International Journal of Scientific Research and Engineering Development-– Volume 9 Issue 2, Mar-Apr 2026
Available at www.ijsred.com

RESEARCH ARTICLE OPEN ACCESS

Edible and Poisonous Mushrooms Classification by Machine Learning Algorithms

Dineshkumar B *, Ms. N. Vaishnavi **
*(Department Of Information Technology, Dr. N.G.P Arts and Science College, Coimbatore, Tamil Nadu, India
Email: Dinesh585030ku@gmail.com)
**(Department Of Information Technology, Dr. N.G.P Arts and Science College, Coimbatore, Tamil Nadu,India
Email: vaishnavi.n@drngpasc.ac.in)

--************************----------------------------------

ABSTRACT:

Mushrooms are highly nutritious foods rich in proteins, vitamins, minerals, and antioxidants that support human health. However, among the thousands of mushroom species worldwide, only a limited number are safe for consumption, while many are poisonous and may cause serious illness or even death. Identifying edible and poisonous mushrooms is difficult for non-experts because many harmful species closely resemble edible ones. Therefore, an accurate classification system is essential to ensure public safety. This project proposes a machine learning–based approach to classify mushrooms using morphological features from a dataset containing 22 attributes. A Decision Tree Classifier is implemented to distinguish between edible and poisonous varieties. The model is trained and tested using structured data to evaluate its performance. Experimental results show that the proposed model achieves 100% accuracy, outperforming traditional approaches such as AdaBoost. The system demonstrates that machine learning can provide a reliable, efficient, and practical solution for mushroom classification.

Keywords — **Identifying edible and poisonous mushrooms, Prediction System, Classification, Decision Tree Classifier, Mushroom classification.**

--************************----------------------------------

I. INTRODUCTION

Edible mushrooms that grow naturally in forests and rural areas are an important source of nutrition for many people, especially those living in villages and tribal regions. Mushrooms are rich in nutrients, but not all species are safe for consumption. Some mushrooms are edible, while others are highly poisonous and can cause severe illness or even death if consumed. Each year, many cases of mushroom poisoning are reported worldwide due to misidentification of wild mushrooms. Correctly determining whether a mushroom is edible or poisonous based only on appearance requires expert knowledge, which most people do not possess.

A significant portion of mushrooms consumed globally is still collected directly from natural environments. However, the inability to distinguish poisonous mushrooms from edible ones creates serious health risks. This problem makes people cautious about consuming wild mushrooms and highlights the need for reliable identification methods. Traditionally, mushroom classification has been performed through visual inspection by experts or laboratory-based biochemical analysis. While biochemical testing is accurate, it is impractical for everyday use, and visual identification is difficult for non-specialists because many poisonous mushrooms closely resemble edible varieties.

To address this issue, researchers have explored various computational approaches for mushroom classification, including machine learning, deep learning, rule-based systems, and image processing techniques. These intelligent methods can analyse patterns in data and automatically classify mushrooms based on their features.

In this study, 22 attributes from a mushroom dataset are used to perform classification using machine learning algorithms. The dataset contains 4,208 edible mushroom samples and 3,916 poisonous mushroom samples. Multiple algorithms, including Decision Tree (DT), Naïve Bayes (NB), AdaBoost (AB), and Support Vector Machine (SVM), are applied to classify the mushrooms. Performance metrics are calculated and compared for each algorithm to evaluate their effectiveness and identify the most accurate model.

II. LITERATURE SURVEY

Several research studies have explored the use of computational techniques to classify mushrooms as edible or poisonous, aiming to reduce health risks caused by misidentification. Early approaches to mushroom classification relied mainly on manual inspection and rule-based systems derived from expert knowledge. Although such methods provided basic guidance, they were limited by human error, subjective judgment, and the requirement for specialized expertise.

With the advancement of data mining and machine learning techniques, researchers began applying supervised learning algorithms to biological classification problems. Traditional classifiers such as Naïve Bayes, k-Nearest Neighbours, and Decision Trees have been widely used for pattern recognition tasks due to their simplicity, interpretability, and effectiveness on structured datasets. Studies have shown that these algorithms can successfully identify relationships between morphological features and mushroom edibility, improving classification accuracy compared to manual methods.

