High Level Design Document

Project Name: Mushroom Classification

Name: Venkata Phani Srikar Pillalamarri

E-mail: pvpsrikar@gmail.com

Table of Contents

Abstract

The Audubon Society Field Guide to North American Mushrooms offers detailed descriptions of 23 species of gilled mushrooms belonging to the Agaricus and Lepiota families. Every species is categorized into two distinct groups: unequivocally safe for consumption (Edible)or hazardous (Poisonous) (including mushrooms that are either unquestionably dangerous or perhaps edible but not advised). This consolidation highlights the guide's focus on prudence, emphasizing the intricacy and the hazards associated with mushroom identification. Unlike some plants that may be readily classified as either safe or dangerous based on simple guidelines, such as the "leaflets three, let it be" saying for Poisonous Oak and Ivy, mushrooms do not possess clear-cut markers to determine their suitability for consumption.

The main goal is to provide a dependable technique for forecasting whether a certain mushroom is toxic or safe for consumption. The task at hand is to develop a precise and practical prediction model that accurately distinguishes between edible mushrooms and hazardous ones, taking into account their varied and sometimes nuanced properties. Implementing such a strategy will greatly improve the safety of foragers and hobbyists by decreasing the likelihood of unintentional poisoning. The objective of this research is to use the comprehensive descriptions and classifications in the book in order to create a prediction framework. This will help answer the guide's claim that there is no straightforward criterion for determining whether a mushroom is edible or not.

Chapter – 1

Introduction

For this project, a high-level design document is essential because it gives direction and clarity, guaranteeing that all team members and stakeholders are aware of the goals, specifications, and limitations of the project. This shared knowledge lowers the possibility of misunderstandings and helps focus efforts toward a single objective. The paper also provides an organized method by decomposing complex procedures into smaller, more manageable sections and outlining the system's architecture, important modules, and functional relationships. Better scheduling, resource allocation, and planning are made possible by this organization, guaranteeing that every stage of the project is effectively and efficiently managed.

Additionally, relatively early in the project's development, the high-level design document helps with risk management by identifying possible hazards and impediments. This enables proactive risk reduction techniques, guaranteeing a project's smooth development without significant setbacks. In addition to promoting cooperation and review, the document acts as a point of reference for stakeholders and team members to provide input and guarantee the solidity of the design. Moreover, it establishes the foundation for comprehensive design and execution stages, directing the development of technical specifications, coding, testing, and deployment strategies. In order to ensure the project's long-term viability, the document also functions as crucial reference material for the future. It facilitates maintenance, scalability, and the onboarding of new team members.

Chapter – 2

Description

2.1 Problem Perspective

The issue at hand is determining with accuracy which members of the Agaricus and Lepiota families of mushrooms are edible and which are hazardous. This is done using information from The Audubon Society Field Guide to North American Mushrooms. This is a crucial problem as misidentification may have dangerous repercussions, including poisoning and major health hazards. The difficulty of this process is increased by the lack of clear-cut guidelines for determining whether a mushroom is edible or severely hazardous, since minute morphological variations might indicate whether a mushroom is safe to eat or not. Therefore, the main task is to create a trustworthy prediction model that can precisely categorize mushrooms according to their specific traits, improving the security and self-assurance of mushroom hunters and aficionados in recognizing edible species. In order to develop such a model and manage the inherent risks and uncertainties in mushroom identification, this study intends to make use of thorough species descriptions and classification data.

2.2 Problem Statement

Descriptions of fictitious samples that match to 23 species of gilled mushrooms in the Agaricus and Lepiota Family Mushroom may be found in the Audubon Society Field Guide to North American Mushrooms (1981). Every species has a label indicating whether it is certainly toxic, definitely edible, or maybe edible but not advised. The poisonous category and this final category were combined. The Guide states unambiguously that there is no "leaflets three, leave it be" criteria that can be used to poisonous oak and ivy mushrooms to determine if they are edible. Determining which mushrooms are edible and which are harmful is the primary objective.

