Explainable Edibles: An Exploration of XAI methods for Computer Vision

Christopher Rico[1] and Saul Alarcon[2]

School of Professional Studies, Northwestern University

[1] christoph.rico@gmail.com
github.com/christophrico

[2] alarcon.saul@gmail.com
github.com/SaulAlarcon

Abstract

With increasing numbers of critical decisions being made by increasingly complex AI models, many Explainable AI (XAI) methods now exist to elucidate the inner workings of AI-based decisions. This applies four XAI techniques to a custom, complex Computer Vision (CV) model and examines the extracted features that determine whether a mushroom is poisonous. By directly comparing several XAI methods with what are found as important features in predicting if a mushroom is poisonous in a separate dataset, we can validate the "explainability" of our CV models.[3]

Keywords: Classification, Explainable Artificial Intelligence, Computer Vision, Logistic Regression, Mushroom

Introduction & Problem Statement

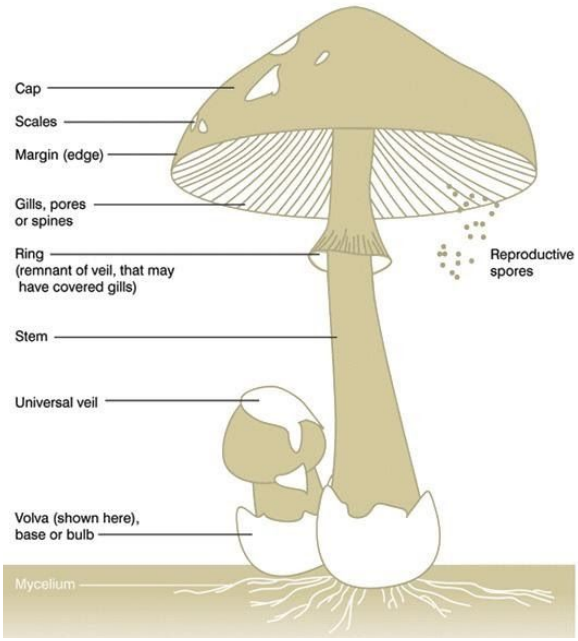
The importance of correctly interpreting a predictive model's output cannot be overstated. Lucid explanation engenders greater user trust, guides engineers into how a machine learning (ML) model can be improved, and supports a greater understanding of the modeled process (Arrieta et al., 2020). As modeled processes grow ever-more complex, the benefit of using complex models has also grown, bringing the tradeoff between accuracy and interpretability to the fore. With model-driven decisions further permeating everyday life, prediction explainability continues to be an important area of research (Russell and Norvig, 2021). Different approaches have been proposed to tackle this problem, but an understanding of their applicability and efficacy remains lacking.

This paper will investigate several XAI techniques applied to the toy problem of a CV classification system designed to identify poisonous or edible mushrooms. While many XAI techniques are available for CV classification systems, most of them center around highlighting the most explanatory features, i.e., which regions of an image contribute the most to an AI system assigning it a classification. By comparing the extracted explanatory features from each XAI technique to a "ground-truth" Kaggle

[3] The Github repo with our experiment code can be found at: github.com/Edible-ai/mushie-classifier

dataset containing textual data for classifying mushrooms as either edible or poisonous, we will attempt to examine both whether the trained CV system has learned tangible features of each class, as well as whether each XAI method identifies similar explanatory features. We hope that this examination will lead to a larger discussion about the techniques and tools available to assess the trustability of ML systems.

With thousands of mushroom species existing in our ecosystem, it is exhausting to categorize each kind's attributes. With the difference between some species being sometimes minor, it can be difficult for even experienced foragers and mycologists to easily identify mushrooms and discern whether it is poisonous. By using a variety of Machine Learning techniques to identify key attributes of what makes a mushroom poisonous, it is hoped we can collectively become better at identifying poisonous mushrooms, resulting in fewer accidental illnesses and/or deaths.



Additionally, because there are so many discernable and distinguishable mushroom features(i.e., gills, stalk, caps, color), it is an excellent candidate for XAI. We can more easily notice the types of features that our XAI models pick up on, as well as the ones that are being excluded.

