A Multi-Modal AI Framework for pre-trained ML models

**Overview**  
This project implements a machine learning-based diagnostic tool for crop disease detection. It leverages image data, textual symptom descriptions, and numeric environmental factors to identify diseases and suggest possible remedies.

**Problem Statement**  
Traditional methods of crop disease diagnosis rely on expert knowledge and are often not scalable to rural or low-resource settings. Farmers face difficulty identifying diseases early due to a lack of accessible and accurate tools.

**Scope**

* Multi-modal input support: images, text, numerical data
* Automatic algorithm selection based on data type
* Disease prediction and classification
* Platform-independent (Web, Mobile, Cloud)

**Objectives**

* Provide early disease diagnosis
* Automate the input preprocessing and algorithm selection
* Achieve high classification accuracy using state-of-the-art models

**Input Types Supported**

| **Type** | **Description** |
| --- | --- |
| Image | Leaf photos uploaded by farmers |
| Text | Symptom descriptions in local language or English |
| Numbers | Temperature, humidity, pH level, rainfall, etc. |

**ML Algorithms Used**

| **Data Type** | **Algorithm** |
| --- | --- |
| Image | CNN (MobileNetV2 / ResNet50) |
| Text | DistilBERT for sequence classification |
| Numbers | Random Forest Classifier / XGBoost |

1. System Flow Diagram

User Input  
  │  
  ├── Image → CNN Model  
  │  
  ├── Text/Numeric → BERT / Random Forest  
  │  
  → Output → Diagnosis and Report

**Requirements**

Software

* Python 3.8+
* TensorFlow, Transformers, OpenCV
* Scikit-learn, NumPy, Pandas
* Google Colab or Jupyter Notebook

Hardware

* GPU (optional for faster image model training)

**Modules**

* Data Input Handler – Accepts multi-modal input
* Preprocessing – Normalizes and tokenizes input
* Model Selector – Chooses ML model based on input
* Inference Engine – Predicts disease
* Results Interface – Displays diagnosis and suggestions

1. Data Cleaning & Feature Selection

Image

* Resize, normalize, remove corrupt images
* Feature extraction using CNN

Text

* Lowercasing, tokenization, removal of stopwords
* Use of DistilBERT tokenizer

Numeric

* Imputation for missing values
* Feature selection using correlation and variance

**Supported Platforms**

* Web App
* Android Mobile App (via TensorFlow Lite)
* Cloud-based APIs (Flask/FastAPI + ML models)

**Dataset Sources**

* Images: PlantVillage Dataset
* Text: Custom CSV Dataset via Google Sheets or Farmer Input
* Numeric: Simulated data or real-world agricultural databases

**Evaluation Metrics**

* Accuracy
* Precision, Recall, F1-Score
* Confusion Matrix

**Future Enhancements**

* Multi-language symptom input
* Integration with weather APIs
* Explainable AI for transparency