Report for Kaggle competition—Hongyu Zhou

A. Introduction

This is a bird classification task where there are 1082 original training data in RGB colors and 20 classes. Before doing the task, some explorations into the images as well as the labels shows that the color of some parts of the bird such as the wings, the head and so on, is crucial for the identification of the birds. In addition, the shape of tails and beak can also be important. The difficulty can come from occlusion of important parts, small size in the image or similar color with the background.

Since the number of data set is limited, data augmentation is required to add on the variety and the difficulty of training. Pretrained models on ImageNet are used to save the training time and to avoid overfitting to some extent. Self supervised learning on unlabelled dataset is also taken into consideration. UNET with resnet encooder is applied in the self-supervised learning task. As for pretrained models, multiple networks have been tested including ResNet, VGG and ViT. The model is firstly run locally and then on Colab to increase the batch size and input image size if the model is promising.

B. Methods

B.1. Data augmentation

Among multiple data augmentation approaches, I chose the rotation, flipping, changing the constrast, adding gaussian noise with $\sigma = 10$, and Gaussian blurring.

B.2. Pretrain model

Pretrained models on ImageNet are mainly chosen in the torchvision package based on classification tasks. Models on the semantic segmentation task such as Deeplab is also tested, but without better results(82% in validation set).

The pretrained models tested are VGG19, ResNet152 and ViT base model with 16 hidden layers. with weights of the best accuracy score. The selection of models is based on the accuracy score and its number of parameters. For models more than 100M, the memory space is not enough even on Colab.

B.3. Self supervised learning

The dataset of self-supervised learning is Nabirds unlabelled images, which contains about 50000 images whose total size is 9GB. Since the color and the shape of parts is the key to identify a bird, the pretext task is chosen to restore RGB images from grey-scale images yet the number of channels of the grey-scale image is set to be 3. I applied a UNET to restore the grey-scale images, whose encoder is ResNet50 due to the limitation of memory. The code is in the ssl function in main.py. Thanks to fastai, the implementation of UNET is simpler when the encoder is decided. Moreover, to train faster in the pretext task, I also initialized the encoder with pretrained models on ImageNet.

	VGG19	**ResNet152**	**ViT base16**
Accuracy	85	89	93

	ResNet50 ImageNet	**ResNet50 ssl**
Accuracy	88	90

Table 1

C. Results

The following results are on the validation set.

The first one is a comparison on the power of data augmentation. I first applied ResNet152 without data augmentation, giving me an accuracy of 83%, which was a good start but not very high. Then after data augmentation, there was just a jump in accuracy to be 91%. And in the Kaggle test, without data augmentation I got about 63% but with it I had 78%.

Then it is a comparison of three models and the model obtained by self-supervised learning in table 1. The ViT is really outstanding and the best result I got in the Kaggle test evaluation comes from this model.

As for hyperparameters, I found that the learning rate of the pretrained model should be very small, which I set it to be 1e-5. The larger the batch size, the better the result. So I made it 64 or 32 in short of memory. In self-supervised learning, however, I had to make it 8, so I update the weights every 4 steps. The seed is set to be 3047 as suggested by the paper [1]. The optimizer is Adam with a linear scheduler.

Time is not a problem in pretrained models but it took an hour for each epoch in self-supervised learning, which I run for 2 epochs to see the result converging. Due to the size of dataset, it is difficult to run it on Colab, and I fail to apply the most decent model as the backbone of UNET due to the limitation of memory. But there is still a little improvement on the performance.

References

References

[1] David Picard, *Torch.manual seed(3407) is all you need: On the influence of random seeds in deep learning architectures for computer vision*,arXiv:2109.08203