Long Tailed Fine-Grained Food Recognition: Auxiliary Classifier and Multi-Modality Approaches

Hsiao-Tzu Hung
洪筱慈
R08922A20
Master Program of Artificial Intelligence
r08922a20@ntu.edu.tw

Bo-Yu Chen
陳柏昱
R10944032
Graduate Institute of Networking and Multimedia
r10944032@ntu.edu.tw

Tun-Min Hung
洪敦敏
RA in Music AI Lab, Academia Sinica
allenhung@iis.sinica.edu.tw

Payko Liu
劉欣平
R09946030
Data Science Degree Program
r09946030@ntu.edu.tw

1 Introduction

The Long-tailed distribution and fine-grained images are the two main challenges in this work. Based on the chosen backbone, TransFG, we introduce two methods for each challenge. For the long-tailed problem, we implement two methods with auxiliary classifier, trying to provide the model with the information about the occurrence frequency of images and avoiding it being overfit to the frequent categories. For the fine-grained problem, we also implement two methods which utilize the information given by the Chinese names of categories.

2 Methodology or Model Architecture

We tackle the long tailed and fine-grained problems separately. For long tailed problem, we add an additional auxiliary classifier to predict the frequency of the input image. As for the fine-grained problem, we consider the Chinese label of the input image as an additional information and use it for multi-modality learning. In the following sections, we will first introduce the backbone model in section 2.1, and describe the auxiliary classifier approaches in section 2.2, 2.3. Finally, the multi-modality approaches will be introduced in section 2.4 and 2.5.

2.1 Backbone: TransFG

The backbone is TransFG [1], a model that is based on Vision Transformer(ViT). The purpose of the model is to handle fine-grained classification task, and it achieved SOTA performance on several benchmarks, including CUB-200-2011, Stanford Dogs, and NABirds. The novelty of the paper is that it proposed a mechanism called Part Selection Module(PSM). The module chose the index with the highest attention with every attention head and concatenated those selected tokens as input to the last transformer layer. By doing so, it can distinguish subtle differences between two images by localizing important features and discarding redundant ones. Because challenge 3 is also a fine-grained classification task, we chose the SOTA model as the starting point of our project.

Deep Learning for Computer Vision (Fall 2022) , National Taiwan University.

2.2 Auxiliary classifier in multi-task fashion

The ordinary classification model usually overfits the head class in long-tailed distribution, leading to performance degradation in the tailed class. We propose a multi-task approach to let the model know which kind of data they currently deal with to mitigate this problem. To be more specific, we add an auxiliary classifier after the feature extractor but parallel with the class classifier. The goal of the auxiliary classifier is to predict the frequency class as accurately as possible. Furthermore, we were inspired by the work proposed by Kang [2] that decomposes long-tailed classification into two-stage. In the first stage, they apply random sampling to train a suitable feature extractor. In the second stage, they freeze the feature extractor and use the class-balanced sampling to train a good classifier. As a result, we apply random sampling in the first stage to train a suitable feature extractor. Next, we add an auxiliary classifier parallel with the class classifier and train it by class balanced sampler in the multi-task fashion.

2.3 Auxiliary classifier as task separator

In this method, we use an additional classifier to predict the frequency category of an input image. The input of the classifier is the predicted logits given by the baseline backbone with the labels as r, f, c, which indicate rare, freq, comm respectively. We additionally trained three TransFG models using only images in rare (freq, or comm) set. Then, we use the classifier as a gate to separate input images into three tracks, i.e., if an image is classified as a rare image, it will next be predicted by the model trained on the rare set.

