpreters are available for installations on many operating systems, allowing python code execution on a wide variety of systems.

Python is Interpreted − Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.

Python is Interactive − You can actually sit at a Python prompt and interact with the interpreter directly to write your programs.

Python is Object-Oriented − Python supports Object-Oriented style or technique of programming that encapsulates code within objects.

Python is a Beginner's Language − Python is a great language for the beginnerlevel programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games.

3.4.2 HISTORY OF PYTHON

Python was created by Guido van Rossum who is a Dutch programmer. He was also known as the "Benevolent dictator for life" (BDFL) for python, until he stepped down from the position in July 2018. He Worked at the Centrum Wiskunde & Informatica (CWI) and contributed a globe () routine to BSD Unix during 1986 and helped develop the ABC programming language. But later during December 1989, he was looking for a hobby project around the time in Christmas and started working on a new interpreter for a new language. He later named this language as Python after choosing the word form a TV serial named Monty Python's Flying Circus. He admits that ABC language is the predecessor of Python language. The first release of the language was done in 1991.

3.4.3 FEATURES OF PYTHON

Python is one of the most widely used language over the web. Some of the features are

- **Easy-to-learn** − Python has few keywords, simple structure, and a clearly defined syntax. This allows the student to pick up the language quickly.
- **Easy-to-read** − Python code is more clearly defined and visible to the eyes.
- **Easy-to-maintain** − Python's source code is fairly easy-to-maintained.
- **A broad standard library** − Python's bulk of the library is very portable and cross-platform compatible on UNIX, Windows, and Macintosh.
- **Interactive Mode** − Python has support for an interactive mode which allows interactive testing and debugging of snippets of code.
- **Portable** − Python can run on a wide variety of hardware platforms and has the same interface on all platforms.
- **Extendable** – It can add low-level modules to the Python interpreter. These modules enable programmers to add to or customize their tools to be more efficient.
- **Databases** − Python provides interfaces to all major commercial databases.

- **GUI Programming** − Python supports GUI applications that can be created and ported to many system calls, libraries and windows systems, such as Windows MFC, Macintosh, and the X Window system of Unix.
- **Scalable** − Python provides a better structure and support for large programs than shell scripting.

3.4.4 LIBRARIES USED

- Numpy
- Pandas
- Scikit-learn

Numpy

Numpy is a Python library used for working with arrays. Numpy y stands for Numerical Python. It also has functions for working in domain of linear algebra, fourier transform, and matrices. In Python we have lists that serve the purpose of arrays, but they are slow to process. NumPy aims to provide an array object that is up to 50x faster than traditional Python lists.

Installation command: $ pip install numpy

Pandas

Pandas is a software library written for the python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series. It is free software released under the three-clause BSD license.

Installation command: $ pip install pandas

Scikit-learn

Scikit-learn is a free software machine learning library for the python programming language. It features various classification, regression and clustering

algorithms including support vector, random forests, gradient boosting, k-means and DBSCAN, and is designed to interoperate with the python numerical and scientific libraries Numpy and Scipy.

Installation command: $ pip install Scikit-learn

3.5 Git

Git is a distributed version-control system for tracking changes in source code during software development. It is designed for coordinating work among programmers, but it can be used to track changes in any set of files. Its goals include speed, data integrity, and support for distributed, non-linear workflows.

Git was created by Linus Torvalds in 2005 for development of the Linux kernel, with other kernel developers contributing to its initial development. Its current maintainer since 2005 is Junio Hamano.

As with most other distributed version-control systems, and unlike most client–server systems, every Git directory on every computer is a full-fledged repository with complete history and full version-tracking abilities, independent of network access or a central server. Git is free and open-source software distributed under the terms of the GNU General Public License version 2

Git will be integrated with deploying platforms to make the deployment easier to deploy updates in Application can done by updating the git version, updated git version will automatically updated in deployment platform, and also reflected in the Applicaiton

