



清华大学
Tsinghua University

用ResNet18对数据集进行分类

报告人：潘明轩

报告时间：2023-4-20

1 环境配置

- 在命令行输入：conda create -n media python=3.9 也可以选择你喜欢的python版本
- conda activate media 激活环境
- 安装pytorch

PyTorch Build	Stable (2.0.0)		Preview (Nightly)	
Your OS	Linux	Mac	Windows	
Package	Conda	Pip	LibTorch	Source
Language	Python		C++ / Java	
Compute Platform	CUDA 11.7	CUDA 11.8	ROCm 5.4.2	CPU
Run this Command:	<pre>conda install pytorch torchvision torchaudio cpuonly -c pytorch</pre>			

- 根据自己的电脑配置选择最合适的，有显卡的电脑推荐自学安装CUDA



1 环境配置

- 在命令行输入：python
- 查看torch以及torchvision是否正常安装

```
>>> import torch
>>> torch.__version__
'2.0.0'
>>> import torchvision
>>> torchvision.__version__
'0.15.0'
>>> |
```

- 其他本次实验可能会用到的包：
- matplotlib, numpy, opencv-python
- 如果之后经常用这台电脑编程，这些包要常备。
(助教的电脑配置：12代i5无独显)



2 数据集

- 数据集是iNaturalist的子集



Accipiter cooperii



Uta stansburiana elegans