Literature Review

The arena of deep neural networks (DNN) has experienced an explosion of variation in the past several years. Several advancements to the original multilayer perceptron model include convolutional neural networks, recurrent neural networks, and LSTM modules. Convolutional neural networks, originally proposed by (Krizhevsky et al., 2012), use convolutional layers to condense local features of an image so that a DNN can then perform classification on portions of it. The function of convolutional layers has been thoroughly decomposed and examined by (Zeiler et al., 2013) even before Tensorflow library Keras introducing a built-in suite of layer visualization tools.

Explainable AI is a subfield of artificial intelligence with two camps that must be differentiated between: interpretable modeling and post-hoc explainability techniques (Grégoire, Samek, and Müller, 2018). The former centers around building simple, interpretable ML models from square one using techniques like linear regression or decision trees. The latter centers around extracting explanations from so-called “black box” models, such as DNNs. Because we’ve selected a DNN modeling approach, we will focus only on post-hoc explainability methods in this paper.

Explainability methods for deep convolutional networks can be loosely classified into four camps: explanation by simplification, feature relevance explanation, visual explanation, and architecture modification (Arreita et al., 2020). We will perform both feature relevance and visual explanation techniques on the model. The most intuitive of these methods is simple visualization of a given input as it exits specific activation layers, exploring which feature maps are activated by a given input in the model. We have previously explored this approach on generative adversarial network deconvolution layers utilizing the Keract library.

The TF-Explain library implements several approaches, including occlusion sensitivity, which visualizes how parts of the image affect the neural network's confidence by occluding parts iteratively (Zeiler et al., 2013); Grad-CAM (short for Gradient-weighted Class Activation Mapping), which visualizes how parts of the image affect the CNN’s output by looking into the gradient backpropagated to the class activation maps (Selvaraju et al., 2017). These are all qualitative approaches that center around understanding the decision process by mapping back the output in the input space to see which parts of the input were most discriminative for the output (Arreita et al., 2020).

Using supervised learning to classify mushrooms as poisonous or non-poisonous has been researched with the UCI Machine Learning Repository Mushroom Dataset (Chelliah et al., 2018). Researchers in this study utilized a collection of supervised learning methods, including Support Vector Machines (SVM), Logistic Regression, and Decision Trees. They conclude that while all models resulted in highly accurate classification, Decision Trees were the most reliable.

Dataset

The image dataset consists of roughly 10,000 JPEG images of 81 different mushroom species. We have specifically chosen these species because information about their edibility is readily available. As this study is concerned primarily with using ML tools to identify and assess mushroom features that indicate edibility, our dataset must contain mushroom species whose edibility is known. Because no large pre-existing dataset of mushroom images is available, we chose to curate this dataset independently.

Initially, we used the BeautifulSoup HTML parsing library to scrape species names, edibility data, and images of 142 mushroom species from Mushroom.world, a mycology database. This yielded 659 images, which was far too few images to train a performant computer vision model. This proved to be one of the first and largest hurdles that surfaced early in our project. We knew that since our XAI models would need to identify particular mushroom features, a sub-par computer vision model would not suffice. We then built a more robust image scraper using the Selenium library, which scraped Google Images for (unlicensed) photos of each of the 142 species. This expanded the dataset roughly twenty times in size.

For each of these 142 species, Mushroom.world assigns one of seven edibility categories: inedible, edible, poisonous, edible and good, lethally poisonous, edible and excellent, or edible when boiled. Because this study is concerned primarily with the features that distinguish between edible and poisonous mushrooms, we initially collapsed edibility into a binary classification, with inedible species grouped in with poisonous species. However, after many model training attempts that yielded poor precision and recall within the poisonous class, we reconsidered the inclusion of the 61 inedible species, and removed them from the dataset altogether.

The other dataset that is being used in this project has been downloaded from Kaggle. This dataset is hosted on Kaggle but originally from the UCI Machine Learning Repository. Interestingly, this is the same dataset utilized in the supervised learning study mentioned earlier (Chelliah et al., 2018), allowing us to compare results easily. In this dataset, there are 23 features that describe various properties of 23 different mushroom species, which come from the Agaricus and Lepiota Family. There are 8123 observations of these species. Unlike the image database, however, the species for these observations are not labeled. Several variables are available in this dataset concerning the size, shape, color, and more of

the gills, cap, stem, etc. They also include visual (i.e., cap color, gill attachment) and non-visual (i.e., odor) attributes of mushrooms.

