Vgg 模型 地址

C:\Users\DELL/.cache\torch\hub\checkpoints\vgg16-397923af.pth

Ctrl+entel 直接换行

第0层又被称为输入层，最后一层又被称为输出层，中间的若干层被称为隐藏层。

神经网络的层数是从隐藏层开始计数的，输入层不计入总层数。

损失函数LossFunction、误差函数ErrorFunction、代价函数CostFunction，及其缩写L、E、C表达的是同一个函数。

损失函数里的系数1/2，仅仅是为了令函数求导后的系数为1，为了方便计算而设定的，没有实际意义。

. 表示当前目录  
.. 表示当前目录的上一级目录。  
./表示当前目录下的某个文件或文件夹，视后面跟着的名字而定  
../表示当前目录上一级目录的文件或文件夹，视后面跟着的名字而定。  
正则化可以加快神经网络的训练速度  
例如：  
文件夹 a  
下面有  文件夹b c 和文件 d。  
文件夹b下面有e.php 和文件f。  
  
则e中的 . 表示 文件夹b  
./f 表示b下面的文件f。  
..  表示a文件夹。  
../d 表示a文件夹下的d文件。

前向传播，就是将样本数据输入到神经网络模型，经由神经网络计算，最终输出预测值的过程。

Pip install scipy 下载一个库

1 GCN

图卷积神经网络

2 bp神经网络，按照误差逆向传播算法训练的多层前馈神经网络。

由于神经网络结构复杂、参数多，导致传统的偏导数和梯度下降法无法直接被使用。直至反向传播算法应运而生，它大大提高了求解神经网络模型的最优化参数的效率

TensorFlow™是一个采用数据流图（data flow graphs），用于数值

计算的开源软件库，是python的一个库。可以训练出一个模型，

实现垃圾识别分类的流程是：用输入设备输入图片，使用TensorFlow中卷积神

经网络训练好的模型对图片进行识别，输出分类

全连接层就是一个分类器，

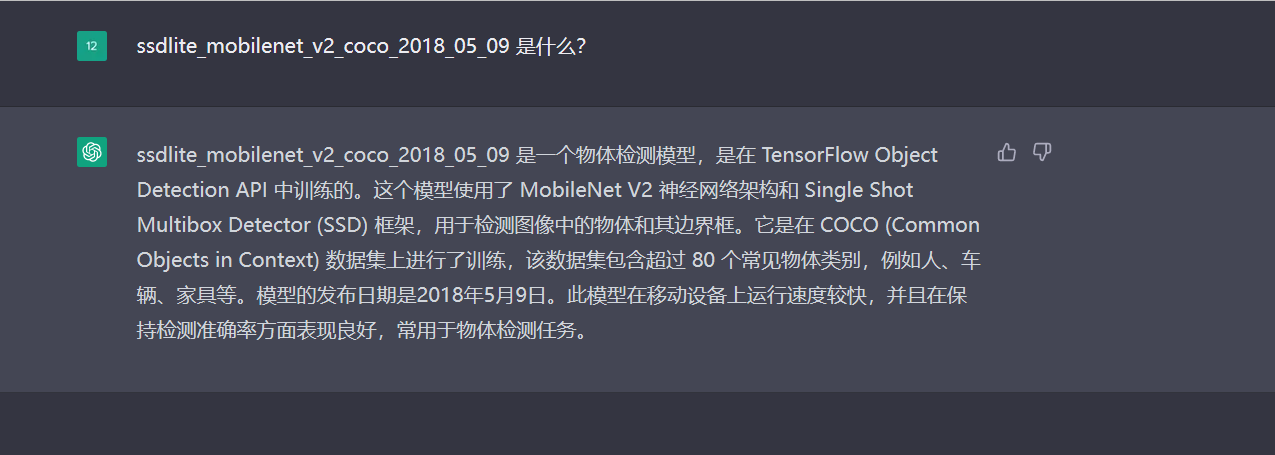
卷积神经网络比传统的神经网络有共享卷积核的优点，对高维数据处理起来没有压力。

监督学习是我们提前知道训练的输入和输出，我们的模型训练目标自然是希望模型在接收输入后，可以得到和我们提前知道的一样的输出。

sigmoid函数也叫s型函数，用作神经网络的激活函数，将变量映射到0,1之间。

Win+V 历史剪切板

出现红线 鼠标放在红线上 ALT 可以出现小灯泡 debug



SSDLite论文  
SSDLite 是 Google 在 CVPR2018 论文 MobileNetV2: Inverted Residuals and Linear Bottlenecks 中提出的轻量级检测模型

