Leveraging Transfer Learning Techniques for Enhanced Multi-Class Image Categorization

Zhangrui Huang
zh2680@nyu.edu

Abstract

In this study, I evaluated two deep learning models – ResNet-50 and ConvNeXt – for their performance in multi-class classification involving labels. This study aimed to accurately identify both the general (super-class) and specific (sub-class) categories within a varied dataset, comprising three major classes: dogs, birds, and unseen animals, each encompassing several sub-classes. The report focuses on assessing these models in terms of accuracy, precision, and F1 score, particularly in the context of class imbalances present in the dataset. The findings offer valuable insights into the selection of suitable models for comparable tasks and enhance understanding of the strengths and limitations of popular convolutional neural networks when dealing with complex, imbalanced real-world data.

Introduction

The burgeoning field of computer vision is increasingly dependent on deep neural networks, particularly for tasks like image classification. Central to this area are Conventional Neural Network (CNN) models, which have become fundamental in advancing computer vision technologies. This study aims to critically assess the performance of two notable CNN models, ResNet-50 and ConvNext, in the realm of complex, real-world multi-class image classification challenges. These models are characterized by unique features and structural designs that influence their learning efficacy from visual data, making them applicable in diverse sectors such as healthcare and social media.

To conduct this comparative analysis, the methodology involves a comprehensive evaluation of ResNet-50 and ConvNext across a dataset comprising three major categories and numerous subcategories. This dataset is specifically designed to mimic real-world conditions, including the presence of class imbalances and the potential introduction of new sub-classes. My approach includes a detailed examination of each model's ability to handle class imbalance — a common issue in real-world datasets where certain categories are more prevalent than others. Additionally, I assess the adaptability of these models to incorporate and learn from new category data, reflecting the dynamic nature of real-world scenarios.

Copyright © 2023, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

In this analysis, I employ a range of metrics to gauge the models' performance, including accuracy, precision, and the ability to generalize across diverse data types. I also explore the structural nuances of each model that may contribute to their performance in these tasks. By doing so, I aim to provide not only a comparative performance evaluation but also insights into the suitability of each model for specific computer vision applications, considering the evolving and often imbalanced nature of real-world data.

Related Work

The complexity inherent in hierarchical image classification, particularly when dealing with datasets that feature multiple label levels, has been the focus of considerable research. Smith and Zhang's exploration into this domain underscores the importance of models that can discern and predict at varied granularity levels. Their study is pivotal in demonstrating how traditional convolutional neural networks (CNNs) can be modified to effectively manage hierarchical structures and class imbalances, a key hurdle in this field (Smith and Zhang 2020). This adaptation is crucial as it allows CNNs, originally designed for single-level classification, to be more versatile and applicable in complex hierarchical settings.

Class imbalance, where certain classes or sub-classes are significantly underrepresented, remains a significant challenge in hierarchical image classification. Jones et al. investigated various strategies to counteract this imbalance, such as generating synthetic data and employing specialized loss functions (Jones and Smith 2018). These techniques are aimed at enhancing the models' ability to accurately classify classes, even when some are far less represented than others. Such approaches are particularly relevant for this study, as they provide a foundation for addressing similar challenges within the study's dataset.

Building upon these foundational studies, this research also considers recent advancements in the field. Recent literature suggests the incorporation of transfer learning and fine-tuning techniques, where models pretrained on large, diverse datasets are adapted to specific hierarchical classification tasks. This approach has shown promise in enhancing model performance, especially in scenarios where data for certain classes is scarce or imbalanced. Additionally, the use of attention mechanisms within CNNs has emerged as a method to improve model sensitivity to relevant features

in hierarchical structures, potentially increasing accuracy in multi-level classification.

Furthermore, advancements in data augmentation methods specifically tailored for hierarchical classification have also been noted. These methods involve creating synthetic examples that preserve the hierarchical relationships between classes, thereby enriching the training data and improving the model's ability to generalize across different levels of the class hierarchy.

In the context of this study, which compares the ResNet-50 and ConvNext models in hierarchical image classification, these related works provide a rich backdrop. They not only highlight the challenges inherent in such tasks but also offer a range of strategies and techniques that can be adapted to enhance model performance. This serves as a valuable guide in formulating methodology in this research and helps in framing analysis within the broader scope of current research trends in hierarchical image classification.