2.3 Project Solution

Creating a machine learning-based prediction model that can reliably identify mushrooms belonging to the Agaricus and Lepiota families as either hazardous or edible is the answer to this challenge. The first step in the procedure is gathering data, which is done by using The Audubon Society Field Guide to North American Mushrooms' comprehensive descriptions and classifications. To extract pertinent information including physical traits, environment, and other differentiators, this data will undergo preprocessing.

To create and assess the model, the provided data will be divided into training and testing sets. To find the best method for classification, a variety of machine learning algorithms will be investigated, including Logistic Regression, SVM, KNN, Naive Bayes, Decision Trees, and Artificial Neural Networks. To guarantee solid and trustworthy predictions, the model's performance will be assessed using measures like accuracy, precision, recall, and F1-score.

The model's prediction capacity will be increased by using feature selection and engineering approaches to determine the key characteristics that set edible mushrooms apart from hazardous ones.

A user-friendly interface or application that allows users to enter mushroom attributes and instantly estimate an edible mushroom's state will also be a part of the solution. In addition to

providing explanations for the predictions, this interface will reveal which factors affected the categorization choice.

2.4 Technical Requirement

There are several ways to categorize the technical needs for this project: data requirements, model creation, software and tools, user interface, testing, and validation. Every category makes certain that the project is carried out effectively, producing a strong and trustworthy prediction model.

Data:

1. Data Source:
   The data is taken from The Audubon Society Field Guide to North American Mushrooms. link
2. Features of the Data:
   - Cap Shape: This attribute describes the shape of the mushroom cap and includes the following categories: bell (b), conical (c), convex (x), flat (f), knobbed (k), and sunken (s).
   - Cap Surface: This attribute indicates the texture of the mushroom cap surface: fibrous (f), grooves (g), scaly (y), and smooth (s).
   - Cap Colour: This attribute represents the colour of the mushroom cap: brown (n), buff (b), cinnamon (c), grey (g), green (r), pink (p), purple (u), red (e), white (w), and yellow (y).
   - Bruises: This attribute specifies whether the mushroom has bruises or not: bruises (t), no bruises (f)
   - Odour: This attribute describes the smell of the mushroom: almond (a), anise (l), creosote (c), fishy (y), foul (f), musty (m), none (n), pungent (p), and spicy (s).
   - Gill Attachment: This attribute indicates how the gills are attached to the mushroom stem: attached (a), descending (d), free (f), and notched (n).
   - Gill Spacing: This attribute describes the spacing of the gills: close (c), crowded (w), and distant (d).
   - Gill Size: This attribute specifies the size of the gills: broad (b) and narrow (n).
   - Gill Colour: This attribute indicates the colour of the gills: black (k), brown (n), buff (b), chocolate (h), grey (g), green (r), orange (o), pink (p), purple (u), red (e), white (w), and yellow (y).
   - Stalk Shape: This attribute describes the shape of the mushroom stalk: enlarging (e) and tapering (t).
   - Stalk Root: This attribute indicates the type of root system: bulbous (b), club (c), cup (u), equal (e), rhizomorphs (z), rooted (r), and missing (?).
   - Stalk Surface Above Ring: This attribute represents the texture of the stalk surface above the ring: fibrous (f), scaly (y), silky (k), and smooth (s).
   - Stalk Surface Below Ring: This attribute describes the texture of the stalk surface below the ring: fibrous (f), scaly (y), silky (k), and smooth (s).

- Stalk Colour Above Ring: This attribute indicates the colour of the stalk above the ring: brown (n), buff (b), cinnamon (c), grey (g), orange (o), pink (p), red (e), white (w), and yellow (y).
- Stalk Colour Below Ring: This attribute represents the colour of the stalk below the ring: brown (n), buff (b), cinnamon (c), grey (g), orange (o), pink (p), red (e), white (w), and yellow (y).
- Veil Type: This attribute describes the type of veil: partial (p) and universal (u).
- Veil Colour: This attribute indicates the colour of the veil: brown (n), orange (o), white (w), and yellow (y).
- Ring Number: This attribute represents the number of rings present: none (n), one (o), and two (t).
- Ring Type: This attribute describes the type of ring: cobwebby (c), evanescent (e), flaring (f), large (l), none (n), pendant (p), sheathing (s), and zone (z).
- Spore Print Colour: This attribute indicates the colour of the spore print: black (k), brown (n), buff (b), chocolate (h), green (r), orange (o), purple (u), white (w), and yellow (y).
- Population: This attribute describes the mushroom population: abundant (a), clustered (c), numerous (n), scattered (s), several (v), and solitary (y).
- Habitat: This attribute indicates the habitat where the mushroom is typically found: grasses (g), leaves (l), meadows (m), paths (p), urban (u), waste (w), and woods (d).

