Shroom for Improvement: Comparing Performance and Interpretability of Mushroom Classification Models

Derrick Sun
Purdue University
West Lafayette, Indiana, USA

Jaison Joseph George
Purdue University
West Lafayette, Indiana, USA

Krishna Dudani
Purdue University
West Lafayette, Indiana, USA

Omkar Pote
Purdue University
West Lafayette, Indiana, USA

Abstract

Between 1999 and 2016, approximately 133,700 mushroom exposure cases and 52 fatalities were reported in the U.S. [4], creating a the need for more accurate identification of poisonous mushrooms. While classification models such as neural networks and decision trees achieve high accuracy, they often lack interpretability. This study uses models including decision trees, Naive Bayes classifiers, and neural networks on the UCI Mushroom dataset. In addition to accuracy, we propose evaluating the interpretability of our models in the case of feature attribution using LIME on K-medoids . Our findings suggest that while simpler models like Naive Bayes are surprisingly effective, interpretability methods enhance our understanding of feature importance and the nuanced differences between different models with similar accuracy.

Keywords

Mushroom classification, Interpretability, Naive Bayes, Neural Networks, Decision Trees, LIME

ACM Reference Format:
Derrick Sun, Jaison Joseph George, Krishna Dudani, and Omkar Pote. 2018. Shroom for Improvement: Comparing Performance and Interpretability of Mushroom Classification Models. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation email (Conference acronym 'XX)*. ACM, New York, NY, USA, 6 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

The accurate identification of poisonous mushrooms is a real-world classification problem with serious consequences. Between 1999 and 2016, the United States reported over 130,000 mushroom exposure cases and dozens of fatalities. While machine learning models have been widely applied to this problem and can achieve near-perfect accuracy, many of these models operate as black boxes, making it difficult to understand their decision-making process.

In this project, we use the UCI Mushroom dataset [1] to explore the balance between model performance and interpretability. We evaluate three supervised learning models — a decision tree classifier implemented from scratch, a Naive Bayes classifier, and a four-layer neural network with embedding layers. While the dataset is known to be easily separable, our focus extends beyond raw accuracy to include the explainability of model predictions.

To evaluate local interpretability, we apply LIME [6] (Local Interpretable Model-Agnostic Explanations) to a set of representative test instances selected using clustering. This method allows us to summarize how each model makes decisions across diverse regions of the input space.

Our findings reinforce that simpler models can match neural networks in accuracy on this dataset, while also providing greater transparency. The application of LIME across representative clusters further reveals how each model prioritizes different features when classifying mushrooms as edible or poisonous.

The rest of the paper is organized as follows. Section 2 describes the dataset and preprocessing. Section 3 introduces the models used. Section 4 presents our interpretability analysis using LIME. Section 5 discusses results and conclusions.[1]

2 Dataset and Preprocessing

The UCI Mushroom dataset consists of 8124 instances and 22 categorical attributes, each representing a feature of a mushroom (e.g., odor, cap shape, gill color). The target variable indicates whether a mushroom is edible or poisonous.

Since all features are categorical, we applied label encoding to convert string-based categories into integer representations. To preserve interpretability, each column was encoded using a separate 'LabelEncoder', allowing for reversible mappings between encoded values and original categories. This preprocessing step enabled compatibility with both traditional models and neural networks using embedding layers.

3 Models Overview

3.1 Decision Tree Classifier

We implemented a decision tree classifier from scratch using the ID3 algorithm [5] to promote transparency and maintain full control over the model's learning process. At each node, the algorithm selects the feature that maximizes information gain and splits the