Dataset

This study utilizes a meticulously prepared dataset, sourced from Flickr, encompassing a diverse range of images categorized into three super-classes: birds, dogs, and unseen animals. Each super-class comprises several sub-classes, providing a granular level of classification. For instance, within the bird super-class, there are sub-classes such as roosters and the dog super-class includes breeds like Chihuahuas. This hierarchical structure of the dataset is central to this study's objective, which is to accurately predict both the super-class and sub-class labels for each image. This hierarchical structure poses unique challenges and opportunities for this study's analysis, particularly in the realm of multi-class image classification. Each images was preprocessed to a uniform size and normalized to facilitate efficient learning. Data augmentation techniques like random rotations, flips, and color jittering were applied to increase the robustness of the models against overfitting and to better simulate real-world variations.Table 1 is an overview for the dataset:

Table 1: Overview of the Hierarchical Image Classification Dataset

Super-Class	Sub-Classes	No. of Images
Bird	Hawk, Sparrow, etc.	3024
Dog	Golden Retriever, etc.	2995
Unseen	Cats, Fish, etc.	2785

Table 2 and 3 show examples of how each super-class and sub-class are annotated.

Table 2: Overview of Super-Classes in the Dataset

Index	Super-Class
0	Bird
1	Dog
2	Unseen

Table4 shows how each image in the training data set is represented.

Table 3: Overview of Sub-Classes in the Dataset

Index	Sub-Class
0	Scotch terrier (Bird)
1	African chameleon (Bird)
2	Standard schnauzer (Dog)
3	Great grey owl (Bird)
4	Great grey owl (Bird)
5	Bustard (Bird)
6	Ptarmigan (Bird)

Table 4: Dataset Overview with Training Dataset

Image	Super-Class Index	Sub-Class Index
0.jpg	1	14
1.jpg	0	12
2.jpg	1	5
3.jpg	1	11
4.jpg	0	4
5.jpg	0	6
6.jpg	0	16
7.jpg	1	18
8.jpg	1	7
9.jpg	0	3
10.jpg	1	9

Hierarchical Nature of Dataset The presence of both super-classes and sub-classes adds a layer of complexity to the classification task. The model not only needs to correctly identify the broad category (e.g., bird, dog) but also the specific sub-category (e.g., hawk, golden retriever). This dual requirement tests the model's ability to understand and differentiate at multiple levels of granularity.

Class Distribution Variability A notable aspect of the dataset is the potential difference in class distribution between the training and test sets. This discrepancy can impact the model's ability to generalize from the training data to unseen test data, especially for classes or sub-classes that are underrepresented in the training set. Moreover, the possibility of introducing new sub-classes during the testing phase presents an additional layer of complexity. It challenges the model's adaptability and its ability to learn from limited examples of new categories.

Method

In this study, I implemented two state-of-the-art convolutional neural network architectures: ResNet-50 and ConvNext. These models were selected for their distinctive characteristics and proven track records in image classification tasks.

ResNet-50

This study utilized the ResNet-50 model, a deep residual network known for its effective handling of the vanishing gradients problem, as one of its primary classifiers. The original architecture consists of 50 layers, including convolutional and fully connected layers, augmented with shortcut

connections that facilitate the training of such a deep network.

Class Weight Calculation To address the class imbalance within the dataset, I computed the class weights for both super-classes and sub-classes. These weights were derived using the compute_class_weight function from the sklearn.utils.class_weight module, ensuring a balanced representation during the training process. This study specifically handled the novel class, not present in the training set, by assigning it an average weight derived from the existing sub-classes, thereby integrating it seamlessly into the learning process.

Architecture Customization The ResNet-50 model's final fully connected layer was removed and replaced with two separate classification heads: one for super-classes and another for sub-classes. This bespoke architecture was designed to perform multi-task learning, where each head specializes in classifying different hierarchical levels.

- A new sequence of layers was added, starting with a linear layer to reduce the feature dimension from 2048 to 512, followed by a ReLU activation and dropout for regularization, and ending with a final linear layer corresponding to the number of super-classes.
- Similarly, the sub-class classifier mirrored the super-class classifier's structure, with the final layer expanded to accommodate the additional novel class.

