BloomScope: A Botanical Classification System for Plant Identification"

Problem Statement:

Describe the challenges people face in identifying plants accurately, such as Limited botanical knowledge, difficulty in distinguishing between similar species, and lack of accessible resources for identification.

Proposed System/Solution:

Introduce BloomScope, a botanical classification system that utilizes image recognition technology to identify plant species.

Explain how BloomScope will work by allowing users to take photos of plants and receive accurate identifications through a user-friendly interface.

Highlight the system's ability to provide additional information about identified plants, such as habitat, growth patterns, and common uses.

System Development Approach:

Detail the steps involved in developing BloomScope, including data collection, model training, and software development.

Discuss the technology stack and tools used for building the system, such as deep learning frameworks for image recognition and web development frameworks for the user interface.

Algorithm and Deployment:

Describe the deep learning algorithm used for plant identification, such as convolutional neural networks (CNNs), and explain how it was trained on a dataset of plant images.

Explain the deployment strategy for BloomScope, which may involve hosting the system on cloud infrastructure for scalability and accessibility.

Program :

import numpy as np

from sklearn.datasets import load\_iris

from sklearn.model\_selection import train\_test\_split

from sklearn.ensemble import RandomForestClassifier

from sklearn.metrics import accuracy\_score

# Load the Iris dataset

iris = load\_iris()

X = iris.data

y = iris.target

# Split the data into training and testing sets

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

# Train a Random Forest classifier

rf\_classifier = RandomForestClassifier(n\_estimators=100, random\_state=42)

rf\_classifier.fit(X\_train, y\_train)

# Function to predict the species of a flower based on its measurements

def predict\_species(sepal\_length, sepal\_width, petal\_length, petal\_width):

features = np.array([[sepal\_length, sepal\_width, petal\_length, petal\_width]])

prediction = rf\_classifier.predict(features)

species = iris.target\_names[prediction][0]

return species

# Sample output

sample\_sepal\_length = 5.1

sample\_sepal\_width = 3.5

sample\_petal\_length = 1.4

sample\_petal\_width = 0.2

predicted\_species = predict\_species(sample\_sepal\_length, sample\_sepal\_width, sample\_petal\_length, sample\_petal\_width)

print("Predicted species:", predicted\_species)

Output :

Prediction species : setosa

Results:

Present the performance metrics of BloomScope, including accuracy, precision, and recall, based on testing with a diverse set of plant images.

Include user feedback and satisfaction ratings to demonstrate the usability and effectiveness of the system.

Conclusion:

Summarize the key findings of the project, emphasizing the significance of BloomScope in addressing the challenges of plant identification.

Discuss potential future enhancements or extensions to the system, such as incorporating real-time updates and expanding the database of plant species.

References:

Provide citations for any relevant research papers, datasets, or software libraries used in the development of BloomScope.