Experimental Outline and Methods

This experiment was executed using a Jupyter Lab kernel running inside a Python 3.8 virtual environment. Various industry-standard Python data science and ML libraries, including Pandas and PIL, are used to manipulate and explore both the Kaggle textual dataset and the mushroom image data. Keras and TensorFlow are used to build, train, and evaluate the CNN models. Exact package specifications can be found on the aforementioned Github project page.

To accelerate the development of a performant CNN model, we opted to use MobileNetV2 (Sandler et al., 2018) as a base model to perform transfer learning. MobileNetV2 consists of 16 convolution "blocks" which each contain several stacked convolution, batch normalization, and global average pooling layers. To aid in quick training and minimize overfitting, we trained only the top convolution block to 85% accuracy and 92% AUROC. After this, we fine-tuned the model by unfreezing every layer and training at a low learning rate until accuracy reached ~90% and AUROC reached 97%.

XAI visualizations were generated using three techniques from the TF-Explain library. We opted to instantiate and compare Grad-CAM, Integrated Gradients, and Occlusion Sensitivity. SmoothGrad was also implemented, but visualizations were practically unintelligible.

To analyze the Kaggle dataset, a Logistic Regression Model is implemented. Logistic Regression was chosen because it can output probabilities as opposed to only a Yes/No classification. Those with a lower risk tolerance for poisonous mushrooms will want to be almost certain that a mushroom is edible (99.99%+), while others may be okay with a slightly lower probability. All of the hyperparameters in the logistic regression are the ones provided by default from the Scikit-Learn package. This included using 'l2' as our penalty type and 'lbfgs' as our solver.

The Kaggle dataset consisted entirely of 23 categorical variables. To ensure that these variables could be used as predictor variables for our model, they are all transformed using one-hot encoding into

binary variables. This resulted in a total of 115 variables to use as predictors for our logistic regression. No other data cleansing/modifications were done as there were no missing values.

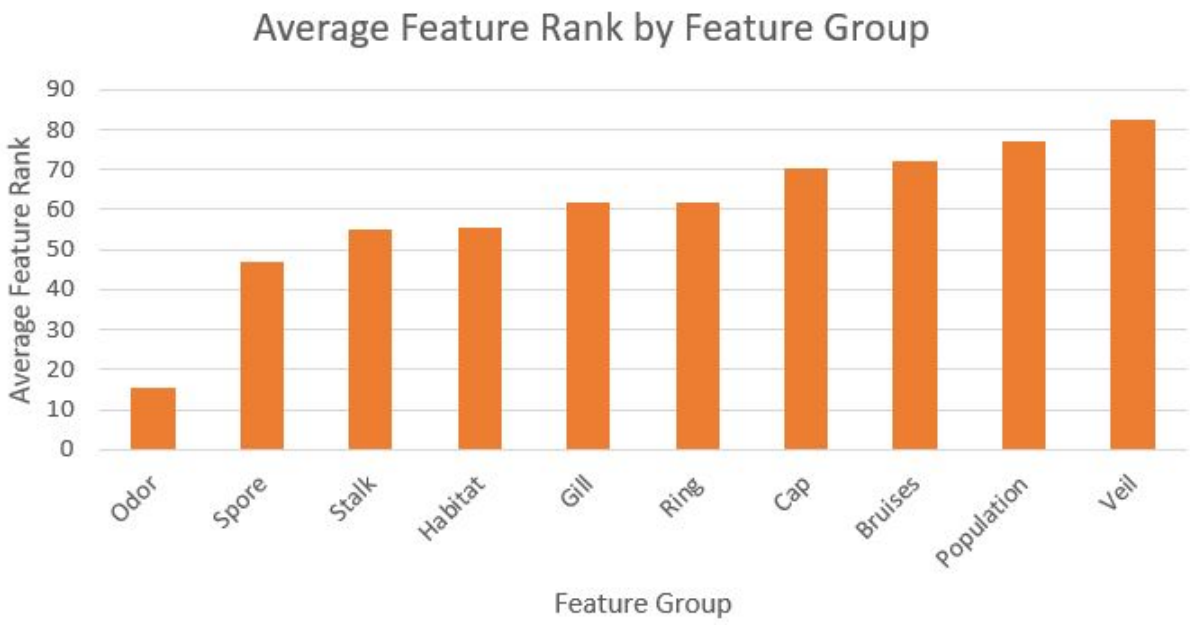
75% of the Kaggle dataset observations are used for training, while the remaining 25% are reserved for testing. After training, a collection of metrics are used to assess the model. These included precision, recall, accuracy, and the AUROC.