CHAPTER 4

SYSTEM DESIGN

4.1 SYSTEM ARCHITECTURE DIAGRAM

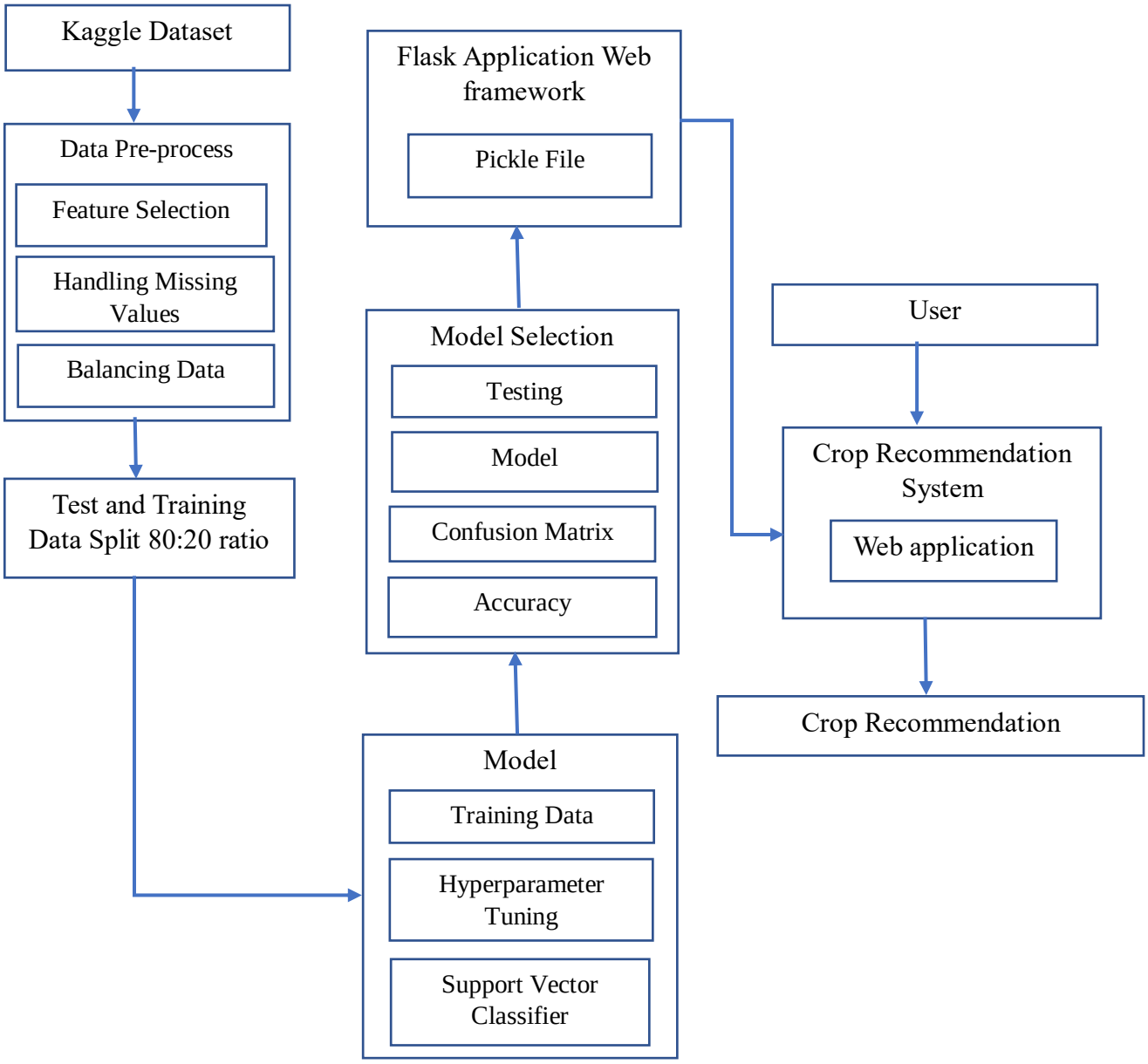


FIG 4.1 ARCHITECTURE DIAGRAM

4.1.1 ARCHITECTURE DESCRIPTION

In Fig 4.1 Consider a crop recommendation system data has been collected from the agricultural website and also used kaggle data. The dataset contains wide range of inputs for recommending the crop for cultivation, that contains crop duration, temperature required for a crop potassium, nitrogen, ph, rainfall amount of water need, water source that most suits for this kind of crop. Collected data has preprocessing before analyzing, it can refer to manipulation or dropping of data before it is used in order to ensure or enhance performance. Exploratory Data Analysis is a next step of data preprocessing, exploratory data analysis is an important machine learning step that involves learning about the data without spending too much time or getting lost in it. Exploratory Data Analysis consist of renaming the columns, finding data types of columns, missing values treatment, dropping the duplicate rows, variable transformation, dropping the missing or null values. It also uses visualizations techniques like heat map to find relevance of independent fields and dependent fields, in missing value treatment used is mean, mode for different fields. System encodes the categorical values using one hot encodingmethod for using data in system, it also includes checking whether the data is balanced or not, the unbalanced data reduces accuracy of a system. Algorithm selection the dataset has been split into a particular ratio as training and testing data. Training data has been used to train the machine learning model, after training the model using machine learning algorithm has testing using the test data to validate the model using the accuracy score of confusion matrix, if the accuracy of model is low different algorithms is used based on the EDA, different machine algorithms examined to get best accuracy. In state of getting best accuracy there also a possibility of model getting overfitted. The most suitable algorithm has been built into model and that model has been stored into a pickle file for system to be used when recommendation of crop needs. The web application has been created using a web framework flask, web application contains code for getting user input and sending it to the model and getting result from model by using a pickle file. The web application has been deployed in an online platform for user.