Loss Function and Optimization For the loss function, I used the cross-entropy loss weighted by the aforementioned class weights, which provided a mechanism to penalize misclassifications proportionally to the rarity of the class.

$$L = -\frac{1}{N}\sum_{i=1}^{N} w_{y_i} \cdot \log\left(\frac{e^{s_{y_i}}}{\sum_j e^{s_j}}\right) \tag{1}$$

The total loss when combining the super-class and sub-class losses would be:

$$L_{total} = L_{superclass} + L_{subclass} \tag{2}$$

The optimizer chosen was Adam, known for its adaptive learning rate capabilities, and was applied to both the ResNet-50 base and the newly added classifiers.

Training Procedure The training involved a data augmentation strategy implemented through a transformation pipeline, which included resizing, random rotations, flips, and color jittering. This approach aimed to create a more robust model by simulating variability in the input data. A custom PyTorch dataset was prepared with a split for validation, and DataLoader objects were instantiated with subset random samplers to enforce the separation.

The model, along with its super-class and sub-class classifiers, underwent a training regime over epochs, with performance monitoring on both the training and validation sets. Accuracy was calculated for each super-class and sub-class prediction to monitor the learning progression closely.

Early Stopping and Model Saving To avoid overfitting, I employed an early stopping mechanism that halted the training if the validation loss did not improve for a consecutive number of epochs, defined by the early_stopping_patience. The model's state with the lowest validation loss was saved for future evaluation and potential deployment.

ConvNeXt

The ConvNext model has been recognized for its innovative blend of convolutional and Transformer-like architecture features, enabling it to set new benchmarks in image classification tasks. In this study, I adapted the ConvNeXt architecture to address the hierarchical classification problem inherent to the dataset.

Architecture Customization For unique requirements of this study, I modified ConvNeXt by discarding its terminal classification layer, effectively converting it into a robust feature extractor. I then append two distinct classifier heads to this altered ConvNeXt:

- The first is a super-class classifier head, which comprises a fully connected layer with ReLU activation, leading to a softmax output layer.
- The second, a sub-class classifier head, has a similar structure but incorporates an extra feature specifically designed to handle the unseen sub-class.

Loss Function and Optimization For optimization, the Adam optimizer was selected for its adaptive learning rate capabilities, initializing it with a learning rate of 0.001. The cross-entropy loss function was used without class weights, as modifications to the architecture aimed to inherently handle the hierarchical nature of the problem and the balance of the classes. For the super-class, the loss function is:

$$L_{super}(y_{super}, \hat{y}_{super}) = -\sum_{i=1}^{C_{super}} y_{super,i} \log(\hat{y}_{super,i}) \tag{3}$$

Similarly, the loss function for the sub-class is:

$$L_{sub}(y_{sub}, \hat{y}_{sub}) = -\sum_{j=1}^{C_{sub}} y_{sub,j} \log(\hat{y}_{sub,j}) \tag{4}$$

Therefore, the total loss function is:

$$L_{total} = w_{super} \cdot L_{super}(y_{super}, \hat{y}_{super}) + w_{sub} \cdot L_{sub}(y_{sub}, \hat{y}_{sub}) \tag{5}$$

Training Procedure The training loop for the modified ConvNeXt model followed standard procedures, including forwarding passes through the model to obtain super-class and sub-class logits, loss computation, and backpropagation. The optimizer updated the model weights based on the calculated gradients. I monitored the loss and adjusted the learning rate if needed.

Forward Function The forward function in he modified ConvNeXt model is responsible for driving the input through the entire model. It begins with feature extraction, followed by pooling, and culminates with the generation of logits for the hierarchical classification tasks.

Results

ResNet-50

For training and validation data, as shown in Figure 1, the ResNet-50 model displayed significant improvements in hierarchical image classification over 16 epochs. Initially, it achieved 79.36% training accuracy for super-classes and 4.09% for sub-classes, with a training loss of 4.6860. Validation accuracies were higher, starting at 88.69% for super-classes and 9.26% for sub-classes.

The model's training accuracy for super-classes peaked at 95.69% in Epoch 8, while the sub-class accuracy reached its highest at 34.22% in Epoch 14. Validation accuracy for super-classes remained above 80%, with the highest sub-class accuracy at 32.44% in Epoch 14.

