Improving of identifying whether mushrooms are poisonous or edible using EfficientNet B0

Anup Budhathoki, Viktoria Huszar, and Harith Elamin

Department of Computer Science, Oslo metropolitan university, Norway

May 2024

1 Abstract

This research aims to find out how could modern advanced machine learning architecture improve AlexNet model in classification of Poisonous and Edible Mushrooms. The background of this project is a paper titled as "A New Deep Learning Model for the Classification of Poisonous and Edible Mushrooms Based on Improved AlexNet Convolutional Neural Network"[2]. This paper presented a deep learning model for classifying poisonous and edible mushrooms. Thu auther precented an improved version of the AlexNet convolutional neural network architecture. However, despite the limitations of the data set, the author showed the experimental results that show that the proposed model can accurately classify edible and poisonous mushrooms and can shorten training and testing times. In this research, we have explained the reason for improving the AlexNet model to EfficientNetB0 model. We tuned the model parameters to significantly improve performance. We also used techniques such as random search, or default optimization in finding the optimal hyperparameters, and then have evaluated the model performance using cross-validation techniques to ensure the model's generalization ability and reliability. EfficientNet B0 model with 0.0001 learning rate, and with four dense layers and dropout in the test dataset with 90

Keywords: EfficientNetB0; convolutional neural network; deep learning; mushroom;

2 Introduction

The classification of mushrooms, a fundamental task spanning culinary arts, ecological studies, and public health initiatives, is pivotal for distinguishing between species that are either edible or poisonous [1]. Ensuring the accuracy of such classification is paramount for safeguarding food safety and preserving biodiversity [2]. Traditionally, this task has relied on manual inspection, a labor-intensive process prone to errors and requiring specialized expertise in mycology [3].

However, recent advancements in machine learning, particularly in the realm of computer vision, have transformed the landscape of mushroom classification. Deep learning techniques, notably convolutional neural networks (CNNs), have emerged as powerful tools for image classification tasks [4]. CNNs excel in learning hierarchical representations of image data, enabling them to capture intricate patterns and features crucial for discriminating between different mushroom species [5].

Moreover, the concept of transfer learning has revolutionized the development of classification systems by leveraging pre-trained models to expedite the training process and enhance classification accuracy [6]. By fine-tuning pre-trained models on specific tasks, transfer learning enables the transfer of knowledge from large-scale datasets to domain-specific applications, thereby mitigating the need for extensive labeled data [7].

Within the realm of CNN architectures, EfficientNet has garnered significant attention for its remarkable performance and computational efficiency [8]. EfficientNetB0, the smallest variant in the EfficientNet family, strikes an optimal balance between model size and performance, making it an ideal candidate for mushroom classification tasks [9]. By leveraging transfer learning with EfficientNetB0 as the backbone, we aim to harness the power of pre-trained models to develop a highly accurate and efficient mushroom classification system.

In this study, we embark on a comprehensive exploration of transfer learning and the EfficientNetB0 architecture for mushroom classification. Through meticulous experimentation and analysis, we seek to elucidate the intricate dynamics of transfer learning and the unique characteristics of the EfficientNetB0 architecture in the context of mushroom classification. By shedding light on the efficacy and potential limitations of these techniques, we endeavor to contribute to the advancement of machine learning methodologies in mycology and beyond.

Our research not only addresses the pressing need for automated mushroom classification systems but also lays the groundwork for future applications in food safety, environmental conservation, and public health initiatives.

3 Related Work

3.1. Convolutional Neural Network (CNN)

3.2. EfficientNet B0

xxxx[3] 3.3. Convolutional Neural Network (CNN)

4 Methods

Our journey to develop a robust mushroom classification system began with a comprehensive search for a dataset that would serve as the foundation of our machine learning project.

We conducted extensive research, leveraging academic repositories, online datasets, and domain-specific forums to identify a suitable corpus of mushroom images. This process involved a meticulous examination of each dataset's composition, quality, and relevance to our research objectives.

After thorough deliberation and consultation with domain experts, we settled on a dataset comprising approximately 100,000 images sourced from diverse sources such as online repositories, scientific publications, and citizen science initiatives.

On closer examination, however, we encountered several challenges inherent in the selected data set.

Among these challenges, was the presence of multiple classes, which include poisonous, edible, conditionally edible, and deadly mushrooms, which complicated the classification task.

Furthermore, the dataset suffered from significant class imbalance, with the 'conditionally edible' class disproportionately represented, posing a potential bias in model training.