Conference acronym 'XX, Woodstock, NY

ACM ISBN 978-1-4503-XXXX-X/2018/06
https://doi.org/XXXXXXX.XXXXXXX

[1] Our code can be found at https://github.com/DerrickhSun/CS573MushroomProject

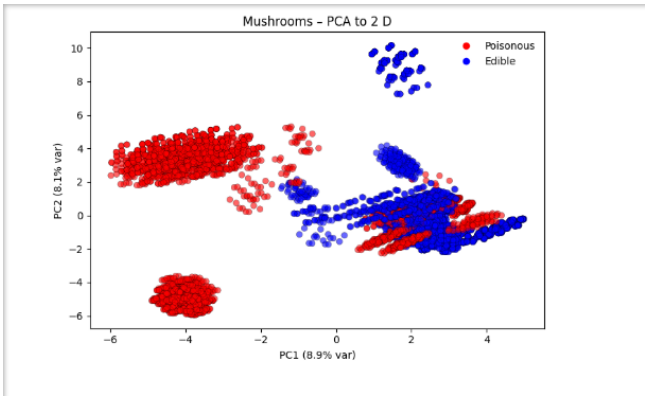

Figure 1: Visualization of the mushroom dataset in 2 dimensions using Principal Component Analysis

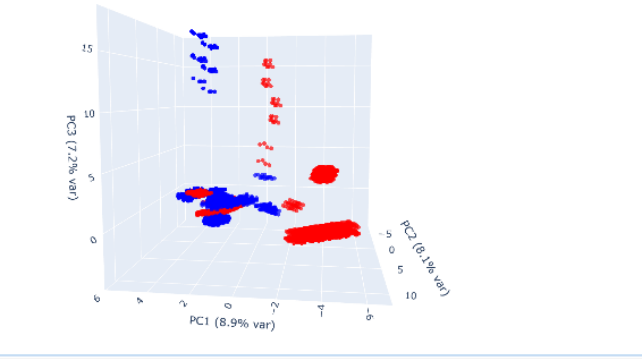

Figure 2: Visualization of the mushroom dataset in 2 dimensions using Principal Component Analysis

dataset accordingly to reduce class entropy. The recursion terminates when all samples at a node belong to the same class, no features remain, or information gain drops to zero. The implicit assumption made when using the model is that Each internal node of the tree represents a decision rule based on a single feature, and each path from the root to a leaf node corresponds to a complete classification rule. This structure makes decision trees inherently interpretable, allowing users to trace the exact splits for each feature that lead to a particular prediction.

Objective Function. Given a dataset D with features $X_1, X_2, \ldots, X_n$ and a target variable Y, the objective of the ID3 algorithm is to construct a decision tree that maximizes classification accuracy by recursively selecting the feature that provides the highest information gain at each node.

Formally, the objective is:

$$\text{Select feature } X_i \text{ such that } IG(D, X_i) = \max_j IG(D, X_j)$$

The recursive splitting continues until one of the following conditions is met:

Figure 3: A visual representation of our decision tree model

- All instances in the subset belong to the same class.
- There are no remaining features to split on.
- The information gain from further splitting is zero.

Entropy and Information Gain. Let D be a dataset with n samples and a target variable Y with k distinct classes. The entropy of Y is defined as:

$$H(Y) = -\sum_{i=1}^{k} p_i \log_2 p_i$$

where p_i is the proportion of instances in class i.
For a feature X, the entropy of Y after splitting on X is:

$$H(Y|X) = \sum_{v \in \text{Values}(X)} \frac{|D_v|}{|D|} H(Y_v)$$

where:

- $\text{Values}(X)$ is the set of unique values of feature X
- D_v is the subset of D where $X = v$
- Y_v is the target labels in subset D_v

The information gain obtained by splitting on feature X is:

$$IG(D, X) = H(Y) - H(Y|X)$$

To further enhance interpretability, we print the tree structure in a human-readable format, exposing the most discriminative features and the specific attribute values that determine whether a mushroom is classified as edible or poisonous.

The UCI Mushroom dataset is known to be linearly separable, and decision trees have been shown to achieve 100% accuracy on it without overfitting. Our scratch implementation validates this, reaching 100% accuracy on the full training data. This confirms both the effectiveness of the model and the suitability of this dataset as a benchmark for interpretable classification.

