DETAIL PROJECT REPORT

Mushroom Prediction

Revision Number : 1.2

Last Date of Revision: 26/09/2022

Document Version Control

Date	Version	Description	Author
26/09/2022	1.0	Abstract Introduction General Description	Aluvala Anand
26/09/2022	1.1	Technical Requirements Data Requirements Data Pre-processing Design Flow	Aluvala Anand
26/09/2022	1.2	Data from User and its validation Rendering the Results Deployment Conclusion	Aluvala Anand

Contents

Abstract

To help people to know which mushroom is poisonous or which mushroom is not

1. Introduction

1.1 Why this DPR Document?

The main purpose of this DPR documentation is to add the necessary details of
the project and provide the description of the machine learning model and the
written code. This also provides the detailed description on how the entire
project has been designed end-to-end.

Key points:

- Describes the design flow
- Implementations
- Software requirements
- Architecture of the project
- Non-functional attributes like:
- Reusability
- Portability
- Resource utilization

2. General Description

2.1 Problem Perspective

The mushroom prediction is a machine learning model that helps users to understand mushroom is poisonous or not

2.2 Problem Statement

The main goal of this model is to predict mushroom is poisonous or eatable based on some input data like capshape,capcolor, veil color etc.

2.3 Proposed Solution

To solve this problem we made an user interface where
User can enter the file path and output file is generated
In that path

3. Technical Requirements

As technical requirements, we don't need any specialized hardware for
virtualization of the application. The user should have a device that has the access
to the web and the fundamental understanding of providing the input. And for
the backend, we need a server to run all the required packages to process the
input and predict the desired output.

3.1 Tools Used

Python programming language and frameworks such as NumPy, Pandas,
Scikit-learn, Flask, VS Code and are used to build the whole model.

• VS Code is used as IDE.
• For visualization of the plots, Matplotlib, Seaborn and Plotly are used. •
Heroku is used for deployment of the model.
• Front end development is done using HTML/CSS
• Python Flask is used for backend development.
• GitHub is used as version control system.

4. Data Requirements

4.1 Data Collection

The data for this project is collected from the Kaggle Dataset, the URL for the dataset is

Mushroom Classification | Kaggle

4.2 Data Description

Mushroom Classification dataset publicly available on Kaggle. The information in the dataset is present in one csv files named as mushrooms.csv. Dataset contains 8000+ rows which shows the information such capshape,capsurface etc.

4.3. Data Pre-processing

• Checked for info of the Dataset, to verify the correct datatype of the Columns. •
Checked for Null values, because the null values can affect the accuracy of the
model.
• Performed One – Hot encoding on the desired columns.
• Checking the distribution of the columns to interpret its importance.
Now, the info is prepared to train a Machine Learning Model

5. Design Flow

5.2 Logging

In logging, at each time an error or an exception occurs, the event is logged into the system log file with reason and timestamp. This helps the developer to debug the system bugs and rectify the error.

5.3 Data from User

The data from the user is retrieved from the created HTML web page.

5.4 Data Validation

- The data provided by the user is then being processed by app.py file and validated.
- The validated data is then sent to the prepared model for the prediction.

5.5 Rendering the Results

The data sent for the prediction is then rendered to the web page.

6. Deployment

Deployment is done on local host

7. Conclusion

This mushroom prediction will predict mushroom is poisonous or not

8. Frequently Asked Questions (FAQs)

Q1) What’s the source of data?

The data for training is provided by the client in multiple batches and each batch contain multiple files.

Q2) What was the type of data?

The data contain only Categorical values.

Q4) After the File validation what you do with incompatible file or files which didn’t pass the validation?

Files like these are moved to the Achieve Folder and a list of these files has been shared with the client and we removed the bad data folder.

Q5) How logs are managed?

We are using different logs as per the steps that we follow in validation and modelling like File validation log, Data Insertion, Model Training log, prediction log etc.

Q6) What techniques were you using for data pre-processing?

- Removing unwanted attributes.
- Visualizing relation of independent variables with each other and output variables.
- Checking and changing Distribution of continuous values.
- Removing outliers
- Cleaning data and imputing if null values are present.
- Converting categorical data into numeric values.

Q7) How training was done or what models were used?

- Before dividing the data in training and validation set, we performed
pre-processing over the data set and made the final dataset.
- As per the dataset training and validation data were divided.
- Algorithms like SVM, Decision Tree, Random Forest, XGBoost ,Knn were used based on the recall, final model was used on the dataset and we saved that model.