# Harvestify

**System Requirement Specification (SRS)**

Project work Phase 1 (EAI753)

**BACHELOR OF TECHNOLOGY (CSE)**

**Submitted to: Submitted By:**

Mr. Ashish Bishnoi Atishay Jain (TCA1959018)

Assistant Professor Charu Saxena (TCA1959050)

CCSIT, TMU

Dec, 2022



**FACULTY OF ENGINEERING & COMPUTING SCIENCES**

**TEERTHANKER MAHAVEER UNIVERSITY, MORADABAD**

# Table of Contents

[Table of Contents 2](#_Toc121880354)

[1 Introduction 3](#_Toc121880355)

1. [Problem Statement 3](#_Toc121880356)
2. [Scope of the Project 4](#_Toc121880357)
3. [Definitions, Acronyms, and Abbreviations 4](#_Toc121880358)
4. [Refrences 5](#_Toc121880359)

[2 Project Description 5](#_Toc121880360)

1. [Scope of the work 5](#_Toc121880361)
2. [Project Modules 6](#_Toc121880362)
3. [User Characteristics 14](#_Toc121880364)
4. [Constraints of project 15](#_Toc121880365)
5. [Assumptions and Dependencies 15](#_Toc121880366)

[3 Specific Requirement 16](#_Toc121880367)

1. [Performance Requirements 17](#_Toc121880368)
2. [Standard Compliance (diagrams) 18](#_Toc121880371)
3. [Software and Hardware Requirements 19](#_Toc121880374)

# Introduction

Harvestify is a website. Simply, it is a hybrid platform of crop recommendation, fertilizer recommendation and diseases analysis and prevention. Harvestify provides many applications on a single platform with the advanced technology like Harvestify provides a recommendation system also which helps the user to choose the best crop and fertilizer according to their need. Harvestify application developed on the Machine Learning and Deep Learning modules and its front-end is designed using the HTML, CSS, JS and back-end is designed using Python language using flask server.

## Problem Statement

Currently, agriculture is facing a hideous problem. Where in spite of all right knowledge consumed, an agriculture sector is facing a huge loss. Why? To put simply the crops aren’t supervised properly. classification analysis can help you with finding the right area for your crop, further resulting to control damage and more revenue generation. To be able to successfully yield crops, foremost and major key role is proper irrigation functionality. Machine learning algorithm can help with better irrigation resulting in following ways – maintain a desired soil water range in the root zone that is optimal for plant growth, low labor input for irrigation process management.

Farming is about risk calculation but what if the risk can be calculated and cured beforehand. Anomaly analysis can help you with identifying the weakness and strength of the soil, resulting in more revenue generation and saving ample amount of time.

Climate is now a data problem,” says Claire Monteleone. Earlier, Improper weather predictions lead to many crops lost- resulting in loss of money and time invested. But technology has evolved over years leading businesses to higher stable growth. Regression analysis will help you with better production forecasting using weather**.** conditions.Farming is one of the major sectors that influences a country’s economic growth.

In country like India, majority of the population is dependent on agriculture for their livelihood. Many new technologies, such as Machine Learning and deep Learning, are being implemented into agriculture so that it is easier for farmers to grow and maximize the yield.In this project, I present a website in which the following applications are implemented; Crop recommendation, Fertilizer recommendation and plant disease prediction, respectively.

* In the crop recommendation application, the user can provide the soil data from their sideand the application will predict which crop should the user grow.
* For the Fertilizer recommendation application, the user can input the soil data and the type of the crop they are growing, and the application will predict what the soil lacks or has excess of and will recommend improvements.
* For the last application, that is the plant disease prediction application, the user can input an image of a diseased plant leaf, and the application will predict what disease it is and will also give a little background about the disease and suggestion to cure it.

The works done till now only concentrated on crop prediction using different soil properties and Data Mining Techniques. Fertilizer Recommendation is not taken into consideration. So, it is necessary to develop crop yield prediction and fertilizer recommendation system which predicts crop yield based on soil nutrients crop yield data and recommend fertilizer for selected crop based on different datasets like fertilizer data, location data and crop yield data.

## Scope of the Project

Harvestify is important to drive farming efficiency. As earlier the soil erosion, Manures, Fertilizer and Biocides, etc. is less used by the farmers but now these products will increase the nitrogen, phosphorus, potassium and also increases the ph. level of the soil. So, this project will help those farmers who are not aware from it so they simply give their soil details, and also upload their crop image for best crop recommendation, fertilizer and disease prediction with prevention and solution.