Algorithm 1 ID3 Algorithm

function ID3(D, F)
  if all labels in D are the same **then**
    return Leaf node with that label
  end if
  if F is empty **then**
    return Leaf node with majority label in D
  end if
  $f_{\text{best}} \leftarrow \arg\max_{f \in F} \text{InformationGain}(D, f)$
  Create a node N that splits on f_{best}
  for all values v of f_{best} **do**
    $D_v \leftarrow \{x \in D \mid x[f_{\text{best}}] = v\}$
    if D_v is empty **then**
      Add a leaf with majority label in D to N
    else
      $F' \leftarrow F \setminus \{f_{\text{best}}\}$
      Add ID3(D_v, F') as a child of N for value v
    end if
  end for
  return N
end function

3.2 Naive Bayes Classifier

Naive Bayes is a probabilistic classifier based on Bayes' Theorem, which assumes that features are conditionally independent given the class label. While this assumption is often violated in real world data sets, this model can and often does still yield good performance - especially on datasets with clear separability, such as the UCI Mushroom dataset.

For our implementation, we observed that not all output classes were associated with every possible feature value. This sparsity introduced the possibility of zero probabilities, which would cause the classifier to incorrectly assign a probability of zero to a class. To address this, we used Laplace smoothing (i.e., pseudocounts) to ensure all feature-class combinations had non-zero probabilities. We calculated the prior probabilities for each output class using frequency-based estimation: we divide the count of instances of a particular output class by the total number of training instances.

This model is particularly well-suited to categorical data, as it operates directly on feature distributions rather than requiring numerical representations. Given the simplicity of the dataset and its near linear separability (based on visualization of the data after it had been transformed with Principal Component Analysis into 3 dimensions), the Naive Bayes model was able to achieve perfect training accuracy, matching the performance of more complex models like decision trees.

Naive Bayes also offers some inherent interpretability. Each feature's contribution to the final prediction can be understood by examining the likelihoods it assigns to each class. While this interpretability is less direct than a decision tree's, it still allows for insight into which features (e.g., odor, gill color) drive the prediction.

3.3 Neural Network

3.3.1 Implementation. We implemented a four-layer neural network to classify mushrooms based on categorical attributes. To handle categorical input, we used an embedding layer that learned dense vector representations of each feature value. This enabled the network to capture semantic relationships among feature categories, such as similarities in odor, cap color, or stalk shape.

The architecture consisted of 4 layers with 1000 neurons each, primarily using ReLU activation functions, though some layers also employed sigmoid activations. The final layer produced logits for binary classification, which were transformed into probabilities via softmax only during evaluation. We used the cross-entropy loss function, as it performed better than both mean squared error (MSE) and binary cross-entropy (BCE) in our testing. Moreover, MSE is ill suited in general for binary classification. Training was conducted using stochastic gradient descent over 300 epochs.

3.3.2 Performance. Early training runs with smaller hidden layers (e.g., under 100 neurons) led to poor performance, with the network producing near-uniform predictions close to 50% for both classes. Increasing the layer size to 200 showed minor improvements, but the model remained hesitant to commit to confident predictions. Only after scaling the hidden layers to 1000 neurons did the network begin learning distinct decision boundaries, producing more varied probability outputs (e.g., 20–80%). At this configuration, the model achieved 97% classification accuracy on held-out data.

Despite its strong performance, the neural network required significantly more training time than our other models, and its internal representations were less interpretable. We therefore applied LIME to extract local feature contributions and better understand the rationale behind its predictions.

4 Interpretability with LIME

While past work [3] has applied LIME to individual instances for local explanation, we found that relying on a single sample may not sufficiently reflect model behavior across the input space. To address this, we implemented a clustering-based sampling strategy to select representative data points for interpretation.

Specifically, we applied K-Medoids [2] clustering to the test set (encoded as numeric vectors) to identify central points from distinct regions of the feature space. For K-medoids, the assumption is that there are K-clusters in the dataset. For each resulting cluster center, we used LIME to generate local explanations and extract the top contributing features to the model's prediction.

This approach allowed us to explore a diverse and interpretable subset of the test distribution, highlighting consistent patterns in how different models make decisions across mushroom subtypes. For instance, across all clusters classified as "poisonous," features such as `odor`, `spore print color`, and `gill size` repeatedly appeared as the dominant explanatory factors. In contrast, clusters corresponding to "edible" classes more frequently emphasized `cap shape`, `stalk color above ring`, and `habitat`.

