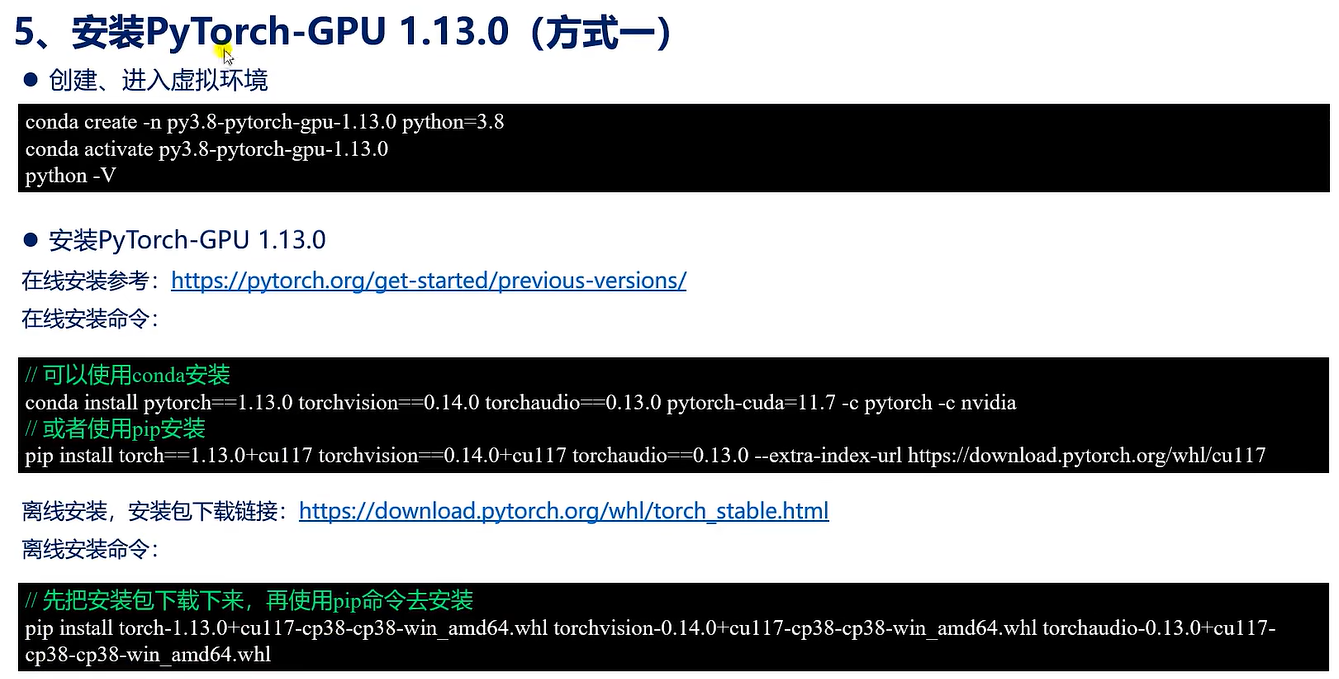
上一节已经安装了Tensorflow-GPU版本了，安装完成之后，我们训练模型发现是没有问题的。

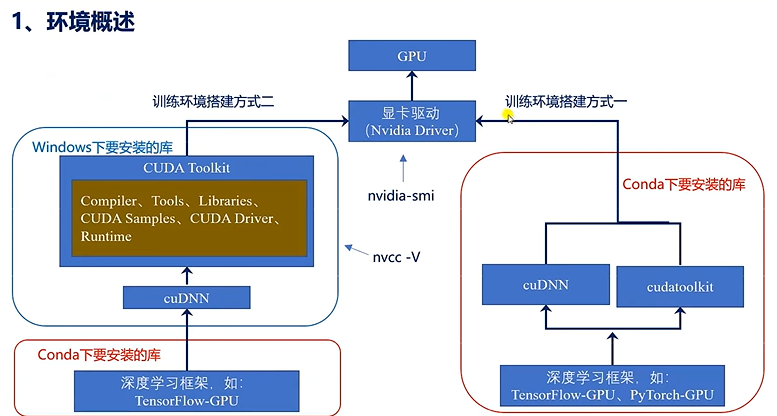
本讲课程我们就来讲解怎么安装PyTorch-GPU版本。

/\*

这种方式的话就是安装PyTorch-GPU的方式：



也是前面提到的方式一安装:



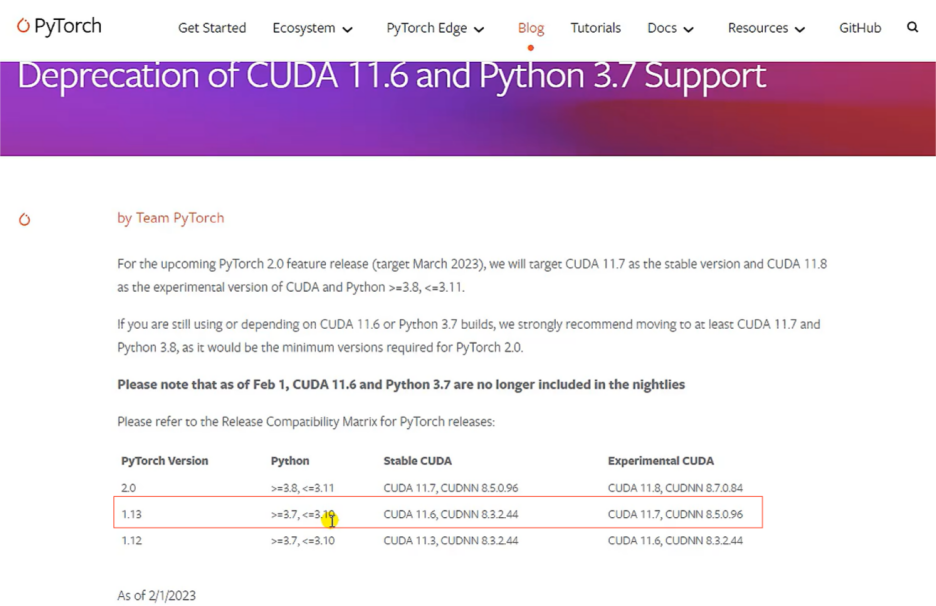
也就是说在Anaconda下我们先用Conda命令去安装这两个依赖库，安装好了这两个依赖库之后，再安装PyTorch-GPU。然后再验证PyTorch-GPU是否可以用到显卡，并正常训练模型。

/\*

1.首先新建、进入虚拟环境：

由于前面要求PyTorch-GPU要求Python至少是3.7版本，最大3.10版本，所以可以先新建一个3.7~3.10版本的环境。再去安装PyTorch。这里直接选择安装Python 3.8。

<https://pytorch.org/blog/deprecation-cuda-python-support/>



在前面我们新建了TensorFlow训练模型的虚拟环境。

然后在这里我们又需要新建另一个PyTorch-GPU的训练模型的环境。这两个训练环境，我们是分开来搭建的，之所以这么做，主要是考虑到可能TensorFlow和PyTorch的某些依赖库可能版本会发生冲突。所以选择将这两个分别安装在独立的环境下。

执行命令新建虚拟环境：

conda create -n py3.8-pytorch-gpu-1.13.0 python=3.8

conda activate py3:8-pytorch-gpu-1.13.0

python -V

1. 根据官网的安装教程执行对应的命令安装PyTorch-GPU：

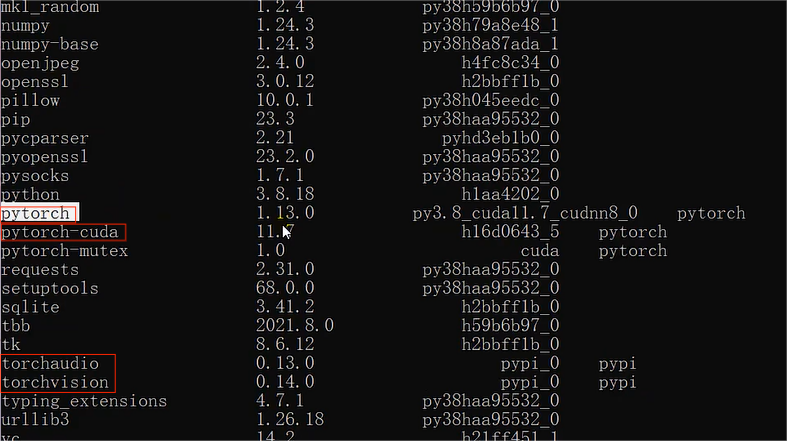
<https://pytorch.org/get-started/previous-versions/>

