Bag of Tricks and a Strong Baseline for FGVC

Jun Yu[1], Hao Chang[1], Keda Lu[1,2], Guochen Xie[1], Liwen Zhang[1], Zhongpeng Cai[1], Shenshen Du[1], Zhihong Wei[1,2], Zepeng Liu[1,2], Fang Gao[3] and Feng Shuang[3]

[1]University of Science and Technology of China), Hefei, Anhui, China
[2]Ping An Technology Co., Ltd, Shenzhen, Guangdong, China
[3]Guangxi University, Nanning, Guangxi, China

Abstract

Fine-grained visual classification (FGVC), as a subclass classification task under the superclass, brings more challenges. However, in addition to fine-grained features, the FungiCLEF 2022 dataset is also characterized by imbalance and rich meta-information. This motivates us to explore the impact of different methods and components in fine-grained classification on FungiCLEF 2022. We explore the impact of different data augmentations, backbones, loss functions, and attention mechanisms on classification performance. Additionally, we explore different metadata usage scenarios. In the end, we win second place in the CVPR2022 FGVC Workshop FungiCLEF 2022 challenge. Our code is available at https://github.com/wujiekd/Bag-of-Tricks-and-a-Strong-Baseline-for-Fungi-Fine-Grained-Classification.

Keywords

fine-grained, class-imbalanced, meta information

1. Introduction

In contrast to general image classification, the goal of fine-grained image classification is to correctly classify subclasses that belong to the same superclass (birds, cars, etc.). FGVC has long been considered a challenging task due to small differences between classes and large differences within classes. Publicly available datasets and benchmarks accelerate machine learning research and allow quantitative comparisons of new approaches. In the fields of deep learning and computer vision, rapid progress over the past decade has been largely driven by the publication of large-scale image datasets. The same is true for the problem of FGVC, where datasets, such as iNaturalist [1], have helped develop and evaluate new methods for fine-grained domain adaptation. But there has been a lack of research on the automatic classification of fungi.

Generally, the main methods of FGVC mainly focus on how to make the network focus on the most discriminative regions, such as part-based models and attention-based models. Inspired by human observational behavior, these methods introduce localization-induced biases

CLEF 2022: Conference and Labs of the Evaluation Forum, September 5–8, 2022, Bologna, Italy
changhaoustc@mail.ustc.edu.cn (H. Chang); wujiekd666@gmail.com (K. Lu)
0000-0002-3197-8103 (J. Yu); 0000-0001-8123-683X (H. Chang); 0000-0002-6328-2653 (K. Lu); 0000-0001-6494-6362 (G. Xie); 0000-0002-4757-9473 (L. Zhang); 0000-0002-2540-3215 (Z. Cai); 0000-0002-2683-3313 (S. Du); 0000-0002-4100-5700 (Z. Wei); 0000-0003-3052-1328 (Z. Liu); 0000-0003-1816-5420 (F. Gao); 0000-0002-4733-4732 (F. Shuang)

to neural networks with complex structures. In addition, some data augmentation methods and loss functions can also make the model pay more attention to fine-grained feature regions. When some species are visually indistinguishable, some extra-visual information can assist fine-grained classification, such as spatiotemporal priors and textual descriptions. But none of these methods have been explored on fungi-related datasets. This motivates us to explore the combined effect of various modules of deep neural network(DNN) for fungi's fine-grained classification method.

To address the fungi FGVC problem, we experiment a bag of tricks for image classification. First, we do an exploratory data analysis on the data set of FungiCLEF 2022, and deal with the imbalance and fine-grained characteristics of the data set accordingly. We apply the current state-of-the-art(SOTA) data augmentation methods to expand the richness of the dataset, and use large-parameter Convolutional Neural Network(CNN) models and transformer models to extract image features. Furthermore, we explore various loss functions, attention mechanisms, and two-stage training approach to deal with indistinguishable features and data imbalance, respectively. Since the raw data contains rich metadata, we apply various methods to improve the accuracy of the model after processing the metadata of different modalities.

In terms of experiments, we first use SwinTransformer [2] as the baseline model, F_1 score of the Public Leaderboard is 75.01%, and then introduce various SOTA data augmentation and tricks for evaluation to improve the single mode by nearly 5%. After determining the optimal solutions for various settings, we use multiple CNN or Transformer-based models to extract features, and finally model fusion is used to obtain the final result. The F_1 score of the Public Leaderboard is 83.12%, while the final F_1 score of the Private Leaderboard is 79.6% in the fungi competition [3] of LifeCLEF 2022 Workshop [4, 5].