4.1 LIME for NBC

We clustered the test set into 40 medoids (Manhattan-distance K-Medoids), then ran LIME on each medoid under three different

random seeds (120 local explanations total). In Figure 4, `pct_in_top5` is the percentage of explanations in which that feature appeared among the top five, and `mean_signed_weight` is its average signed LIME coefficient (positive pushes prediction toward "poisonous," negative toward "edible").

	feature	pct_in_top5	mean_signed_weight
1	ring-type=p	67.5	-0.15450106379149192
2	talk-surface-above-ring=s	66.66666666666667	-0.12091624657236666
3	odor=n	62.5	-0.37925939510057527
4	gill-size=b	52.5	-0.12288219606124273
5	population=v	35.833333333333336	0.11263952265854241
6	talk-surface-below-ring=s	21.666666666666668	-0.11114122612661463
7	odor=f	20.0	0.3890307048634066
8	talk-surface-above-ring=k	17.5	0.15021719563805694
9	talk-surface-below-ring=k	15.833333333333334	0.14558679518531278
10	gill-color=b	15.0	0.30963868189510385
11	ring-type=l	15.0	0.27461485191400536
12	bruises=t	15.0	-0.10886595107819227
13	spore-print-color=h	14.166666666666666	0.16609034824218621
14	gill-size=n	14.166666666666666	0.12381579420630912
15	spore-print-color=n	11.666666666666666	-0.12603680451525587
16	odor=s	7.5	0.2694566386938517
17	bruises=f	7.5	0.10420116032064639
18	spore-print-color=k	7.5	-0.13140467496555408
19	gill-color=n	7.5	-0.15845079042221105
20	stalk-color-above-ring=g	7.5	-0.2694349832562712
21	odor=y	2.5	0.2586940168710254
22	population=n	2.5	-0.23713621094723894
23	stalk-color-below-ring=g	2.5	-0.27946965079453
24	odor=l	2.5	-0.33040899169867327
25	odor=a	2.5	-0.34330581137732813
26	gill-spacing=c	1.6666666666666667	0.13693081107759952
27	population=a	1.6666666666666667	-0.23130774228969397
28	gill-color=w	0.8333333333333334	-0.12063054211130463
29	gill-spacing=w	0.8333333333333334	-0.13361292129612531

Figure 4: LIME summary of Naïve Bayes important features (40 medoids × 3 seeds).

Across all runs, only 29 distinct dummies ever cracked the top five. The five most frequent are:

(1) **ring-type = pendant** (`ring-type=p`): appeared in 67.5% of explanations, mean weight -0.1545
(2) **stalk-surface-above-ring = smooth** (`stalk-surface-above-ring=s`): 66.7%, weight -0.1209
(3) **odor = none** (`odor=n`): 62.5%, weight -0.3793
(4) **gill-size = broad** (`gill-size=b`): 52.5%, weight -0.1229
(5) **population = several** (`population=v`): 35.3%, weight +0.1124

In plain English:

- Mushrooms with a *pendant* ring or a *smooth* stalk-surface above the ring are more likely to be classified as *edible*. - Lack of odor ("no odor") strongly pushes the model toward "edible," while any odor (e.g. odor = foul, pungent, etc.) would push toward "poisonous." - Broad gills also favor "edible." - Conversely, a *several* population pattern (cluster size "several") slightly increases the chance of "poisonous."

All other features—such as bruising, spore-print colors, habitat, etc.—occurred far less often in the top five, indicating that NB relies on this small core of categorical signals for over 93% accuracy on held-out data.

4.2 LIME for Decision Trees

To probe our Decision Tree's behavior on held-out examples, we again selected 40 medoids from the one-hot–encoded test set (Manhattan-distance K-Medoids, $k = 40$, seed 0) and generated LIME explanations under three random seeds (0, 1, 2), yielding $40 \times 3 = 120$ local fits. In each explanation, we requested the top five one-hot dummies and filtered out any feature = 0 entries, so that only active categories (feature = 1) appear.

