● 无障碍



# pytorch重写Dataset, 加载到Dataloader



3 人赞同了该文章

深度学习训练几个步骤,第一个就是要加载数据,pytorch自带的工具流程一般是这样:

```
# 找到数据集
train_data = torchvision.datasets.CIFAR10(root="your root",download=True)
# 把数据加载进来
train_loader = torch.utils.data.DataLoader(data=train_data)
```

我们做研究时如果不想用pytorch自带的数据集,想加载自己的数据集怎么办?那么就要通过重写一个继承了Datasets的MyDataset类(随便叫什么名字,这是自己的类)来放置自己的数据集。简单来说由于Dataloader只认识Dataset形式的数据集,所以我们要用自己的数据就也要把我们的数据变成那样。

MyDataset主要而且必须重写3个类就可以使用了,即:

```
from torch.utils.data import Dataset
class MyDataset(Dataset): #继承Datasets
    def __init(self): # 初始化一些用到的参数,一般不仅有self
        pass
    def __len__(self): # 数据集的长度
        pass
    def __getitem__(self, idx): # 按照索引读取每个元素的具体内容
        pass
```

#### def init (self):

\_\_init\_\_(self)找到我们的数据集,这里按行读取csv文件并存放在list中,读取txt也是一样的,数据的划分可以用split,最后存成self.data留着用。

```
def __init__(self, save_path):
    self.path = save_path # 数据所在路径
    data = [] # 空列表留着存放数据
    with open(save_path + 'data.csv', 'r') as f: # 按行读取csv
        reader = csv.reader(f)
        for row in reader:
            data, label = row
            data.append((data, label))
    self.data = data
```

## def len (self):

\_\_len\_\_(self)返回数据集的长度,即有多少条数据,加载数据会用到。

```
def __len__(self):
    return len(self.data)
```

## def getitem ():

\_\_getitem\_\_(self, idx)是真的要取数据了,图像要在这里读成tensor,文字也要embedding成tensor,也就是把非结构化的数据形式都变成数字数据,才能被dataloader调用进行之后的训练。

▲ 赞同 3 ▼ ● 添加评论 4 分享 ● 喜欢 ★ 收藏 △ 申请转载 …

def \_\_getitem\_\_(self, idx):
 data, label = self.data[idx]
 return data, label



编辑于 2022-02-12 21:32

深度学习 (Deep Learning) PyTorch

## 文章被以下专栏收录



#### 多模态数据融合

多模态数据融合过程中一些深度学习方法的记载和感悟

Python

## 推荐阅读



Pytorch中的Dataset和 DataLoader

机器学习社... 发表于机器学习社...



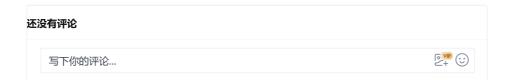
pytorch笔记5-数据读取机制 DataLoader

缓缓飘落的... 发表于R/Pyt...



完整版Pytorchi Dataset篇

丹尼尔小博...



🖴 申请转载