Bolete Species and Edibility Classification

November 2020

Liza Bialik
University of Massachusetts, Amherst
rbialik@umass.edu

Jack Kenney
University of Massachuetts, Amherst
jnkenney@umass.edu

1. Introduction

In this project we investigate the problem of Bolete identification. Boletes are mushrooms that carry their spores in tubules called pores, as opposed to gills. Boletes can be difficult to identify at a species level, even for expert mycologists.

Bolete identification by humans usually involves filtering by characteristics exhibited by the mushroom like color, shape, and texture to narrow down potential matches [1]. We compare the performance of models with and without an explicit representation of such characteristics.

Many of these characteristics are textural such as netting or dots [10]. For this reason, we compare models that are trained on low resolution 256×256 images and high resolution 512×512 images.

We challenge the generally agreed upon idea that a positive species identification is necessary to accurately determine edibility by comparing classification models trained on species and edibility status.

In total, we create and compare three models: one that learns characteristics, one that directly learns species, and one that directly learns edibility from images. This is done for both high and low resolution data.

1.1. Background

Mushroom identification is a common task in the machine learning community. There are numerous papers from 1995 to 2015 such as [15], [2], [12], [3], and [14] that introduce novel ideas using the UCI Mushroom Dataset [5] and other characteristic-to-species datasets.

Classifying *images* of mushrooms, however, is less common in the literature, primarily due to the lack of data. A 2018 paper uses a neural network on a mobile devices to classify 82 edible and 18 poisonous mushrooms from 1020 images [16] with reasonable accuracy. Another recent paper from 2019 demonstrates effective image classification of 35 edible mushroom species and 10 poisonous mushroom species using a dataset of approximately 6,000 and 2,556 images, respectively [11]. Their results demonstrate successful classification of the species from images, which made it clear that with enough data, our larger task of classifying 172 species could be possible.

A challenge facing Bolete classification in particular is that they generally have a similar form and structure, and share many characteristics like the presence of pores, a cap, and a stem [1] that are not necessarily shared by other types of mushrooms. This makes the Bolete classification task more difficult than more general mushroom classification that includes classes from other groups of mushrooms.

Figure 1: Example images of Boletes: From left to right, Aureoboletus-auriflammeus, Tylopilus-minor, Aureoboletus-russellii, and Austroboletus-subflavidus. [10]

There is a large amount of ambiguity surrounding Boletes, as evidenced by the fact that there are 5 different edibility classes, one of which is "iffy." Experts and enthusiasts alike often to fail to agree on identifications, thwarted by "crypto-species," "clades," and "groups" that have are often loosely defined.

There is much yet to be discovered, and new information from DNA sequencing is rapidly changing the classification of Boletes, often splitting, merging, and creating new genera and species [10].

2. Problem statement

2.1. Data

Thorough Bolete identification often requires more than just visual information. Factors such as smell, taste, location, and size can be critical for an accurate identification.

However, many of the key identifying features *are* visual, so image data should be satisfactory for this task.

2.1.1 Photos, characteristics, and IDs

The dataset is constructed from the Western Pennsylvania Mushroom Club's Bolete Filter [10] which is compiled from the field guide *Boletes of Eastern North America* [1] and other sources.

Each image in the dataset has a species label which maps to a list of characteristics and an edibility status. Supplementary photo-species pairs are gathered from a Bolete identification Facebook group identified by experts in the field, including the authors of [1].

2.1.2 Look-alikes

Some species of Boletes are relatively unique, while others fall into a group of "look-alikes," or sets of species that share a large number of characteristics and often are only distinguishable via microscopic examination [8] and/or DNA sequencing [4]. To account for this in evaluation, we use a quantitative metric δ that measures how many species are similar to the given species, with similarity $p \in [0, 1]$.