Gill-attachment = free appears in 100% of explanations with a large positive weight (+0.58), strongly pushing predictions toward *poisonous*. Veil-color = white occurs in nearly 60% of local fits with a negligible negative weight (−0.0007), indicating a slight shift toward edible. Ring-number = one and gill-color = brown also frequently influence the decision, though with much smaller weights. Surprisingly, the global "odor" split does not appear among the top dummies here: most medoids lie in pure-leaf regions where flipping odor alone does not alter the local prediction.

This analysis reveals that, while *odor* is the tree's very first split globally, gill-attachment, veil-color, and ring-number dominate the local decision boundaries for most representative medoids. It underscores the difference between global split importance and the local sensitivity captured by LIME.

4.3 LIME for Neural Networks

We extended our medoid-based LIME pipeline to the neural-network classifier by first one-hot encoding all 22 categorical predictors and selecting 40 representative medoids from the test set via K-Medoids clustering (Manhattan distance, $k = 40$, seed = 0). Because the network was trained on integer-encoded inputs with embedding layers, each one-hot medoid was converted back into its original 22-dimensional label-encoded form before being fed to the model. We then generated local explanations under three different random seeds (0,1,2), requesting all one-hot features but filtering each explanation to retain only the first five "active" dummy variables (those whose one-hot value is 1). Finally, we aggregated across the resulting 120 explanations (40 medoids×3 seeds) by computing, for each dummy feature, (1) the percentage of explanations in which it appeared among the top five most important features (`pct_in_top5`) and (2) its average signed LIME weight across those runs (`mean_signed_weight`).

Figure 6 shows that the smooth surface above the ring (`stalk-surface-above-rings`) is the most consistently influential feature, appearing in 70% of explanations with a positive mean weight of approximately 0.012. Absence of odor (`odor_n`) follows closely at 62.5% presence and a mean weight of +0.038, indicating that mushrooms without a detectable smell drive the model slightly toward a "edible" prediction. The network also frequently relied on the white stalk surface below the ring (stalk-color-below-ring_w,

Delimiter: ,

	feature	pct_in_top5	mean_signed_weight
1	gill-attachment_f	100	0.5830159
2	veil-color_w	59.16666667	-0.000652996
3	ring-number_o	36.66666667	0.000257393
4	gill-color_n	24.16666667	0.015248761
5	gill-color_b	15	0.018897261
6	stalk-surface-below-ring_s	13.33333333	-0.001285491
7	gill-size_b	12.5	0.002759246
8	gill-spacing_c	12.5	0.0019932
9	stalk-surface-above-ring_s	10.83333333	-7.75E-05
10	bruises_t	9.166666667	0.000897394
11	cap-shape_x	9.166666667	0.000452047
12	stalk-color-below-ring_w	8.333333333	-0.001194784
13	stalk-root_b	8.333333333	-0.001289796
14	gill-color_u	7.5	0.00449755
15	population_v	7.5	0.001268288
16	stalk-shape_t	7.5	0.000693324
17	ring-type_p	6.666666667	0.001397131
18	cap-color_n	6.666666667	-6.13E-06
19	odor_n	6.666666667	-0.000141681
20	cap-surface_y	6.666666667	-0.000188448
21	habitat_d	6.666666667	-0.001047368
22	spore-print-color_k	6.666666667	-0.002242235
23	cap-shape_f	5.833333333	0.003222167
24	spore-print-color_n	5.833333333	0.000757571
25	stalk-color-above-ring_w	5.833333333	-0.000672325
26	gill-color_g	5	0.014622568
27	stalk-color-above-ring_p	5	0.000196375
28	stalk-surface-above-ring_k	5	-8.53E-05
29	odor_s	5	-0.000847434
30	bruises_f	5	-0.001378089
31	cap-surface_f	4.166666667	0.0002552
32	stalk-color-below-ring_p	4.166666667	-0.000551768
33	population_y	4.166666667	-0.000713262
34	odor_f	3.333333333	0.000502755
35	gill-color_w	3.333333333	-0.000681453
36	stalk-shape_e	3.333333333	-0.00177515
37	gill-color_p	3.333333333	-0.002643409
38	cap-color_e	3.333333333	-0.004362244
39	gill-color_k	2.5	0.011778621
40	ring-type_l	2.5	0.004459581
41	stalk-root_c	2.5	0.003208084
42	ring-type_e	2.5	0.001963513
43	cap-color_g	2.5	-0.001490138
44	habitat_p	2.5	-0.001515379
45	odor_l	1.666666667	0.009925497
46	stalk-color-above-ring_n	1.666666667	0.00920996
47	stalk-color-above-ring_b	1.666666667	0.005382196
48	stalk-color-above-ring_g	1.666666667	0.005374075
49	stalk-surface-below-ring_k	1.666666667	0.004074417
50	stalk-color-below-ring_b	1.666666667	0.003941373
51	gill-spacing_w	1.666666667	0.00170666
52	habitat_g	1.666666667	0.001363961
53	cap-color_y	1.666666667	0.000423591
54	odor_y	1.666666667	-0.000155546
55	spore-print-color_h	1.666666667	-0.000618519
56	spore-print-color_w	1.666666667	-0.000884728
57	stalk-color-below-ring_n	1.666666667	-0.001204103
58	odor_a	1.666666667	-0.001288831
59	odor_p	1.666666667	-0.002907049