CHAPTER 5

SYSTEM IMPLEMENTATION

5.1 LIST OF MODULES

5.1.1 Data collection

5.1.2 Exploratory data analysis

5.1.3 Data modeling

5.1.4 Deployment

5.1.1 DATA COLLECTION

Data is important part of this project; all the recommendations are made by the model that uses dataset for the training. Dataset must be more accurate to make a good recommendation, as data base part of recommendation dataset most not biased. Dataset must not be an old because every decade the climate conditions has been changing and agriculture cycle also change, data must be in a current to get best result for user. Dataset from Kaggle is used in the model building.

In [3]:
```
import pandas as pd
import numpy as np
import seaborn as sns
```

In [4]:
```
dataset=pd.read_csv('C:/Users/karthikeyan/Desktop/dataset/Crop_recommendation.csv')
```

In [5]:
```
dataset.head()
```

Out[5]:

	N	P	K	temperature	humidity	ph	rainfall	label
0	90	42	43	20.879744	82.002744	6.502985	202.935536	rice
1	85	58	41	21.770462	80.319644	7.038096	226.655537	rice
2	60	55	44	23.004459	82.320763	7.840207	263.964248	rice
3	74	35	40	26.491096	80.158363	6.980401	242.864034	rice
4	78	42	42	20.130175	81.604873	7.628473	262.717340	rice

In [6]:
```
dataset.shape
```

Out[6]:
```
(2200, 8)
```

FIGURE 5.1 SAMPLE DATASET

5.1.2 EXPLORATORY DATA ANALYSIS

This step includes getting some concept about the answer and elements affecting it, earlier than constructing the real model. Distribution of data inside distinctive variables of a character are explored graphically the usage of bar-graphs, Relations between distinct aspects are captured via graphical representations like scatter plots and warmth maps. Many data visualization strategies are considerably used to discover each and every characteristic individually and by means of combining them with different features. Exploratory data analysis is an approach to analyze the data using visual techniques. Heat maps, Distribution graphs are used in visualization. It is used to discover trends, patterns, orto check assumptions with the help of statistical summary and graphical representations.

In this process handling all the data manipulation, which contains treatment of missing values, handling imbalanced dataset by drop the rows, if dataset contains huge number of rows or adding extra row for low amount of data. Missing values has been treated by dropping the records, replacing the null values with minimum, maximum or median values. It finds pattern in the data. Visualizations like heat maps, bar charts, scatter plots, heat maps are used to find the significance between the fields, and bar charts are used to find distribution of data. It helps choosing the right machine learning algorithm.

Data that has been split into training and testing data after performing exploratory data analysis. Data has been split into a particular ratio as testing and training data as 75:25 respectively.

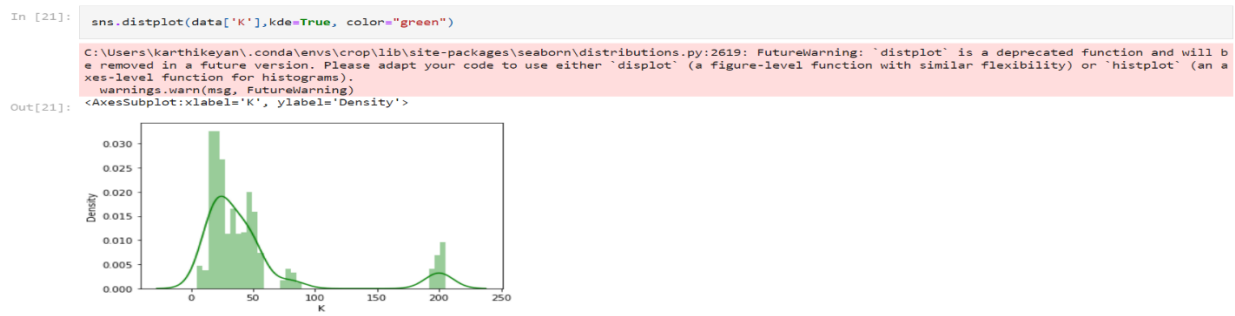


FIGURE 5.2 POTASSIUM DISTRIUBTION VISUALIZATION

```
In [11]:
corr = dataset.corr()
print(corr)
sns.heatmap(corr,
            xticklabels=corr.columns,
            yticklabels=corr.columns)
```

```
                    N         P         K  temperature  humidity        ph  \
N            1.000000 -0.231460 -0.140512     0.026504  0.190688  0.096683
P           -0.231460  1.000000  0.736232    -0.127541 -0.118734 -0.138019
K           -0.140512  0.736232  1.000000    -0.160387  0.190859 -0.169503
temperature  0.026504 -0.127541 -0.160387     1.000000  0.205320 -0.017795
humidity     0.190688 -0.118734  0.190859     0.205320  1.000000 -0.008483
ph           0.096683 -0.138019 -0.169503    -0.017795 -0.008483  1.000000
rainfall     0.059020 -0.063839 -0.053461    -0.030084  0.094423 -0.109069

             rainfall
N            0.059020
P           -0.063839
K           -0.053461
temperature -0.030084
humidity     0.094423
ph          -0.109069
rainfall     1.000000
<AxesSubplot:>
```