## Definitions, Acronyms, and Abbreviations

* **Acronyms**

**MSIE:** Microsoft Internet Explorer.

**DFD:** Data Flow Diagram

**ER Diagram:** Entity Relationship Diagram

**DAC:** Department of Agriculture and Cooperation

**NIC:** National Informatics Centre

**SRS:** Software Requirement Specifications

* **Definitions**

|  |  |  |
| --- | --- | --- |
| **Name** | **Version** | **Purpose** |
| Python | 3.7 | Primary Programming Languages |
| Google Collab | 1.0.0 | Editor or IDLE |
| JavaScript | 13th Edition | For Making responsive |
| Sklearn | .21.0 | Machine Learning Libraries |
| Matplotlib | 3.0 | Used for graph and plotting |
| Pandas | 1.3.5 | Data Analysis Tool |
| Bootstrap | 5.2 | Web Development tool |
| Pytorch | 1.21.1 | Used for Deep learning neural network |
| TourchVision | 0.13.1 | Neural Network |
| Pickle | 3.5 | For Saving Model in file/data structure |

## Refrences

* <https://www.ijeast.com/papers/371-376,Tesma405,IJEAST.pdf>
* <https://www.irjet.net/archives/V5/i2/IRJET-V5I2479.pdf>

# Project Description

The system mainly consists of Harvesting information containing web server. Fertilizer and Crop details with their soil property is stored in dataset and diseases is also stored in form of image into the dataset

## Scope of the work

Harvestify will help to those farmers who are not aware about the best crop to grow or use fertilizer and the diseases in the field so this project come into play which will recommend best crop and fertilizer based on the soil testing and also predict the disease in the crop by taking only image as input and classify the disease and give the prevention method related to that crop.

## Project Modules

## 

Data Collection and Preprocessing

Model Building

Cloud Deployment

Web Application

1. **Data Module**

It is a very most first module of this project. To train the Machine learning model it’s is important to collect the necessary dataset. In this module the data collection and data preprocessing has been done. With the help of pre-processing, it selects the features and find the correlation between the attributes of the dataset. In this firstly we download the dataset of crop prediction, fertilizer prediction and disease prediction from Kaggle. We load this the CSV file in our model and see the head to know how the data look like and group it by area type. Data cleaning process is start with handling the NA values in which we use is null function. In this we have used different function like drop NA values, unique value function so that we can differentiate between them and lambda function.

Data Preparation

Feature Extraction

Prediction

1. **Feature Engineering**

This module is also called as dimensionality reduction. The main motive of this model is used remove the redundant features or the error from the give dataset. In feature engineering we create new feature which is helpful in future for outlier detection and removal in later stage. In our dataset label (crop type) is the categorial feature which cause problem if we have so many labels (crop type) at the same time. It creates problem in searching the best crop, fertilizer and disease analysis.

1. **Outlier Detection**

Outlier are the data points which are the data errors or sometimes they are not the errors but they represent the extreme variations in the dataset. So, it is better to remove them on time otherwise it creates problem in further processing. For removing the outlier, we can use different technique like standard deviation or we can use simple or domain knowledge. In this we set some threshold value and compare these values by using standard deviation method.