After the regression, recursive feature elimination (RFE) is used to identify the most important features in predicting whether a mushroom is poisonous. Those features are then compared with those found using the computer vision model.

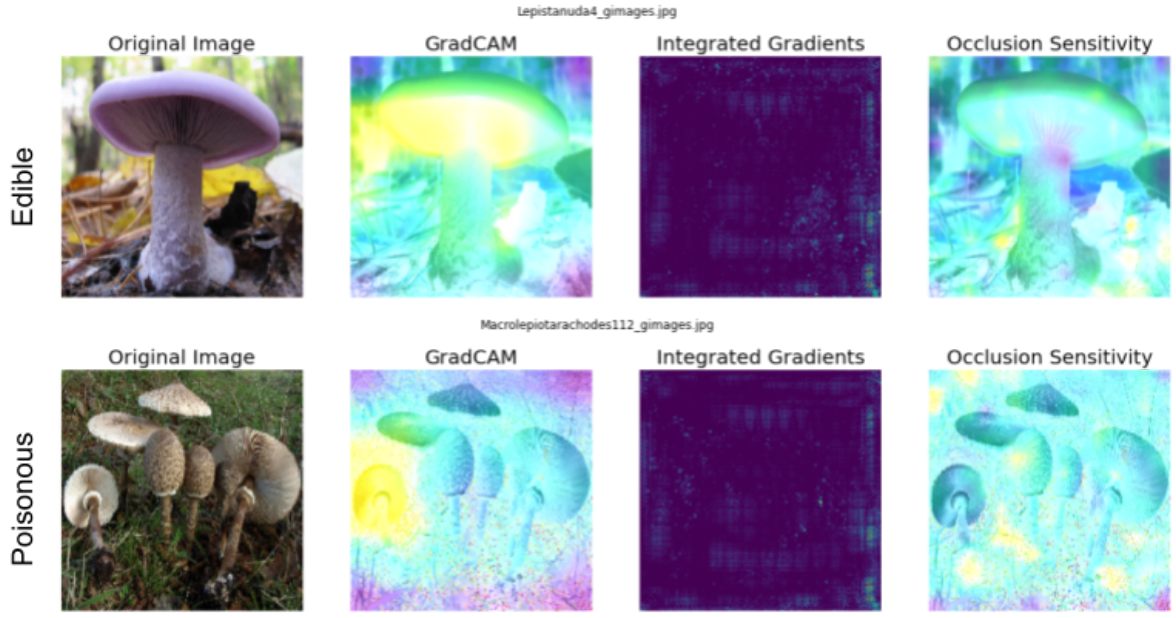
Results

After compiling the logistic regression algorithm as described in the methods section, our model is able to achieve 100% accuracy in the test dataset, suggesting that the features present in our dataset provide essential information in determining whether or not a mushroom is poisonous. Due to achieving perfect accuracy, no hyperparameter tuning is performed.

After training the logistic regression, we could move on to RFE to identify the most important attributes. By running the RFE we can rank the features from our dataset from most to least important. A complete list of these feature ranks are shown in the classification Jupyter notebook within the associated Github page. Because there were over 100 variables, it wasn't easy to internalize the most important features to look for from a holistic standpoint. To address this, all of the features were placed into different groups, and the feature ranks of the groups were averaged. Knowing that a lower feature rank indicates a more important variable, we can interpret the average feature ranks below.