The contribution of this paper are summarized as follows:

- We explore the impact of different combinations of components in image classification on the FungiCLEF 2022 dataset. Appropriate processing methods are applied separately for imbalanced and fine-grained features to alleviate model bias.
- We explore different ways of using metadata and achieve effective improvements on some backbones.
- We achieve an F_1^m score of 83.12% and 79.06% on the public and private test sets of the FungiCLEF 2022 dataset, respectively, winning second place in the fungi competition.

2. Related work

2.1. Fine-grained classification

Existing fine-grained classification methods can be divided into visual-only classification methods and multimodal classification methods. The former relies entirely on visual information to solve the problem of fine-grained classification, while the latter tries to use multimodal data to build a joint representation that merges multimodal information to facilitate fine-grained classification. Fine-grained classification methods that rely solely on vision can be broadly classified into two categories: localization methods [6] and feature encoding methods [7].

Early work [8] used partial annotations as supervision to make the network notice subtle differences between certain species and suffer from their expensive annotations. RA-CNN [9] was proposed to amplify subtle regions to recursively learn to distinguish region attention and region-based feature representations at multiple scales in a mutually reinforcing manner. NTSNet [10] proposed a self-supervised mechanism to efficiently localize information regions.

Feature encoding methods are dedicated to enriching feature representation capabilities to improve the performance of fine-grained classification. CAP [11] designed context-aware attention pools to capture subtle changes in images. TransFG [12] proposed a part selection module that applies a visual transformer to select discriminative image patches. Compared with localization methods, feature encoding methods are difficult to clearly distinguish the distinguishing regions between different species.

To distinguish these challenging visual categories, additional information, i.e., geographic location, attributes, and textual descriptions, can be helpfully utilized. Geo-Aware [13] introduced geographic information prior to fine-grained classification and systematically examined various previous approaches using geographic information, including post-processing, whitelisting, and feature modulation. Presence-only [14] also introduced a spatio-temporal prior to the network, which was shown to be effective in improving the final classification performance. CVL [15] proposed a two-branch network in which one branch learned visual features and the other branch learned textual features, and finally combined these two parts to obtain the final latent semantic representation. All the above methods were designed for specific prior information and cannot be flexibly adapted to different auxiliary information.

2.2. Fungi image classification

Image analysis tool[16] was published for mushroom species identification in 1992, analyzing morphological characters such as length, width and other shape descriptors. Computer vision can also be used to classify microscopic images of fungal spores. Microscopic image datasets of fungal infections and classification methods[17,18] were proposed to speed up medical diagnosis, thus avoiding additional expensive biochemical tests. A visual identification system based on deep CNNs was presented for 1,394 species of fungi and it is used in citizen science projects. The system allowed users to automatically identify observed specimens while providing valuable data to biologists and computer vision researchers.

3. Methodology

As shown in Figure 1, we first carry out exploratory data analysis on the data, and then introduce PIM [16], two-stage training [17], Meta Information and other information as auxiliary information. On the basis of the basic image classification model, we use the abundant information to improve the accuracy of model classification.