在线安装

使用conda安装：

conda install pytorch==1.13.0 torchvision==0.14.0 torchaudio==0.13.0 pytorch-cuda=11.7 -c pytorch -c nvidia

执行命令conda list可以看到对应的版本:



使用pip安装：

pip install torch==1.13.0+cul17 torchvision==0.14.0+cu117 torchaudio==0.13.0 --extra-index-url <https://download.pytorch.org/whl/cu117>

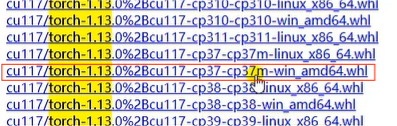
离线安装：

先去下载对应的安装包，再用pip命令通过安装包来安装PyTorch：

<https://download.pytorch.org/whl/torch_stable.html>

离线安装命令：

pip install torch-1.13.0+cu117-cp38-cp38-win\_amd64.whl torchvision-0.14.0+cu117-cp38-cp38-win\_amd64.whl torchaudio-0.13.0+cul17-cp38-cp38-win\_amd64.whl







3.安装好之后，就可以初步验证PyTorch-GPU是否可以使用

初步验证：

1. 首先进入python环境下：执行python
2. 导入PyTorch相关的一些库：

import torch

import torchvision

//导入后就可查看版本：

print(torch.\_\_version\_\_)

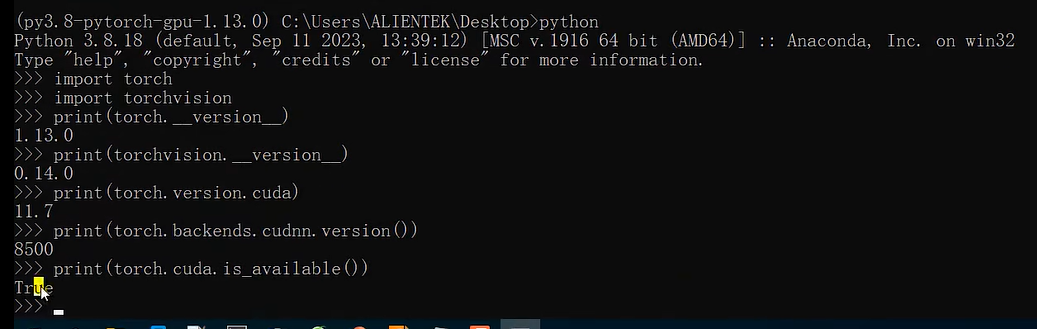
print(torchvision.\_\_version\_\_)

print(torch.version.cuda)

print(torch.backends.cudnn.version())

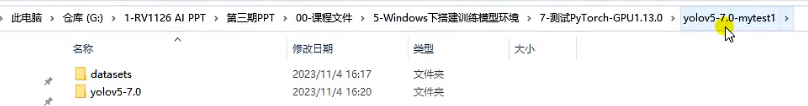
print(torch.cuda.is\_available())