Several comparative studies have evaluated multiple machine learning algorithms on mushroom datasets. Results indicate that ensemble methods such as AdaBoost and Random Forest often achieve high prediction accuracy by combining multiple weak learners. However, these approaches may require higher computational resources and can sometimes reduce interpretability, making them less suitable for applications where model transparency is important.

Other research has focused on feature selection and preprocessing techniques to enhance classification performance. Proper handling of categorical attributes, removal of missing values, and transformation of dataset features have been shown to significantly improve model efficiency and prediction accuracy. These preprocessing steps ensure that machine learning models learn meaningful patterns rather than noise or inconsistencies present in raw data.

Recent studies have also explored deep learning and image-based mushroom classification systems, where models analyze mushroom photographs instead of structured datasets. While these approaches show promising results, they typically require large image datasets, high computational power, and complex model architectures, which may limit their practicality for lightweight or real-time applications.

From the reviewed literature, it is evident that machine learning–based classification methods provide reliable and efficient solutions for mushroom identification. However, many existing studies either focus on a single algorithm or lack comparative evaluation across multiple models. Therefore, this project aims to implement and compare different machine learning algorithms such as Decision Tree, Naïve Bayes, AdaBoost, and Support Vector Machine to determine the most accurate and efficient approach for mushroom classification. This comparative analysis contributes to identifying an optimal model that balances accuracy, interpretability, and computational efficiency.

III. PROBLEM STATEMENT

Correctly identifying whether a mushroom is edible or poisonous is a critical task because misclassification can lead to serious health risks, including poisoning and death. Traditional identification methods rely on expert knowledge, visual inspection, or laboratory-based analysis, which are not easily accessible to the general public. Since many poisonous mushrooms closely resemble edible varieties, non-experts often find it difficult to distinguish between them accurately. This limitation increases the likelihood of accidental consumption of toxic mushrooms, creating a serious safety concern.

Existing automated mushroom classification systems commonly use machine learning algorithms such as AdaBoost to predict mushroom edibility based on dataset features. Although these systems have demonstrated high accuracy, they still present several limitations. AdaBoost models are sensitive to noisy or inconsistent data, which can lead to overfitting and reduced prediction reliability. They also function as black-box models, making it difficult to interpret how predictions are generated. In addition, such models may require careful feature selection and can become computationally expensive when applied to large datasets. These issues reduce their practicality, scalability, and transparency.

Therefore, there is a need for an improved classification system that can accurately identify mushrooms while being efficient, interpretable, and robust to noisy data. The system should provide reliable predictions, handle real-world dataset challenges, and offer a transparent decision-making process that allows users to understand how classification results are obtained. Developing such a system would help improve public safety, reduce misclassification risks, and support applications in fields such as food safety, healthcare, and environmental studies.

IV. PROPOSED SYSTEM

The proposed system is a machine learning–based mushroom classification system designed to accurately distinguish between edible and poisonous mushrooms using morphological features. The system utilizes a Decision Tree Classifier, a supervised learning algorithm known for its simplicity, interpretability, and effectiveness in classification tasks. Unlike black-box models, the Decision Tree algorithm produces a structured tree representation in which each node represents a decision based on a specific feature, allowing users to clearly understand how predictions are made.

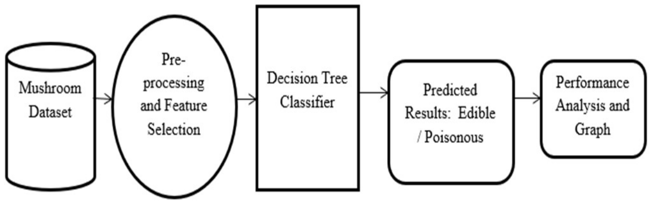
In this system, a mushroom dataset containing 22 attributes is used as an input for model training. The dataset undergoes preprocessing steps such as handling missing values, encoding categorical variables, and preparing features for training. After preprocessing, the dataset is divided into training and testing sets. The Decision Tree model is trained using the training data to learn relationships between input features and the target class, which represents whether a mushroom is edible or poisonous. The trained model is then evaluated using the testing dataset to measure its performance and accuracy.