3. Data preprocessing:
   Techniques to clean, normalise, and encode categorical data for machine learning algorithms
4. Model Deployment:
   Using of the different models such as Logistic Regression, SVM, Random Forest and Decision Tree Classifier to determine the best model by splitting the data sets and evaluating the model using precision, Recall, F1-Score and confusion matrix.
5. Software and Tools:
   - Language used: Python
   - Libraries used: Pandas, Numpy, Matplotlib, Scikit Learn, Data Manipulation and Visualization
   - Environment used to develop: Jupyter Notebook.
6. User Interface:
   - Application Development: Creation of a user-friendly interface, possibly a website, where users can input mushroom characteristics and receive edibility predictions.
   - Frameworks: Use of Flask framework for website creation.
   - Explanation Module: Integration of a module that provides explanations for the model's predictions, enhancing user understanding and trust
7. Deployment:
   Deployed using the Flask framework.

2.5 Constraints:

The accuracy and accessibility of the data from The Audubon Society Field Guide to North American Mushrooms are crucial to the project's success. Restraints might possibly impact the prediction model's performance and dependability if the data is imprecise, limited, or incomplete. Selecting relevant characteristics for the prediction model may provide challenges for the project. The accuracy with which the model can discriminate between edible and deadly mushrooms may be impacted by some characteristics that are stated in the field guide but are difficult to measure or have limited variability.

Important limitations are the prediction model's accuracy and dependability. To guarantee the security of users depending on its forecasts, the model has to reach high standards of precision, recall, and overall accuracy. If the model doesn't achieve certain performance measures, constraints could appear. One major limitation that might arise from machine learning models is their computational complexity, particularly when dealing with huge datasets and intricate algorithms. Inadequate computing resources might affect how quickly and effectively models are developed and assessed.

Regulations and ethical issues pertaining to food safety and data privacy must be followed by the initiative. Requirements like the Food and Drug Administration's (FDA) food safety rules may put limitations on the project's deployment plans and procedures. There could be limitations with creating an intuitive user interface that explains the model's predictions and conclusions to users. It is critical to guarantee usability, accessibility, and clarity for people with varying backgrounds and skill levels.

Requirements for the prediction model's dependability and generalizability include sufficient validation and testing. In order to enhance the model's performance, the project must evaluate its predictions in real-world settings and take user and domain expert comments into account. To guarantee smooth integration and the project's long-term viability, deployment environment constraints like scalability, compatibility with various operating systems, and maintenance needs must be taken into account.

2.6 Assumptions

The Audubon Society Field Guide to North American Mushrooms is considered to include accurate and trustworthy descriptions and classifications of mushrooms. Any errors or inconsistencies in the data might have an impact on how well the prediction model performs.

It is believed that the information from the field guide is typical of the larger population of North American mushrooms belonging to the Agaricus and Lepiota families. Even if a wide variety of species may be included in the guide, there could still be some variants or uncommon species that are not included in the data.

It is believed that the predictive model created using the field guide's data would perform well when applied to mushrooms that are not included in the dataset. This assumption is predicated on the idea that the characteristics included in the categorization process serve as indicators of toxicity or edibility for various species.

Certain morphological traits included in the field guide—like cap form, color, and smell—are thought to be reliable markers of a mushroom's toxicity or edibility. Although many species

may share similar traits, the model may need to take into consideration exceptions or deviations.

