代码原型：<https://github.com/bubbliiiing/yolox-pytorch>

运行环境：

Python=3.8.17 （太新的python好像不太行）

Spikingjelly==0.0.0.14

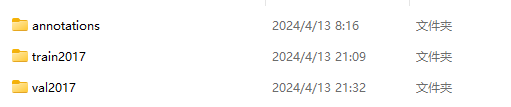
Pytorch==2.2.0+cu118（应该是2.0以上的pytorch）

Opencv-python==4.8.0.74 (太新的不行)

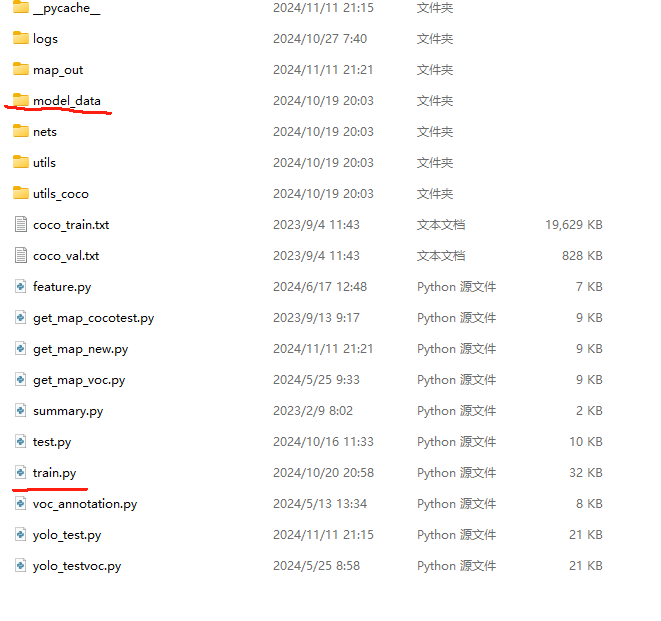
Numpy==1.24.4 (numpy版本不能上2.0)

上面这些是我没记错的话需要安装注意的包，不是无脑按最新的，其他的少什么运行的时候应该直接按就行

Coco数据集设置：COCO-2017，一般我就直接把它跟train文件放在一个路径下面了，这样不用考虑绝对路径的问题了



Annotations是标注，剩下俩文件夹是数据集图像，coco\_dataset和运行文件的路径关系是这样的：



运行coco\_annotations.py，生成训练数据集列表coco\_train.txt和coco\_val.txt（看上面）

训练：python train\_noenc.py，具体训练参数这些看这里代码改就行。

model\_data文件夹下面有预训练权重yolox\_s.pth，训练“s”结构模型可加载（给phi那个参数设置成’s’）。这里面还有一个best\_epoch\_weights.pth是可以直接做inference的，但是不能当作预训练权重加载，莫名其妙训练性能不好。



测试：

测试代码分好几部分

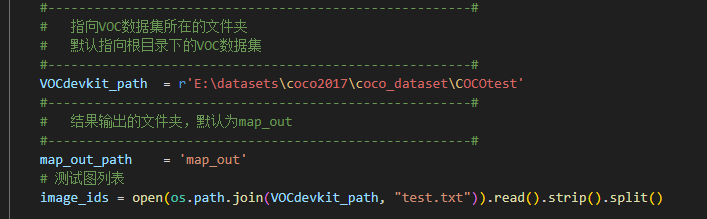
测试网络结构、权重设置：yolo\_test.py，不可直接运行。phi设置模型大小，要跟训练的结构/加载的权重一致，现在用’s’就够用了。

Yolo\_test.py设置完以后，测试代码有两个：

text\_noenc.py: 这个用的少，它只会生成结果，如果需要生成带检测框的图可以运行这个，给yolo\_test.py设置好以后直接python text\_noenc.py运行就行

get\_map\_new.py: 一般用这个，可以直接在COCO val数据集上得到定量结果。Map\_mode一般先设置为0得到检测结果后，在设置成4进行检测框计算。

重点设置：



VOCdevkit\_path：COCO2017测试集的位置，这个我会打个包发一下，里面的东西跟直接用数据集和标签略有不一样

下面的test.txt也会在这个压缩包里，只需要给上面的文件路径改一下就行，不放心就用绝对路径。

网络具体代码在nets文件夹里，backbone在darknet.py，FPN和检测头都在yolo.py，想看结构可以去那