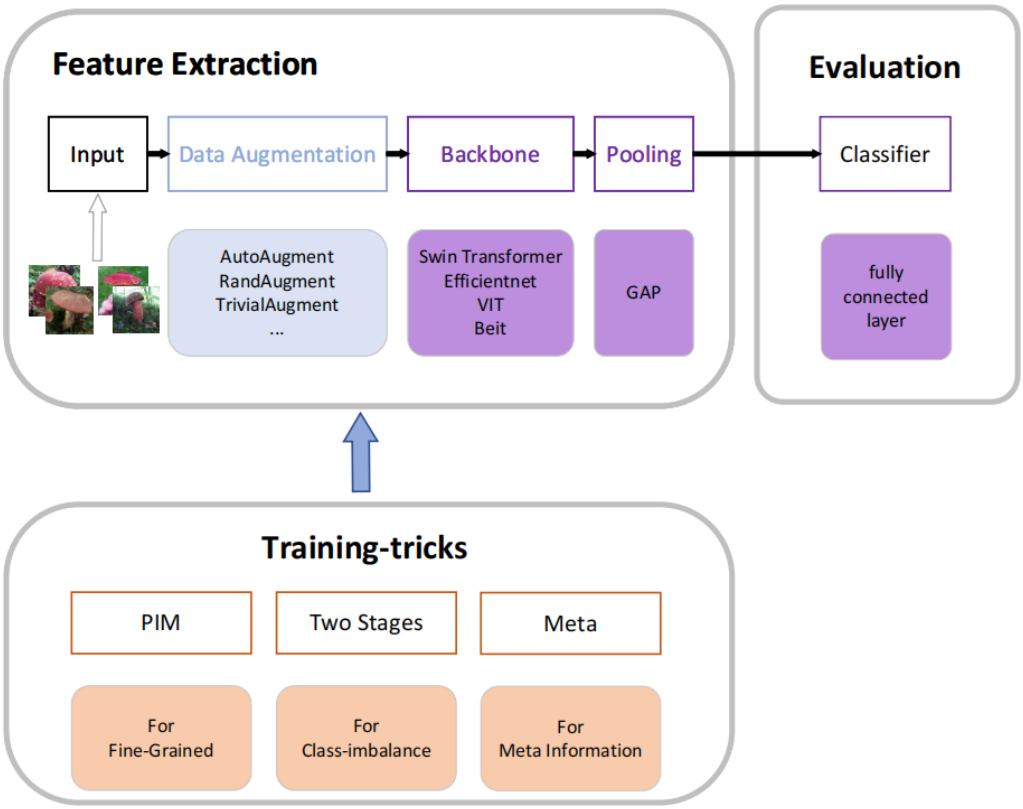

Figure 1: The framework of the method we used. We adopt models of CNN and Transformer with different architectures to extract image features, and introduce a variety of Tricks to improve the accuracy of classification.

3.1. Fungi challenge Dataset

3.1.1. Exploratory data analysis

The Fungi dataset contains 295,938 training images including 1,604 species. Dataset Features: Accurate class labels, highly imbalanced long-tailed class distribution. The test set for this competition includes 59,420 objects, 118,676 images, and 3,134 species, covering the full year of 2021, including observations collected across all substrate and habitat types. (Note: For out-of-range classes, use -1 as the predicted label). As shown in Figure 2, some images of fungi data set [18] are shown.

In addition to the image data information, the challenge also provides metadata, including abundant observational information such as habitat, time, and location. We are conducting a preliminary analysis of this data. There are 33 attributes in the training set, but only 10 attributes in the test set. By comparing one by one, the common features are month, day, countryCode, level0Name, level1Name, level2Name, substrate, habitat. The details are as shown in Table 1.

Figure 2: Examples of intra- and inter-class similarities and differences for selected species of three taxonomically distinct fungi families [18]. The similarity holds on the species and the family level. Left: Russulaceae, center: Boletaceae, right: Amanitaceae.

Table 1
Description of the provided metadata (observation attributes).

Attribute	Description
Month and day	Time information, 12 months, 31 days;
CountryCode	Country information, covering 30 countries;
Locality	More precise location information;
Level0Name, Level1Name and Level2Name	The position information of the observer. The three levels correspond to values of 30, 115, and 317, which present position information in refined granularity;
Substrate	Natural substance on which to live, such as bark, soil, etc. There are 32 values in total;
Habitat	Observations of the environment. A total of 32 values including mixed woodland and so on.

Generally speaking, there are few attributes that can be used. There are mainly time information, more detailed locality information, substrate and habitat information strongly related to the living environment of various fungi.

3.1.2. Data Augment

We do not experiment with subtle data augment combinations on the FungiCLEF 2022 dataset, but use NAS-based or improved data augment methods, namely AutoAugment, RandAugment, and TrivialAugment.

AutoAugment. AutoAugment [19] is a simple search-based data augmentation method. Its main idea is to create a search space of data augment strategies and evaluate the quality of a particular strategy directly on some datasets. In the experiments of AutoAugment, the authors

design a search space where each strategy consists of many substrategies, and in each batch a substrategy is chosen randomly for each image. The substrategies contain two operations, each of which is an image processing method, such as translation, rotation or clipping, and for each operation there is a set of probabilities and magnitudes to characterize the nature of the use of this operation. Using a search algorithm, search for the best policy that enables the neural network to achieve the highest validation set accuracy on the target dataset. AutoAugment achieves the same performance as the semi-supervised approach without using any unlabeled data. Finally, the strategies learned from one dataset can be directly transferred to other similar datasets and perform equally well. For example, the strategy learned in ImageNet can achieve state-of-the-art accuracy on the FGVC dataset Stanford Cars without training the pre-trained model with new data.

RandAugment. RandAugment [20] investigates a data augment strategy based on NAS method search, which provides a significant improvement in search cost compared to earlier NAS search data augment strategies. NAS-method-based data augment strategies (e.g., AA and other methods) suffer from two major drawbacks. First, they use separate search phases, thus increasing the complexity of training and greatly increasing the consumption of computational resources. In addition, since the search phases are separate, these methods cannot adjust the regularization strength according to the model or dataset size. That is, we usually train small models by training them on small datasets and then apply them to train large models. The proposal of RandAugment (later referred to as RA) solves these two problems by significantly reducing the search space and allowing training directly on the target task without a proxy task, and by tailoring the regularization strength of data augment to different datasets and models.