2.4 Semantic CLS token and contrastive learning

The first multi-modality approach is to use Chinese label to make label embedding have a semantic distance relationship. In TransFG, the CLS token embedding is learned from the image patches.However, in food images, it is common that two kinds of food share similar image feature but have very different Chinese name. As a result, we want to align the CLS token embedding with the language embedding and separate categories that look similar. Specifically, given a sample x_i from class i with Chinese name C_i, the CLS token is initialized by an random parameter h_i in TransFG. We first use the BERT model to extract the averaged hidden state of C_i, denoted as H_i. During training, we pass both h_i and the hidden state H_i through the Transformer module to get z_i and z_hi respectively. Then we calculate the contrastive loss of the CLS token embedding and the BERT-based embedding. Formally, the semantic contrastive loss over a batch size B is denoted as:

$$\mathcal{L}_{con\_CLS} = \frac{1}{B^2}\sum_{i}^{B}[\sum_{j:y_i=y_j}^{B}(1-\mathit{Sim}(z_i, z_{hi}) + \sum_{j:y_i\neq y_j}^{B} max(\mathit{Sim}(z_i, z_{hj}) - \alpha), 0))] \quad (1)$$

where α is a constant margin. Finally, the model is trained with the sum of the three losses which can be expressed as $\mathcal{L} = \mathcal{L}_{cross}(y, y') + \mathcal{L}_{con}(z) + \beta\mathcal{L}_{con\_CLS}(z)$, where $\mathcal{L}_{cross}(y, y')$ is the cross-entropy loss between the predicted label y' and the ground-truth label y, $\mathcal{L}_{con}(z)$ is the original contrastive loss over a batch of size B, and the β is a tunable hyperparameter.

2.5 Fine-tune CLIP model

The motivation behind using the CLIP[3] model is that the accuracy of CLIP is usually competitive with the fully supervised method. Furthermore, based on the experiment in the paper, it showed that applying CLIP zero-shot on food-101 has 22.5 performance gain than supervised method using ResNet-50 features. To use the CLIP model, we should translate Chinese label into English label. As a result, We utilized Google Translate api to translate Chinese label into English label. On the other hand, we have the training set at our disposal. We used the training set to fine-tune the CLIP model. To be more specific, we transformed the label into caption "This is a photo of {label}, a type of food." as the text input.

3 Implementation Details

In this section, we first introduce the detailed preprocessing setup for the dataset. Hyperparameter choices of the four approaches are then described in section 3.2 for reproducibility.

3.1 Data preprocessing

The image is resized to (600, 600) with bilinear interpolation method, random crop to (448, 448), random horizontal flip, and normalize with mean (0.485, 0.456, 0.406) and std (0.229, 0.224, 0.225).

3.2 Hyperparameter Choices

TransFG For training, the total step is 100000. We used SGD optimizer and set the initial learning rate as 0.03. The Cosine Warmup scheduler is used with 500 steps warming up with linear increase and cosine decay in the following steps. The batch size is 16. The estimated training time is 15 hours on four GTX-1080Ti 8G GPUs.

Auxiliary classifier in multi-task fashion In this multi-task approach, we simply add a single-layer linear model directly after the feature extractor. The input and output dimension is 2048 and 3 to classify the frequency category.

Auxiliary classifier as task separator In the task separator method, the classifier we used is a simple linear model with 4 linear layers whose hidden dimensions 512, 256, 128 and the dimension of the output is 3. For training, we use Adam optimizer and set the initial learning rate as 0.001. A step scheduler is used and the step size is 10 with the decreasing rate as 0.1. The batch size of train set is 16 and the one of validation set is 8. Moreover, to accelerate the training process by saving the time used for calculating gradients, we freeze the backbone model while training the frequency classifier.

Semantic CLS token and contrastive learning We train the TransFG with semantic contrastive loss for 80000 steps with batch size to be 16, and set the β as 0.001, α as 0.4.

Fine-tune CLIP model When training CLIP, the batch-size is 256. We used the Adam optimizer with betas=(0.9, 0.98), ϵ=1e-6, learning rate as 5e-6, and weight_decay=0.05. The vocabulary size is 49408.

4 Experiments and Discussion

4.1 Classification result

Approaches	Freq.	Comm.	Rare	Main
Baseline	**0.904**	0.704	0.275	0.726
Aux: multi-task	0.891	**0.759**	**0.399**	**0.766**
Aux: 3-stage	0.377	0.535	0.216	0.441
MM: cls token	**0.904**	0.722	0.287	0.736
MM: CLIP (finetune)	0.624	0.470	0.224	0.499

Table 1: Classification result.