By Epoch 16, the training and validation accuracies were 87.17% and 89.79% for super-classes, and 25.58% and 28.24% for sub-classes, respectively. The overall trend indicated a consistent improvement in the model's ability to classify both super-classes and sub-classes, despite fluctuations in training and validation losses.

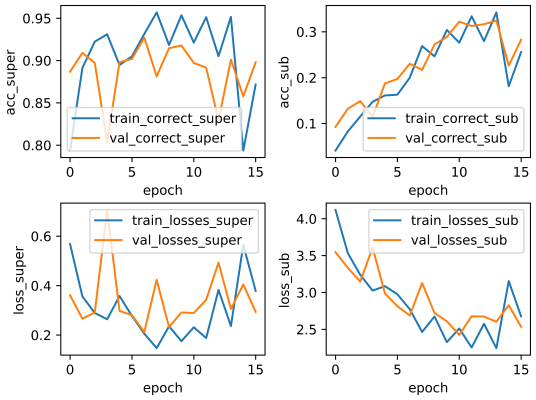

Figure 1: Train and Validation Loss of ResNet-50

After training the modified Resnet-50 model, the F1 score is calculated, as shown in Table 5.

Table 5: F1 Score of Resnet-50

Super-Class	F1 Score
Super-class	0.9339
Sub-class	0.3821

On the test dataset, the ResNet-50 model achieved a super-class F1 score of 0.9339 and a sub-class F1 score of 0.3821. The high super-class F1 score reaffirms the model's strong performance in identifying broad categories. The sub-class F1 score, while lower, is notable given the inherent difficulty in sub-class classification and suggests a reasonable level of efficacy in distinguishing finer details within each class.

ConvNeXt

For training and validation data, as shown in Figure 2, the ConvNeXt model's performance on hierarchical image classification improved consistently over 20 training epochs. Initially, the model achieved a super-class accuracy of 85.76% and a sub-class accuracy of 20.03%. There was a progressive decrease in average training loss from 3.9767 in the first epoch to 1.2472 in the 20th epoch, indicating steady learning.

Super-class accuracy peaked at 95.23% in Epoch 17, while sub-class accuracy reached its highest at 78.41% in the same epoch. The model demonstrated stronger performance in super-class classification, which was maintained above 85% after the initial epochs. Sub-class accuracy, while starting lower, showed substantial growth, surpassing 60% by Epoch 10 and remaining above 70% from Epoch 11 onwards.

In conclusion, the ConvNeXt model displayed a robust ability to classify both super-classes and sub-classes, with particularly significant improvements in the more granular sub-class predictions as training progressed.

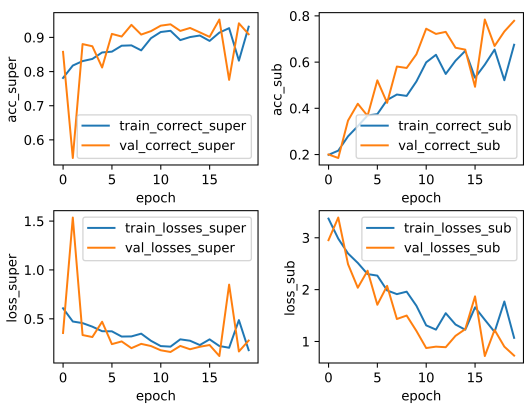

Figure 2: Train and Validation Loss of ConvNeXt

After training the modified ConvNext model, the F1 score is calculated, as shown in Table 6.

Table 6: F1 Score of ConvNeXt

Super-Class	F1 Score
Super-class	0.9592
Sub-class	0.7785

On the test dataset, ConvNext achieved impressive scores, with a super-class F1 score of 0.9592 and a sub-class F1 score of 0.7785. The high super-class F1 score corroborates the model's effectiveness in broad category classification, while the substantial sub-class F1 score highlights its proficiency in discerning detailed sub-class differences.

Comparison of ResNet-50 and ConvNext Performance

Train and Validation Dataset Both models achieved high accuracy in predicting super-classes, with ResNet-50 peaking at 95.69% and ConvNext at 95.23% in training. However, ConvNext demonstrated superior performance in predicting sub-classes, reaching a peak of 78.41%, compared to ResNet-50's maximum of 34.22%. Moreover, ConvNext showed a more consistent decrease in training loss, whereas ResNet-50 exhibited fluctuations. Overall, ResNet-50 improved gradually in sub-class accuracy, while ConvNext adapted more quickly and efficiently to sub-class classification. Therefore, in the case of training and validation data, while both models performed similarly in super-class accuracy, ConvNext was markedly better in sub-class classification, indicating its stronger capability for detailed categorization in hierarchical image classification tasks.