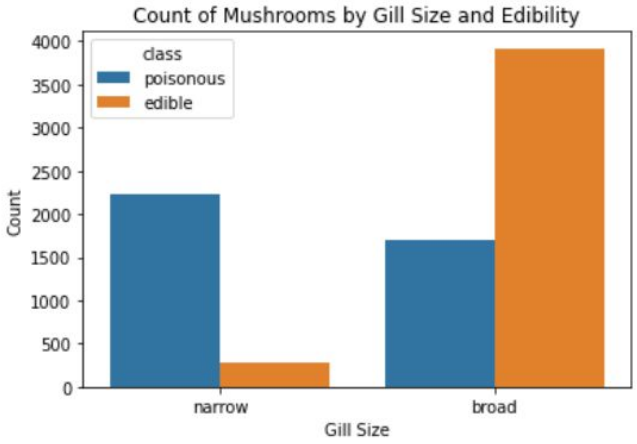
While this chart allows us to see the forest from the trees, it is important not to forget to see the trees from the forest. Many of the groups may have the potential of being influenced by one or two features within that group. For example, two of the most important features discovered were gill-size_narrow and gill-size_broad. That may not be obvious, seeing as the chart places the gill category as the 5th most important feature group.

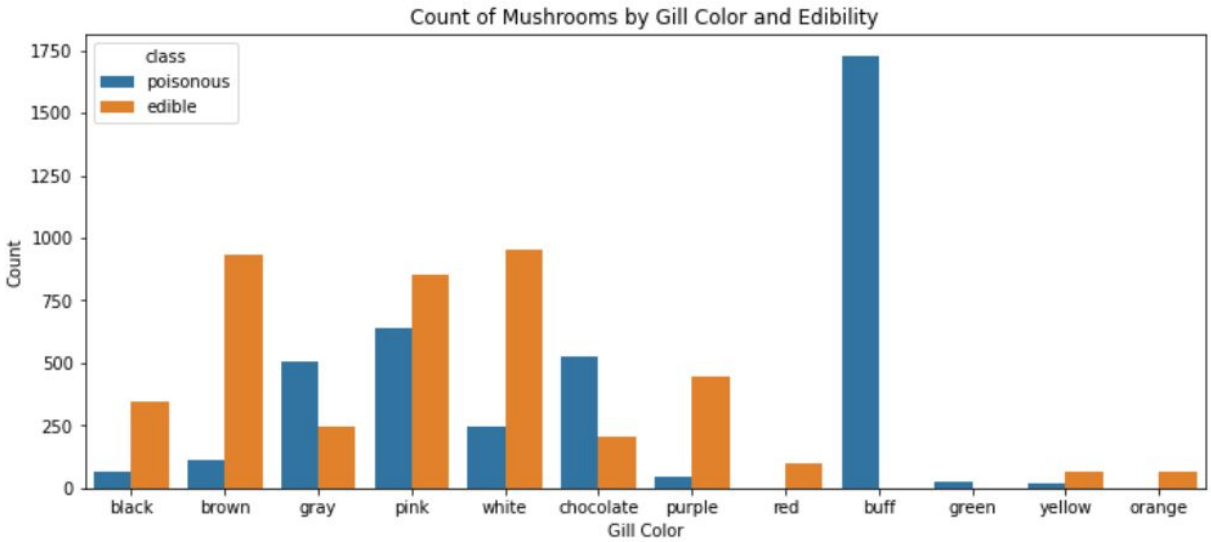


Grad-CAM

Of the three XAI techniques implemented, Grad-CAM yields the most intuitive explanations for the edible class. Grad-CAM explanations consistently highlight concise, continuous regions that correlate well with several of the most explanatory features from the Kaggle dataset: gill characteristics and habitat. For example, in the figure above, Grad-CAM highlights the portion of the mushrooms with gills exposed. While it is difficult to tell which gill feature is specifically explanatory (color, spacing, size, etc.), we hypothesize that it may have more to do with the gill spacing (close) and gill color (buff). These were both features that were identified as strong predictors in the Kaggle dataset. The following charts from the Kaggle dataset illustrate this.