Out[11]:

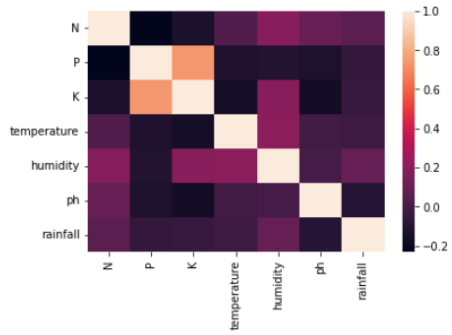


FIGURE 5.3 HEATMAP

5.1.3 DATA MODELING

Data modeling is the coronary heart of data analysis. A model takes the organized data as input and gives the preferred output. This step consists of selecting the suitable kind of model whether the problem is a classification problem or a regression problem or a clustering problem. After deciding on the model family, amongst the number of algorithms amongst that family need to cautiously pick out the algorithms to put into effect and enforce them. System used Random Forest. Algorithm for recommendation System. System needs to tune the hyper parameters of every model to obtain the preferred performance. System additionally need to make positive there is the right stability between overallperformance and generalizability. Confusion matrix has been used to find the accuracy of model, if the accuracy of the model is low hyper parameter tuning has been done to get maximum result, if the accuracy remains low the machine learning algorithm changed based on the exploratory data analysis.

In [63]:

```
from sklearn.svm import SVC
svm_model_linear = SVC(kernel = 'linear', C = 1).fit(x_train, y_train)
svm_predictions = svm_model_linear.predict(x_test)

# model accuracy for X_test
accuracy = svm_model_linear.score(x_test, y_test)
print(accuracy)

# creating a confusion matrix
cm = confusion_matrix(y_test, svm_predictions)
print(cm)
```

```
0.975
```

FIGURE 5.4 SUPPORT VECTOR CLASSIFIER ALGORITHM ACCURACY

5.1.4 DEPLOYMENT

The user accesses the crop recommendation system using the web application. Recommendation system webpage has been created using HTML and CSS for front end and flask used as a web framework of webpage. Webpage for crop recommendation has fields for potassium, nitrogen, rainfall and other factors for prediction. The inputs given by the user has been sent to model through the web API contains the file that has pickle file contain model for recommendation. The web application has been deployed in the streamlit platform. The deployment of web application consists of pickle file, webpage, and FLASK web application framework for the web application development. Deployment plays a vital role in a project any user from anywhere can get the benefit of the web application for crop recommendation system. As the webapplication is deployed it will be very useful to user to get recommendation in any place.