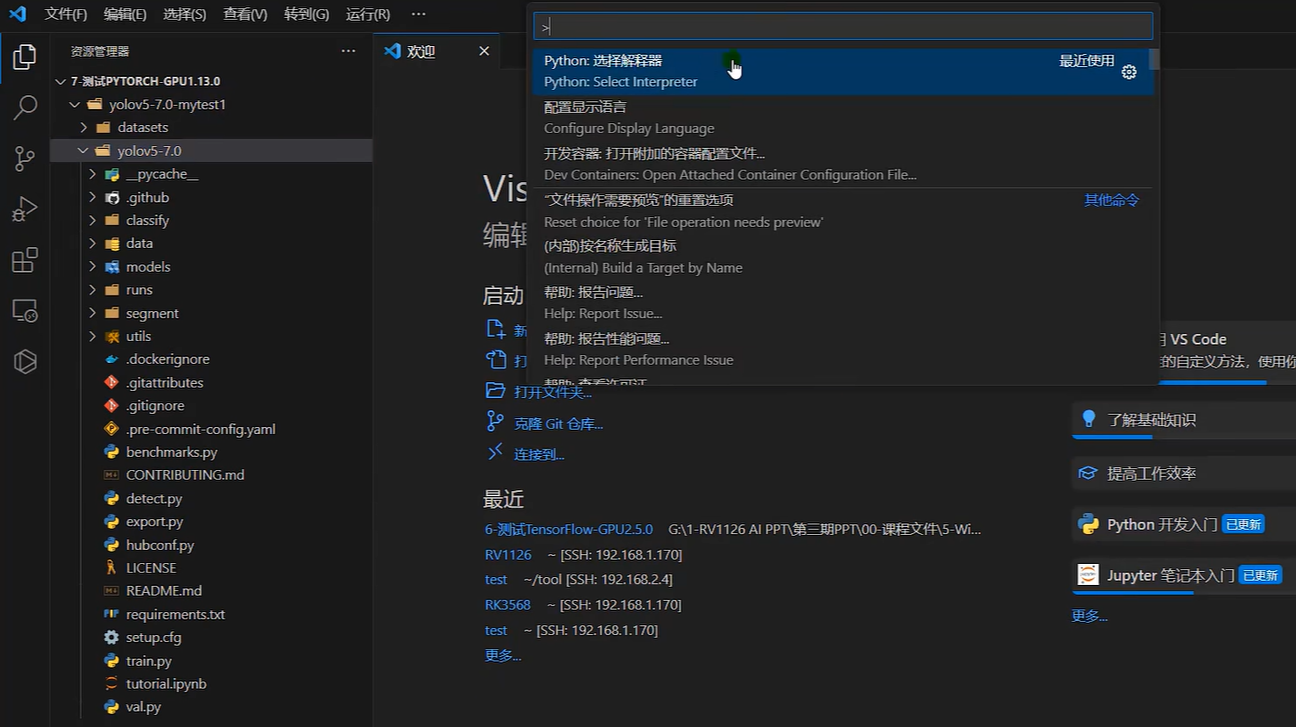


4.进一步验证的话就是先训练一个模型，（以YOLOV5 V7.0版本为例子验证PyTorch-GPU 1.13.0是否可以使用）：

测试文件目录：



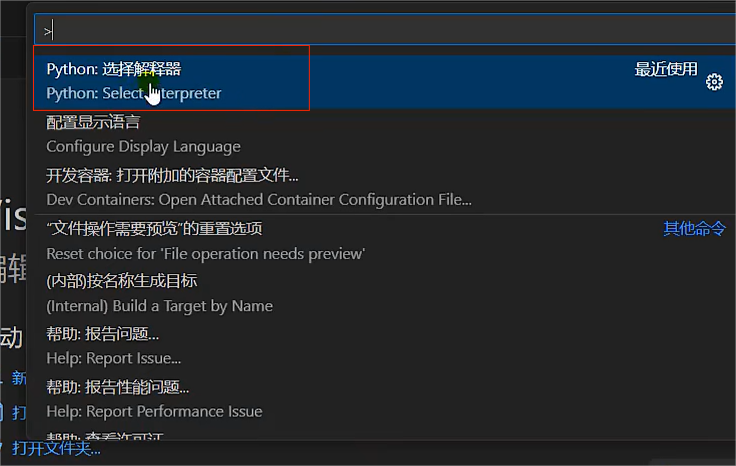
使用VScode打开：



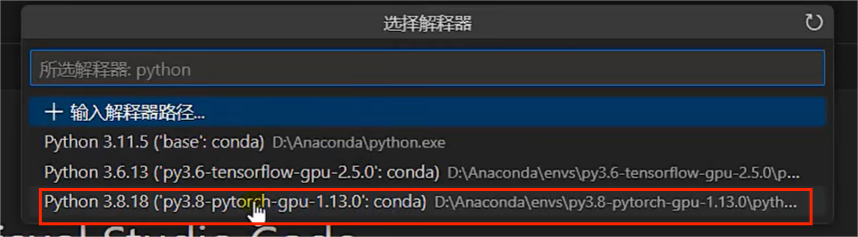
