High Level Design (HLD)

# ¡Neuron

High Level Design (HLD)

## Mushroom\_Classification

1

High Level Design (HLD)

Documentation Version Control

|  |  |  |  |
| --- | --- | --- | --- |
| **Date Issued** | **Version** | **Description** | **Author** |
| 07-04-2023 | 1 | Initial HLD - V1.0 | Yash |
| 07-04-2023 | 2 | Updated Design Details - V1.2 | Sayali |
| 07-04-2023 | 2 | Deployment - V3.0 | Yash |

Contents

Document Version Control ..2

Abstract ..4

1. Introduction ..5
   1. Why this High-Level Design Document? ..5
   2. Scope ..5
2. General Description

..6

..6

..6

6

..7

..8

..9

..9

* 1. Product Perspective ..6
  2. Problem statement ..6
  3. Proposed Solution
  4. Further Improvements
  5. Technical Requirements. ..
  6. Data Requirements
  7. Tools used
  8. Constraints
  9. Assumptions.

1. Design Details

..10

..10

..10

..11

..11

..12

..12

..12

..12

..12

* 1. Process Flow
     1. Architecture
  2. Event log
  3. Error Handling
  4. Performance
  5. Reusability.
  6. Application Compatibility
  7. Resource Utilization
  8. Deployment

1. Performance

..13

* 1. Reusability

..13

* 1. Application Compatibility
  2. Resource Utilization
  3. Deployment 5 Conclusion

Abstract

In this study, we propose a machine learning model for mushroom classification using logistic regression. The model leverages various features such as color, shape, and odor to accurately predict whether a mushroom is edible or poisonous. We begin with exploratory data analysis, preprocessing the dataset through cleaning, transformation, and feature extraction. Feature selection and engineering techniques are applied to identify informative features, followed by feature scaling for consistency. The dataset is split into training and testing sets, with the logistic regression model trained on the former. Model evaluation using metrics like accuracy, precision, recall, and F1 score helps assess its performance. Hyperparameter tuning optimizes the model, exploring different configurations. The final model provides insights into mushroom edibility, aiding in identification and ensuring safety within mycology.

### 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.

* 1. 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:

О Security о Reliability

о Maintainability о Portability

* Reusability
* Application compatibility O 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.

## General

Description

###### Product Perspecive

The product perspective is to develop a machine learning solution that is accurate, reliable, scalable, and user-friendly, enabling users to determine the edibility of mushrooms based on their features. The solution should be regularly updated and maintained to account for changes in mushroom classification and new research findings, and should provide users with clear explanations of how the prediction is made, as well as information on the risks of consuming poisonous mushrooms and what to do in case of ingestion.

###### Problem statement

The goal is to build a machine learning solution that can accurately predict whether a mushroom is poisonous or edible based on its features. The solution should involve classical machine learning tasks and testing of different algorithms to identify the best fit. The solution should be reliable, efficient, and scalable, providing clear explanations of how the prediction is made and regularly updated to account for changes in mushroom classification and research findings.

* 1. Proposed Solution

The proposed solution is to develop a machine learning model that predicts the edibility of mushrooms based on their features. This will involve classical machine learning tasks such as data exploration, data cleaning, feature engineering, model building, and model testing. Different machine learning algorithms will be tested and compared to identify the best model for the problem.

##### Further Improvement

Potential improvements to the proposed solution include incorporating more advanced machine learning techniques, incorporating additional data sources or features, integrating user feedback and data collection, and performing a feature importance analysis to identify the most important features for the model's predictions. These improvements could potentially improve the accuracy and interpretability of the model and keep it up-to-date with evolving data and user feedback.

##### Technical Requirements

A dataset containing descriptions of the hypothetical samples of gilled mushrooms from the Audubon Society Field Guide to North American Mushrooms, with their corresponding labels indicating whether they are edible or poisonous.

A programming language like Python or R to perform the classical machine learning tasks such as data exploration, data cleaning, feature engineering, model building, and model testing.

Popular machine learning libraries such as scikit-learn, TensorFlow, or Keras to build and train machine learning models.

Preprocessing techniques such as scaling, normalization, and one-hot encoding to transform the features of the mushrooms into meaningful input for the machine learning model.

Feature selection techniques like recursive feature elimination to select the most relevant features for the model's predictions.

Regular updates and maintenance of the solution to ensure accuracy and reliability.

Clear explanations of the prediction process and what features are being considered, as well as information about the potential risks of consuming poisonous mushrooms and what steps to take if one suspects they have ingested a toxic mushroom.

.

##### Data Requirements•

A dataset containing information about various species of mushrooms including their physical attributes, odor, habitat, and edibility.