A set of look-alikes $\delta(u, p)$ is calculated for some characteristics vector u and proportion p as the number of species in the dataset that have at least p percent of the exhibited characteristics in common. An exhibited characteristic is one that has a value of 1. There is some set of matching species $\delta(u, p)$ such that:

$$\delta(u, p) = \{d(u, v) \geq p * d(u, u)\}, \quad \forall v \in D \tag{1}$$

where D is the dataset of species characteristics vectors and d is a similarity metric between two characteristics vectors, or the number of exhibited characteristics they have in common. More concretely:

$$d(u, v) = \sum_{i=1}^{C} u_i * v_i = u \cdot v \tag{2}$$

where $u_i, v_i \in \{0, 1\}$, and C is the total number of characteristics. Because the entries are always either 0 or 1, the calculation of $d(u, v)$ can be simplified to $u \cdot v$.

To illustrate the metric, consider the following examples for some species u:

1. $\delta(u, p = 1.0)$ is the set of species that share the exact same list of characteristics, or a true lookalike.
2. $\delta(u, p = 0.8)$ is the set of species that share at least 80% of the characteristics.
3. $\delta(u, p = 0.0)$ is the set of all species in the dataset.

2.2. Evaluation

2.2.1 Species Prediction

For species prediction, both the characteristic and the direct models produce a set of predicted species $\hat{\tau}$ using some similarity p. The set $\hat{\tau}$ is defined in Equations 6 and 8, for the characteristic and direct models, respectively. A model is accurate if the correct species t is in $\hat{\tau}$. A model with a smaller $\hat{\tau}$ is more precise than a model with a larger one.

The overall performance $\mathcal{P}$ of a model is measured as a combination of its accuracy and precision in the range [0,1) defined as:

$$\mathcal{P}(\hat{\tau}, t) = \frac{(M - |\hat{\tau}|)\, \mathbb{1}(t \in \hat{\tau})}{M} \tag{3}$$

where M is the number of species in the dataset and $\mathbb{1}$ is the indicator function.

We plot the performance scores of each model over the domain of p values to gain a better understanding of their relative performance.

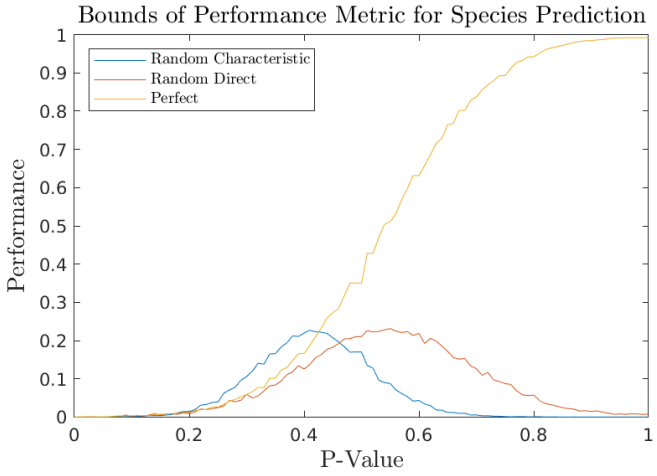

Figure 2: Performance function (P) boundaries from perfect and randomly sampled data. One way to compare the performance curves is by comparing the areas under the curve (AUC). For perfect prediction, the AUC is 0.4547. For random characteristic prediction, the performance AUC is 0.0535. Lastly, for random species prediction, the performance AUC is 0.0817.

2.2.2 Edibility Prediction

The performance of the edibility models is defined simply as the model's accuracy because the target is just a single class instead of a set.

The direct model outputs the edibility status directly, which is then compared with the ground-truth edibility status associated with the image.

For the characteristic model, an example is correct if the mode edibility in $\hat{\tau}$ for the characteristic prediction is equal to the ground-truth.

3. Technical Approach

Since our dataset is relatively small, we use pre-trained models in a transfer learning [6] task. Each of the models uses a GoogLeNet [13] architecture with pre-trained, frozen weights and learn the last fully connected layer, which outputs scores of size of the number of classes.