It is apparent that Grad-CAM nails several of the features that would likely be used by a mycologist to identify mushrooms in the field. On the other hand, Grad-CAM seems to be less able to pick out stalk characteristics as explanatory features. With stalk color and size ranking high up on the list of explanatory features from the Kaggle dataset, it may be more of an indictment of the CNN model itself that none of our tested XAI libraries can identify stalk characteristics as explanatory features.

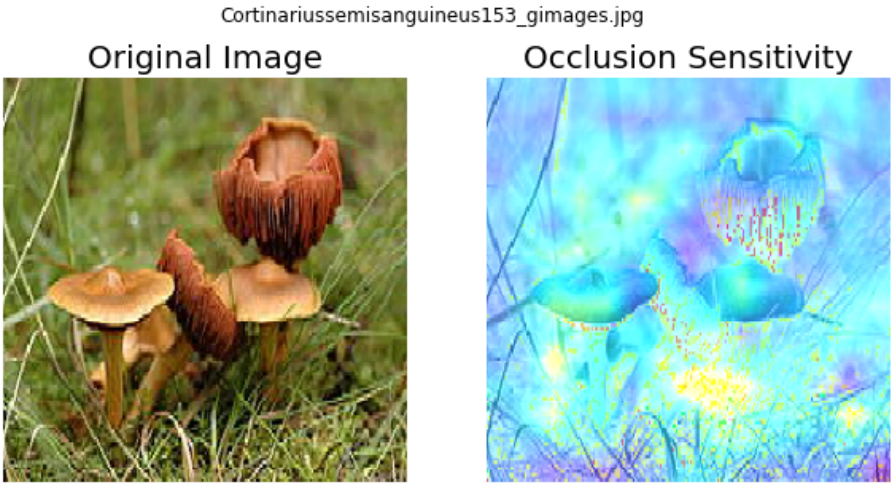




Integrated Gradients

Integrated gradients offer up the least lucid results of any method implemented. The great majority of its outputted visualizations suffer from some artifacts on the bottom right corner, and image regions are rarely highlighted in a way that offers an intuitive explanation. The best examples typically come from images wherein a cluster of pixels highlights an area without much continuity.

Occlusion Sensitivity



Occlusion sensitivity tends to pick out explanatory regions that are spottier than those highlighted by grad-CAM. Many of the explanatory regions it highlights correlate less with features of the mushroom itself but instead with the regions around the mushrooms, particularly within the poisonous class. This suggests that occlusion sensitivity can identify the growing habitat as an explanatory feature, nailing another one of the Kaggle dataset's explanatory features. However, it seems less able to highlight large, contiguous regions of the mushroom itself, which, unfortunately, does not engender confidence in its use as a standalone XAI method for this particular application.

Discussion

Of the three XAI methods we chose to examine, grad-CAM stands out as easily the best fit for our particular problem. Not only does it offer clear, concise highlighting of explanatory regions, but it's highlighted regions also appear to correlate well with the most explanatory features identified by the Kaggle dataset. Considering that grad-CAM works – as the name suggests – by taking an average of activation layer values weighted by gradient values, it makes sense that its localization maps would be the most contiguous and general of the three methods we chose to examine. It is not hard to imagine that grad-CAM might not be as effective for offering explanations regarding images whose explanatory elements are less monolithic than a mushroom, for example. Because grad-CAM takes an average of the activation layer values, its aptitude for generating explanations on images with many spatially disparate explanatory features may be less.

Occlusion Sensitivity, on the other hand, offers a more heuristic approach to determining explanatory features. By iteratively masking portions of the image, running inferences on the altered image, then comparing the inference confidences, this method can suss out explanatory regions of an image in a piecewise manner. The method's very nature causes it to be more likely to catch explanatory regions that are spatially disparate, which is exactly what we saw when we employed this method on our computer vision model. However, it was less able to pick out the higher-ranking explanatory features, instead seeing the trees for the forest, perhaps.

None of this is to say that one XAI method is overall *better* than any other method. After all, choosing the right tool for the job is an enormous determiner for success at any task. However, it does suggest that multiple XAI methods working in concert may yield more thorough results compared to a single method on its own. This idea is not new: ensemble ML models combine inferences from several smaller, biased models to yield startlingly accurate results. This approach's power lies in biases canceling each other out; each model filling in the others' weak spots. Perhaps, then, the era of ensemble XAI methods is nearing.