To mitigate these issues and streamline our research focus, we made the strategic decision to narrow our scope to the binary classification of mushrooms as either 'poisonous' or 'edible'. This decision not only aligned with the primary objectives of our study but also facilitated a more balanced and focused analysis. With the dataset curated and class imbalance addressed through careful selection, our attention turned to preprocessing the images to uniformity and compatibility with our chosen machine learning model.

The initial inspection revealed inconsistencies in image sizes, necessitating resizing to a standardized format of 224x224 pixels. Additionally, we identified discrepancies in image formats, with some images in JPEG, JPG, and PNG formats. Of particular concern was the presence of PNG images, which introduced an additional alpha channel representing transparency, thereby deviating from the RGB color space used by the majority of the dataset. To rectify this discrepancy and ensure consistency in input data, we systematically converted all PNG images to the JPEG format, thus harmonizing the color channels across the entire dataset. Moreover, during the preprocessing stage, we encountered a subset of images that were corrupted or of insufficient quality for meaningful analysis. To maintain data integrity and prevent noise from influencing model performance, these corrupted images were systematically identified and removed from the dataset. This meticulous approach to data preprocessing was essential to ensure the reliability and robustness of our machine learning pipeline.

After carefully preparing the dataset, our next crucial step was model selection and architecture design. Recognizing the importance of utilizing state-of-the-art techniques for image classification, we conducted a comprehensive survey of modern models and architectures. While we first assessed the ResNet model, which has shown success in various image classification tasks, our preliminary experiments revealed suboptimal performance on our mushroom classification task. In response, we shifted our focus to EfficientNetB0, a recent and highly efficient convolutional neural network (CNN) architecture known for its superior performance on image classification.

With the EfficientNetB0 architecture selected as our backbone, we proceeded to fine-tune the model using transfer learning—a technique that leverages pre-trained models on large-scale datasets to accelerate learning on domain-specific tasks with limited labeled data. By transferring knowledge from the pre-trained EfficientNetB0 model, which was trained on ImageNet—a vast dataset comprising millions of labeled images spanning thousands of classes—we aimed to expedite the training process and enhance the model's ability to generalize across diverse mushroom images.

In summary, our methodology encompassed a comprehensive approach to dataset acquisition, preprocessing, and model selection, culminating in the adoption of the EfficientNetB0 architecture for transfer learning—a strategic choice informed by empirical evidence and domain expertise. This meticulous methodology laid the groundwork for subsequent experimentation and model refinement, positioning us to tackle the complex challenge of mushroom classification with rigor and precision.

5 Experiment

Our experimental journey encompassed a meticulous exploration of various configurations, hyperparameters, and optimization strategies to refine our mushroom classification model and maximize its performance. Central to our experimentation was the systematic tuning of hyperparameters, a process that involved iteratively adjusting key variables to identify the optimal settings for our model. One of the primary hyperparameters under scrutiny was the learning rate—a critical parameter governing the magnitude of updates to the model's weights during training. Recognizing the profound impact of learning rate on model convergence and performance, we conducted a series of experiments spanning a wide range of values, from 0.1 to 0.0001, to elucidate its effects on training dynamics and final classification accuracy. In parallel, we explored the effectiveness of different optimization algorithms in guiding the model's weight updates towards the optimal solution. Two prominent optimizers, RMSprop and Adam, were selected for evaluation based on their widespread adoption and proven effectiveness in training deep neural networks. Through rigorous experimentation, we sought to discern nuanced differences in optimization behavior and performance between these algorithms, thereby informing our choice of optimizer for subsequent training iterations. Beyond hyperparameter tuning, our experimentation encompassed the systematic exploration of model architectures and configuration settings to unlock the full potential of our chosen backbone, the EfficientNetB0 architecture. Recognizing that the depth and complexity of neural network architectures can significantly influence model performance, we embarked on a journey of model pruning, systematically removing layers from the pre-trained EfficientNetB0 model to tailor its architecture to our specific classification task. This process involved iteratively removing different numbers of layers, ranging from one to seven, and evaluating the resulting architects' performance on our validation dataset.

