Assignment 3: Bird image classification competition

Bastien DECHAMPS
Ecole des Ponts ParisTech
bastien.dechamps@eleves.enpc.fr

Abstract

This report will present the approach I made for the image classification challenge of the Object Recognition and Computer Vision MVA course. The problem was to classify birds images into 20 different breeds with a very small annotated dataset (around 60 images per breed). I tried many differents architechtures and came to my best score with an pretrained Inception-v3 model [4], which scored 0.8516 on the public leaderboard.

1. Introduction

A common approach to deal with fine-grained image classification with very few data is to fine-tune a pretrained classifier which have been trained on bigger datasets such as Imagenet. Among the models which gave me good results, I tried Resnet [1], an improved version of it called Resnext [5], Inception-v3 [4] and Resnext-WSL [3], a weakly supervised resnext trained by Facebook on 940 million public images.

2. Pipeline

In this section, I will present the data preprocessing steps as well as the best models I used, implemented in Pytorch.

2.1. Bird detection

As every image is not necessarily centered on the birds, I used a pretrained SSD model [2] to detect them with bounding boxes, with which I cropped the image. The model was implemented in the `gluoncv` library. Unfortunately, this preprocessing step reduced my accuracy on the test set, even though it my validation score was sligthly better, so I did not used it for my final submissions.

2.2. Models

The models I used that gave me the best accuracies were Inception-v3 and Resnext-WSL. The Resnext-WSL uses a Resnext architecture and was trained in a weakly supervised way on a huge dataset. As it has a lot of parameters, I trained only the last prediction layer along with the last convolution layer for 20 epochs, using a momentum of 0.9 and a learning rate of 0.005 decayed by 0.9 at each epoch.

The best accuracy I had was from the Inception-v3 model, but with custom weights trained on the iNaturalist dataset available here[1]. I trained the model by fine-tuning the last prediction layer for 5 epochs before unfreezing the whole network and training it 10 epochs. As mentioned in [4], I used RMSprop with decay of 0.9 and $\epsilon = 1$ and a learning rate of 0.045 decayed every two epochs by an exponential rate of 0.94.

3. Results

I trained the different models using the k-fold cross validation method with $k = 5$ on a RTX 2070. The obtained accuracy are listed in the table 1.

	Resnext-WSL	Inception-v3
val.	0.9417	**0.9515**
test	0.8452	**0.8516**

Table 1. Accuracies for the different models

References

[1] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015. 1

[2] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott E. Reed, Cheng-Yang Fu, and Alexander C. Berg. SSD: single shot multibox detector. *CoRR*, 2015. 1

[3] Dhruv Mahajan, Ross B. Girshick, Vignesh Ramanathan, Kaiming He, Manohar Paluri, Yixuan Li, Ashwin Bharambe, and Laurens van der Maaten. Exploring the limits of weakly supervised pretraining. *CoRR*, 2018. 1

[4] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. 1

[5] Saining Xie, Ross B. Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. *CoRR*. 1

[1] https://github.com/macaodha/inat_comp_2018