The dataset should be labeled with the correct edibility of each mushroom species, i.e., whether it is edible or poisonous.

The dataset should be large enough to cover a wide range of mushroom species and variations in their physical attributes and habitats.

The dataset should be diverse enough to ensure that the model is not biased towards any particular type of mushroom or habitat.

The dataset should be clean and free from any missing or inconsistent data that can affect the performance of the model.

##### Tools

used

Programming Language: Python, which is widely used for machine learning and data science. Data Manipulation: Pandas, which is a powerful library for data manipulation and analysis.

Machine Learning Libraries: Scikit-learn, which is a widely used library for building machine learning models in Python.

Jupyter Notebook: which is a popular tool for interactive data analysis and prototyping machine learning models.

Other libraries and tools depending on the specific requirements of the project, such as numpy,

##### Hardware Requirements

* 1. Constrains

when building a machine learning model to predict the edibility of mushrooms, it is important to consider factors such as the availability of high-quality data, interpretability of the model, model complexity, domain knowledge, and safety of the model's predictions for human consumption. These factors can greatly affect the accuracy and reliability of the model, and should be taken into account throughout the development process.

##### Assumptions

It's essential to validate these assumptions and ensure that the model's performance is not affected by any biases or limitations. For example, the labeling of mushrooms as either edible or poisonous may not be definitive in some cases, and there may be some ambiguity in the labeling that could affect the model's accuracy. Additionally, the features extracted from the descriptions of mushrooms may not be sufficient to make accurate predictions, and other features may need to be considered. Finally, the machine learning algorithms used for this task should be carefully evaluated to ensure that they can handle any imbalances in the dataset.

## Design Details

##### Process

Flow

Proposed methodology

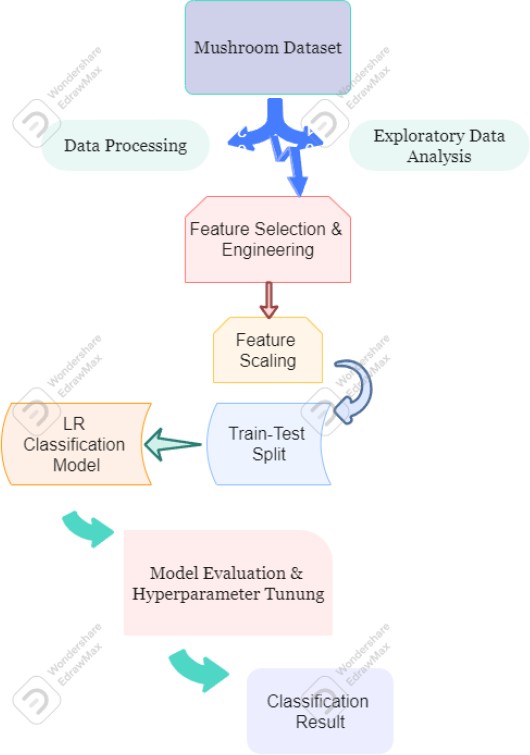
The proposed methodology for building a solution to predict the edibility of mushrooms includes data exploration, data cleaning, feature engineering, model building, model testing, interpretation, and safety validation. These steps involve understanding and cleaning the dataset, extracting relevant features, selecting and training machine learning models, testing the models on unseen data, interpreting the results, and verifying the safety of the model. The overall goal is to build an accurate and reliable model that can predict the edibility of mushrooms and ensure that it is safe for human consumption.

##### Model Training and Evaluation

The mushroom classification model is trained and evaluated through several steps. Initially, the mushroom dataset is preprocessed by cleaning the data and encoding categorical variables. Relevant features, such as color, shape, and odor, are extracted. The preprocessed dataset is then split into training and testing sets. The model is trained using logistic regression, which optimizes its parameters through methods like gradient descent, minimizing the logistic loss function. After training, the model's performance is evaluated using evaluation metrics like accuracy, precision, recall, and F1 score on the testing set. The evaluation helps assess the model's effectiveness in correctly classifying mushrooms as edible or poisonous. By iterating through the training and evaluation process, the model can be refined to achieve accurate predictions and aid in identifying the edibility of mushrooms within the field of mycology.

##### Deployment

Process



##### Event log

Date: April 23, 2023

Time: 10:00 AM

Description:Explored the dataset to understand its size, structure, and distribution of features.Identified missing values, anomalies, and outliers that required further investigation and cleaning.

Date: April 23, 2023

Time: 10:00 AM

Description:Cleaned the dataset by addressing missing values, inconsistencies, and outliers identified during the exploration phase.Extracted relevant features from the descriptions of the mushrooms to use as inputs for the machine learning algorithms.