/\*

首先需要选择python解释器：

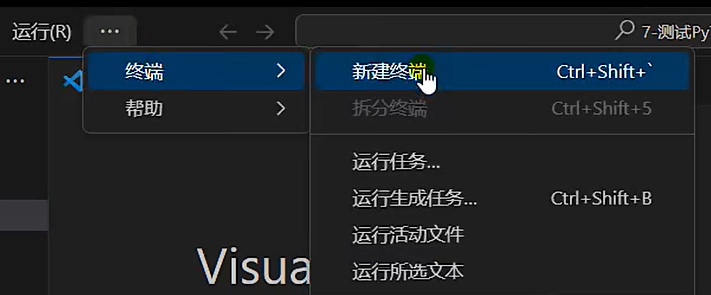
按Ctrl + shift + P然后选择Python解释器：



再选择刚刚搭建的虚拟环境：



再打开终端：



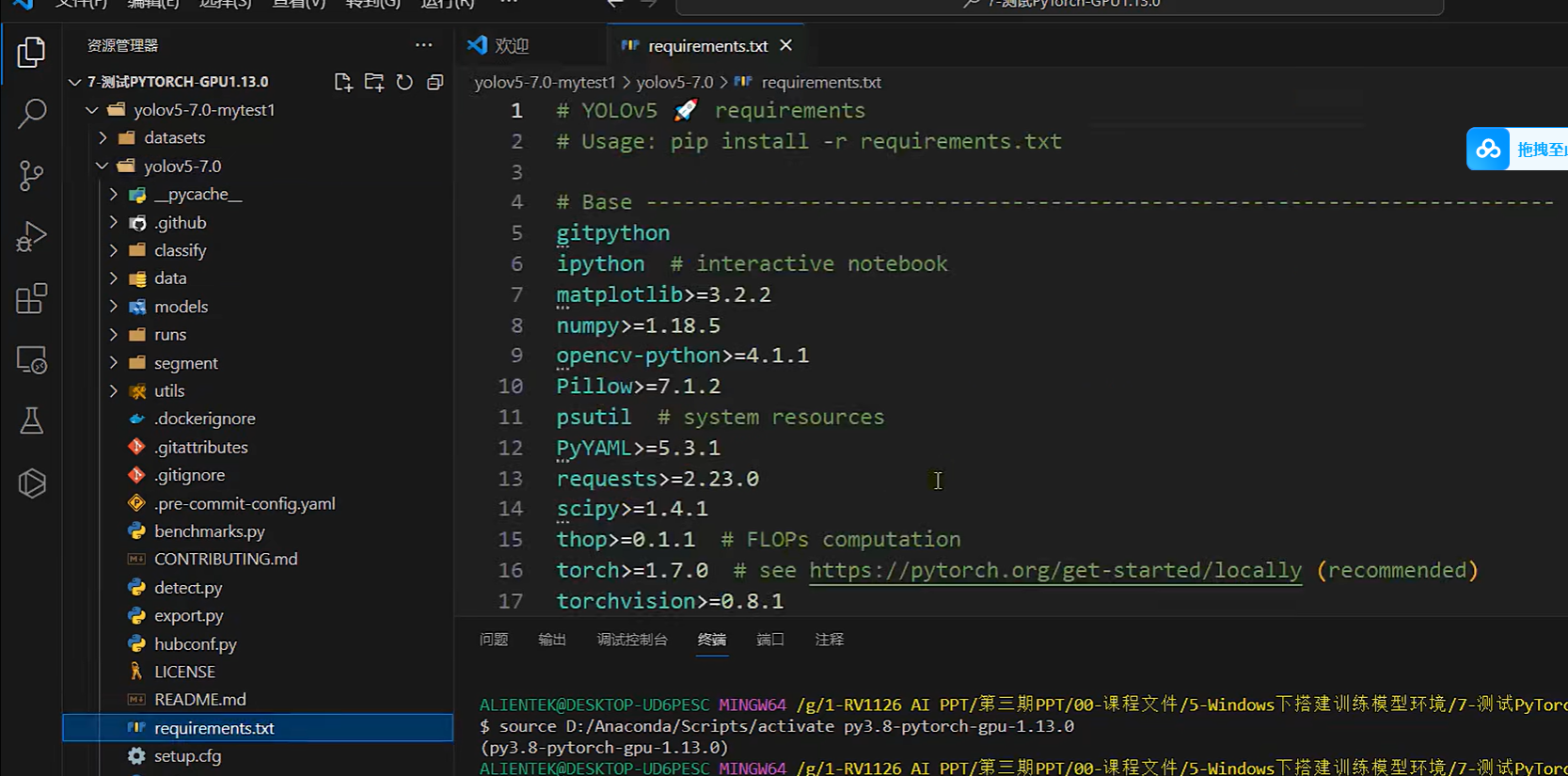
选择使用Git bash:



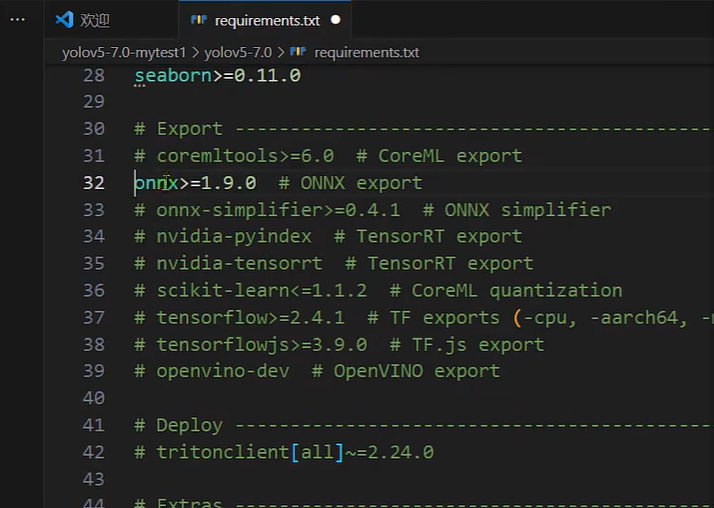
\*/

/\*

然后在训练之前需要先安装YOLOV5如下提示的依赖包：



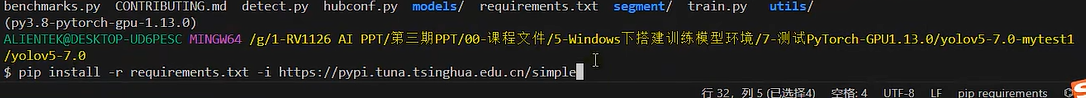
这些包可以根据这个文件进行安装，如果在后续要把训练出来的pb文件转化成onnx文件的话，就需要安装onnx:



那么我们就直接通过这个文件来安装:

进入对应的目录后执行：

pip install -r requirements.txt -i <https://pypi.tuna.tsinqhua.edu.cn/simple>



这样，他就会根据这个文本来安装依赖库。

\*/

/\*

安装好后就可以训练YOLOV5的模型:



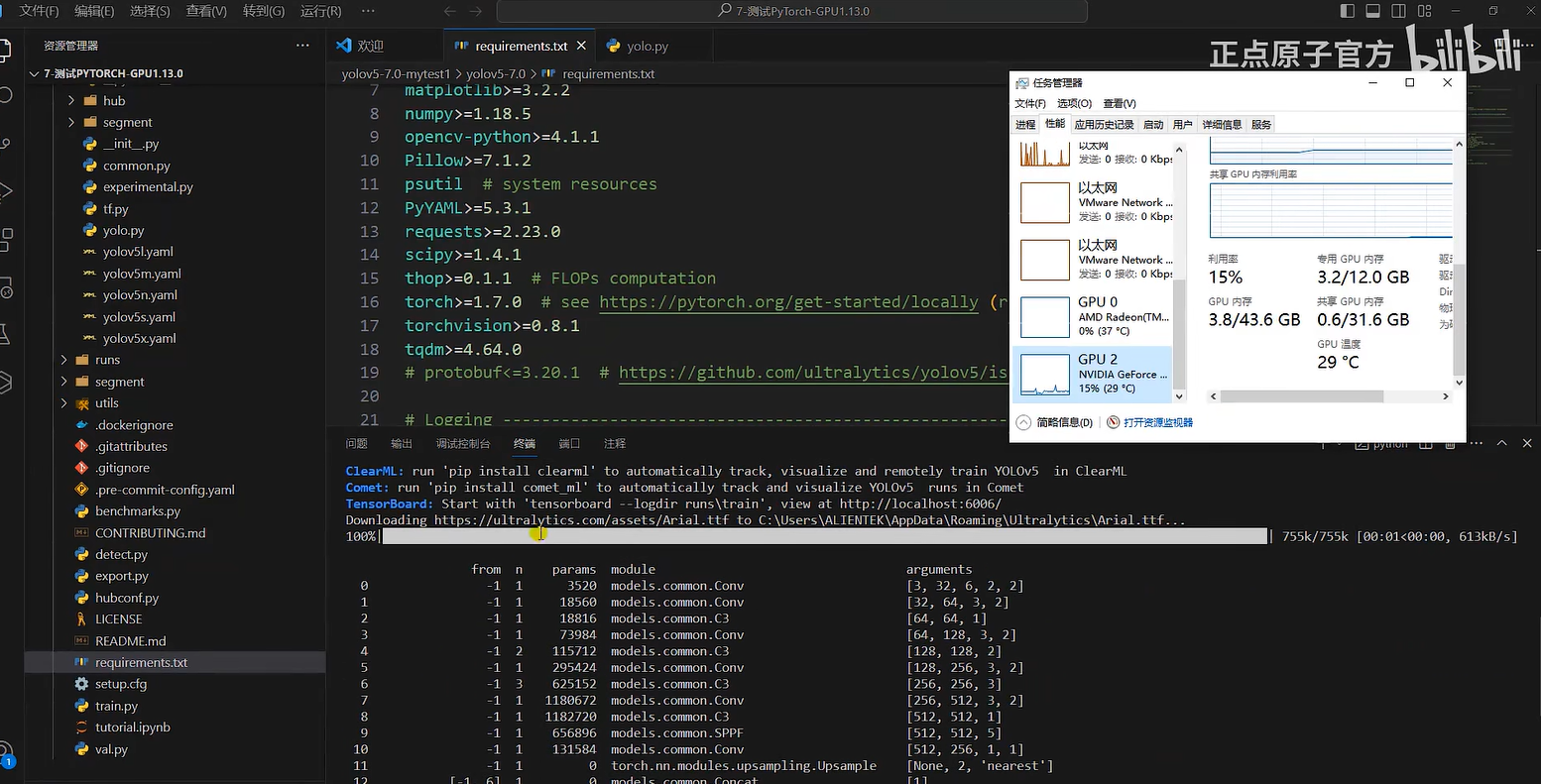
python train.py --data coco128.yaml --weights’’--cfgyolov5s.yaml --img640 --epoch 1

注意：

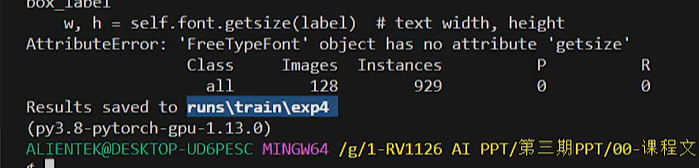
如果是在cmd/Git bash 终端运行这个训练的命令的话，直接使用上面的即可。后面的epoch 1表示，训练一个Epoch。

但如果打开的是powershell则使用命令：

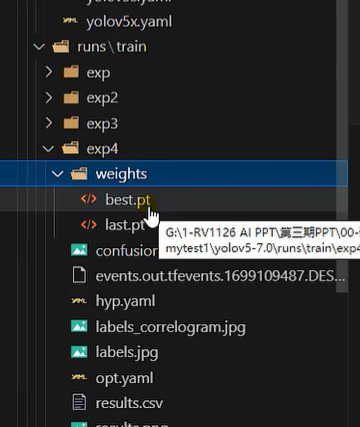
python train.py --data coco128.yaml --weights[string]::Empty --cfgyolov5s.yaml --img640 --epoch 1



训练时会在线下载一些东西。训练的同时GPU显存的占用率逐渐上升。



训练的权重保存到了exp4里：



best.pt就是训练得到的pt文件，后续我们将会把pytorch训练得到的pt文件转化成rknn文件。然后部署到开发板上。

\*/

\*/

\*/

下一节我们讲述安装TensorFlow-GPU版本的另外一种方式。（方式二）