The models utilize GoogLeNet's multi-scale design for detecting coarse details like overall shape, as well as finer details like netting and dots. For the same reason, we train two versions of each model: one on low resolution images and one on high resolution images to compare trade-offs between image detail and computational complexity.

The data is shuffled and 3-fold cross validation is used with stratified splits over species on the training data. During cross validation we fine-tune hyper-parameters such as the learning rate, mini-batch size, the optimizer [7], and the number of epochs that maximize validation accuracy. The number of epochs is limited to implement early stopping techniques discussed in *Deep Learning* (p. 239). Using early stopping approximates an L2 regularization for the model as discussed on page 242 of [6].

To get the final results, we train on the whole training set used in cross-validation and evaluate performance on a held-out test set which is 30% of the total dataset.

The codebase for the project can be found at https://github.com/JackKenney/bolete-classifier.

3.1. Data Preprocessing

3.1.1 Images

After removing all duplicate images and species for which there are less than 6 images, there are 172 Bolete species in the dataset with 1,868 images in total. The images are padded and scaled to get uniform input image dimensions: $(512 \times 512 \times 3)$ for high-resolution and $(256 \times 256 \times 3)$ for low-resolution.

3.1.2 Characteristics

Characteristics are combined into one list of 38 items. For each species we construct a vector in $\{0, 1\}^{38}$ of characteristics that it does and does not exhibit. The characteristics are not mutually exclusive and can appear together, for example a "red cap" and "white netting" can coexist in the same species.

3.2. Characteristic Based Model

A multi-label logistic regression classifier is trained on the images with an output layer of size 38. After running an image through the model, we find all species that match the resulting list of exhibited characteristics (within some similarity p) and return the set as the result.

This model uses the multi-label soft margin loss:

$$\begin{cases} \log(\sigma(\hat{y}_c)) & \text{if } y_c = 1 \\ \log(\sigma(1 - \hat{y}_c)) & \text{if } y_c = 0 \end{cases}$$

Which can be summed into a cross-entropy loss over each characteristic:

$$L = -\sum_{c=0}^{C} (y_c \log(\sigma(\hat{y}_c)) + (1 - y_c) \log(1 - \sigma(\hat{y}_c))) \quad (4)$$

where C is the number of characteristics,
y_c is the true label value of the characteristic c,
$\hat{y}_c \in \mathbb{R}$ is the raw score of the characteristic c,
and σ is the sigmoid function which maps $\mathbb{R} \to (0, 1)$:

$$\sigma(\hat{y}_c) = \frac{1}{1 + e^{-\hat{y}_c}} \quad (5)$$

A characteristic vector $u \in \{0, 1\}^{38}$ is created from hard-classification. Using u and some similarity $p \in [0, 1]$, the set of predicted species is:

$$\hat{\tau} = \delta(u, p) \quad (6)$$

3.3. Direct Model

A multi-class convolutional network is trained with an output layer of size 172. The softmax loss of scores s for true label y is defined as:

$$L = -\log\left(\frac{e^{s_y}}{\sum_{j=1}^{M} e^{s_j}}\right) \quad (7)$$

Which is the negative log of the score of the correct species, y, exponentiated and divided by the sum of the exponentiated score for each species.

Instead of returning the single most-likely species, $\hat{y} = \arg\max(s)$, the model outputs a set of several species $\hat{\tau}$ so that this method can be comparable to the characteristic-based model. This set $\hat{\tau}$ is the top k most likely species, where k is the number of lookalikes for the most likely class $\hat{y}$, for some proportion p.

$$\hat{\tau} = \text{top\_k}(s, k), \quad k = |\delta(\hat{y}, p)| \quad (8)$$

3.4. Edibility prediction

Characteristic-Based Model: Using $\hat{\tau}$ from Equation 6, the mode edibility status in the set is compared with the correct species edibility status to calculate accuracy.

Direct Model: The direct model uses a similar model to the one described in 3.3, but with edibility status as class labels instead of species ids.

