Plant Leaf Diseases Using Tensorflow

Nishchal Krishnappa (0609414)
MS in Business Analytics
Machine Learning for Predictive Analytics
Golden Gate university

Table of Contents

Primary Objective.

- Plant diseases may reduce agricultural output and food security. Leaf disease, caused by fungus, bacteria, and viruses, is a frequent plant disease. Poor crop yields, quality, and economic losses may result from these diseases.
- Effective leaf disease management and prevention need early identification and precise diagnosis. Image recognition and machine learning have allowed automated systems to swiftly and correctly diagnose leaf diseases. These platforms enable farmers and agricultural specialists choose treatment and preventative methods.
- in this project, we apply machine learning skill to predict the disease in plants. over approach leverage python, along with libraries like tensorflow,pandas, seaborn, matplolib, and various other libraries to process and analyze the image of leaf. For agricultural systems to survive, plant disease detection and management research must continue. We can safeguard our crops and maintain food security for future generations by using cutting-edge science and technology.

Introduction to the data

Data Source

I have extracted the data from Kaggal website which contains train, validation, and test image in which they are 14 different plants leaf which are tomato, straw Berry, Squash, Soy Bean, Raspberry, Potato, Pepper, Peach, Orange, Grape, Corn, Cheery, Blue Berry, Apple this all plant continue different type of disease.

Plant Diseases

The dataset includes 38 different plant disease classes, such as 'Apple_Apple_scab', 'Apple_Black_rot', 'Apple_Cedar_apple_rust', 'Apple_healthy', 'Blueberry_healthy', and many more. Each disease category contains over 2000 images, providing a robust dataset for training and validating the machine learning model.

Data Preparation

The data has been carefully organized into train, validation, and test folders, allowing for proper model evaluation and preventing data leakage. The images are in RGB color mode and have been resized to a consistent size of 128x128 pixels to meet the input requirements of the deep learning model.

Image Integration Process

1

Importing Libraries

The first step in the image integration process is to import all the required libraries. This includes libraries like TensorFlow and Keras, which are essential for loading and preprocessing the image data. By importing these libraries, the code gains access to the necessary tools and functions to effectively handle the image data and prepare it for the machine learning model.

2

Loading Image Data

After importing the libraries, the next step is to load the image data using TensorFlow and Keras. The code utilizes the 'tf.keras.utils.image_dataset_from_directory' function, which is a convenient utility provided by TensorFlow to load image data from a directory and prepare it for training a neural network model. This function allows the code to specify the directory containing the training images, infer the labels from the directory structure, and configure various settings such as batch size, image size, and color mode.

3

Dataset Creation

The image integration process culminates in the successful creation of the training dataset. After applying the specified configurations, the code recognizes a total of 69,768 files belonging to 38 different classes. This indicates that the dataset has been successfully created, containing a diverse set of plant leaf images across multiple disease categories. The availability of a well-structured and comprehensive dataset is crucial for training an effective machine learning model to accurately identify and classify plant diseases.

Approach

- This project leverages a deep learning process using TensorFlow, convolutional neural network , Keras, and many more process
- The dataset contains 14 different plant species with various disease conditions, including healthy leaves.
- The data is split into training, validation, and test sets to train and evaluate the model.

Model Architecture

The model uses a convolutional neural network (CNN) architecture, including:Convolutional layers to extract features from the plant leaf images

- Pooling layers to reduce spatial dimensions and control overfitting
- Dropout layers to prevent overfitting
- Fully connected layers to combine the extracted features for classification
- A final softmax layer to output probabilities for the 38 disease classes

Training and evaluation

- The model is trained for 5 epochs, showing improvements in both training and validation accuracy and loss.
- The training and validation accuracy converge, indicating the model is generalizing well and not overfitting.
- The final validation accuracy reaches 94.36%, demonstrating the model's strong performance.

