Mushroom Classification

Revision Number: 1.0

Last date of revision: 14/04/2024

# Document Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| Date Issued | Version | Description | Author |
| 14/04/2024 | 1 | Initial HLD — V1.0 | Aman Srivastava |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Contents

[Document Version Control. 2](#_TOC_250021)

[Abstract. 4](#_TOC_250020)

1. [Introduction 5](#_TOC_250019)
   1. [Why this High-Level Design Document? 5](#_TOC_250018)
   2. [Scope. 5](#_TOC_250017)
   3. [Definitions 5](#_TOC_250016)
2. [General Description. 6](#_TOC_250015)
   1. [Product Perspective 6](#_TOC_250014)
   2. [Problem statement 6](#_TOC_250013)
   3. [PROPOSED SOLUTION 6](#_TOC_250012)
   4. [FURTHER IMPROVEMENTS 6](#_TOC_250011)
   5. [Technical Requirements. 6](#_TOC_250010)
   6. [Data Requirements](#_TOC_250009) 6
   7. [Tools used.](#_TOC_250008) 7
      1. Hardware Requirements. 8
3. [Design Details](#_TOC_250005) 8
   1. [Process Flow.](#_TOC_250004) 8
      1. [Model Training and Evaluation](#_TOC_250003) 8
      2. [Deployment Process 1](#_TOC_250002)0
   2. Performance. 11
   3. Reusability. 11
   4. Application Compatibility 11
   5. Resource Utilization 11
4. Conclusion 11

## Abstract

Although this dataset was originally contributed to the UCI Machine Learning repository nearly 30 years ago, mushroom hunting (otherwise known as "shrooming") is enjoying new peaks in popularity. Learn which features spell certain death and which are most palatable in this dataset of mushroom characteristics. And how certain can your model be?

1. **Introduction**

###### 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: o Security
      * Reliability
      * Maintainability
      * Portability
      * Reusability
      * Application compatibility
      * Resource utilization
      * Serviceability

##### 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. **Definitions**

Term

ML

*SVC*

*Description*

Machine Learning

Support Vector Classification

### General Description

#### Product Perspective

#### This dataset includes descriptions of hypothetical samples corresponding to 23 species of gilled mushrooms in the Agaricus and Lepiota Family Mushroom drawn from The Audubon Society Field Guide to North American Mushrooms (1981). Each species is identified as definitely edible, definitely poisonous, or of unknown edibility and not recommended. This latter class was combined with the poisonous one. The Guide clearly states that there is no simple rule for determining the edibility of a mushroom; no rule like "leaflets three, let it be'' for Poisonous Oak and Ivy.

* 1. Problem statement

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. PROPOSED SOLUTION

Model trained on SVC model gives best accuracy among multilinear regression, decision tree classification, logistic regression and SVC.

* 1. FURTHER IMPROVEMENTS

Hyperparameter tuning can be done to explore more better accuracy with same models.

#### Technical Requirements

* + - Requires knowledge of-
    - Machine learning
    - Standard Scaler
    - SVC
    - Logistic Regression
    - Jupyter notebook
    - Scikit-Learn
    - Decision Tree Classifier
    - Multilinear regression
  1. **Data Requirements**

Requires significant numbers of records to train our machine learning models and so that it doesn’t overfit or underfit, resulting in a good model instead of dummy or biased model.

* 1. Tools used

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



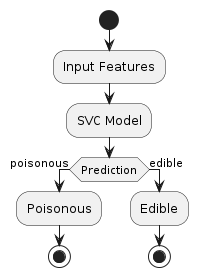
* + - VSCode is used as IDE.
    - For visualization of the plots, Matplotlib, Seaborn and Plotly are used.
    - Gradio is used for deployment of the model.
    - GitHub is used as version control system.
    1. Hardware Requirements
* Memory and disk space required per user: 1GB RAM + 1GB of disk + .5 CPU core.
* Server overhead: 2-4GB or 10% system overhead (whatever is larger), .5 CPU cores, 10GB disk space.
* Port requirements: Port 8000 plus 5 unique, random ports per notebook.

## Design Details

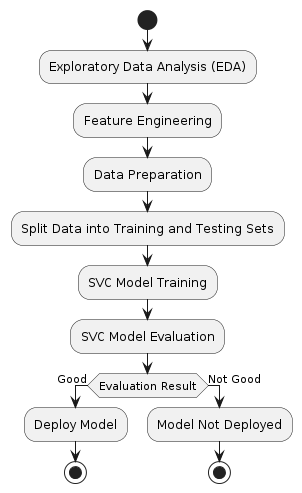
##### Process Flow

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

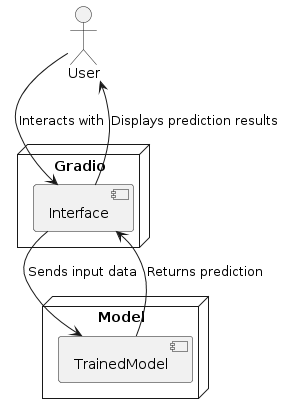
Proposed methodology



##### Model Training and Evaluation



##### Deployment Process



## Performance

Multilinear Regression gave accuracy of 0.7447525632220096.

LogisticRegression gave accuracy of 0.9559865721745617.

DecisionTreeClassifier gave accuracy of 0.9063782170831779.

SVC gave accuracy of 0.9682954121596419.

Hence, SVC is best model for the given dataset.

#### 3.3 Reusability

The code written and the components used should have the ability to be reused with no

problems.

#### Application Compatibility

The different components for this project will be using Python as an interface between them. Each component will have its own task to perform, and it is the job of the Python to ensure proper transfer of information.

#### Resource Utilization

When any task is performed, it will likely use all the processing power available until that function is finished.

## Conclusion

## SVC is best model for the given dataset giving accuracy of 0.9682954121596419.

## References

1. https://en.wikipedia.org/wiki/Unmanned ground vehicle
2. Scikit-learn
3. Python
4. Pandas
5. Numpy