4. Evaluation and Results

4.1. Results

All of the models all make improvements over baseline accuracies, which are defined as always predicting the most common class(es). However, the models' validation accuracies tend to be much lower than their training accuracies, partly because the accuracy metric is too strict for the task, allowing only a single prediction without considering similarity.

In addition, because the characteristic and direct models have such different learning tasks, their accuracies are on drastically different scales, and are not directly comparable.

For these reasons we supplement the models' accuracies over epochs with the performance metric (3) across p-values to get a better sense of their relative success.

The high resolution and low resolution models produced very similar results, so in this section we only discuss the model accuracies and $\mathcal{P}$-performances from the high-resolution models. The low resolution model results are included in the appendix.

4.2. Species

4.2.1 Baseline

The most common species in the database makes up about 0.016 of the database. So, if a model were to learn to always greedily choose the most probable class, the expected accuracy would be about 1.6%.

Similarly, predicting the most common characteristic for each category produces a model accuracy of 70.6%.

4.2.2 Characteristic Model Performance

Both the training and the test accuracies are higher than the baseline expectation. From this, we conclude that some of the characteristics were able to be learned from the images. However, there is also a gap between the accuracies of the training set (0.74) and the held out test set (0.71), suggesting the model is not successfully generalizing. Importantly, the test accuracy hovered just above the baseline and did not rise and then drop back down after a certain number of epochs. This indicates that while the model appears to be over-fitting to the training data, it could be due to a lack of available data more-so than a limitation of the model itself.

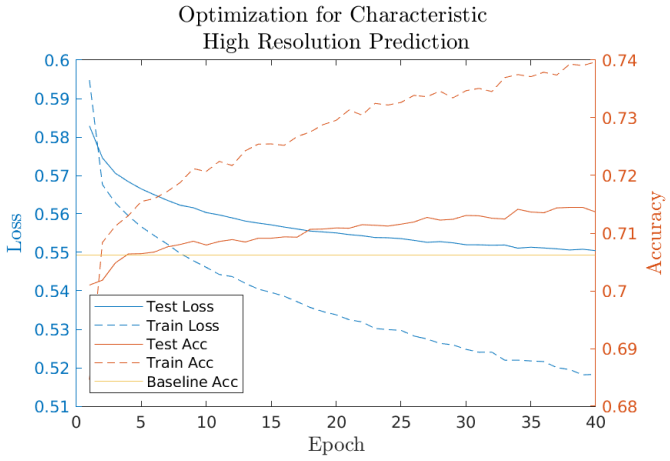

Figure 3: The descending curves represent the loss while the ascending curves correspond to accuracy. Training is denoted using dashed lines and testing with solid lines. Loss is plotted on the left y-axis and accuracy on the right y-axis. The baseline accuracy (0.706) for always predicting the most likely characteristics in the dataset is the solid horizontal line.

4.2.3 Direct Model Performance

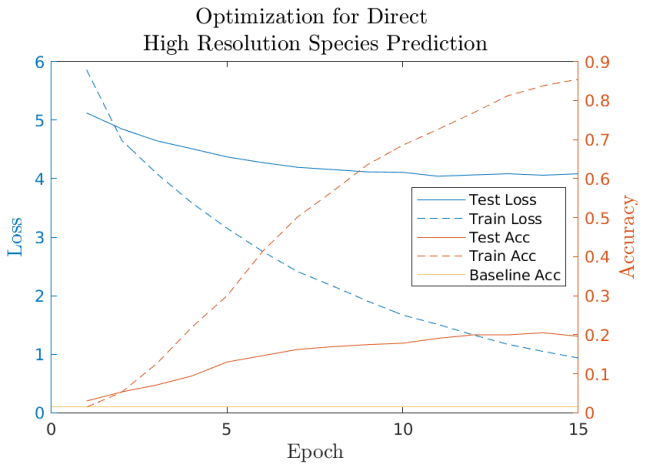

Figure 4: The legend and plotting for this figure is the same as in Figure 3. A baseline accuracy (0.016) for predicting the most likely class in the dataset is provided in solid yellow at the bottom of the figure.