Delimiter: ,

	feature	pct_in_top5	mean_signed_weight
1	stalk-surface-above-ring_s	70.0	0.011635743743592126
2	odor_n	62.5	0.03832892209933725
3	stalk-color-below-ring_w	46.666666666666664	0.011111319373823184
4	spore-print-color_k	37.5	0.022033829009933264
5	population_v	33.333333333333336	-0.010626135878764167
6	spore-print-color_n	30.0	0.02790019730812244
7	gill-color_n	25.0	0.029301746063994955
8	stalk-color-above-ring_p	20.0	-0.01337119623347995
9	odor_f	20.0	-0.06481533856634991
10	gill-color_w	17.5	0.015621191109186857
11	ring-number_o	17.5	-0.010365264666885978
12	stalk-surface-below-ring_k	17.5	-0.01653745368816409
13	spore-print-color_w	17.5	-0.01765831302470791
14	stalk-surface-above-ring_k	17.5	-0.04057920396768128
15	veil-color_w	16.666666666666668	0.011430955506384812
16	stalk-color-below-ring_p	15.0	-0.010737026284955037
17	gill-color_b	15.0	-0.017516151040778867
18	spore-print-color_h	15.0	-0.021388325036701638
19	stalk-color-above-ring_n	2.5	-0.01953680640504606
20	odor_l	2.5	-0.02076992281013498
21	gill-color_u	0.8333333333333334	0.010352085282435032

Figure 6: Aggregated LIME summary of Neural Network important features (120 local explanations). Note: in this case positive mean signed weights mean more likely to predict edible instead of poisonous, unlike the other 2 models

46.7%), which was similar to the naive bayes model. As for the black spore-print color (`spore-print-color_k`, 37.5%), and clustered population (`population_v`, 33.3%; mean weight −0.011), the latter's negative weight reflecting a tendency to push the local prediction toward "edible." These results closely mirror the patterns observed in our Naive-Bayes analyses, suggesting that odor absence and stalk characteristics such as stalk-surface-above-ring are robust, model-agnostic indicators of lack of mushroom toxicity.

Interestingly, the mean signed weights of LIME applied to neural networks seem to be significantly smaller for every single feature outputted in the summary. This suggests that the neural network tried to give more equal importance to every variable while maintaining high accuracy, unlike the decision tree.

4.4 Cross-Model Comparison

All three classifiers—decision tree, Naive Bayes, and neural network—were evaluated on the same single 70/30 train–test split. On that split, our handcrafted decision tree achieved 100% test accuracy, the Naive Bayes classifier reached 94.59%, and the neural network about 93.8%. Despite these similar overall scores, the LIME summaries in Figures 4, 5, and 6 uncover important distinctions in how each model uses the input features as well as some commonalities. The decision tree had some commonalities with the other models, seeming to be more likely to predict poisonous when with certain

stalk-surface traits such as the stalk-surface-above-ring and stalk-surface-below-ring. However, these did not seem to be as prioritized as other feature values, such as a white veil color, having one ring number, and a free gill attachment. There is a noticeable difference in accuracy from the decision tree and the other two models, and the tree itself was very shallow and had a depth of 4 only, which initially suggests that the results of using LIME on decision trees may be superior.