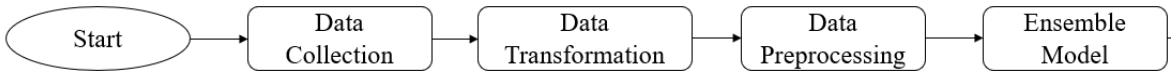
It is believed that those who use the prediction model are familiar with the fundamentals of mushroom identification and the dangers involved with eating wild mushrooms. The forecasts and explanations provided by the model are intended to enhance users' understanding and aid in decision-making, not to completely replace the advice of experts.

Chapter – 3

Design

The core functionality of the Mushroom Classification application revolves around predicting whether a mushroom is edible or poisonous based on its attributes. To achieve this, a decision tree classifier machine learning model was employed. This model is chosen for its interpretability and effectiveness in handling categorical data.

3.1 Process Flow



Data Collection

A comprehensive dataset of mushroom characteristics is utilized for training the machine learning model. This dataset includes features such as cap shape, cap surface, cap color, gill attachment, gill spacing, gill size, and several other attributes that describe the physical properties of mushrooms.

Data Preprocessing

The preprocessing of the dataset involves several critical steps to ensure it is in the right format for the algorithm:

- **Data Cleaning**: Any missing or inconsistent data is handled appropriately to ensure a clean dataset.
- **Encoding Categorical Variables**: Since the mushroom dataset contains categorical features, these are encoded into numerical values using techniques such as one-hot encoding.
- **Feature Scaling**: Although not always necessary for decision tree algorithms, feature scaling can be applied to normalize the data.

Model Training

The decision tree classifier is trained using the following steps:

- **Splitting the Data**: The dataset is divided into training and testing sets to evaluate the model's performance.
- **Training the Model**: The decision tree algorithm is trained on the training set, learning the relationships between the features and the target variable (edible or poisonous).
- **Hyperparameter Tuning**: Various hyperparameters of the decision tree, such as maximum depth and minimum samples per leaf, are tuned to optimize the model's performance.

Prediction

Once the model is trained, it can be used to predict the edibility of mushrooms based on new input data:

- **User Input**: Users input the characteristics of a mushroom through the application's user interface.
- **Data Processing**: The input data is pre-processed in the same way as the training data to ensure consistency.
- **Model Inference**: The processed input data is fed into the trained decision tree classifier, which outputs a prediction indicating whether the mushroom is edible or poisonous.

Model Evaluation

To ensure the reliability and robustness of the model, several evaluation metrics are used:

- **Accuracy**: Measures the overall correctness of the model's predictions.
- **Precision and Recall**: Evaluates the model's performance, particularly in correctly identifying poisonous mushrooms.
- **Confusion Matrix**: Provides insight into the number of true positive, true negative, false positive, and false negative predictions.

3.2 Logging

An essential component of every online application is logging, which aids in system monitoring, troubleshooting, and maintenance while offering insights into the behaviour of the program. Logging may record information regarding data input, model predictions, failures, and system performance for a web application that uses machine learning.

Logging will keep track when the user has logging and what page has, he accessed. When the user logs in, it will tell us whether the model has been loaded or not. When the user inputs the data, that data gets stored and the prediction of the model i.e., the result of the model is also saved along with date and time. It logs the errors in the system. It is crucial for fixing the system.

Chapter – 4

Performance

For your web application to provide consumers a quick, responsive, and seamless experience, performance optimization is essential. Performance factors for a machine learning web application include low latency, rapid model predictions, effective data processing, and best use of system resources. System scalability, cost-effectiveness, and user pleasure are all impacted by high performance.

For the system developed:

Sure, let's discuss the importance and effects of performance on your Mushroom Classification System using a Flask application and a Random Forest model.

1. Accuracy and Precision

- **Accuracy**: This metric indicates how often the model correctly classifies mushrooms as edible or poisonous. High accuracy ensures users get reliable predictions, which is critical when their health could be at risk.
- **Precision**: Precision in this context measures the proportion of true positive edible classifications out of all positive edible classifications. High precision ensures that when the model predicts a mushroom as edible, it is indeed safe to consume, minimizing false positives.