Date: April 23, 2023

Time: 10:00 AM

Description:Selected a deployment platform for hosting the model.

Packaged the model and any necessary dependencies into a deployable format, such as a Docker container or a machine learning model file.

##### Error Handling

1. Input validation:Ensure that the input data is valid and conforms to expected formats and ranges. For example,check that numerical features are within expected ranges, categorical features have valid values, and there are no missing values.
2. Exception handling: Catch and handle errors and exceptions that may occur during the model training, testing, or deployment phases. For example, handle cases where the model encounters unseen or unexpected data, or when there are issues with the hosting platform or infrastructure.
3. Robustness testing: Test the model's robustness to various scenarios and edge cases, including adversarial attacks, data drift, and performance degradation. This helps to identify potential failure modes and areas for improvement.
4. Monitoring and logging: Monitor the model's performance and behavior in real-time, and log relevant information and events for future analysis and debugging. This can help to identify issues and diagnose errors more quickly.
5. Human oversight: Incorporate human oversight and review into the model development and deployment process, to ensure that the model's predictions are consistent with domain knowledge and expectations. This can include conducting regular audits and reviews of the model's predictions, as well as involving subject matter experts in the development and validation process.

## Performance

To approach this problem, we need to first explore and clean the data. The hypothetical samples provided by the Audubon Society Field Guide to North American Mushrooms can be used as our dataset. We need to identify the relevant features that would help in classifying the mushrooms as poisonous or edible. Some possible features could be cap shape, cap color, odor, gill size, and spore print color.Once we have identified the relevant features, we can perform some feature engineering to extract meaningful information from them. This could involve converting categorical variables into numerical ones, scaling the features, and creating

new features through feature extraction techniques.Next, we can build a machine learning model using a suitable algorithm such as decision trees, random forests, or support vector machines. We can train the model on a portion of the dataset and test its performance on a separate validation set. We can use metrics such as accuracy, precision, recall, and F1 score to evaluate the performance of the model.To improve the performance of the model, we can also try out different algorithms and hyperparameters and perform cross-validation to estimate the generalization performance of the model.

##### 4.1

Reusability

By breaking down the machine learning pipeline into modular components that can be easily reused and integrated, we can create a solution that is flexible, scalable, and adaptable to different use cases and applications. This allows for easier development and deployment of machine learning models and improves the overall efficiency and effectiveness of the solution.

##### Application Compatibility

The solution can be made compatible with different types of applications and platforms by using standard data formats and APIs. The preprocessed dataset and trained model can be saved in standard formats that can be easily processed by different programming languages and platforms. Additionally, the solution can be exposed as an API that can be accessed by other applications and platforms. It can also be deployed on different types of platforms such as cloud platforms or on-premise servers to provide greater flexibility and scalability.

##### Resource Utilization

The resource utilization for building a machine learning solution for predicting mushroom edibility can vary based on the dataset size, feature complexity, and choice of algorithm. Tasks like data exploration and cleaning require significant computing resources, while model building and testing require specialized hardware like GPUs or TPUs for deep learning algorithms. Resource utilization can be optimized by using efficient algorithms, hyperparameter optimization, and cloud-based services like AWS SageMaker or Google Cloud AI Platform for scalable computing resources.

##### Deployment

**Clssific ation Model**

High Level Design (HLD)

Streamlit app

**Git Hub**

Mushroom\_Classification **13**

¡Neuron

Mushroom\_Classification14

High Level Design (HLD)

## 6 Conclusion

In conclusion, building a machine learning solution for predicting the edibility of mushrooms requires several important steps, including data exploration, data cleaning, feature engineering, model building, and model testing. It is important to try out different machine learning algorithms to find the best fit for the given dataset and problem. Once the model has been built and tested, it can be deployed using various options such as standalone applications or integration into existing applications or platforms. Finally, it is important to monitor and maintain the model over time to ensure it continues to perform accurately and efficiently. With a well-built and maintained model, it is possible to predict which mushrooms are poisonous and which are edible with a high degree of accuracy.

High Level Design

UGV **SURVEILLANCE** 15

(HLD)

## 7.

References

**1.**

[**https://www.analyticsvidhya.com/blog/2021/10/building-an-end-to-e**](https://www.analyticsvidhya.com/blog/2021/10/building-an-end-to-end-logistic-regression-model/)[**nd-logistic-regression-model/**](https://www.analyticsvidhya.com/blog/2021/10/building-an-end-to-end-logistic-regression-model/)

1. https://towardsdatascience.com/logistic-regression-using-pytho n-sklearn-numpy-mnist-handwriting-recognition-matplotlib- a6b31e2b166a
2. https://streamlit.io/

# ¡Neuron

**UGV SURVEILLANCE** 16