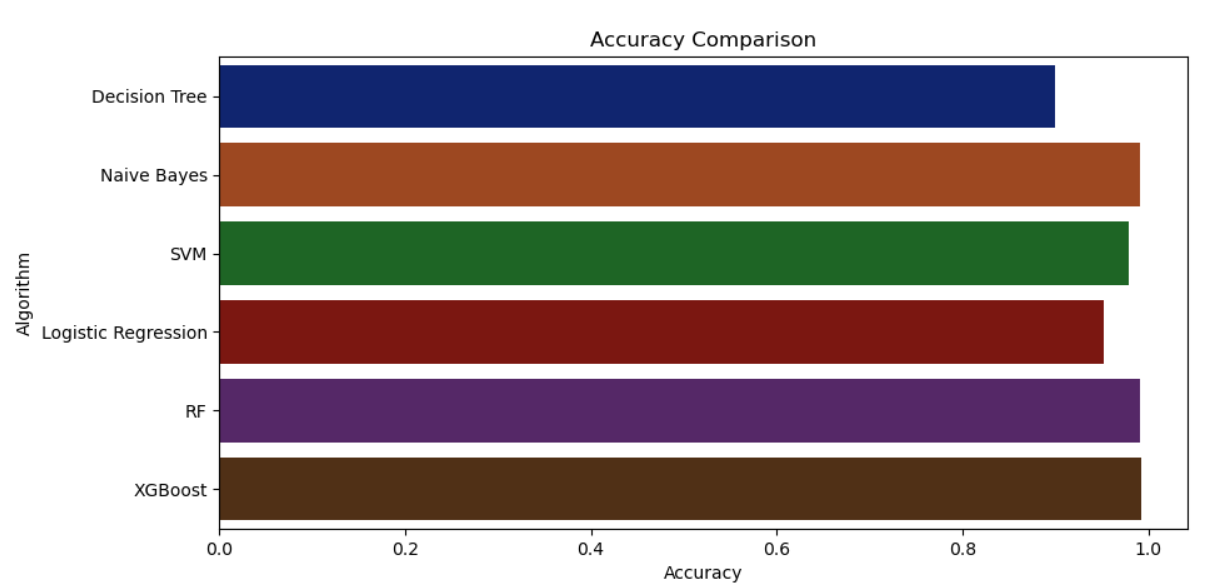
1. **Model Module**

In this module, firstly we train different-different machine learning algorithm model on the same data and check their accuracy and select the best fit algorithm which give high accuracy result and prediction as a crop recommendation algorithm and use for further process.

For Fertilizer Recommendation we use the same algorithm which is used by crop recommendation as they need same data as an input like Nitrogen, Potassium, Phosphorus, Ph value, rainfall in mm, location to predict the crop and fertilizer.

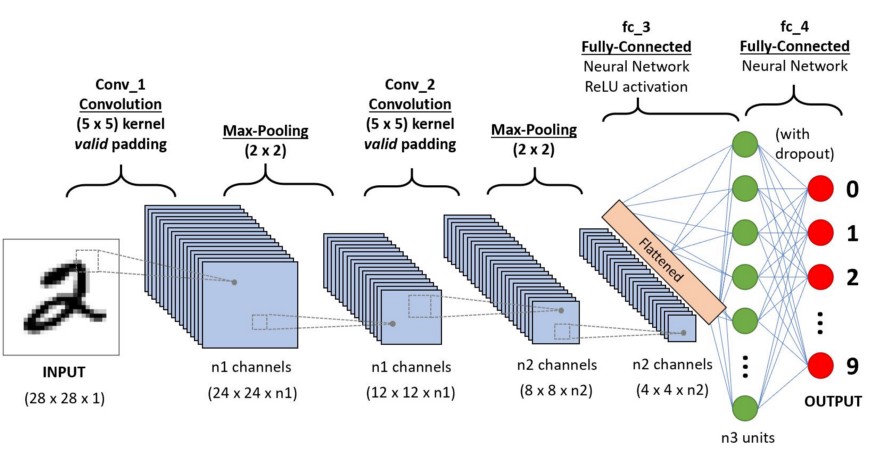
For Disease Prediction which is a part of image classification we use the CNN and RS-Net to process the image and predict the result for this we use GPU to train our model for this GPU is not provided on every platform to train the model again and again so we train the model only once’s and save that model for the future uses.   
The algorithm we check for best accuracy is: -

* Decision Tree Classifier
* Support Vector Machine Classifier
* Random Forest Classifier
* Logistic Regression
* Gaussian Naïve Bayes
* XG Boost