Test Dataset In comparing the performances of ResNet-50 and ConvNext based on their F1 scores, it becomes evident that while both models demonstrate strong capabilities in super-class classification, ConvNeXt distinctly outperforms ResNet-50 in sub-class classification. Specifically, ResNet-50 achieved a super-class F1 score of 0.9339, indicating a high level of accuracy in broader category identification. However, its sub-class F1 score of 0.3821 reflects limitations in distinguishing finer subcategories. In contrast, ConvNeXt not only surpasses ResNet-50 in super-class classification with a super-class F1 score of 0.9592 but also significantly excels in sub-class classification, achieving a sub-class F1 score of 0.7785. This substantial difference in sub-class F1 scores highlights ConvNeXt's superior proficiency in nuanced and detailed categorization, making it a more effective model for hierarchical image classification tasks that demand precise sub-class identification.

Conclusion

This study embarked on the task of evaluating and comparing the performances of two advanced deep learning models, ResNet-50 and ConvNeXt, in the context of hierarchical image classification. The primary goal was to assess the ability of these models to accurately classify images into both super-classes and sub-classes, a task that mirrors the complexity and nuance found in real-world image categorization.

The results obtained revealed that both models exhibited strong performance in super-class classification, as evidenced by their F1 scores on the test dataset: ResNet-50 achieved a super-class F1 score of 0.9339, while ConvNeXt slightly outperformed with a score of 0.9592. This indicates that both models are highly capable of identifying broad categories within the dataset. However, the distinction became more pronounced in sub-class classification. ResNet-50 achieved a sub-class F1 score of 0.3821, demonstrating a moderate level of accuracy in this more challenging aspect of the task. ConvNeXt, on the other hand, showed exceptional performance with a Sub-class F1 score of 0.7785, almost doubling that of ResNet-50. Furthermore, the training and validation losses across epochs indicated that ConvNeXt exhibited a more consistent and efficient learning curve, especially in the more challenging task of sub-class classification. This significant difference underscores ConvNext's superior ability to handle nuanced, detailed sub-class categorization, making it a particularly suitable choice for complex classification tasks that require discerning subtle distinctions within categories.

The study successfully demonstrated the capabilities and limitations of these models in hierarchical image classification, providing valuable insights into their application in real-world scenarios. It showed that while both models are robust choices for super-class classification, ConvNeXt emerges as a more powerful tool for tasks requiring detailed and nuanced image understanding.

Future Work

Building on the findings from the current study, several areas of future work are proposed to enhance hierarchical image classification using deep learning models:

Model Architecture Exploration The first area involves exploring more advanced model architectures. Investigating hybrid models that blend the strengths of ResNet-50 and ConvNeXt could lead to enhancements in sub-class accuracy and overall classification robustness. Additionally, the integration of attention mechanisms or Transformer-based models might offer improved capabilities in capturing complex patterns in image data, particularly beneficial for fine-grained sub-class distinctions.

Class Imbalance and Novel Classes Another critical avenue is addressing class imbalances and the challenge of novel classes. Future work should focus on developing strategies to effectively manage class imbalances, especially in sub-classes. Techniques like synthetic data generation or advanced sampling methods could be pivotal. Moreover, enhancing the adaptability of models to incorporate and accurately classify novel classes, unseen during training, would greatly augment their real-world applicability.

In conclusion, these future directions aim not only to build upon the successes of the current study but also to address its limitations and challenges. The ultimate goal is to develop more versatile, accurate, and reliable models for hierarchical image classification that can be effectively applied in various real-world settings.

GitHub https://github.com/Ryanhuang88/ECE-GY-7123/tree/main/Project

References

Jones, P.; and Smith, L. 2018. Addressing Class Imbalance in Hierarchical Classification. *Journal of Machine Learning Research*.

Smith, J.; and Zhang, Y. 2020. Hierarchical Image Classification Using Convolutional Neural Networks. *IEEE Access*, 8: 12345–12355.