TrivialAugment. The TrivialAugment [21] data augment strategy originates from the NAS approach and outperforms NAS, implementing SOTA's data augment strategy in a simpler way. Although the approach of data augment using NAS method for automatic search is effective, the limitation lies in the need to trade-off the search efficiency and the performance of data augmentation. To solve this problem, TrivialAugment data augment strategy (later referred to as TA) is proposed. Compared with previous data augment strategies, TA is parameter-free and uses only one data augmentation method per image, so its search cost is almost free compared to AA and even RA, and it achieves SOTA results. TA uses the same data augment method as RandAugment. Specifically, data augment is defined as consisting of a data augment function a and the corresponding intensity value m (some data augment functions do not use intensity values). For each image, TA samples a data augment function and an intensity value uniformly, and then returns the enhanced image. In addition, while previous methods tend to overlay multiple data augment methods, TA only uses a single data augment method for each image. Using such an approach, the TA-enhanced dataset can be viewed as a single image enhanced separately using all data augment methods, and then uniformly sampled from it.

3.1.3. Meta Information

Appearance information alone is often insufficient to accurately distinguish some fine-grained species. When an image of a species is given, human experts also take advantage of the additional information to help make the final decision. In the related literature, some works [18] have applied meta data to the post-processing of probability predicted by the image classification

model, and some work have integrated meta data and image data as the input of multimodal learning to train a large model which has achieved more excellent results.

Intuitively, species distribution shows a trend of clustering geographically. Different species have different living habits, and spatiotemporal information can assist the fine-grained task of species classification. When considering latitude and longitude, we firstly need geographic coordinates to circle the earth. To do this, we convert the geographic coordinate system to a Cartesian coordinate system, i.e. $[lat, lon] \rightarrow [x, y, z]$. Likewise, December to January is closer than October. Therefore, we perform the mapping of $[month] \rightarrow [sin(\frac{2\pi month}{12}), cos(\frac{2\pi month}{12})]$.

When using attributes as meta information, we initialize the attribute list as a vector. For example, for nominal properties such as substrate and habitat, there are 32 different values, so a 32-dimensional feature vector can be generated.

3.2. Image feature extraction backbones

Backbone is crucial for feature extraction of FungiCLEF 2022 dataset. Different bcakbones have different learning ability for features and different focus on the dataset, and the choice of different models can bring us richer data features. In this challenge, we tried the following models.

3.2.1. Vision Transformer

Vision Transformer(viT) [22] is a model proposed by Google in 2020 to directly apply transformer to image classification. The input of the transformer is a sequence of token embeddings, so the patch embeddings of the image are fed into the transformer and then the features are extracted and classified. Research shows that when enough data is available for pre-training, ViT outperforms CNN and breaks the limitation of the transformer's lack of inductive bias to obtain better migration results in downstream tasks. We loaded ViT pre-trained models on ImageNet dataset for training and achieved good results on our dataset.

3.2.2. SwinTransformer

There are two main challenges in the application of Transformer to the image field. Visual entities are highly variable, and the visual Transformer performance may not be very good in different scenes Image resolution is high. With many pixel points, Transformer's global self-attention-based computation leads to a large computational effort. To address these two problems, SwinTransformer [2] proposes a Transformer with a hierarchical design that includes a sliding window operation, which consists of a non-overlapping local window and an overlapping cross-window. Restricting the attention computation to a single window can introduce the localization of CNNs convolution operations on the one hand and save computation on the other. The overall architecture of Swin Transformer is hierarchical, with four stages, each of which reduces the resolution of the input feature map and expands the perceptual field layer by layer like CNNs. There are several places where it is handled differently than ViT. ViT will position-encode the embedding on the input, while Swin-Transformer is here as an option (self.ape). Swin-Transformer does a relative position encoding when calculating Attention. ViT will add a learnable parameter separately as a token for classification, while Swin-Transformer directly

averages and outputs classification, which is somewhat similar to the final global average of CNN. pooling layer. On the ImageNet22K dataset, the accuracy rate of Swin Transformer can reach an astonishing 86.4%, which is one of the current SOTA models.

3.2.3. Efficientnet

The traditional practice of model scaling is to arbitrarily increase the depth or width of the CNNs, or to use a larger input image resolution for training and evaluation. While these approaches do improve accuracy, they typically require long periods of manual tuning and still often yield sub-optimal performance. A new approach to model scaling is to use a simple and efficient composite coefficient to scale CNNs in a more structured way. Unlike traditional methods that arbitrarily scale network dimensions such as width, depth, and resolution, EfficientNets [23] uniformly scales network dimensions with a fixed set of scale scaling factors. By using this novel scaling method and AutoML techniques, the authors call this model EfficientNets, which is up to 10 times more efficient (smaller and faster). To understand the effect of network scaling, the authors systematically studied the effect of scaling different dimensions on the model. While scaling individual dimensions can improve model performance, the authors observe that balancing all dimensions of the network based on available resources can maximize overall performance. In addition, the effectiveness of model scaling relies heavily on the baseline network. To further improve performance, the authors also develope a new baseline network that optimizes accuracy and efficiency by performing neural structure search using the AutoML MNAS framework. The final architecture uses moving inverse bottleneck convolution (MBConv).