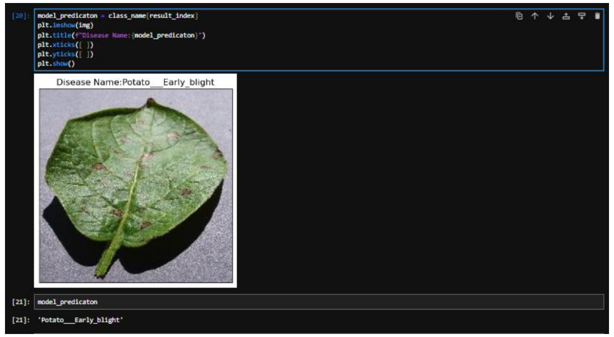
Results

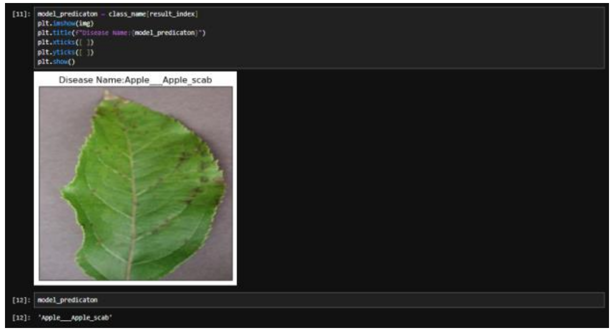
The trained model is tested on a separate test dataset.

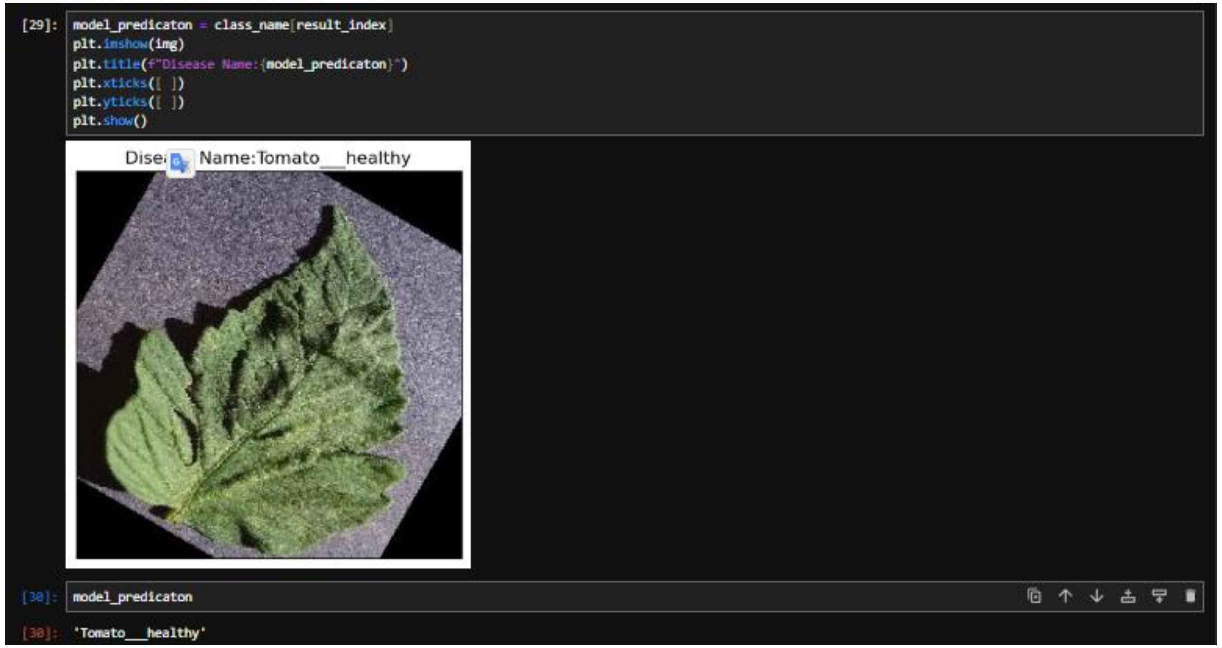
The model's predictions are analyzed, showing high confidence in many classifications.

Detailed performance metrics, including precision, recall, F1-score, and support, are provided for each of the 38 plant disease classes.

The overall macro-average F1-score is 0.94, indicating excellent model performance.







References

- (n.d.). Retrieved from https://keras.io/api/data_loading/image/
- Agudelo, M. B. (2008). *A case study concerning the extraction and identification of plant-parasitic nematodes*. APS.
- Jajoo, P. (26 May 2022). *A Case Study on Machine Learning Techniques for Plant Disease Identification*. Springer Link.
- SAMIR. (Dataset). *New Plant Diseases Dataset*. Kaggle. Retrieved from https://www.kaggle.com/datasets/vipoooool/new-plant-diseases-dataset/code