Like the characteristic model, the direct model sees overall improvement over the baseline accuracy, with a large gap between training (0.85) and testing (0.20) accuracies.

Unlike the characteristic model, the baseline for the direct model accuracy is much lower at approximately 1.6%. While its improvement of almost 20% during testing might seem significant in comparison to the characteristic model, we can not make any claims about their relative performance from accuracies and losses alone, because the scales of the tasks are so different.

4.2.4 $\mathcal{P}$ – Performance

$\mathcal{P}$ charts are used to compare the performance of the characteristic and the direct models in the same space over p. Performance is calculated according to Equation 3 for $p \in [0, 1]$. Ideal performance curves have high values when p is large. Such a curve means the models predicts the correct species with the fewest number of look-alikes.

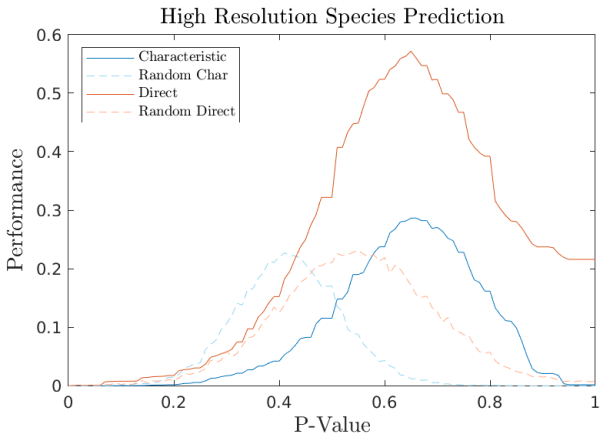

Figure 5: The direct model's performance is plotted with solid orange, the characteristic model's performance with solid blue, and the performance of random predictions with dashed blue and orange over all values of p.

Figure 5 shows that while both models perform better than random, the direct model strongly outperforms the characteristic model on species prediction for the entire domain that p spans. Additionally, the characteristic model performs better than its random counterpart, suggesting that it has learned some valuable information about the characteristics.

4.3. Edibility

4.3.1 Baseline

The distribution of edibility statuses in the dataset is:

Avoid	Bitter	Iffy	Good	Choice
0.047	0.063	0.267	0.544	0.079

This table shows that if a model were to learn to always greedily choose the most probable class the expected accuracy would be about 54%.

4.3.2 Model Accuracy

Similar to the model performance for species prediction, with edibility prediction there is only a slightly better test accuracy performance than the baseline that does not diverge with later epochs due to early stopping. Due to the gap in training (0.60) and testing (.56) accuracy and loss we conclude that there was insufficient data to learn this prediction.

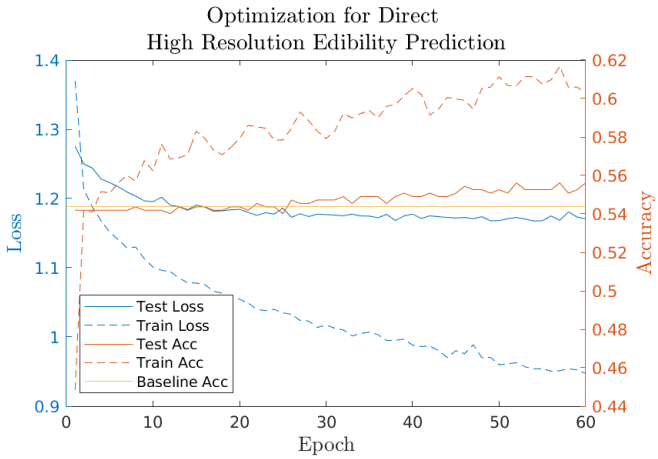

Figure 6: This plot is created via the same method as Figure 3. The baseline accuracy for mode edibility class prediction is plotted horizontally in yellow at 54.4%.