3.2.4. Bidirectional Encoder representation from Image Transformer

Following the development of BERT in the field of natural language processing, Bao et al. [24] proposed a masked image modeling task to pretrain visual Transformers(BEiT). Specifically, in image classification field, each image has two views, namely image patches (e.g. 16×16 pixels) and visual tokens (i.e. discrete tokens). We first "tokenize" the original image into visual tokens. Then randomly mask some image patches and feed them into the backbone Transformer. The goal of pre-training is to recover original visual tokens from corrupted image patches. After pretraining BEiT, we directly fine-tune model parameters on downstream tasks by appending task layers on the pretrained encoder. Experimental results on image classification and semantic segmentation show that the model achieves better results than previous pre-training methods. For example, base-size BEiT achieves 83.2% top-1 accuracy on ImageNet-1K, significantly outperforming DeiT (81.8%) trained from scratch under the same settings. Furthermore, large-size BEiT achieves 86.3% accuracy using only ImageNet-1K, even better than ViT-L (85.2%) with supervised pretraining on ImageNet-22K.

3.3. Loss function

For the task of fungal classification, the loss function we use is a cross-entropy loss function for training. In addition, we use Focal Loss [25] and Seesaw Loss [26] to mitigate the problem of long-tailed distribution of the dataset. Where Focal Loss uses a modulating factor to the cross-entropy loss to reduce the loss contribution from easy examples and elevate the importance of

hard examples, SeesawLoss achieves a relative balance of positive and negative sample gradients by dynamically reducing the weight of the excessive negative sample gradients imposed by the head category on the tail category.

3.4. Fine-tuning to improve F_1^m score

3.4.1. Attentional mechanism

We used the fine-grained classification method of PIM [16] to help us with the task of fungal classification.PIM [16] is an excellent method for fine-grained classification that automatically finds the most discriminative regions and uses local features to provide features that are more helpful for classification. PIM [16] is a plug-in module, so it can be integrated into very many common CNN or Transformer-based network backbone, such as swin-transformer, effintnet, etc. The plugin module can output pixel-level feature maps and fuse filtered features to enhance FGVC. It can be briefly explained by selecting appropriate output feature maps from the backbone's blocks to input to a weakly supervised selector to filter out regions with strong discriminative power or regions with little relevance to classification, and finally fusing the features from the selector's output with a combiner to obtain prediction results.

3.4.2. Two-stage training

Two-stage training consists of unbalanced training and balanced fine-tuning. In this section, we will focus on different approaches to balance fine-tuning. CNNs trained on imbalanced datasets without any reweighting or resampling methods can learn good feature representations but have poor classification accuracy on underrepresented tail categories. Fine-tuning these networks on balanced subsets makes that features learned from unbalanced datasets are transferred and rebalanced across all classes[31]. These fine-tuning methods [27] can be divided into two parts: deferred rebalancing via resampling (DRS) and via reweighting (DRW).

- DRS first uses a vanilla training schedule and then applies resampling method in the fine-tuning stage. To obtain a balanced subset for fine-tuning, the resampling method described in the related work [28] on "Resampling Methods";
- DRW first uses a vanilla training schedule and then applies reweighting method in the fine-tuning stage. The fine-tuning stage will employ the reweighting method described in the related work [28] on "Reweighting Methods".

3.5. Metadata Use

3.5.1. Conditional probability post-processing

We refer to someone else's method [18], a simple way to use metadata to improve classification performance – similar to the spatio-temporal priors used. For metadata (D) and image (I) of a given type, the corresponding conditional probability is calculated, and then the post-processing is carried out for softmax output of the last layer of the model.