However, it is also important to consider that the decision tree's global split on `odor` manifests locally in pure-leaf regions of the decision tree, which causes most medoid explanations to the aforementioned secondary attributes related to gill attachment, veil color, and ringe number. . (Figure 5), even though odor was its very first split. By contrast, Neural Networks produces the smallest (and potentially more interpretable) explanation set: only 21 one-hot dummies ever appear among the top five features across 120 medoid explanations (Figure 6) with stalk-surface-above-ring, no odor, white stalk color below ring, and black spore print color dominating.

The Naive Bayes Classifier sits between the two: it again highlights odor absence (`odor_n`, present in 62.5% of explanations) but also elevates a partial ring type and some stalk surface traits—most notably `talk-surface-above-ring_s` (66.67%) which is common to the neural networks summary as well. (Figure 6).

In summary, while all models learn the same separability in the data, the tree's local behavior defers to its deep leaves, NBC's explanations remain highly concentrated on a handful of features, and the network captures a richer mix of primary and secondary signals. This illustrates the trade-off between parsimony and nuance when choosing an interpretable model for safety-critical mushroom classification.

5 Conclusion

This study explored the balance between predictive performance and interpretability for classifying mushrooms as edible or poisonous using the UCI Mushroom dataset. Three distinct classification models—decision trees, Naive Bayes classifiers, and neural networks—were evaluated, complemented by interpretability analysis through LIME and K-medoids methods.

Our experiments confirm that simpler, inherently interpretable models like decision trees and Naive Bayes can achieve exceptional accuracy, comparable to more complex neural networks. Specifically, our handcrafted decision tree achieved perfect accuracy on this dataset, highlighting its effectiveness for easily separable categorical data. Despite neural networks' robust predictive capabilities, their black-box nature necessitated additional interpretability methods like LIME, which successfully provided insights into feature importance and decision rationale.

Interpretability analysis revealed nuanced differences in how each model leveraged dataset features. While odor consistently emerged as a critical feature across all models, the decision tree prominently emphasized gill attachment and veil color locally due to its hierarchical structure. In contrast, Naive Bayes and neural network models consistently identified odor absence and specific stalk surface characteristics as dominant predictors. Notably, the neural network balanced its reliance across multiple features, reflecting a richer, albeit less transparent, decision process.

Ultimately, our findings underline the importance of selecting models aligned with interpretability requirements, especially in safety-critical applications like mushroom identification. Incorporating interpretability frameworks such as LIME and K-medoids significantly enhances trust and usability, guiding future efforts toward transparent, explainable AI systems.

6 Contributions

Derrick coded the neural network model. Jaison wrote the Naive Bayes Classifier from scratch. Krishna performed data analysis, model analysis, and wrote most of the final paper. Omkar coded decision trees.

References

[1] 1981. Mushroom. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5959T.

[2] 2008. Partitioning around medoids (program PAM). In *Finding Groups in Data*. John Wiley & Sons, Inc., Hoboken, NJ, USA, 68–125.

[3] Md. Sabbir Ahmed, Sadia Afrose, Ashik Adnan, Nazifa Khanom, Md Sabbir Hossain, Md Humaion Kabir Mehedi, and Annajiat Alim Rasel. 2022. Comparative Analysis of Interpretable Mushroom Classification using Several Machine Learning Models. In *2022 25th International Conference on Computer and Information Technology (ICCIT)*. 31–36. doi:10.1109/ICCIT57492.2022.10055555

[4] William E Brandenburg and Karlee J Ward. 2018. Mushroom poisoning epidemiology in the United States. *Mycologia* 110, 4 (July 2018), 637–641.

[5] J R Quinlan. 1986. Induction of decision trees. *Mach. Learn.* 1, 1 (March 1986), 81–106.

[6] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (San Francisco, California, USA) *(KDD '16)*. Association for Computing Machinery, New York, NY, USA, 1135–1144. doi:10.1145/2939672.2939778