Interestingly, both the high resolution and the low resolution models achieved the same performance, despite making different mistakes on different images. Out of the 561 test images, the high and low resolution direct models predict 249 values differently.

The characteristic model's optimization in Figure 3 is the same for edibility as for species because it predicts characteristics in both cases.

4.3.3 Accuracy over p

Both models' edibility predictions are plotted in the space of accuracy over p-value in Figure 7. With edibility, the y-axis is accuracy, not performance, because the prediction is a single classification, not a set. The characteristic model's $\hat{\tau}$ is mapped to an edibility by taking the mode of the edibility status of $\hat{\tau}$.

Because the direct edibility prediction does not depend on p, it has a higher accuracy AUC (55.6%) than the characteristic model (48.9%) on the interval [0,1]. However, the 6.7% accuracy advantage of the direct model is misleading when considering that it is only 0.6% above the baseline.

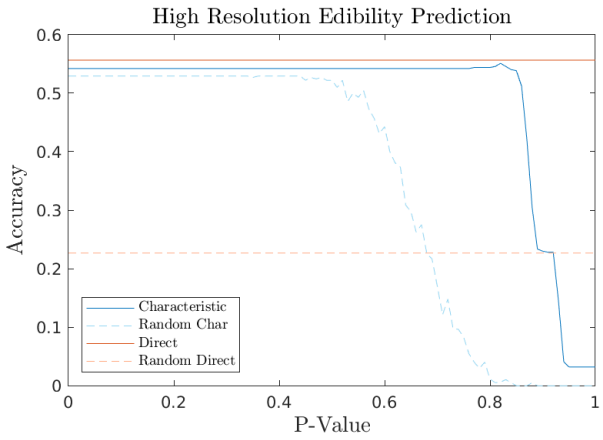

Figure 7: Both orange lines are horizontal because the direct model's edibility prediction does not depend on p. The blue lines plot the characteristic model's accuracy, which does depend on p to create the set $\hat{\tau}$ from which the predicted edibility is extracted.

4.4. Look-alikes

Surprisingly, the characteristic-based look-alikes are quite different from the direct model look-alikes, meaning that species that share most of their characteristics do not tend to actually look similar enough to confuse the direct model.

In Figure 8, the percent of similarity between the $\hat{\tau}$ from the direct and characteristic model outputs for the evaluation data is plotted. In this chart it becomes clear that as p increases, regardless of the image resolution, the $\hat{\tau}$ sets become increasingly disjoint, until they are entirely separate at $p = 1$.

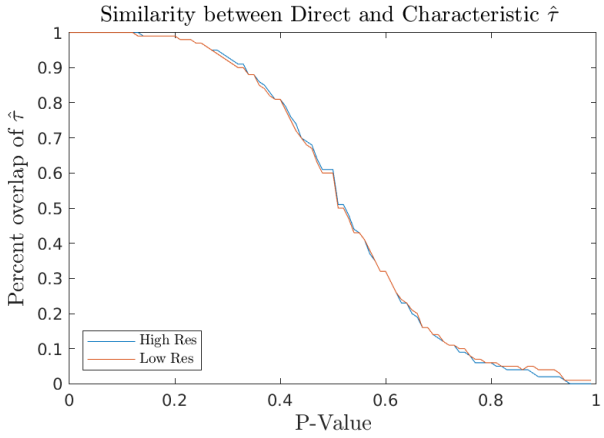

Figure 8: In both the high and low resolution models, the predicted set of species produced by the characteristics and direct models have a low percent similarity on average. At $p = 0.6$, they share only about 30% of $\hat{\tau}$ and at $p = 0.8$ only about 10%.

5. Conclusions

5.1. Data

It is clear from the differences between training and testing model accuracies that more data is needed for this high complexity task. Reducing input dimensionality by a factor of 4 does not affect the performance, suggesting the model is not able to take advantage of the high-resolution details in the large images. More data would also help disambiguate the large number of classes in both tasks.