3.5.2. Metaformer

FGVC is the task that requires the classification of objects belonging to multiple subordinate classes of a superclass. Recent state-of-the-art approaches usually design complex learning pipelines to solve this task. However, visual information alone is usually insufficient to accurately distinguish FGVC. Nowadays, meta-information (e.g., spatio-temporal priors, attributes, and textual descriptions) usually appears together with images. Is it possible to use a uniform and simple framework to leverage various meta-information to aid fine-grained classification? To address this question, Metaformer explores a unified and powerful meta-framework for FGVC. In practice, MetaFormer provides a simple and effective way to address the joint learning of visual and various meta-information. In addition, MetaFormer provides a powerful baseline for FGVC without the bells and whistles. Numerous experiments have shown that MetaFormer can effectively exploit various meta-information to improve the performance of fine-grained classification. In a fair comparison, MetaFormer can outperform current SOTA methods with only visual information on the iNaturalist2017 and iNaturalist2018 datasets. After adding meta-information, MetaFormer can outperform the current SOTA method by 5.9% and 5.3%, respectively.

3.5.3. The application of MLP

Due to the large difference between the meta data and the image data, directly using the multi-modal method to learn them at the same time will cause the model to be difficult to converge or to converge too slowly. Therefore, the features and meta information extracted by the image model can be used through MLP. The extracted features are further interacted, which can speed up the convergence speed, and can not waste the information of the meta data.

4. Experiments

Firstly, we train various well-known architectures such as EfficientNets, SwinTransformer, Vision Transformer, and BEiT on the CNNss and Transformers families respectively. Furthermore, two-stage fine-tuning is introduced and evaluated. Finally, the models of various different architectures are integrated. Additionally, the effects of different metadata and their combinations on the final prediction performance of CNNs and ViT are evaluated.

4.1. Setup

In this section, we describe the complete training and evaluation procedure, including the training strategy, image augment, and testing procedures. All architectures are initialized with publicly available pre-trained checkpoints and the same policies are trained using the PyTorch framework in a Tesla A100 graphics card. All neural networks are optimized using stochastic gradient descent with momentum set to 0.9. The starting learning rate (LR) is set to 0.01 and is further reduced by Reduce LR On Plateau strategy (the dataset is split 9:1 between training and validation) and the mutilstep adjustment strategy (training with the full dataset). We trained 300 epochs in the training phase of the verification set, and selected checkpoints according to

the best results of the divided verification set for fine-tuning in the next stage. In addition, the full training phase trained 21 epochs for fine-tuning. The default parameter settings are used for data augment. In order to speed up the efficiency of the model, the batch size is determined according to the vidio memory during training, which is between 4 and 64. For training, in addition to the conventional data augment methods, we also adopt a more advanced automatic augment technology. More specifically, we use random horizontal flip with 50% probability, random vertical flip with 50% probability, random adjustment crop with 0.8 - 1.0 scale, random brightness/contrast adjustment with 40% probability. All images are resized to the desired network input size: in the CNNs performance experiments, the input size is 600 × 600. For Swin transformer, the input size is 384 × 384, while on BEiT, it is 512 × 512. In the testing phase, in addition to resizing and normalizing operations, we also use TTA, specifically 10-Crop verification.

On the basis of the above training, we replace various loss functions for evaluation. In addition, on the basis of the full amount of extremely unbalanced dataset training used in the first stage, PIM [16] and two-stage fine-tuning are introduced respectively. Finally, model ensemble is applied, which is fused separately from the result layer and the feature layer. In addition, the impact of meta data on the fungi dataset of this challenge is evaluated by post-processing and MLP, respectively.

4.2. Metrics

According to the requirements of the competition, the evaluation index used is the macro-average F_1. F_1^m, which is not affected by the class frequency and is more suitable for the long-tailed class distribution observed in nature. Given that the dataset is highly imbalanced and has long-tailed distributions, the learning process may ignore the least present species. So, F_1^m allows to easily assign a cost value to each label's two error types and measure more task-related performance. Define F_1^m as the mean of various F_1 scores:

$$F_1^m = \frac{1}{N} \sum_{S=1}^{N} F_1^S, \tag{1}$$

where N represents the number of classes and S is the species index.

4.3. Results

In this section, we compare the performance of well-known CNN models and Transformer-based models on the F_1^m metric. Then, two-stage fine-tuning is introduced. Additionally, we discuss the impact of metadata on the performance of image classification models. Finally, the best results are obtained using model ensemble.

The impact of different data augmentations. We try different data augmentation methods on the basic SwinTransformer, such as RandAugmet [20], TrivialAugment [21] and CutMix and random erasure, as shown in Table 2.

Two-Stage training Evaluation. We further explore on the basis of the previous model, namely SwinTransformer. The attention mechanism PIM [16] is introduced, and two-stage

Table 2
The effect of different data augments

Augment method	F_1^m score	Score fluctuation
Baseline	75.01%	-
RandAugment	75.42%	+0.41%
TrivialAugment	75.50%	+0.49%
Cutmix and Random Erasing	75.59%	+0.58%

Table 3
The effect of different Stage2 method

Stage2 method	F_1^m score	Score fluctuation
Baseline	75.01%	-
PIM	75.89%	+0.88%
DRS	75.82%	+0.81%

Table 4
The effect of different loss function

loss function	F_1^m score	Score fluctuation
Baseline(CE)	75.01%	-
FocalLoss	75.89%	+0.88%
SeesawLoss	75.45%	+0.44%

fine-tuning using balanced data is attempted. As shown in Table 3, we can see a significant improvement over the original baseline.