CHAPTER 6

SAMPLE CODING

```
import pandas as pd
import numpy as np
import seaborn as sns
dataset=pd.read_csv('C:/Users/karthikeyan/Desktop/dataset/Crop_recommendation.csv
')
dataset.head()
print("Number of Instances and Attributes:", df.shape)
print('Dataset columns:',df.columns)
print('Data types of each columns: ', df.info())dataset.count()
dataset.isnull().sum()
count_dups = df.duplicated().sum()
print("Number of Duplicates: ", count_dups)
    if count_dups >= 1:
        df.drop_duplicates(inplace=True)
        print('Duplicate values removed!')
    else:
        print('No Duplicate values')
import matplotlib.pyplot as plt
feature_columns=[dataset['N'],dataset['P'],dataset['K'],dataset['temperature']]
fig = plt.figure(1, figsize=(10,8))
ax = fig.add_subplot(111)
Ps = ax.boxplot(feature_columns)
```

```
corr = dataset.corr()
print(corr)
sns.heatmap(corr,
        xticklabels=corr.columns,
        yticklabels=corr.columns)
"Age Distribution barchart(figure_3)"
bins = [100,150,200,250]
plt.hist(dataset['rainfall'], bins, histtype='bar', rwidth=0.5)
plt.title('Age Distribution')
plt.xlabel('Age')
plt.ylabel('No of Persons')
plt.show()
dataset.tail()
dataset.reindex()
shuffled_dataset = dataset.sample(frac=1, random_state=1).reset_index()
shuffled_dataset.head()
data=shuffled_dataset.drop(['index'], axis = 1)
data.head()
shuffled_dataset.head()
sns.distplot(data['N'], kde=True)
sns.distplot(data['K'],kde=True, color="green")
sns.distplot(data.P,kde=True, color="red")
```

```
data.head()


from sklearn.model_selection import train_test_split
features_columns=['N', 'P', 'K', 'temperature', 'humidity', 'ph', 'rainfall']
predicted_class=['label']
x=data[features_columns].values
y=data[predicted_class].values
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20,random_state=0)
print(x_train.shape,y_test.shape)


from sklearn.linear_model import LogisticRegression
model=LogisticRegression(max_iter=500)
model.fit(x_train,y_train)
y_pred=model.predict(x_test)
from sklearn.metrics import accuracy_score
print(accuracy_score(y_test,y_pred))
from sklearn.metrics import confusion_matrix
confusion_matrix(y_test, y_pred)
print(cm)


from sklearn.neighbors import KNeighborsClassifier
knn = KNeighborsClassifier(n_neighbors = 7).fit(x_train, y_train)
```

```
# accuracy on X_test
accuracy = knn.score(x_test, y_test)
print (accuracy)

# creating a confusion matrix
knn_predictions = knn.predict(x_test)
cm = confusion_matrix(y_test, knn_predictions)
print(cm)
from sklearn.naive_bayes import GaussianNB
gnb = GaussianNB().fit(x_train, y_train)
gnb_predictions = gnb.predict(x_test)

# accuracy on X_test
accuracy = gnb.score(x_test, y_test)
print(accuracy)

# creating a confusion matrix
cm = confusion_matrix(y_test, gnb_predictions)
print(cm)
from sklearn.svm import SVC
svm_model_linear = SVC(kernel = 'linear', C = 1).fit(x_train, y_train)
```

```
svm_predictions = svm_model_linear.predict(x_test)

# model accuracy for X_test
accuracy = svm_model_linear.score(x_test, y_test)
print(accuracy)

# creating a confusion matrix
cm = confusion_matrix(y_test, svm_predictions)
print(cm)
from sklearn.neighbors import KNeighborsClassifier
knn = KNeighborsClassifier(n_neighbors = 7).fit(x_train, y_train)

# accuracy on X_test
accuracy = knn.score(x_test, y_test)
print(accuracy)

# creating a confusion matrix
knn_predictions = knn.predict(x_test)
cm = confusion_matrix(y_test, knn_predictions)
print(cm)
from sklearn.ensemble import RandomForestClassifier
```

```
classifier = RandomForestClassifier(n_estimators = 10, criterion = 'entropy',
random_state = 0)

classifier.fit(x_train, y_train.ravel())

y_pred=classifier.predict(x_test)

print("RandomForest:{0}".format(accuracy_score(y_test,y_pred)))

import pickle

"saving model as a disk"

pickle.dump(svm_model_linear,open('prediction.pkl','wb'))

"loding model to compare the results"

prediction =pickle.load(open('prediction.pkl','rb'))

print(svm_model_linear.predict([[25,129,195,17,81,5,72]]))


import streamlit as st

import pandas as pd

import numpy as np

import os

import pickle

import warnings

st.beta_set_page_config(page_title=”Crop Recommender”, page_icon=” ”,
layout=’centered’, initial_sidebar_state=”collapsed”)

def load_model(modelfile):

        loaded_model=pickle.load(open(modelfile, ‘rb’))
```

```
    return loaded_model
def main():
    html_temp= “””
    <div>
    <h1 styple=”color:MEDIUMSEAGREEN; text-align:left;”>Crop
    Recommendation <h1>
    </div>
    “””
    St.markdown(html_temp, unsafe_allow_html=True)
    st.subheader(" Find out the most suitable crop to grow in your farm 👦🌾")
    N = st.number_input("Nitrogen", 1,10000)
    P = st.number_input("Phosporus", 1,10000)
    K = st.number_input("Potassium", 1,10000)
    temp = st.number_input("Temperature",0.0,100000.0)
    humidity = st.number_input("Humidity in %", 0.0,100000.0)
    ph = st.number_input("Ph", 0.0,100000.0)
    rainfall = st.number_input("Rainfall in mm",0.0,100000.0)
    feature_list = [N, P, K, temp, humidity, ph, rainfall]
    single_pred = np.array(feature_list).reshape(1,-1)
    if st.button('Predict'):
        loaded_model = load_model('prediction.pkl')
        prediction = loaded_model.predict(single_pred)
```

```
                col1.write('''
                ## Results 🔍
                ''')
            col1.success(f"{prediction.item().title()} are recommended crop for your farm.")
    hide_menu_style = """
    <style>
    #MainMenu {visibility: hidden;}
    </style>
    """
hide_menu_style = """
        <style>
        #MainMenu {visibility: hidden;}
        </style>
        """
st.markdown(hide_menu_style, unsafe_allow_html=True)


if __name__ == '__main__':
        main()
```