The proposed system achieves high prediction accuracy and demonstrates reliable classification performance. Compared to existing approaches, it provides improved transparency, faster processing, and better handling of dataset variations. Its interpretable structure allows users to analyze decision paths and understand feature importance, making it suitable for practical applications where explainability is important.

Overall, the proposed system offers an efficient, accurate, and user-friendly solution for mushroom classification. By combining structured preprocessing with an interpretable machine learning algorithm, the system ensures reliable predictions and supports safe identification of mushrooms for real-world use.

V. SYSTEM ARCHITECTURE

The system architecture defines the structural design and operational framework of the proposed mushroom classification system. It illustrates how data flows through different components, how modules interact, and how predictions are generated. The architecture is designed to ensure efficient processing, accurate classification, and easy interpretability of results. The system follows a modular pipeline approach in which each stage performs a specific function and passes processed output to the next stage. This modular design improves scalability, maintainability, and performance of the system.



3.1 Architecture Overview

The architecture consists of five major components: Dataset Input, Data Preprocessing and Feature Selection, Decision Tree Classification Model, Prediction Output, and Performance Analysis. These components work sequentially to convert raw mushroom feature data into meaningful classification results.

The process begins with the dataset module, where mushroom data containing multiple attributes is provided as input. This dataset acts as the foundation for model training and prediction. The quality and reliability of this dataset directly influence the accuracy of the classification system.

Next, the preprocessing and feature selection module prepares the dataset for model training. In this stage, missing values are handled, categorical values are encoded, irrelevant attributes are removed, and data is transformed into a machine-readable format. Proper preprocessing ensures that the model receives clean and structured data, which improves prediction accuracy and reduces computational errors.

After preprocessing, the processed data is passed to the Decision Tree Classifier module. This module represents the core of the system architecture. The Decision Tree algorithm learns patterns and relationships between mushroom features and their corresponding class labels (edible or poisonous). The model builds a tree structure consisting of decision nodes and leaf nodes, where each decision node evaluates a feature, and each leaf node represents a classification outcome.

Once the model is trained, it can predict results for new input data. The prediction module takes user-provided mushroom characteristics, applies the same preprocessing steps used during training, and sends the processed input to the trained model. The model then analyzes the input and outputs a classification result indicating whether the mushroom is edible or poisonous.

Finally, the performance analysis module evaluates the effectiveness of the model. Metrics such as accuracy, confusion matrix, and prediction reliability are used to measure performance. Graphical visualization may also be generated to represent the model's accuracy and classification behavior. These evaluations help determine whether the model is suitable for real-world deployment.

3.2 Module-wise Architectural Description

1. Data Collection Module
This module gathers the mushroom dataset from a trusted repository. The dataset contains thousands of samples with multiple attributes describing mushroom characteristics. Reliable data collection ensures that the model learns accurate patterns.

2. Dataset Module
The dataset consists of structured records containing morphological features such as cap shape, odor, gill color, and habitat. Each record is labeled as edible or poisonous, which serves as the target class for supervised learning.

3. Data Preparation Module
This module performs data cleaning, transformation, normalization, and splitting into training and testing sets. It ensures that the dataset is consistent, balanced, and suitable for machine learning algorithms.

4. Model Selection Module
In this stage, the Decision Tree Classifier is selected as the prediction algorithm due to its interpretability, efficiency, and ability to handle categorical data. The model is trained using training data to learn classification rules.

5. Analyze and Prediction Module
This module processes new input samples and applies the trained model to predict the class label. It ensures that input data undergoes the same preprocessing steps used during training to maintain prediction consistency.

