Architecture

Mushroom Classification

Written by	Arun sreeram R
Document version	1.0
Last Revision Date	12/02/2023

Document Control

Date	Version	Author	Comments
12/02/2023	Version-1.0	Arun Sreeram R	Introduction and Architecture Defined

Contents

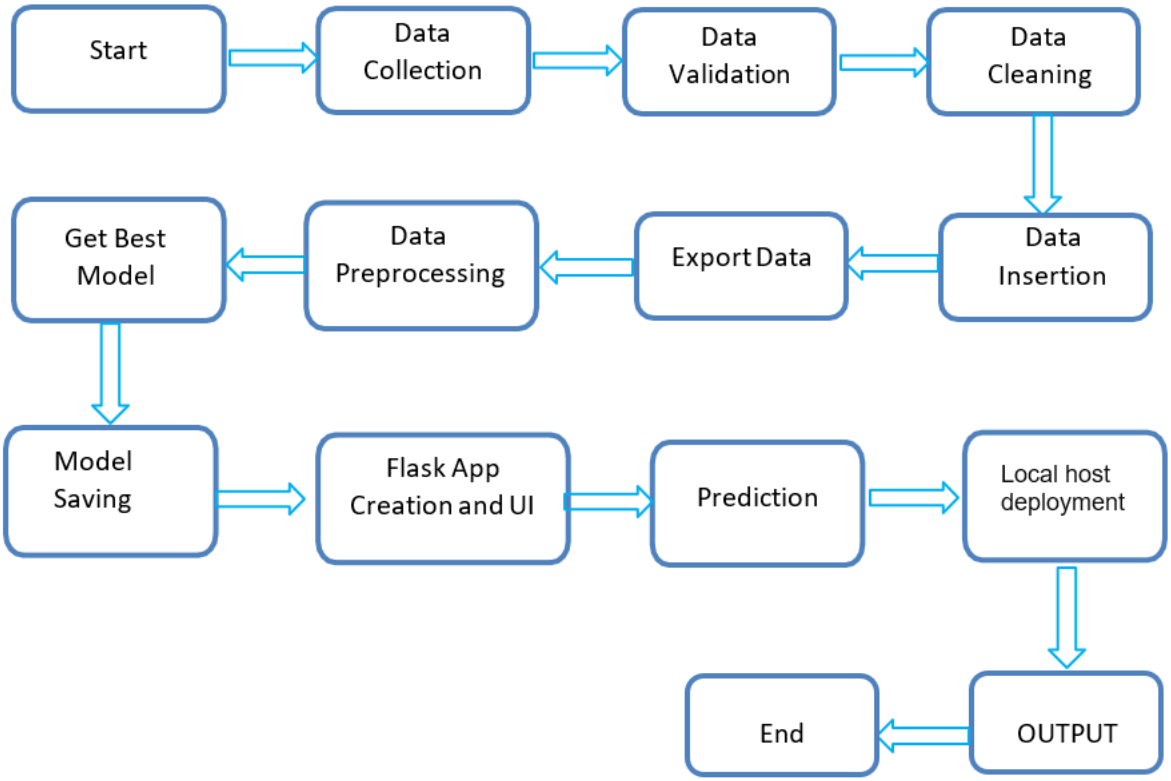
Page no.

1. Introduction

1.1 Why this Architecture Design Document?

With a focus on four essential quality attributes—usability, availability, maintainability, and testability—this work aims to present a complete architecture design of the Mushroom Classification.
The project's history and its architecturally significant function requirements are covered in this document. The purpose of this document is to aid the development team in choosing the system's top-level organisational structure. Finally, during the assessment of the team's work, the project coach can utilise this document to confirm that the development team is fulfilling the predetermined requirements.

1 Architecture



2 Architecture Description

3.1 Data Description

From the Audubon Society Field Guide to North American Mushrooms, this dataset comprises descriptions of hypothetical samples corresponding to 23 species of gilled mushrooms in the Agaricus and Lepiota Family Mushroom (1981). Each species is classified as either unquestionably edible, unquestionably poisonous, or maybe edible but not advised.

3.2 Import Data

Data Import from Database - The data in a stored database is imported as a CSV file to be used for Data Pre-processing and Model Training.

3.3 Data Cleaning

There are no null values in the data and all the variables are categorical, some of the observations meaningless and they are converted into meaningful observation.
Example: "?" is converted into letter "m" (Missing)

3.4 Exploratory Data Analysis

Every independent variable in the dataset is displayed by a multiple bar plot as it relates to the dependent variable's classes of poisonous and edible mushrooms as part of the EDA process.

3.5 Data Preprocessing

Data preprocessing steps are converting categorical variables into numerical variables using label encoding method and train and test split of the data etc.

3.6 Model Building

Following data preprocessing, separate the data into train and test sets (using simple random sampling), and use several machine learning algorithms for classification. The more accurate model is the XGboost model.

3.7 Model Dump

I developed a model and used the pickle module to dump the model in a pickle file format after comparing all accuracy levels and determining the optimal model for the dataset.

3.8 Data from User

Here With the aid of the UI interface, the user must input the names of all the features in the proper order and submit it to the model. The model will be fed the data and determine whether or not the feature set depicting a mushroom is edible.

3.9 Data Validation

Here Data Validation will be done, given by the user.

3.10 Model Call for specific input

A NumPy array will be created based on the User Input and fed to our model after being sent to the backend in variable format. After the pickle file has been loaded, the model will determine whether the input is edible or not and will communicate the conclusion to our html page.

3.11 User Interface

I created a user-interactive page for our application's front end where users can submit their input values. On their front-end page, I created a form with a lovely CSS design. The backend receives this HTML user input data in variable format. Decoupled HTML was used to create this content.

3.12 Deployment

The deployment is done in local host.