The impact of different loss function. In addition, the influence of different loss functions on the model accuracy is also discussed. As shown in Table 4, FocalLoss performs better in fine-grained classification.

The effect of different backbones. As shown in Table 5, we use different backbone training on the full data set of this challenge, and we adopt a high-performing data augment approach. In addition, we replace the loss function with FocalLoss, and use two-stage fine-tuning to obtain the results of the online test set evaluation. It is a pity that we do not use PIM [16] because of computing power, which requires a lot of computing resources and time. It can be seen that the Transformers series is better than the CNNs series as a whole, and BEiT has the best performance. Because the Private Leaderboard's test set accounts for 80%, the evaluation is more authoritative and reliable.

Use of meta data. First, we try to use the prior probability of meta data as post-processing, but the effect is not ideal as shown in Table 6. We believe that the poor performance is due to the large difference between the distributions of the training and test sets. In addition, we use MetaFormer and MLP for interactive learning of meta features and image features respectively, from the perspective of the model. Among them, MetaFormer is difficult to converge during training, and it may also lead to exploding gradients. And in MLP, the effect also decreased, as

Table 5
The effect of different backbones

Backbone	F_1^m score(Private)	F_1^m score(Public)
Efficientnet b6	76.57%	81.58%
Efficientnet b7	76.91%	80.73%
SwinTransformer-large	76.96%	80.02%
SwinTransformer-base	76.79%	79.98%
BEiT	77.64%	80.48%

Table 6
Effect of using meta data

Stage2 method	F_1^m score
Baseline	75.01%
Post-processing	73.7%
MLP	74.55%

Table 7
Effect of Model Ensemble

Fusion layer	F_1^m score
Softmax output layer	80.98%
Feature output layer	80.91%

Table 8
CVPR2022 Challenge-FGVC × Fungi leaderboard score

Stage	Rank	F_1^m score
Public	3	83.12%
Private	3	79.06%

shown in Table 6.

Of course, meta-information is not useless. We find a value different from the only value in the training set in the substrate attribute of the test set, which is spiders. We find the images where the substrate is spiders, reset their label to -1, and test it online, and the results show that we get a certain improvement.

Model Ensemble and Challenge Score. In the initial stage, when we divide the dataset 9:1, we perform two levels of fusion, as shown in Table 7. The output of the Softmax layer uses a simple average fusion, and the feature output layer uses the concat of the features extracted by the pooling of the last layer of each model. Then we train with an MLP to obtain the final result. It can be seen that the effect is not very different, so our final solution selects the first simple and effective fusion method. As shown in Table 8, we achieve excellent results on the leaderboard and 2nd on the Private and Public leaderboards.

5. Conclusions

In this work, we conduct a full exploratory analysis of the fungi dataset and experiment with multiple different types of backbones. On the basis of the previous one, a two-stage fine-tuning is introduced, and the model ensemble is carried out to obtain the optimal result. In addition, we delve into additional meta data and try various protocols with poor results. Our methods win second place in the CVPR2022 "Automatic fungi classification as an open-set machine learning problem" challenge.

References

[1] G. Van Horn, O. Mac Aodha, Y. Song, Y. Cui, C. Sun, A. Shepard, H. Adam, P. Perona, S. Belongie, The inaturalist species classification and detection dataset, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 8769–8778.

[2] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, B. Guo, Swin transformer: Hierarchical vision transformer using shifted windows, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 10012–10022.

[3] L. Picek, M. Šulc, J. Heilmann-Clausen, J. Matas, Overview of FungiCLEF 2022: Fungi recognition as an open set classification problem, in: Working Notes of CLEF 2022 - Conference and Labs of the Evaluation Forum, 2022.

[4] A. Joly, H. Goëau, S. Kahl, L. Picek, T. Lorieul, E. Cole, B. Deneu, M. Servajean, A. Durso, I. Bolon, et al., Lifeclef 2022 teaser: An evaluation of machine-learning based species identification and species distribution prediction, in: European Conference on Information Retrieval, Springer, 2022, pp. 390–399.