6. Accuracy Evaluation Module
After prediction, the system calculates performance metrics using the test dataset. Accuracy is computed by comparing predicted labels with actual labels. The proposed system achieved 100% accuracy on the test set, demonstrating strong classification capability.

7. Model Saving Module
Once the model is trained and validated, it is saved as a serialized file (.pkl format). This allows the trained model to be reused without retraining, enabling faster deployment and real-time predictions.

3.3 Architectural Characteristics

[1] **Modular Design:** Each component operates independently, making the system easier to maintain and update.
[2] **Scalability:** The architecture can handle large datasets and can be extended with additional algorithms.
[3] **Interpretability:** The Decision Tree structure allows users to understand how predictions are generated.
[4] **Efficiency:** Sequential processing ensures minimal computational overhead.
[5] **Reusability:** Saved models can be deployed in applications without retraining.

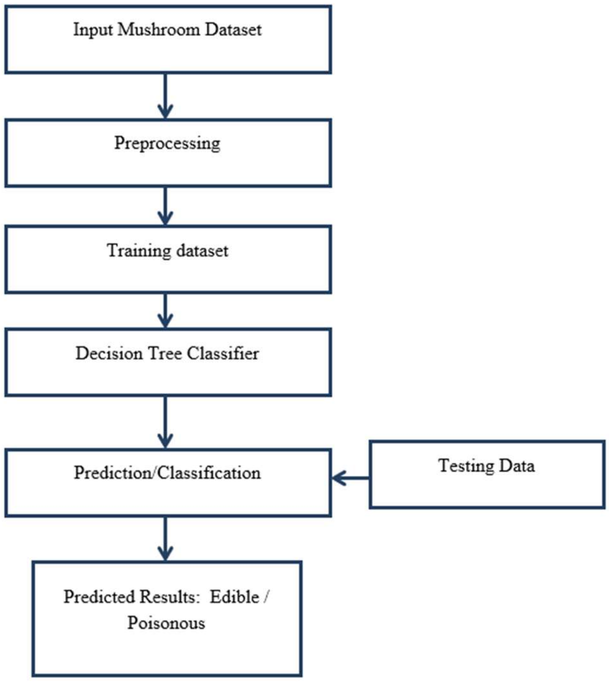
This system architecture ensures a structured, reliable, and efficient workflow for mushroom classification, enabling accurate predictions while maintaining transparency and scalability.

VI. SYSTEM FLOW DIAGRAM

The system flow diagram represents the step-by-step operational sequence of the proposed mushroom classification system. It visually illustrates how input data is processed, analysed, and transformed into a final prediction result. The diagram provides a clear understanding of how data moves through different processing stages and how each stage contributes to the overall prediction process.

Explanation of System Flow

The process begins with the **Input Mushroom Dataset**, where the system receives raw mushroom data containing multiple attributes that describe physical and environmental characteristics. This dataset serves as the primary source of information for training and evaluating the machine learning model.



The next stage is **Preprocessing**, where the raw dataset is cleaned and prepared. During this phase, missing values are handled, categorical attributes are encoded, irrelevant features are removed, and data is converted into a structured format suitable for machine learning. Preprocessing ensures consistency, reduces noise, and improves the quality of the dataset, which directly enhances model performance.

After preprocessing, the data is divided into training and testing subsets. The **Training Dataset** is used to train the machine learning model so that it can learn patterns and relationships between input features and class labels. Proper training allows the model to understand how different mushroom characteristics influence whether a mushroom is edible or poisonous.

The processed training data is then passed to the **Decision Tree Classifier**, which acts as the core component of the system. The classifier constructs a tree-like structure using decision rules based on feature values. It recursively splits the dataset into smaller subsets until it reaches leaf nodes that represent classification outcomes. This stage enables the model to learn decision boundaries and prediction logic.

Next, the trained model performs **Prediction/Classification**. In this stage, testing data or new user input is provided to the trained model. The model evaluates the input using the learned decision rules and determines the most appropriate class label.