When comparing the features identified in the XAI models with those found in the Kaggle Dataset, one observation becomes strikingly clear. The most important feature in determining whether a mushroom is poisonous is the mushroom's odor – a feature that isn't discoverable through our XAI methodology! When it comes to exhibiting these odor features in a way that could be shared similar to the way we have done with Computer Vision, there is not much we can do, but there is some interesting research surrounding artificial scents (Parry, 2018).

As mentioned in the Dataset section, the set of species being observed in the Kaggle dataset differed from those in the image dataset. This led to initial concerns about whether or not the findings from the Kaggle dataset could be sufficiently generalized to those found in the image dataset. By comparing those features between both datasets, however, we are more confident that they can, as these features seem to be exhibited in every biological mushroom family.

Conclusion

This project shows us that, yes, identifying particular mushroom features is both a possible and plausible approach in discerning whether the mushroom is poisonous. Applying XAI methods to help illustrate how our computer vision model chooses sections/areas of mushrooms to make its classifications aided insight into the efficacy and applicability of several available methods.

While it may be difficult to match these attributes with those found in the Kaggle dataset with a measurable degree of confidence, it is difficult to argue that there is no link between the methodologies after observing the striking similarities. Among these important features are traits that a Mycologist or mushroom hobbyist can use to study the wide assortment of mushrooms more effectively. This addresses

the initial problem of helping the general population understand what traits will increase the odds that a mushroom is poisonous.

Directions for Future Work

As with any research project, the findings from current work inspire ideas for future work. Here are a few of the ideas that have been surfaced for future work:

1. Explore additional supervised learning models with the Kaggle dataset. Although we achieved perfect accuracy, different models may give different insights with the feature importance. Decision trees, in particular, may be good at visualizing some of the most important primary features in deciding whether a mushroom is poisonous.
2. Implement the XAI models onto an edge device (such as an iPhone) for easily accessible identification of whether a mushroom is poisonous while traveling.
3. Investigate different hyperparameter options for the XAI methods and implement a wider variety of XAI methods to compare.

References

Barredo Arrieta, Alejandro, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador Garcia, et al. “Explainable Artificial Intelligence (XAI): Concepts, Taxonomies, Opportunities and Challenges toward Responsible AI.” *Information Fusion* 58 (2020): 82–115. https://doi.org/10.1016/j.inffus.2019.12.012.

Chelliah, Balika J, S Kalaiarasi, Apoorva Anand, Janakiram G, Bhaghi Rathi, and Nakul K Warrier. “Classification of Mushrooms Using Supervised Learning Models.” *International Journal of Emerging Technologies in Engineering Research (IJETER)* 6, no. 4 (2018): 1–4.

Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." Advances in neural information processing systems 25 (2012): 1097-1105.

Montavon, Grégoire, Wojciech Samek, and Klaus-Robert Müller. “Methods for Interpreting and Understanding Deep Neural Networks.” *Digital Signal Processing* 73 (2018): 1–15. https://doi.org/10.1016/j.dsp.2017.10.011.

Parry, Wynne. *'Digital Smell' Technology Could Let Us Transmit Odors in Online Chats*. NBC News, November 27, 2018. https://www.nbcnews.com/mach/science/digital-smell-technology-could-let-us-transmit-odors-online-chats-ncna940121.

Russell, Stuart J., and Peter Norvig. *Artificial Intelligence: a Modern Approach*. Hoboken: Pearson, 2021.

Sandler, Mark, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. "MobileNetV2: Inverted Residuals and Linear Bottlenecks." 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018. https://doi.org/10.1109/cvpr.2018.00474.

Selvaraju, Ramprasaath R., Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. "Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization." *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017. https://doi.org/10.1109/iccv.2017.74.

Zeiler, Matthew D, and Rob Fergus. "Visualizing and Understanding Convolutional Networks." arXiv.org, November 28, 2013. https://arxiv.org/abs/1311.2901v3.