5.2. Performance

Though none of the models' test accuracies are significantly above the baseline, the p-value performances shed a little more light on the relative success of each model.

5.2.1 Direct vs. Characteristic Species Classification

For species classification, the direct model out-performs the characteristic model by a large margin, as seen in Figure 5. At its peak, the direct model reaches 58% on high resolution and 60% on low resolution data at around $p = 0.65$. Notably this is a higher p than at the peak of both random models.

With a relatively high required similarity, and subsequently a smaller lookalike set, the direct species model performs better than all other models.

Combined with the fact that the the characteristic and direct models share less than 30% of their predicted $\hat{\tau}$, we conclude the direct model successfully learns important

species-specific features that are more informative than the characteristics-specific features of the other model.

5.2.2 Direct Species vs. Direct Edibility

The direct species model also out-performs the direct edibility model, which is particularly impressive considering the huge difference in the number of classes (172 and 5, respectively) and baseline performances (1.6% and 54%, respectively). From this we conclude that species-specific features are significantly easier to learn from images than edibility-specific ones.

6. Future Work

If more data became available, we expect the gap between training and testing accuracy would diminish and performance to increase.

Further investigation into the differences between $\hat{\tau}$ for the characteristic based models and the direct models is required. What does each consider "similar" species? With more data about which species are commonly considered lookalikes among human mycologists and foragers, comparisons can be made between the lookalikes produced by each model.

Often a single image is insufficient to capture all of the angles and information required for an identification, for example the stem ornaments from the outside and the cross section bluing. For this reason, it would be interesting to use a model that supports multiple input images via either stacked input or recurrent layers, to see whether it improves accuracy.

Whether transfer learning from a characteristic-based model to a direct model would be effective is still an open question. Answering it would entail training a characteristic model as described in this work, freezing the weights, and then adding a fully connected layer of size $[38, 5]$, linearly combining predicted characteristics to into edibility.

Since lookalikes can often be disambiguated using microscopic examination of specimens, a dataset of labeled microscopic images of different species of Boletes could be useful either in conjunction or maybe even instead of our macro-image dataset.

References

[1] A. E. Bessette, W. C. Roody, and A. R. Bessette. *Boletes of Eastern North America*. Syracuse University Press, 2017.

[2] X. Chai, L. Deng, Q. Yang, and C. X. Ling. Test-cost sensitive naive bayes classification. In *Fourth IEEE International Conference on Data Mining (ICDM'04)*, pages 51–58. IEEE, 2004.

[3] T. Chen, T. He, M. Benesty, V. Khotilovich, and Y. Tang. Xgboost: extreme gradient boosting. *R package version 0.4-2*, pages 1–4, 2015.

[4] B. T. M. Dentinger, J. F. Ammirati, E. E. Both, D. E. Desjardin, R. E. Halling, T. W. Henkel, P.-A. Moreau, E. Nagasawa, K. Soytong, A. F. Taylor, R. Watling, J.-M. Moncalvo, and D. J. McLaughlin. Molecular phylogenetics of porcini mushrooms (boletus section boletus). *Molecular phylogenetics and evolution*, 57(3):1276—1292, December 2010.

[5] D. Dua and C. Graff. UCI machine learning repository, 2017.

[6] Y. Goodfellow, Ian amd Bengio and A. Courville. *Deep learning*, volume 1. MIT press Cambridge, 2016.

[7] G. Hinton, N. Srivastava, and K. Swersky. *Lecture 6a: Overview of mini-batch gradient descent, slides 26-30*. Coursera, 2012.

[8] W. R. J. D. R. Largent, David L. *How to Identify Mushrooms to Genus III Microscopic Features: Microscopic Features*. Mad River Press, 1977.

[9] MATLAB. *9.9.0.1467703 (R2020b)*. The MathWorks Inc., Natick, Massachusetts, 2020.

