High Level Design (HLD)

Mushroom Classifier

Revision Number : 0.1
Last date of revision : 30/10/2022

Contents

Document Version Control 3
Abstract 4
1 Introduction 5
1.1 Why this High-Level Design Document ? 5
1.2 Scope 5
1.3 Definitions 6
2 General Description 6
2.1 Product Perspective 6
2.2 Problem Statement 6
2.3 Proposed Solution 6
2.4 Further Improvements 7
2.5 Technical Requirements 7
2.6 Data Requirements 7
2.7 Tools Used 8
2.7.1 Hardware Requirements 8
2.8 Constraints 9
2.9 Assumptions 9
3 Design Details 10
3.1 Process Flow 10
3.1.1 Model Training and Evaluation 10
3.1.2 Deployment Process 11
3.4 Performance 11
3.5 Reusability 11
3.6 Application Compatibility 11
3.7 Resource Utilisation 11
3.8 Deployment 12
4 Conclusion.... 12

Document Version Control

Date Issued	Version	Description	Author
28/10/2022	0.1	Initial HLD - V0.1	Dhinagaran S
30/10/2022	1.0	HLD - V1.0	Dhinagaran S

Abstract

Mushrooms are one of the most regularly used food items for maintaining health. Mushrooms are rich in the

B vitamins: Riboflavin
Niacin and
Pantothenic acid

Even though it has some goodness, in mushrooms there are also some poisonous categories which can lead the humans to death after eating it. It is tough to classify the mushroom as poisonous or edible. InOrder to prevent the death due to eating of poisonous mushrooms, we need to design a system which can classify the mushroom based on the characteristics of the mushroom.

The Audubon Society Field Guide to North American Mushrooms contains descriptions of hypothetical samples corresponding to 23 species of gilled mushrooms in the Agaricus and Lepiota Family Mushroom (1981). Each species is labelled as either definitely edible, definitely poisonous, or maybe edible but not recommended. This last category was merged with the toxic category. The Guide asserts unequivocally that there is no simple rule for judging a mushroom's edibility, such as "leaflets three, leave it be" for Poisonous Oak and Ivy. The main goal is to predict which mushroom is poisonous & which is edible.

1 Introduction

1.1 Why this High-Level Design Document ?

The purpose of this High-Level Design (HLD) Document is to add the necessary detail to the current project description to represent a suitable model for coding. This document is also intended to help detect contradictions prior to coding, and can be used as a reference manual for how the modules interact at a high level.

The HLD will:

- Present all of the design aspects and define them in detail
- Describe the user interface being implemented
- Describe the hardware and software interfaces
- Describe the performance requirements
- Include design features and the architecture of the project
- List and describe the non-functional attributes like:
    - Security
    - Reliability
    - Maintainability
    - Portability
    - Reusability
    - Application compatibility
    - Resource utilisation
    - Serviceability

1.2 Scope

The HLD documentation presents the structure of the system, such as the database architecture, application architecture (layers), application flow (Navigation), and technology architecture. The HLD uses non-technical to mildly-technical terms which should be understandable to the administrators of the system.

1.3 Definitions

Term	Description
MCS	Mushroom Classifier System
Database	Collection of all the information monitored by this system
IDE	Integrated Development Environment
Heroku	Platform Provider (PAAS) for hosting the application

2 General Description

2.1 Product Perspective

The MCS based application is useful to classify whether the mushroom is edible or poisonous, which will help to prevent humans from dangerous

2.2 Problem statement

To create an AI solution for classifying the mushroom

2.3 Proposed Solution

The solution proposed here is an MCS (Mushroom Classifier System) based classifier that can be implemented to perform the mushroom classification. It will get the input from where it is required to predict the category of the mushroom, which is supposed to be classified. The input will be processed and given to the model to predict the output, then the output will be displayed to the web page, where they started the request.

2.4 Further Improvements

MCS can be added with more use cases like getting the input image from users, and using the image we can process it, and can give the output as edible or poisonous. The same system can be implemented using live camera using open source libraries like OpenCV to handle live images, and then processing and giving the output as edible or poisonous

2.5 Technical Requirements

This application is based on python language, so python should be installed and it should be properly configured with the PATH.

