# 环境安装

## Anaconda

### 下载

<https://www.anaconda.com/download>

### 命令行

|  |
| --- |
| 创建环境：conda create -n xxx python=3.10  查看：pip list/conda list  进入环境：conda activate xx  安装torch：在torch的网址生成命令安装  安装jupyter：conda install nb\_conda  启动jupyter：jupyter notebook  查看显卡版本：nvidia-smi |

### 其他

|  |
| --- |
| 查看显卡是否支持CUDA: <https://developer.nvidia.com/zh-cn/cuda-gpus>  查看函数的所有方法：dir(torch)  默认环境位置：C:\Users\name\.conda\envs |

## Pytorch

### 网址

<https://pytorch.org/>

### 接口

|  |
| --- |
| 查看是否支持GPU加速：torch.cuda.is\_available() |

# 数据加载

## Dataset

### 重写特殊方法

1. \_\_getitem\_\_

|  |
| --- |
| 允许用户直接用索引访问对象中的数据。  **def** \_\_getitem\_\_(self, item):  item\_path = os.path.join(self.path, self.label\_ls[item])  img = Image.open(item\_path)  **return** img  当对象为obj时，obj[1]则直接调用\_\_getitem\_\_，并把1当入参到getitem，然后返回方法中的返回值。 |

1. \_\_len\_\_

|  |
| --- |
| 允许用户使用python的len函数求对象的长度时。  **def** \_\_len\_\_(self):  **return** len(self.label\_ls)  当求len(obj)时，会直接调用\_\_len\_\_方法，并返回。 |

1. \_\_add\_\_

|  |
| --- |
| 允许用户通过+运算符操作对象。  **def** \_\_add\_\_(self, other: **'Dataset[T\_co]'**) -> **'ConcatDataset[T\_co]'**:  **return** ConcatDataset([self, other])  当用户使用+运算符操作两个对象时，会把后一个Dataset对象当作入参传给\_\_add\_\_方法，默认返回ConcatDataset后的对象。  扩展：  ConcatDataset方法继承Dataset类，调用时入参对象列表，返回\_\_getitem\_\_方法的返回值，其他方法都和Dataset的方法类似，只不过把每个Dataset对象的这些方法返回的结果进行了相加。  （简版，详细看代码接口）  **class** ConcatDataset(Dataset[T\_co]):  @staticmethod  **def** cumsum(sequence):  **return** “所有Dataset对象的长度和”   **def** \_\_init\_\_(self, datasets: Iterable[Dataset]) -> **None**:  ...   **def** \_\_len\_\_(self):  **return** “所有Dataset对象的长度和”   **def** \_\_getitem\_\_(self, idx):  **return** “二分法判断idx属于哪个对象，再取对象的索引” |

### 演示

|  |
| --- |
| 如下demo：  *#!/usr/bin/env python # -\*- coding: utf-8 -\*- # @Time : 2023/6/3 19:00 # @Author : 刘双喜 # @File : 32.dataset.py # @Description : Dataset类演示* **from** torch.utils.data **import** Dataset, ConcatDataset **import** os **from** PIL **import** Image   **class** GetImgData(Dataset):  **def** \_\_init\_\_(self, src\_dir, label\_dir):  self.src\_dir, self.label\_dir = src\_dir, label\_dir  self.path = os.path.join(src\_dir, label\_dir)  self.label\_ls = os.listdir(self.path)   **def** \_\_getitem\_\_(self, item):  item\_path = os.path.join(self.path, self.label\_ls[item])  # 打开对应列表索引的名称的图片  img = Image.open(item\_path)  **return** img   **def** \_\_len\_\_(self):  **return** len(self.label\_ls)   **def** \_\_add\_\_(self, other):  **return** ConcatDataset([self, other])   **if** \_\_name\_\_ == **'\_\_main\_\_'**:  src\_dir =**'..\\..\\dataset\\32.Dataset\\hymenoptera\_data\\train'** ants = GetImgData(src\_dir, **'ants'**)  bees = GetImgData(src\_dir, **'bees'**)  all = ants + bees  all[1].show() |