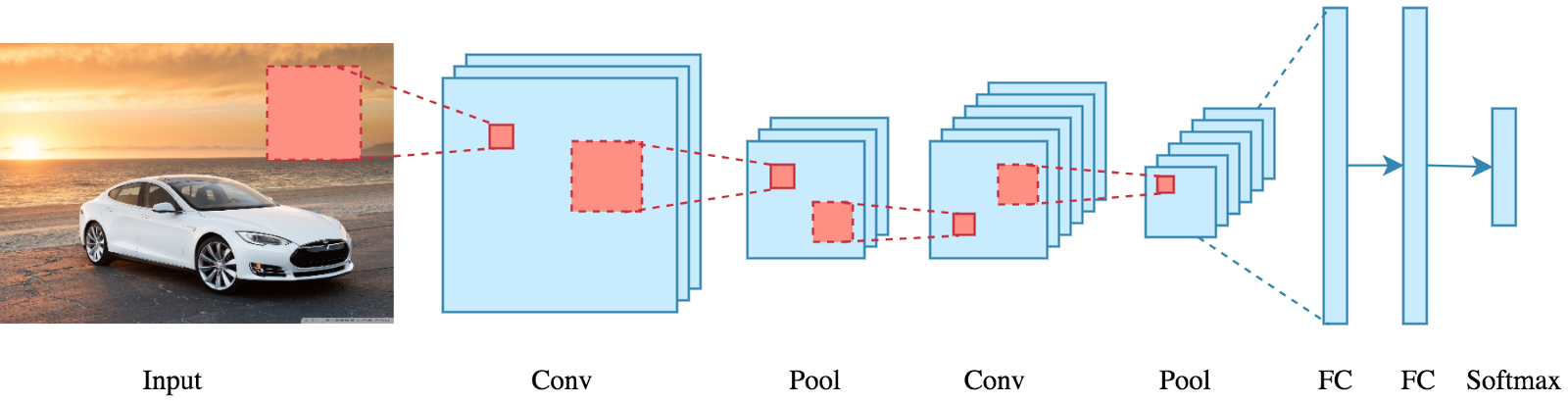


**CNN Model:**

CNNs were first developed and used around the 1980s. The most that a CNN could do at that time was recognize handwritten digits. It was mostly used in the postal sectors to read zip codes, pin codes, etc. The important thing to remember about any deep learning model is that it requires a large amount of data to train and also requires a lot of computing resources. This was a major drawback for CNNs at that period and hence CNNs were only limited to the postal sectors and it failed to enter the world of machine learning.



In deep learning, a convolutional neural network (CNN/ConvNet) is a class of deep neural networks, most commonly applied to analyses visual imagery. Now when we think of a neural network, we think about matrix multiplications but that is not the case with ConvNet. It uses a special technique called Convolution. Now in mathematics convolution is a mathematical operation on two functions that produces a third function that expresses how the shape of one is modified by the other.

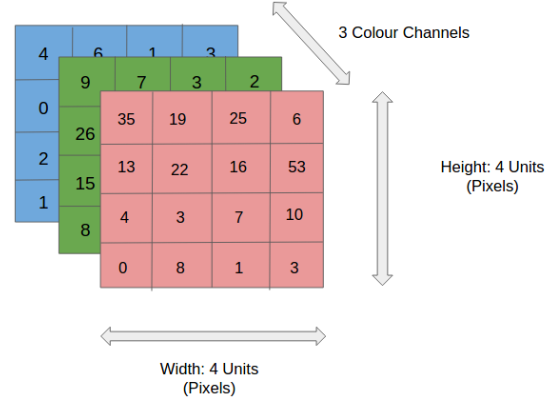


But we don’t really need to go behind the mathematics part to understand what a CNN is or how it works.

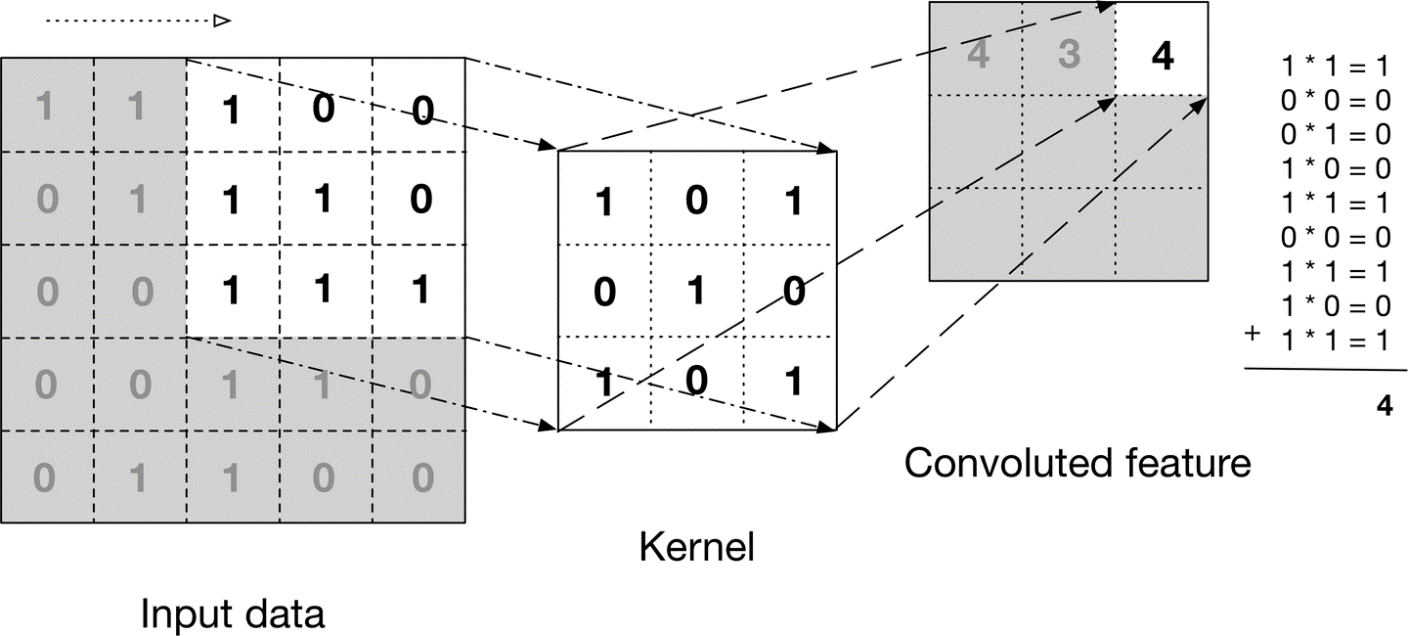
Bottom line is that the role of the ConvNet is to reduce the images into a form that is easier to process, without losing features that are critical for getting a good prediction.