- The application should have the access to use the python environment
- Necessary packages listed in the requirements.txt should be made available for this application

2.6 Data Requirements

This application uses very less data when compared to other applications which are in the market now. It only carries 22 categorical data, which only requires very less data.

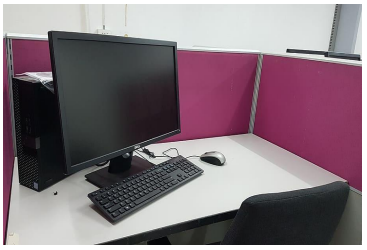
2.7 Tools Used

Python programming language and frameworks such as NumPy, Pandas, Scikit-learn, TensorFlow and Keras and used to build the whole model.

- PyCharm is used as an IDE.
- VSCode is used as an Text Editor
- For visualisation of the plots Matplotlib and Seaborn were used.
- Heroku is used for deployment of the model.
- Front end development is done using HTML/CSS
- Python Flask library is used for backend development.
- GitHub is used as a version control system.

2.7.1 Hardware Requirements

- PC or Laptop or any remote server which can run the application with necessary software which are mentioned in the technical requirements.


2.8 Constraints

The MCS based classification system must be user friendly, and the user not be required to know any of the workings of ML models.

2.9 Assumptions

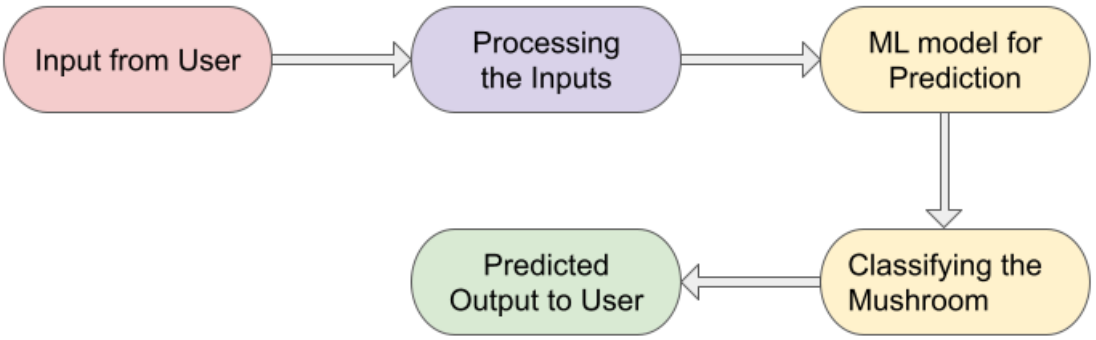
The main objective of the project is to implement the use cases as previously mentioned (2.2 Problem Statement) for a new dataset that comes through the MCS classifier which has the input fields as categorical data. The system can identify all the categories which are mentioned in the input fields without any exception. It is also assumed that all aspects of this project have the ability to work together in the way the designer is expecting.