Table 1 shows the experiment results of the baseline model and four methods. In this multi-task approach, we found that training an auxiliary classifier in a multi-task fashion does improve the performance in most of the tracks, especially rare track after second stage training. However, it is not clear whether the auxiliary classifier makes a difference. As a result, we conduct a comprehensive ablation study to answer this question.

On the other hand, the outcome given by the method with task separator is not as good as our expectation. The possible reason is that the logits given by the pretrained backbone did not bring enough information for the classifier to learn the different between three frequency sets. In addition, the error will be accumulated through the multiple-stage inference process; if the pretrained model gives bad logits or the classifier gives wrong predictions, then the image will ultimately go into the wrong track while the last models are only expert on their own tracks.

As for the semantic CLS token approach, the accuracy of the Comm./Rare/Main tracks is improved, and the accuracy of the Freq track stays the same as the baseline. This result indicates that the contrastive learning performed on the multi-modal CLS token pairs can align the label embedding to the semantic label embedding to a certain level. Finally, the CLIP model performed not as expected even fine-tuning on the training set. There are two possible reasons leading to the result. First, we translate the food name from Chinese to English using the Google Translation API. By doing so, the English counterpart might not illustrate the meaning of the food as much as the Chinese did. Secondly, due to the constraint of the hardware, we fine-tuned the model with batch size 256. On the other hand, the CLIP model was trained with batch size 32768 which was a huge gap with our setup.

4.2 Ablation study

To understand whether auxiliary classifiers improve the performance, we conduct an ablation study which is shown in table 4.2. We follow the Kang [2] decompose the long-tailed problem into two stages. Next, we denote 1-stage as full modeling training with random sampling and 2-stage as class classifier training with class-balanced sampling. Furthermore, we refer to **None** as not training at this stage, **Basic** as the usual training approach, and **Multi-task** as multi-task training with auxiliary classifier. In table 4.2, we can observe that the auxiliary classifier does not improve the performance when training in the 1-stage. However, it does improve the performance if we train by Basic in 1-stage and **Mulit-Task** in 2-stage. Surprisingly, if we train it by **Mulit-Task** in both stages, the performance is almost the same as training it by **Basic** in both stages. As a result, we conclude that training an auxiliary classifier in a multi-task fashion does help them, but it can only be applied in the second stage.

1-Stage	2-Stage	Freq	Comm	Rare	Main
Basic	None	0.904	0.704	0.275	0.726
Multi-Task	None	**0.906**	0.711	0.240	0.725
Basic	Basic	0.890	0.743	0.379	0.754
Basic	Multi-Task	0.890	**0.759**	**0.400**	**0.766**
Multi-Task	Multi-Task	0.895	0.741	0.384	0.754

Table 2: Ablation Study of Multi-tasking.

5 Conclusion

We've tried two approaches to tackle fine-grained and long-tailed problem, respectively. In section 2.4, we can see that with external information from chinese labels and contrastive learning, it showed improvement from the baseline model in table 1 in Comm, Rare, and Main track. Furthermore, in section 2.2, with the help of an auxiliary classifier in the second-stage training, we can see obvious performance gain than baseline model in Comm, Rare, and Main track in table 1. In conclusion, both methods we proposed, **Semantic CLS token and contrastive learning** and **Auxiliary classifier in multi-task fashion** not only have novelty but also improve the accuracy from the baseline model.

References

[1] Ju He et al. "TransFG: A Transformer Architecture for Fine-grained Recognition". In: *arXiv preprint arXiv:2103.07976* (2021).

[2] Bingyi Kang et al. "Decoupling representation and classifier for long-tailed recognition". In: *ICLR* (2020).

[3] Alec Radford et al. "Learning transferable visual models from natural language supervision". In: *arXiv preprint arXiv:2103.00020* (2021).