**How does it work?**

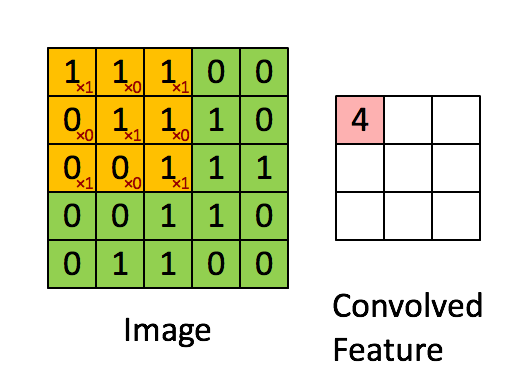
Before we go to the working of CNN’s let’s cover the basics such as what is an image and how is it represented. An RGB image is nothing but a matrix of pixel values having three planes whereas a grayscale image is the same but it has a single plane. Take a look at this image to understand more.



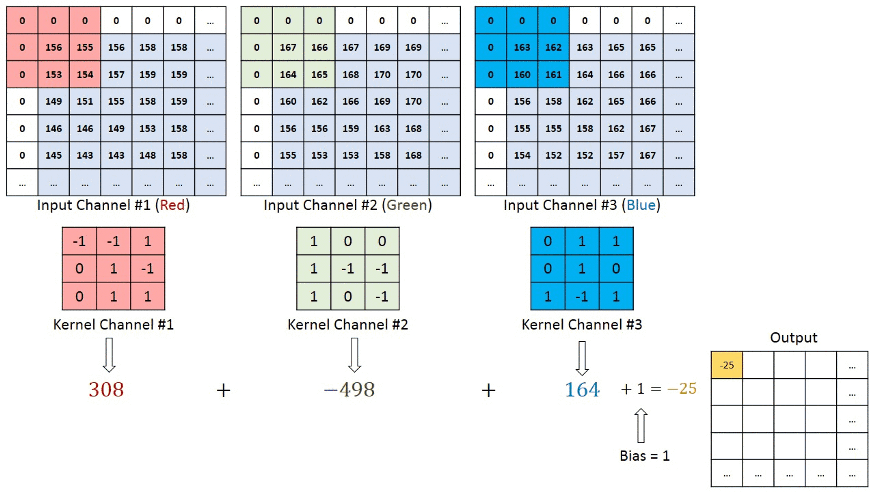
For simplicity, let’s stick with grayscale images as we try to understand how CNNs work.



The above image shows what a convolution is. We take a filter/kernel (3×3 matrix) and apply it to the input image to get the convolved feature. This convolved feature is passed on to the next layer.