3 Design Details

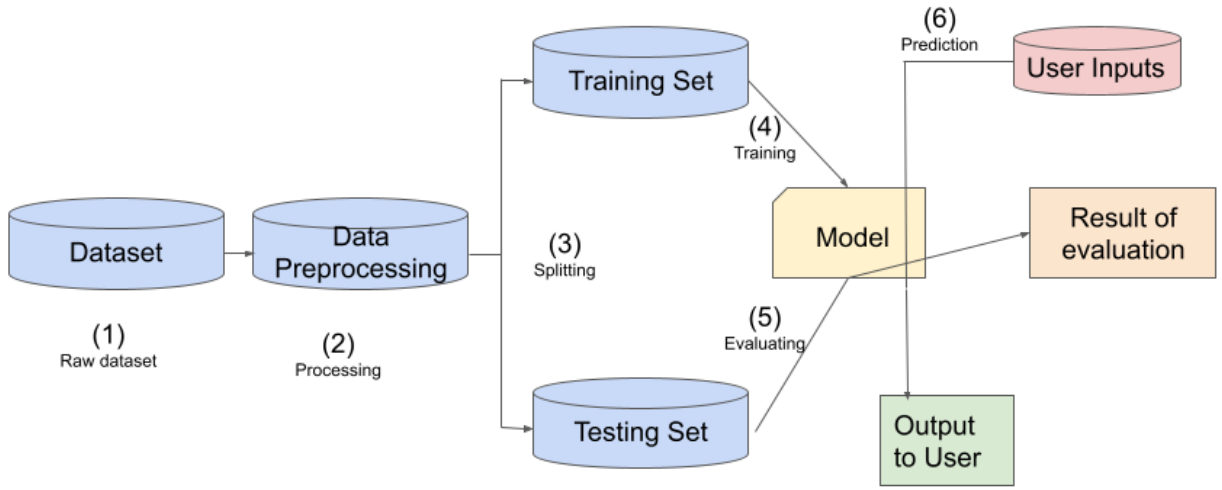
3.1 Process Flow

For identifying the different types of anomalies, we will use a deep learning base model. Below is the process flow diagram as shown below.

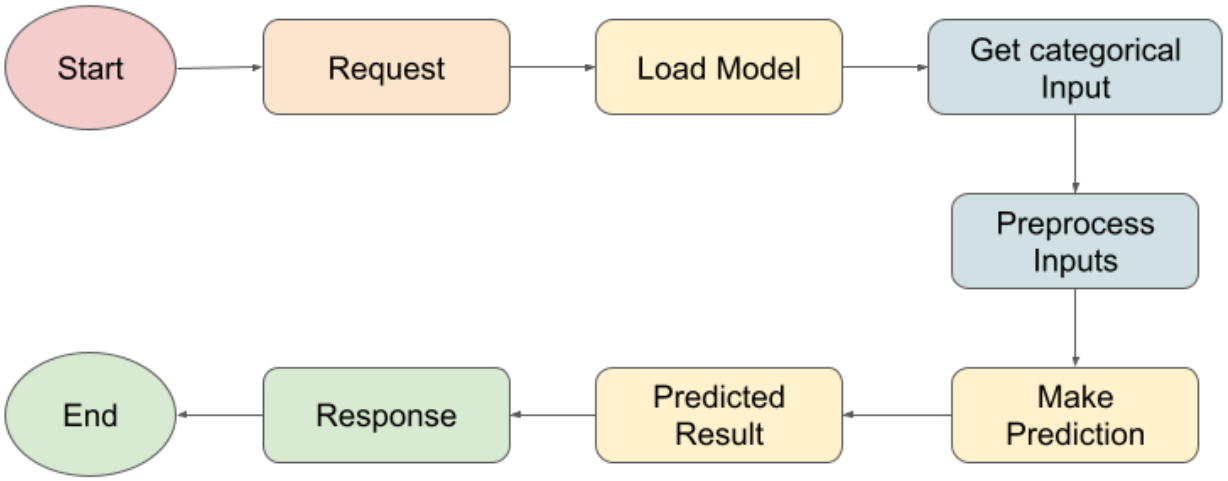
Proposed methodology



3.1.1 Model Training and Evaluation



3.1.2 Deployment Process



3.4 Performance

The MCS based classifier is used for classifying whether the mushroom is edible or poisonous when providing the required inputs. The Accuracy is 1, and the model is performing well.

3.5 Reusability

The code written to check different models based on training is in a modular form like Object. So it can be reused wherever and whenever needed.

3.6 Application Compatibility

All the requirements are listed in the requirements.txt for packages. And the software required for running the packages is Python 3.X >= Python 3.8

3.7 Resource Utilisation

When a request is made, then it will utilise only the main thread. So it will utilise less resources.

3.8 Deployment

4 Conclusion

The Designed MCS (Mushroom Classifier System) will classify the mushroom as edible or poisonous by processing the input given by user. Which will be more helpful to people to consume only edible mushrooms.

5 References

1. Official Logos from their official websites. (For Icons reference)