[5] A. Joly, H. Goëau, S. Kahl, L. Picek, T. Lorieul, E. Cole, B. Deneu, M. Servajean, A. Durso, H. Glotin, R. Planqué, W.-P. Vellinga, A. Navine, H. Klinck, T. Denton, I. Eggel, P. Bonnet, M. Šulc, M. Hruz, Overview of lifeclef 2022: an evaluation of machine-learning based species identification and species distribution prediction, in: International Conference of the Cross-Language Evaluation Forum for European Languages, Springer, 2022.

[6] W. Ge, X. Lin, Y. Yu, Weakly supervised complementary parts models for fine-grained image classification from the bottom up, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 3034–3043.

[7] C. Yu, X. Zhao, Q. Zheng, P. Zhang, X. You, Hierarchical bilinear pooling for fine-grained visual recognition, in: Proceedings of the European conference on computer vision (ECCV), 2018, pp. 574–589.

[8] W. Luo, X. Yang, X. Mo, Y. Lu, L. S. Davis, J. Li, J. Yang, S.-N. Lim, Cross-x learning for fine-grained visual categorization, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 8242–8251.

[9] J. Fu, H. Zheng, T. Mei, Look closer to see better: Recurrent attention convolutional neural network for fine-grained image recognition, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 4438–4446.

[10] Z. Yang, T. Luo, D. Wang, Z. Hu, J. Gao, L. Wang, Learning to navigate for fine-grained

classification, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 420–435.
[11] A. Behera, Z. Wharton, P. Hewage, A. Bera, Context-aware attentional pooling (cap) for fine-grained visual classification, arXiv preprint arXiv:2101.06635 (2021).
[12] J. He, J.-N. Chen, S. Liu, A. Kortylewski, C. Yang, Y. Bai, C. Wang, A. Yuille, Transfg: A transformer architecture for fine-grained recognition, arXiv preprint arXiv:2103.07976 (2021).
[13] G. Chu, B. Potetz, W. Wang, A. Howard, Y. Song, F. Brucher, T. Leung, H. Adam, Geo-aware networks for fine-grained recognition, in: Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, 2019, pp. 0–0.
[14] O. Mac Aodha, E. Cole, P. Perona, Presence-only geographical priors for fine-grained image classification, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 9596–9606.
[15] X. He, Y. Peng, Fine-grained image classification via combining vision and language, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 5994–6002.
[16] P.-Y. Chou, C.-H. Lin, W.-C. Kao, A novel plug-in module for fine-grained visual classification, arXiv preprint arXiv:2202.03822 (2022).
[17] Y. Zhang, X.-S. Wei, B. Zhou, J. Wu, Bag of tricks for long-tailed visual recognition with deep convolutional neural networks, in: Proceedings of the AAAI conference on artificial intelligence, volume 35, 2021, pp. 3447–3455.
[18] L. Picek, M. Šulc, J. Matas, T. S. Jeppesen, J. Heilmann-Clausen, T. Læssøe, T. Frøslev, Danish fungi 2020-not just another image recognition dataset, in: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2022, pp. 1525–1535.
[19] E. D. Cubuk, B. Zoph, D. Mane, V. Vasudevan, Q. V. Le, Autoaugment: Learning augmentation strategies from data, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 113–123.
[20] E. D. Cubuk, B. Zoph, J. Shlens, Q. V. Le, Randaugment: Practical automated data augmentation with a reduced search space, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 2020, pp. 702–703.
[21] S. G. Müller, F. Hutter, Trivialaugment: Tuning-free yet state-of-the-art data augmentation, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 774–782.
[22] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al., An image is worth 16x16 words: Transformers for image recognition at scale, arXiv preprint arXiv:2010.11929 (2020).
[23] M. Tan, Q. Le, Efficientnet: Rethinking model scaling for convolutional neural networks, in: International conference on machine learning, PMLR, 2019, pp. 6105–6114.
[24] H. Bao, L. Dong, F. Wei, Beit: Bert pre-training of image transformers, arXiv preprint arXiv:2106.08254 (2021).
[25] T.-Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollár, Focal loss for dense object detection, in: Proceedings of the IEEE international conference on computer vision, 2017, pp. 2980–2988.
[26] J. Wang, W. Zhang, Y. Zang, Y. Cao, J. Pang, T. Gong, K. Chen, Z. Liu, C. C. Loy, D. Lin, Seesaw loss for long-tailed instance segmentation, in: Proceedings of the IEEE/CVF

conference on computer vision and pattern recognition, 2021, pp. 9695–9704.
[27] K. Cao, C. Wei, A. Gaidon, N. Arechiga, T. Ma, Learning imbalanced datasets with label-distribution-aware margin loss, Advances in neural information processing systems 32 (2019).
[28] Y. Cui, M. Jia, T.-Y. Lin, Y. Song, S. Belongie, Class-balanced loss based on effective number of samples, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019, pp. 9268–9277.