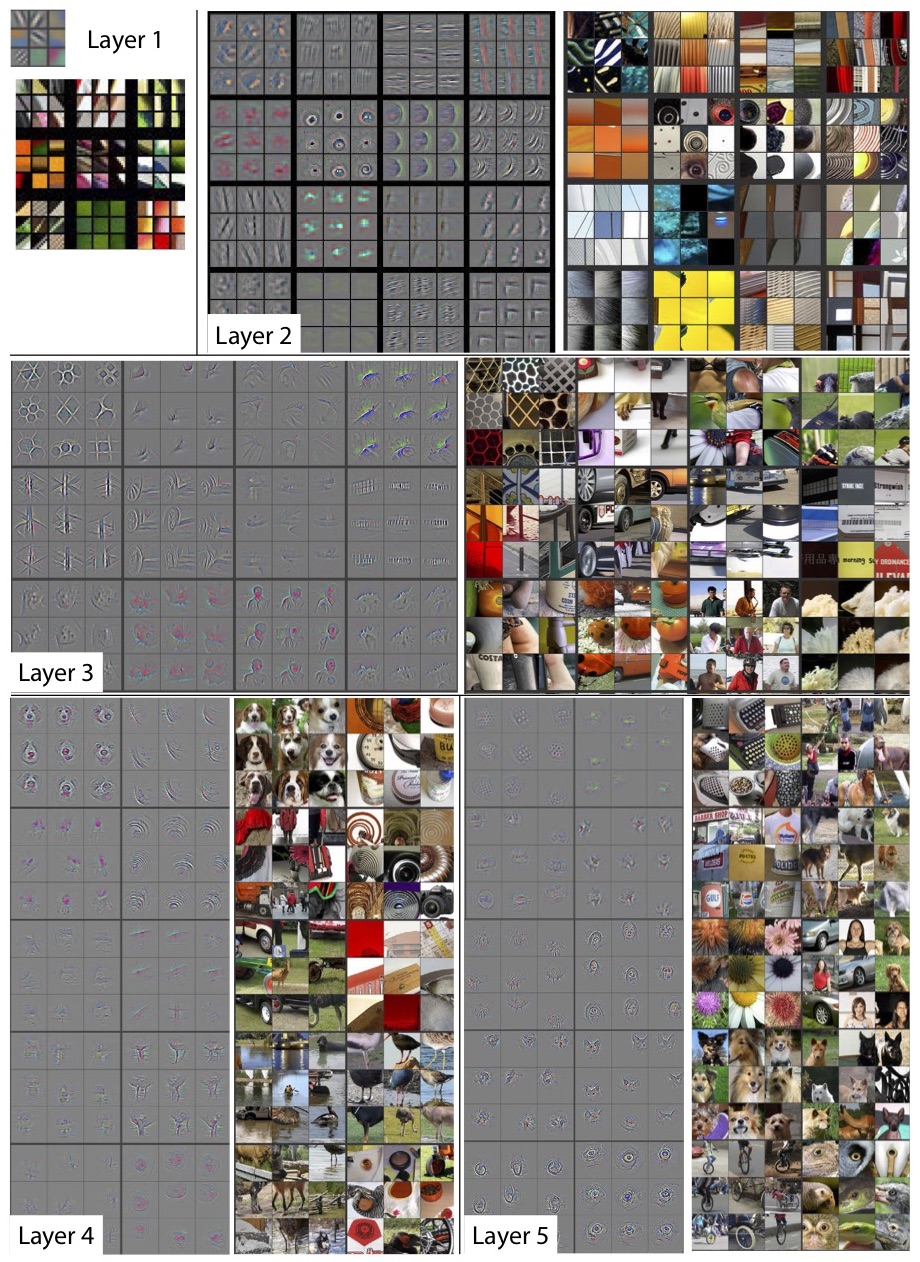


In the case of RGB color, channel take a look at this animation to understand its working

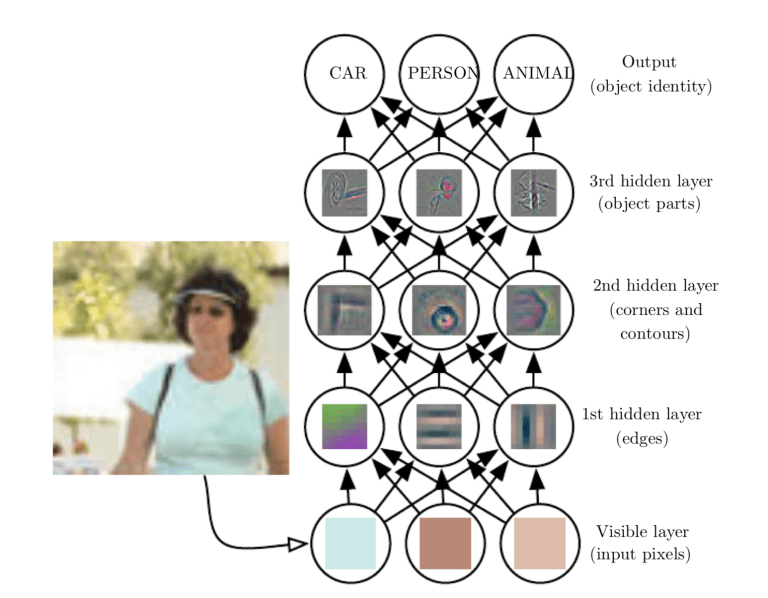


Convolutional neural networks are composed of multiple layers of artificial neurons. Artificial neurons, a rough imitation of their biological counterparts, are mathematical functions that calculate the weighted sum of multiple inputs and outputs an activation value. When you input an image in a ConvNet, each layer generates several activation functions that are passed on to the next layer.