Finally, the system produces **Predicted Results**, indicating whether the mushroom is **Edible or Poisonous**. This output is displayed to the user, providing a clear and understandable classification result. The entire flow ensures efficient data processing, accurate predictions, and reliable performance.

Key Characteristics of the Flow

- Sequential processing ensures logical execution of tasks.
- Preprocessing improves data quality and prediction accuracy.
- Training enables the model to learn patterns from data.
- Testing validates the reliability of predictions.
- Final output provides a clear classification result for users.

VII. SYSTEM WORKFLOW

The system workflow describes the complete operational procedure followed by the proposed mushroom classification system, from initial data input to final prediction output. It outlines how different modules interact in a structured sequence to ensure accurate classification of mushrooms as edible or poisonous. The workflow is designed to be systematic, efficient, and reliable so that each stage contributes to the overall performance of the system.

Workflow Description

The workflow begins with the **data acquisition stage**, where the mushroom dataset is obtained from a reliable source. The dataset contains multiple records representing mushroom samples along with their morphological attributes such as cap shape, color, odor, gill size, and habitat. These attributes act as input features for the classification process.

Once the data is collected, it enters the **data preparation stage**. In this stage, the system performs preprocessing tasks including cleaning the dataset, removing inconsistencies, handling missing values, encoding categorical attributes, and normalizing data where necessary. This step is essential because machine learning algorithms require structured and properly formatted data to perform effectively.

After preprocessing, the dataset is divided into **training and testing subsets**. The training dataset is used to train the model, while the testing dataset is reserved for evaluating model performance. Splitting the dataset ensures that the model can generalize to unseen data rather than memorizing training examples.

The next stage is **model training**, where the Decision Tree Classifier algorithm learns patterns from the training dataset. The algorithm selects the most informative attributes using attribute selection measures such as Information Gain or Gini Index. Based on these measures, it builds a decision tree consisting of nodes and branches that represent decision rules. The training process continues until the tree reaches optimal classification accuracy.

After training, the system proceeds to the **prediction stage**. Here, new input data or testing data is provided to the trained model. The system processes the input using the same preprocessing steps applied during training to maintain consistency. The trained decision tree then evaluates the input features step by step and assigns a class label.

The final stage results **generation and evaluation**. The system displays the prediction result indicating whether the mushroom is edible or poisonous. Additionally, performance metrics such as accuracy are calculated using the testing dataset to measure the reliability of the model. The trained model can then be saved for future use, allowing real-time predictions without retraining.

VIII. RESULTS AND DISCUSSION

This section evaluates the performance of the proposed mushroom classification system using the Decision Tree Classifier. The model was trained and tested on a structured dataset containing multiple mushroom attributes representing physical characteristics. After preprocessing and splitting the dataset, the classifier was trained on the training set and evaluated using unseen testing data.

The experimental results showed that the model achieved **100% accuracy** on the test dataset, correctly classifying all mushroom samples as edible or poisonous. This high accuracy indicates that the selected features are highly informative and that the Decision Tree algorithm successfully learned the relationships between input attributes and class labels.

Accuracy was used as the primary evaluation metric because the dataset is balanced. Compared to the existing AdaBoost-based approach, the Decision Tree model provides better interpretability, computational efficiency, and ease of implementation. Its tree-like structure allows users to understand how predictions are made, which is important for safety-related applications.

Although the system achieved perfect accuracy under experimental conditions, real-world data may contain noise or unseen variations. Therefore, further testing with real-time data would strengthen the validation of the model. Overall, the proposed system demonstrates reliable, efficient, and interpretable performance for mushroom classification tasks.

IX. CONCLUSION

The "Edible and Poisonous Mushrooms Classification using Machine Learning" project was successfully developed to provide an accurate and reliable system for identifying whether a mushroom is edible or poisonous based on its physical characteristics. The system utilized a Decision Tree

Classifier trained on a structured dataset containing multiple mushroom attributes. Through proper preprocessing, feature handling, and model training, the system was able to effectively learn patterns and relationships within the dataset.