CHAPTER 7

SCREENSHOTS

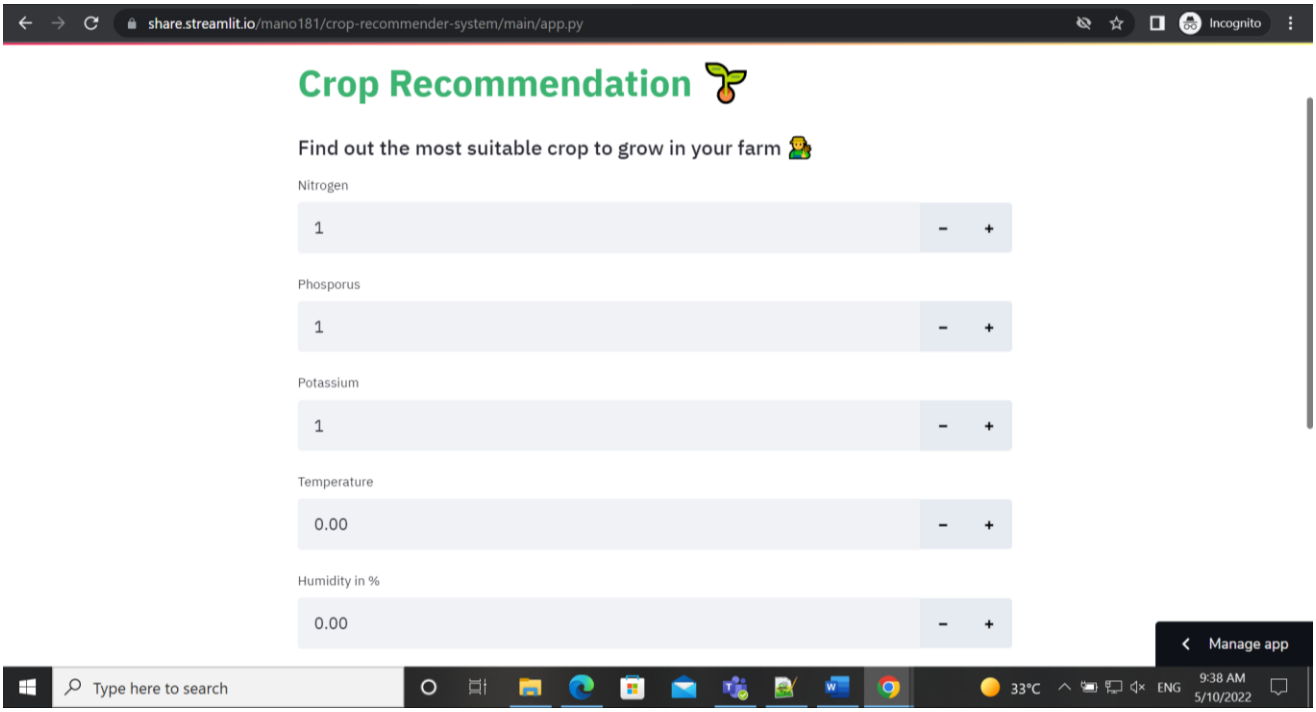


FIGURE 7.1 APPLICATION INPUT IMAGE1

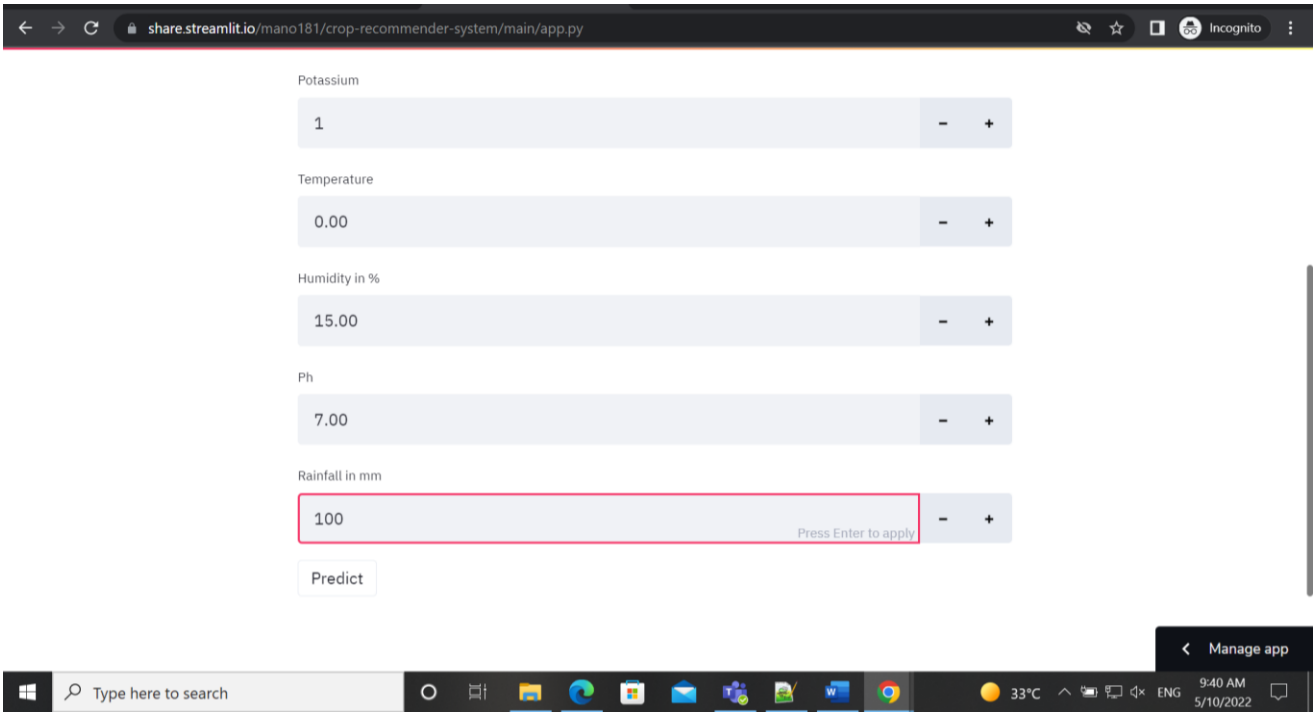


FIGURE 7.2 APPLICATION INPUT IMAGE2

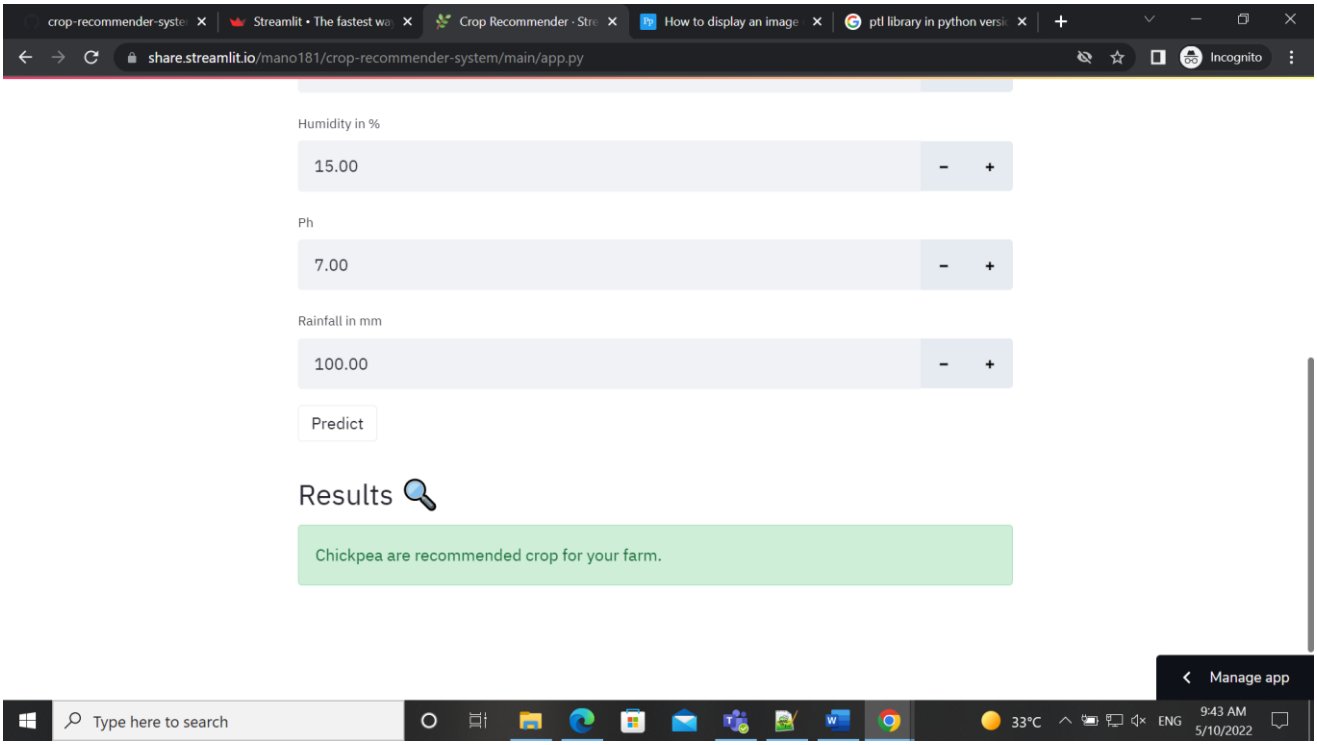


FIGURE 7.3 APPLICATION RESULT IMAGE

CHAPTER 8

CONCLUSION

The emerging trends and the availability of intelligent technologies makes us to develop new models that help to satisfy the needs of emerging world. So, we have developed a crop recommendation system which can possibly contribute to public agriculture. The model proposes an efficient application based on machine learning algorithm to recommend the crop for farming or cultivated the suitable crop for their land. Major characteristics for crop recommendation have been considered build an model for recommendation. The project also contains a web application which is deployed and available to all at any instance. It can recommend suitable crop for your land within few seconds of processing. The model uses support vector classifier for recommendation of crop. The recommendation system works very effectively and efficiently in identifying the most suitable crop for cultivation according to the inputs given by the user.

REFERENCES

[1] S.P. Raja, Barbara Sawicka, Zoran Stankovic, and G.Mariammal "Crop Prediction Based on Characteristics of the Agricultural Environment Using Various Feature Selection Techniques and Classifiers", IEEE Access Journal, Volume:10, 2022.