2. Recall and F1 Score

- **Recall**: Recall measures the proportion of actual poisonous mushrooms that are correctly identified by your model. High recall is crucial because missing a poisonous mushroom can have severe consequences.
- **F1 Score**: The F1 score balances precision and recall. For your system, a high F1 score indicates that the model effectively balances the risk of false positives and false negatives, providing a robust prediction system.

3. Model Generalization

- **Overfitting**: If Random Forest model performs well on training data but poorly on new data, it has overfitted. This means it has learned the noise in the training set, making it unreliable for real-world use.
- **Underfitting**: If the model performs poorly on both training and test data, it has underfitted, meaning it is too simplistic to capture the complexities of the mushroom data.
- **Generalization**: Ensuring that the model generalizes well means it can accurately classify new, unseen mushrooms, which is essential for real-world application.

4. Efficiency and Scalability

- **Computational Efficiency**: ML Model needs to provide predictions quickly, especially if integrated into a real-time application. Efficient models ensure that users do not experience significant delays when interacting with the system.

- **Scalability**: As the user base grows or as more mushroom data is added, your system should handle increased load without performance degradation. This is essential for maintaining a smooth user experience.

5. Robustness and Stability

- **Robustness**: The model should handle noisy or incomplete data gracefully. For instance, if some features of a mushroom are missing or inputted incorrectly, the model should still provide a reasonable prediction.
- **Stability**: The model should consistently perform well across different datasets. This stability builds trust among users, as they can rely on the model's predictions over time.

6. Bias and Variance

- **Bias**: If the model is too simplistic (high bias), it might misclassify mushrooms frequently, leading to underfitting. This could result in unsafe recommendations.
- **Variance**: If model is too complex (high variance), it might be overly sensitive to training data and fail to generalize (overfitting). This could also lead to unreliable predictions.
- **Bias-Variance Tradeoff**: Balancing bias and variance is crucial for minimizing errors. For your system, this balance ensures that the model is neither too simplistic nor too complex, providing accurate predictions.

7. Interpretability and Transparency

- **Interpretability**: Users and stakeholders should understand how the model arrives at its predictions. This is especially important in applications affecting health and safety. Transparency in the model's decision-making process helps build trust and facilitates troubleshooting.
- **Transparency**: By providing clear explanations of how predictions are made (e.g., which features contributed most to a decision), you increase user confidence in the system.

Importance of Performance in Your Mushroom Classification System

1. **User Trust and Adoption**
    - High-performing models ensure that users can trust the system's recommendations, encouraging wider adoption and reliance on the system.
2. **Safety and Health Impact**
    - Accurate predictions directly impact user safety, especially when distinguishing between edible and poisonous mushrooms. Misclassifications can have severe health consequences.
3. **Resource Utilization**
    - Efficient models make better use of computational resources, reducing costs and ensuring that the system can handle real-time predictions effectively.

4. **Continuous Improvement**
    - Monitoring performance metrics helps in continuously refining the model, ensuring it remains accurate and reliable as new data becomes available.
5. **Competitive Advantage**
    - A well-performing system provides a competitive edge by offering reliable and accurate predictions, distinguishing it from other similar applications.

Performance is crucial to ensure accurate, reliable, and efficient predictions. High performance translates to user trust, safety, and scalability, making the system practical and valuable in real-world scenarios. By focusing on various performance metrics and continuously improving the model, we can ensure that your system remains robust and trustworthy for its users.

Conclusion

The Mushroom Classification System, combining a robust Random Forest machine learning model with a user-friendly Flask application, provides a comprehensive solution for accurately classifying mushrooms as edible or poisonous. The system's design ensures high performance, reliability, and user satisfaction through careful consideration of accuracy, efficiency, scalability, and security.

By continuously monitoring and refining both the model and the application, the system remains adaptable and reliable, capable of meeting user needs effectively. This holistic approach ensures that the Mushroom Classification System is a practical and valuable tool, providing accurate predictions and a positive user experience. This comprehensive setup serves as a strong foundation for future enhancements and scaling, positioning the system as a reliable and trusted resource for users.