Aquilegia canadensis

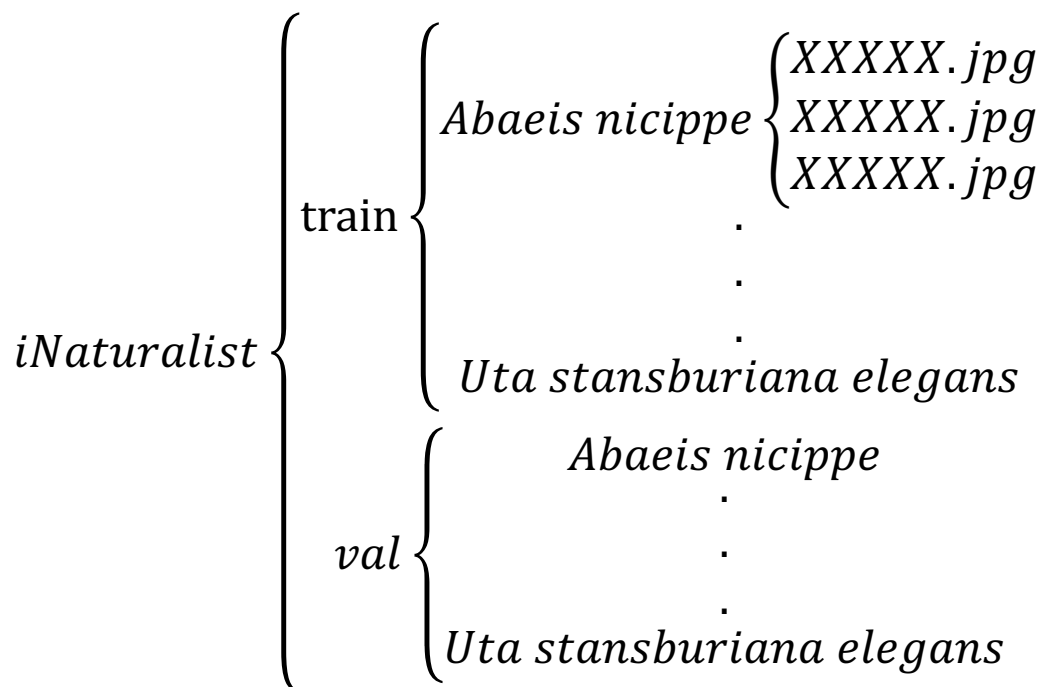


Celithemis eponina



2 数据集

- 数据集格式：



- 包含20个类，每类约100个图片，测试集为约50个图片
- 处理奇怪的类别名称也是数据读取的一部分
- 下载链接：

<https://cloud.tsinghua.edu.cn/f/1d8c3fe01df345ca847f/?dl=1>



3 实验

- 题目描述

1) 训练ResNet18网络，完成数据集的分类任务

我们提供一个Baseline，需要将代码跑起来，比较结果。

2) 调参实验，尝试提升网络的分类效果

有哪些参数是可以调整的？



3 实验

- 数据读取
- Pytorch需要自定义一个继承Dataset的类，并继承三个函数：
 - __init__，初始化数据集
 - __len__，这个函数应当返回数据集长度
 - __getitem__，这个函数应当返回第index的数据
- 此外，还可以用torchvision中的transforms实现对图片的常规处理
Resize, ToTensor, Normalize
- 从dataset中读取的数据应当为tensor，这是torch包中的一般数据格式，包括标签也要转化为tensor形式。最后套一层DataLoader即可转化为一个读取器，每次迭代都会给出一组数据



3 实验

- 网络实现

- 在torchvision的models模块中，为我们提供了许多基本网络结构，可以直接进行调用，只需要一句话：

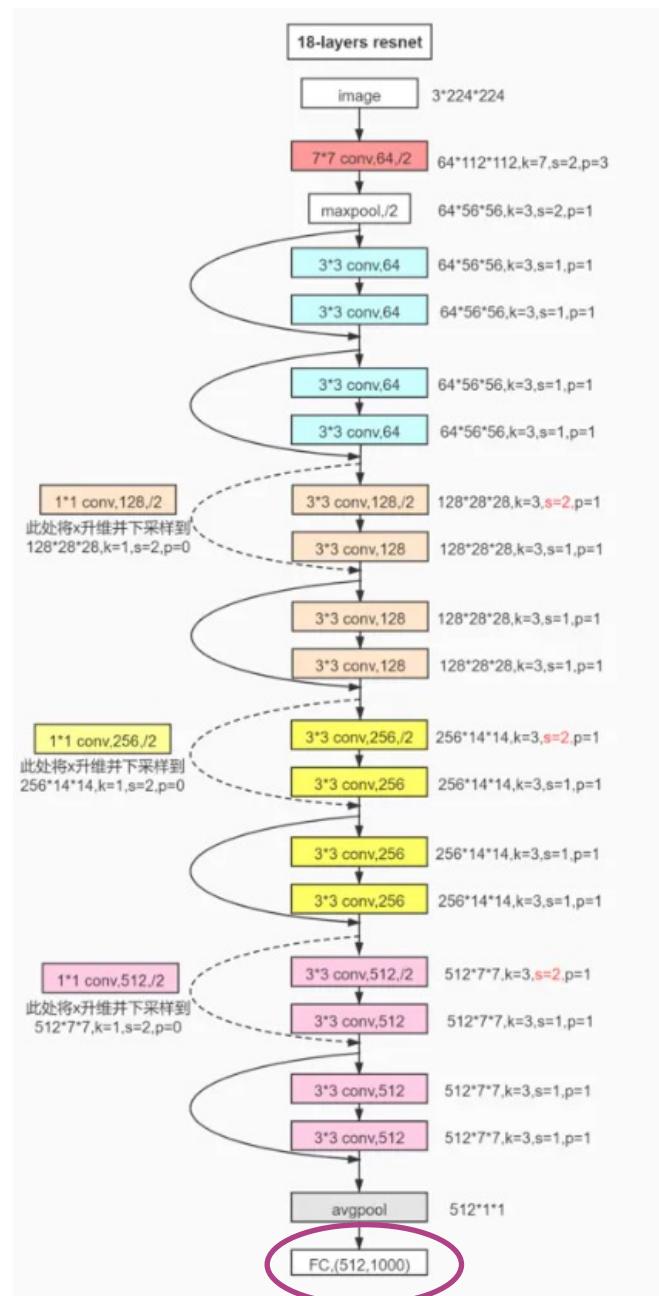
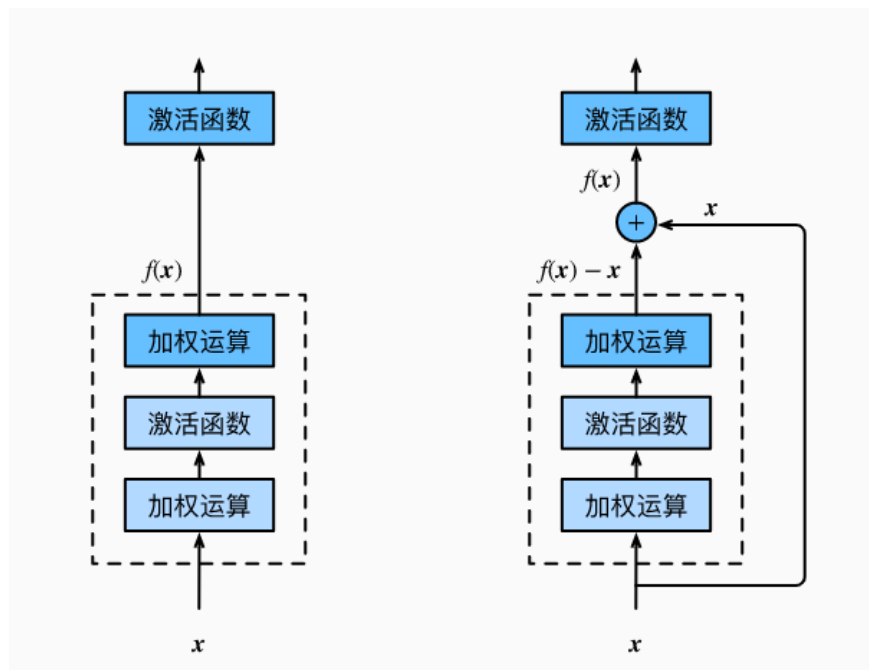
```
from torchvision import models  
model = models.resnet18(pretrained=False)
```