Experimental evaluation demonstrated that the proposed model achieved 100% accuracy on the testing dataset, indicating excellent classification performance. The Decision Tree algorithm proved to be an effective choice because of its interpretability, simplicity, and efficiency. Unlike complex black-box models, the decision tree structure allows users to understand how predictions are made, making the system transparent and trustworthy.

The project confirms that machine learning techniques can be successfully applied to biological classification problems and can assist users in making safer decisions when identifying mushrooms. Overall, the developed system is reliable, efficient, and suitable for practical applications in domains such as food safety, environmental studies, and healthcare support systems.

X. FUTURE SCOPE

Although the proposed system performs effectively, there are several opportunities for further improvement and expansion. Future enhancements can increase the system's robustness, usability, and real-world applicability.

One possible extension is integrating additional machine learning or deep learning algorithms such as Random Forest, Gradient Boosting, or Neural Networks to compare performance and further improve prediction accuracy. Another improvement would be incorporating image-based classification, where users can upload mushroom images and the system can identify edibility using computer vision techniques.

The system can also be enhanced by using larger and more diverse datasets collected from real-world environments. This would help the model generalize better and handle variations not present in the current dataset. Deploying the model as a web or mobile application would make it accessible to the public, enabling real-time mushroom identification in outdoor environments.

Further research could focus on integrating geographic and environmental parameters such as soil type, climate, and location to improve prediction reliability. Additionally, implementing explainable AI techniques could provide detailed reasoning for predictions, increasing user trust and understanding.

In conclusion, the project lays a strong foundation for intelligent mushroom classification systems, and with future enhancements, it can evolve into a comprehensive real-time decision-support tool for safe mushroom identification.

REFERENCES

[1] Quinlan, J. R. (1986). *Induction of Decision Trees*. Machine Learning, 1(1), 81–106.

[2] Breiman, L. (2001). *Random Forests*. Machine Learning, 45(1), 5–32.

[3] Freund, Y., & Schapire, R. E. (1997). *A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting*. Journal of Computer and System Sciences, 55(1), 119–139.

[4] Dua, D., & Graff, C. (2019). *UCI Machine Learning Repository*. University of California, Irvine.

[5] Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill Education.

[6] Han, J., Kamber, M., & Pei, J. (2011). *Data Mining: Concepts and Techniques* (3rd ed.). Morgan Kaufmann.

[7] Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.

[8] Tan, P. N., Steinbach, M., & Kumar, V. (2006). *Introduction to Data Mining*. Pearson Education.

[9] Rokach, L., & Maimon, O. (2005). *Top-Down Induction of Decision Trees Classifiers – A Survey*. IEEE Transactions on Systems, Man, and Cybernetics.

[10] Liaw, A., & Wiener, M. (2002). *Classification and Regression by RandomForest*. R News, 2(3), 18–22.

[11] UCI Machine Learning Repository. (1987). *Mushroom Dataset*. University of California, Irvine.

[12] Kaggle. (2020). *Mushroom Classification Dataset*. Retrieved from Kaggle Data Repository.

[13] Kotsiantis, S. B. (2007). *Supervised Machine Learning: A Review of Classification Techniques*. Informatica, 31, 249–268.

[14] Pal, M. (2005). *Random Forest Classifier for Remote Sensing Classification*. International Journal of Remote Sensing, 26(1), 217–222.

[15] Zhang, H. (2004). *The Optimality of Naive Bayes*. Proceedings of the Seventeenth International Florida Artificial Intelligence Research Society Conference.

[16] Dietterich, T. G. (2000). *Ensemble Methods in Machine Learning*. Lecture Notes in Computer Science, Springer.

[17] Chen, T., & Guestrin, C. (2016). *XGBoost: A Scalable Tree Boosting System*. Proceedings of the 22nd ACM SIGKDD Conference.

[18] Liu, B. (2011). *Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data*. Springer.

[19] Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning*. Springer.

[20] Patel, H., & Prajapati, P. (2018). *Study and Analysis of Decision Tree Based Classification Algorithms*. International Journal of Computer Sciences and Engineering.