To further optimize the computational efficiency and generalization capacity of our model, we employed regularization techniques such as dropout—a popular method for mitigating overfitting by randomly deactivating neurons during training. By varying the dropout rate from 0.1 to 0.9, we explored its impact on model performance and stability, seeking to strike a balance between regularization strength and preservation of valuable information encoded in the network weights. Throughout the experimentation phase, model performance was meticulously evaluated using a comprehensive suite of evaluation metrics, including accuracy, precision, recall, and F1-score. These metrics provide valuable insights into the model's ability to correctly classify mushrooms as either 'poisonous' or 'edible', while also shedding light on potential areas for improvement and refinement. Additionally, to mitigate the risk of overfitting and ensure the robustness of our findings, we employed rigorous validation techniques such as k-fold cross-validation, partitioning the dataset into multiple subsets for training and validation, and averaging the results across folds to obtain a More reliable estimate of model performance. Ultimately, after conducting a myriad of experiments and meticulously analyzing the results, we identified the optimal model configuration—a finely-tuned ensemble of hyperparameters, optimizer settings, and model architecture modifications. This configuration, which involved leveraging the EfficientNetB0 architecture with five layers removed, an Adam optimizer with a learning rate of 0.0001, and a dropout rate of 0.5, emerged as the pinnacle of our experimentation efforts, delivering superior performance on our mushroom classification task. Through meticulous experimentation and empirical validation, we have laid a solid foundation for future research and advancements in mushroom classification and machine learning alike.

6 Discussion

7 Results

This section shows the results of the experiment explained. First, the results of training the source models will be shown, followed by the results from the transfer learning experiments. In fact, in the original model, before the improving, has accuracy of 98.50%, a precision of 99.39%, a recall of 98.79%, an F1 score of 99.09%, and a training time of 1 min 10 s, while the model after improving to efficient db gave the precision score 85 for class 0 (edible) , and 85% of actually edible, precision score was 88% for the class 1 (poisonous mushrooms) and 88% of actually poisonous. The recall score of 86% for class 0 (edible mushrooms), and the recall score of 87% for class 1 (poisonous mushrooms). It's cleasr that the model correctly identified 86% edible to edible. The model correctly identified 87% poisonous to poisonous.

The reason for this could be that the dataset were used in our experiment were more diverse. With hundreds of different-looking mushrooms, it may have been difficult for our model to remember those different patterns. Some types also have very few images, such as 12 to 15 images. With only 12 to 15 images to learn the pattern, it is difficult for the machine. in addition, adding more images in each type folder is a time consuming work and maybe in the next research this strategy can be implemented. But due to time constraints and also due to limitations in GPU power, we decided not to enlarge the images.

7.1 Classification

	precision	recall	f1-score	support
0	0.8535087719298246	0.8622064687638458	0.8578355741679524	2257.0
1	0.8800154320987654	0.8722753346080306	0.8761282888419435	2615.0
accuracy	0.8676108374384236	0.8676108374384236	0.8676108374384236	0.8676108374384236
macro avg	0.866762102014295	0.8672409016859381	0.866981931504948	4872.0
weighted avg	0.8677359715073657	0.8676108374384236	0.8676540160547519	4872.0

7.2 Evaluation

The best results from training the source models can be seen in Fig. xxx.

8 Future works

For future work on this project, we encourage someone to replac more convolutional Layers with more efficient building blocks such as depthwise separable convolutions in order to improve the the performance. use a differente hyperparameters to adjust the depthwise layer, and using a smaller number of weights to increasse the accurancey. However, using generative adversarial networks to generate more fungal image datasets will overcome the data limitations of this research and will contribute to improving the quality and diversity of the training data, ultimately enhancing performance.

9 Conclusion

Our experiment shows that the EfficientNet model was trained on Kegal dataset with 100 southand images, devied into tow classess. Replacing the Convolutional Layers with more efficient building blocks such as depthwise separable convolutions, help the model to work with a better performance.

10 References

[1] A. Krizhevsky, G. Hinton, et al., "Learning multiple layers of features from tiny images," 2009.

[2] Ketwongsa, W.; Boonlue, S.; Kokaew, U. A New Deep Learning Model for the Classification of Poisonous and Edible Mushrooms Based on Improved AlexNet Convolutional Neural Network. Appl. Sci. 2022, 12, 3409. https://doi.org/10.3390/app12073409.

[3] Ul Haq, Husni Fadhilah Dhiya et al. "EfficientNet Optimization on Heartbeats Sound Classification." 2021 5th International Conference on Informatics and Computational Sciences (ICICoS) (2021): 216-221.

[4] J. Gui, Z. Sun, Y. Wen, D. Tao and J. Ye, "A Review on Generative Adversarial Networks: Algorithms, Theory, and Applications," in IEEE Transactions on Knowledge and Data Engineering, vol. 35, no. 4, pp. 3313-3332, 1 April 2023, doi: 10.1109/TKDE.2021.3130191.