The first layer usually extracts basic features such as horizontal or diagonal edges. This output is passed on to the next layer which detects more complex features such as corners or combinational edges. As we move deeper into the network it can identify even more complex features such as objects, faces, etc.

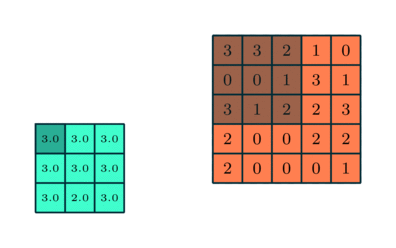


Based on the activation map of the final convolution layer, the classification layer outputs a set of confidence scores (values between 0 and 1) that specify how likely the image is to belong to a “class.” For instance, if you have a ConvNet that detects cats, dogs, and horses, the output of the final layer is the possibility that the input image contains any of those animals.



**Pooling layer: -**

Similar to the Convolutional Layer, the Pooling layer is responsible for reducing the spatial size of the Convolved Feature. This is to decrease the computational power required to process the data by reducing the dimensions. There are two types of pooling average pooling and max pooling. I’ve only had experience with Max Pooling so far, I haven’t faced any difficulties.



So, what we do in Max Pooling is we find the maximum value of a pixel from a portion of the image covered by the kernel. Max Pooling also performs as a Noise Suppressant. It discards the noisy activations altogether and also performs de-noising along with dimensionality reduction.

On the other hand, Average Pooling returns the average of all the values from the portion of the image covered by the Kernel. Average Pooling simply performs dimensionality reduction as a noise suppressing mechanism. Hence, we can say that Max Pooling performs a lot better than Average Pooling.



**Limitations**

Despite the power and resource complexity of CNNs, they provide in-depth results. At the root of it all, it is just recognizing patterns and details that are so minute and inconspicuous that it goes unnoticed to the human eye. But when it comes to understanding the contents of an image it fails.

Let’s take a look at this example. When we pass the below image to a CNN it detects a person in their mid-30s and a child probably around 10 years. But when we look at the same image we start thinking of multiple different scenarios. Maybe it’s a father and son day out, a picnic or maybe they are camping. Maybe it is a school ground and the child scored a goal and his dad is happy so he lifts him.

These limitations are more than evident when it comes to practical applications. For example, CNNs were widely used to moderate content on social media. But despite the vast resources of images and videos that they were trained on it still isn’t able to completely block and remove inappropriate content. As it turns out it flagged a 30,000-year statue with nudity on Facebook.

1. **Application Module**

This module is divided in three parts i.e., Crop Recommendation, Fertilizer Recommendation and Plant Diseases Prediction. For crop recommendation we take several inputs of soil testing and taking location as an input and through weather API we fetch the rainfall condition in that area and recommend the best crop based upon the input. For Fertilizer recommendation we take soil testing result values as an input and the crop that you want to grow and then recommend the fertilizer. For Diseases it will take image as an input and then classify the result and predict the diseases and crop and the prevention method and solutions.

## User Characteristics

* This project if open of all so, that anyone can use that without making their user profile.
* User can go to crop recommendation Portal.
* User can go to Fertilizer Recommendation Portal.
* User can go to Plant Disease Prediction Portal.

## Constraints of project

* + - **Image Dependencies**

The plant disease prediction is based upon the image which is given by the user as the image can be blurred, Distorted, Noisy etc.

* + - **API**

The Weather API which is used in the backend to get the climate condition of that particular area so that the API can face the High Data Traffic, High Data Overloading.

* + - **Internet**

The internet connection is required as it requires internet to fetch the results.

* + - **User Input**

The user-input is also a constraint as the recommendation and prediction is totally depend upon the input which is given by the user If the user-input is wrong then the prediction is also wrong.

* + - **Lack of experience with emerging technologies**

If the user is new to that technology or there is lack of experience to use the website and giving input then this can be also a constraint.

* + - **Weather Forecasting**

Climate is now a data problem,”. Earlier, Improper weather predictions lead to many crops lost- resulting in loss of money and time invested. But technology has evolved over years leading businesses to higher stable growth. Weather can be forecast which means we are not sure about the climate so this can be also a constraint.

## Assumptions and Dependencies

* + - Harvestify depend on the dataset which is used to train our model.
    - Harvestify requires advance hardware to train the model such as GPU.
    - Harvestify does not work without JavaScript support.

# Specific Requirement

This section provides all the detailed functional and non-functional requirements.