[2] Abhinav Sharma, Arpit Jain, Prateek Gupta, Vinay Chowdary "Machine Learning Application for Precision Agriculture", IEEE Access Journal, Volume:9, 2021.

[3] Adel Bakshipour "Cascading Feature Filtering and Boosting Algorithm for Plant Type Classification", IEEE Access Journal, Volume:9, 2021.

[4] Manish Kumar, Ahlad kumar, Vinay S.Palaparthy "Soil Sensors-Based Prediction System for Plant Diseases Using Exploratory Data Analysis and Machine Learning", 2021, IEEE Sensors Journal, Volume:21,Issue:16, 2021.

[5] Gayathri Nagasubramanian, Rakesh Kumar Sakthivel, Rizwan Patan, Muthuramalingam Sankayya, Mahnoud Daneshmand, Amir H.Gandomi " Ensemble Classification and IoT-Based Pattern Recognition for Crop Disease Monitoring System", IEEE Internet of Things Journal, Volume:8, Issue:16, 2021.

[6] Rishi Gupta, Akhilesh Kumar Sharma, Krishna mode, Shahreen Kasim, Zirawani Baharum "Weather Based Crop Prediction in India using Big Data",IEEE Access Journal, Volume:9, 2021.

[7] Andrii Shelestov, Mykola Lavreniuk, Vladimir Vasiliev, Leonid Shumilo, Andrii kolotii "Cloud Approach to Automated Crop Classification Using Sentinel-1 Imagery", IEEE Transactions on Big Data, Volume:6, Issue:3, 2020.

[8] Muhammad Tufail, Javaid Iqbal, Mohsin Islam Tiwana, Muhammad Shahab Alam "Identification of Tobacco Crop Based on Machine Learning for a Precision Agricultural Sprayer", IEEE Access Journal, Volume:9, 2021.