[10] S. Pavelle. *The Bolete Filter*. Western Pennsylvania Mushroom Club, 2020. https://boletes.wpamushroomclub.org/.

[11] J. Preechasuk, O. Chaowalit, F. Pensiri, and P. Visutsak. Image analysis of mushroom types classification by convolution neural networks. In *Proceedings of the 2019 2nd Artificial Intelligence and Cloud Computing Conference*, AICCC 2019, page 82–88, New York, NY, USA, 2019. Association for Computing Machinery.

[12] R. Sikora et al. A modified stacking ensemble machine learning algorithm using genetic algorithms. In *Handbook of Research on Organizational Transformations through Big Data Analytics*, pages 43–53. IGi Global, 2015.

[13] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions, 2014.

[14] Y. Wang, J. Du, H. Zhang, and X. Yang. Mushroom toxicity recognition based on multigrained cascade forest. *Scientific Programming*, 2020, 2020.

[15] G. I. Webb. Opus: An efficient admissible algorithm for unordered search. *Journal of Artificial Intelligence Research*, 3:431–465, 1995.

[16] M. Wulandari, E. M. Kusumaningtyas, and A. R. B. Politeknik. Identification of poisonous fungi basidiomycota macro based on mobile device using neural network. In *2018 International Electronics Symposium on Knowledge Creation and Intelligent Computing (IES-KCIC)*, pages 146–151. IEEE, 2018.

7. Appendices

All plots created using MATLAB [9].

A. Edibility

Table 1: Edibility classifications from dataset.

ID	Category
1	Avoid
2	Bitter
3	Iffy
4	Good
5	Choice

B. Characteristics

Table 2: Species characteristics enumerated.

ID	Category	Feature
1	Cap Color	White, Buff, or Light Gray
2	Cap Color	Yellow to Orange
3	Cap Color	Red, Pink, Purple, or Orange
4	Cap Color	Some Shade of Brown
5	Cap Color	Black, Dark Brown, or Dark Gray
6	Cap Texture	Dry/Smooth (Normal)
7	Cap Texture	Viscid, Sticky, Slimy or Slick
8	Cap Texture	Wrinkled, Pitted or Corrugated
9	Cap Texture	Cracked Beyond Environmental Effects
10	Cap Texture	Spiky or Scaly
11	Pore Color	White, Buff, or Light Gray
12	Pore Color	Yellow to Gold
13	Pore Color	Red, Pink, Purple or Orange
14	Pore Color	Some Shade of Brown
15	Pore Color	Black, Dark Brown, or Dark Gray
16	Pore Staining	Pores Do Not Stain within 30 seconds
17	Pore Staining	Pores Stain Blue
18	Pore Staining	Pores Stain Other than Blue
19	Stem Color	White, Buff, or Light Gray
20	Stem Color	Yellow to Orange
21	Stem Color	Red, Pink, Purple, or Orange
22	Stem Color	Some Shade of Brown
23	Stem Color	Black, Dark Brown, or Dark Gray
24	Stem Decoration	Has No Significant Ornaments
25	Stem Decoration	Has No Significant Ornaments
26	Stem Decoration	Viscid, Sticky, Slimy or Slick
27	Stem Decoration	Has a Ring (Annulus)
28	Stem Decoration	Has Ridges
29	Stem Decoration	Netted (Reticulated) RED
30	Stem Decoration	Netted (Reticulated) WHITE
31	Stem Decoration	Netted (Reticulated)
32	Stem Decoration	Has Scabers/Scales
33	Stem Decoration	Notably Dotted and/or Spotted
34	Cap Flesh Color	White
35	Cap Flesh Color	Yellow
36	Cap Flesh Staining	Does Not Stain within 30 seconds
37	Cap Flesh Staining	Stains Blue
38	Cap Flesh Staining	Stains a Color Other than Blue

C. Additional Figures - Low Resolution Results