- 思考：如果将resnet18用在我们自己的数据集上，还需要做哪一步改动
- 我们还给出了一个resnet18的自主实现，但在其中空缺了一部分，在model.py中，请同学们自主实现缺少的部分，model.py的实际运行结果应和torchvision自带的包差不多。



3 实验

- Resnet18



3 实验

- 训练
- 训练中需要自行设计的参数：

Epochs轮次，lr学习率，batchsize批大小

如果有CUDA的话，device可以改成cuda

自行学习：Loss的选择和Optimizer的选择

本实验中选择了交叉熵损失函数和Adam优化器

```
logits = model(img)
loss = criterion(logits, label)
optimizer.zero_grad()
loss.backward()
optimizer.step()
```



4 作业内容

- 成功运行Baseline，记录在其中遇到的困难
- 是否预训练是否对结果有很大的影响
- 提交的实验报告第一部分应包含：
 - 1) 运行Baseline中遇到的难点以及如何解决（或一些感想）
 - 2) 用没有预训练的ResNet18进行实验，绘制出总/平均Loss和准确率随着Epoch变化的曲线
 - 3) 用预训练的ResNet18进行实验，绘制出总/平均Loss和准确率随着Epoch变化的曲线，它有多大提升？
 - 4) 应介绍你所采用的各项参数，提供一组即可

一个心理预期：40-50，50-60



4 作业内容

- 调整各项参数，比较不同参数对实验结果的影响
- 至少包含： lr 对实验造成的影响
- 提交的实验报告第二部分应包含：
 - 1) 在三组不同的 lr 下，绘制出总/平均Loss和准确率随着Epoch变化的曲线
 - 2) （选做）试着尝试其他参数，如轮次、batchsize和优化器，绘制出总/平均Loss和准确率随着Epoch变化的曲线
 - 3) 总结不同参数对分类任务的影响



谢谢大家！ 欢迎批评指正！