[9] Waleej Haider, Aqeel-Ur Rehman, Nouman M. Durrani, Sadiq Ur Rehman "A Generic Approach for Wheat Disease classification and Verification Using Expert

Opinion for Knowledge-Based Decisions", IEEE Access Journal, Volume:9, 2021.
[10] Bo cheng, Wanyin Wu, Dapeng Tao, Shibo Mei, Ting Mao, Jun Cheng " Random Cropping Ensemble Neural Network for Image Classification in a Robotic Arm Grasping System", IEEE Transactions on Instrumentation and Measurement, Volume:69, Issue:9, 2020.
[11] Rayner Alfred, Joe Henry Obit, Christie Pei- Yee Chin, Haviluddin Haviluddin, Yuto Lim "Towards Paddy Rice Smart Farming: A Review on Big Data, Machine Learning, and Rice Production Tasks", IEEE Access, Volume:9, 2021.
[12] Alsulaiman, Mohammed Arafah, Mohamed Amine Mekhitiche "Intelligent Harvesting Decision System for Date Fruit Based on Maturity Stage Using Deep Learning and Computer Vision", IEEE Access Journal, Volume:8, 2020.
[13] Zengwei Tang, Hong Wang, Xiaobing Li, Xiaohui Li "An Object-Based Approach for Mapping Crop Coverage Using Multiscale Weighted and Machine Learning Methods", IEEE Access Journal, Volume;13, 2020.
[14] Waleej Haider, Aqeel-Ur Rehman, Nouman M. Durrani, Sadiq Ur Rehman "A Generic Approach for Wheat Disease Classification and Verification Using Expert Opinion for Knowledge-Based Decisions", IEEE Access Journal, Volume:9, 2021.
[15] Shantam Shorewala, Armaan Ashfaque, R.Sidharth, Ujjwal Verma "Weed Density and Distribution Estimation for Precision Agriculture Using Semi-Supervised Learning", IEEE Access, volume:9, 2021.