1. **User Interface:**

Harvestify application should contain following user interface

* **Index Page:** It contains the multiple page link through which a user can redirected to different-different pages and use the functionality of the project and get the favorable result.
* **Crop Page:** The crop page requires some soil data and location and based on these values the crop will recommend and the page is very easy to use i.e., User Friendly.
* **Fertilizer Page:** The Fertilizer page requires and the soil data and the crop that you want to grow and according to that it will recommend the best fit fertilizer to you.
* **Plant Disease:** This page only requires image and based on the image it will predict the disease and give the prevention and solution to cure it.

1. **Functions**

Functional Requirement defines a function of a software system and how the system must behave when presented with specific inputs or conditions. These may include calculations, data manipulation and processing and other specific functionality.

Non-functional requirements are requirements which specify criteria that can be used to judge the operation of a system, rather than specific behaviors. This should be contrasted with functional requirements that specify specific behavior or functions. Typical non-functional requirements are reliability, scalability, and cost. Non-functional requirements are often called the utilities of a system. Other terms for non-functional requirements are "constraints", "quality attributes" and "quality of service requirements"

* **Display Crop Prediction:**

In this functionality the user input is required and the based on that input the prediction can be done. This function will help the user to get the result favorable crop based upon their soil testing results such as Phosphorus, Nitrogen, Potassium etc. The user just needs to input the testing result and within a fraction of second it will give you the result i.e., crop name.

* **Fertilizer Prediction:**

In this functionality the user will give the crop that he has and also the values of the soil testing so that the best suited fertilizer will recommend to get better result and saturates the results or the ion into the soil to get higher result, fastest growing rate with low disease rate. The user just waits for some seconds and then user will get the best fertilizer suited for his crop.

* **Disease Prediction:**

In this functionality the user will give the image of the leaf and then it can simply analysis based upon the training and identify the disease and also show prevention step as well as the solution to cure this. The user will not do anything just click the image of the leaf direct upload on the page and click upon the submit button then the automate process will start and the user will get the result within a second.

# Performance Requirements

The following are the key performance requirements.

1. All pages should load within 2 seconds throughout the regions.
2. Search result should be displayed within 1 second.
3. In the case of the recommendation model, the probability of the user to get the right recommendation of crop and fertilizer.

Data Collection

Data Preprocessing

Remove Outliers

Training Dataset

Jupyter Lab

Decision Tree Classifier

Random Forest

Support Vector Machine

XGboost

Gaussian Naïve Bayes

CNN and RS-Net

Model Building

Predicted Value

# Standard Compliance (diagrams)

# 

* More data can be collected manually via web scrapping to make the system more accurate.
* Additional plant images can be collected to make the disease detection part more robust and generalized.
* The system uses supervised and unsupervised Machine learning algorithms and gives best result based on accuracy. The results of the two algorithms will be compared and the one giving the best and accurate output will be selected. Thus, the system will help reduce the difficulties faced by the farmers and stop them from attempting suicides. It will act as a medium to provide the farmers efficient information required to get high yield and thus maximize profits which in turn will reduce the suicide rates and lessen his difficulties.
* The prediction of crop yield based on location and proper implementation of algorithms have proved that the higher crop yield can be achieved. From above work I conclude that for soil classification Random Forest is good with accuracy 86.35% compare to Support Vector Machine. For crop yield prediction Support Vector Machine is good with accuracy 99.47% compare to Random Forest algorithm. The work can be extended further to add following functionality. Web application can be built to help farmers by uploading image of farms. Crop diseases detection using image processing in which user get pesticides based on disease images. Implement Smart Irrigation System for farms to get higher yield.
* We have worked on a sample dataset from Kaggle which has taken into consideration records obtained from a broad agricultural demography. Farmers generally use hit and trial method which leads to wastage of land and resources or even disproportionate growth of crops. We are trying to break all such taxing walls by providing them with an accurate and justified model made by machine learning using random forest classifier to identify the correct crop to be grown in their farms. This will help them in improving their crop production both qualitatively and quantitatively. This will also help them to maintain the quality and nutrition contents of the soil.

# 

# Software and Hardware Requirements

* **Software Requirements**

1. **Front-end**
   1. HTML
   2. JavaScript
   3. CSS
   4. Bootstrap
2. **Back-end**
   1. Python
   2. Tensor Flow
   3. Flask
   4. Heroku
   5. Machine Learning Model
   6. Deep Learning Model
   7. Weather API

* **Hardware Requirements**
  + RAM
  + Hard Disk
  + OS
  + Editor
  + Browser