论文描述了一种新的移动架构MobileNet V2，该架构在多个任务和基准以及不同模型大小的spectrum上提高了移动模型的最先进性能。论文还描述了在称为SSDLite的新框架中将这些移动模型应用于对象检测的有效方法。此外，论文还演示了如何通过简化形式的DeepLabv3构建移动语义分割模型。

MobileNet V2架构基于反向残差结构，其中残差块的输入和输出是thin bottleneck，与传统残差模型相反，传统残差模式在输入中使用扩展表示。MobileNetV2使用轻型深度卷积来过滤中间扩展层中的特征。此外，论文发现，为了保持代表性，去除狭窄层中的非线性非常重要。论文证明，这提高了性能，并提供了导致这种设计的直觉。论文的方法允许将输入/输出域与转换的表达性解耦，这为进一步分析提供了一个方便的框架。论文测量了论文在Imagenet分类、COCO对象检测和VOC图像分割方面的性能。论文评估了精度、乘法加法（MAdd）测量的运算次数以及参数数量之间的权衡

其实是tensorflow在模型库中的对象检测模型，也叫做具有特定神经网络架构的与训练分类器，

TensorFlow在其模型库中提供了几种对象检测模型（具有特定神经网络架构的预训练分类器）。某些模型（例如SSD-MobileNet模型）具有允许更快检测但精度较低的体系结构，而某些模型（例如Faster-RCNN模型）给出的检测速度较慢但精度更高。我最初是从SSD-MobileNet-V1模型开始的。在Faster-RCNN-Inception-V2模型上对检测器进行了重新训练，检测效果明显更好，但速度明显降低。您可以选择训练异物检测分类器所基于的模型。如果您打算在计算能力较低的设备（例如智能手机或Raspberry Pi）上使用对象检测器，

请使用SDD-MobileNet模型。如果要在功能强大的笔记本电脑或台式机上运行检测器，请使用RCNN模型之一。

训练的每一步都会报告损失。它会从高处开始，并随着训练的进行而越来越低。在我对Faster-RCNN-Inception-V2模型的训练中，它从大约3.0开始，然后迅速降至1以下。我建议让您的模型进行训练，直到损失始终低于0.06左右，这将花费比较长的时间（取决于CPU和GPU的功能）。注意：如果使用不同的模型，则损失数将有所不同。MobileNet-SSD的损失大约为20，直到损失始终低于2左右。

您可以使用TensorBoard查看培训工作的进度。

TensorBoard页面提供了信息和图表，它们显示了培训的进度。一个重要的图是损失图，它显示了分类器随时间的总体损失。

那么最后一步就是生成冻结的推理图（.pb文件）。从

TensorFlow对象检测API使用Protobuf，这是一个实现Google协议缓冲区数据格式的软件包。您以前需要从源代码进行编译，但是现在安装很简单！

v现在，我们将从[TensorFlow检测模型库](https://links.jianshu.com/go?to=https%3A%2F%2Fgithub.com%2Ftensorflow%2Fmodels%2Fblob%2Fmaster%2Fresearch%2Fobject_detection%2Fg3doc%2Fdetection_model_zoo.md)中下载SSD\_Lite模型。模型动物园是谷歌的预训练对象检测模型的集合，这些模型具有不同水平的速度和准确性。树莓派的处理器较弱，因此我们需要使用处理能力较小的模型。尽管该模型将运行得更快，但要以降低精度为代价。在本教程中，我们将使用SSDLite-MobileNet，这是最快的模型。

from torch.utils.tensorboard import SummaryWriter

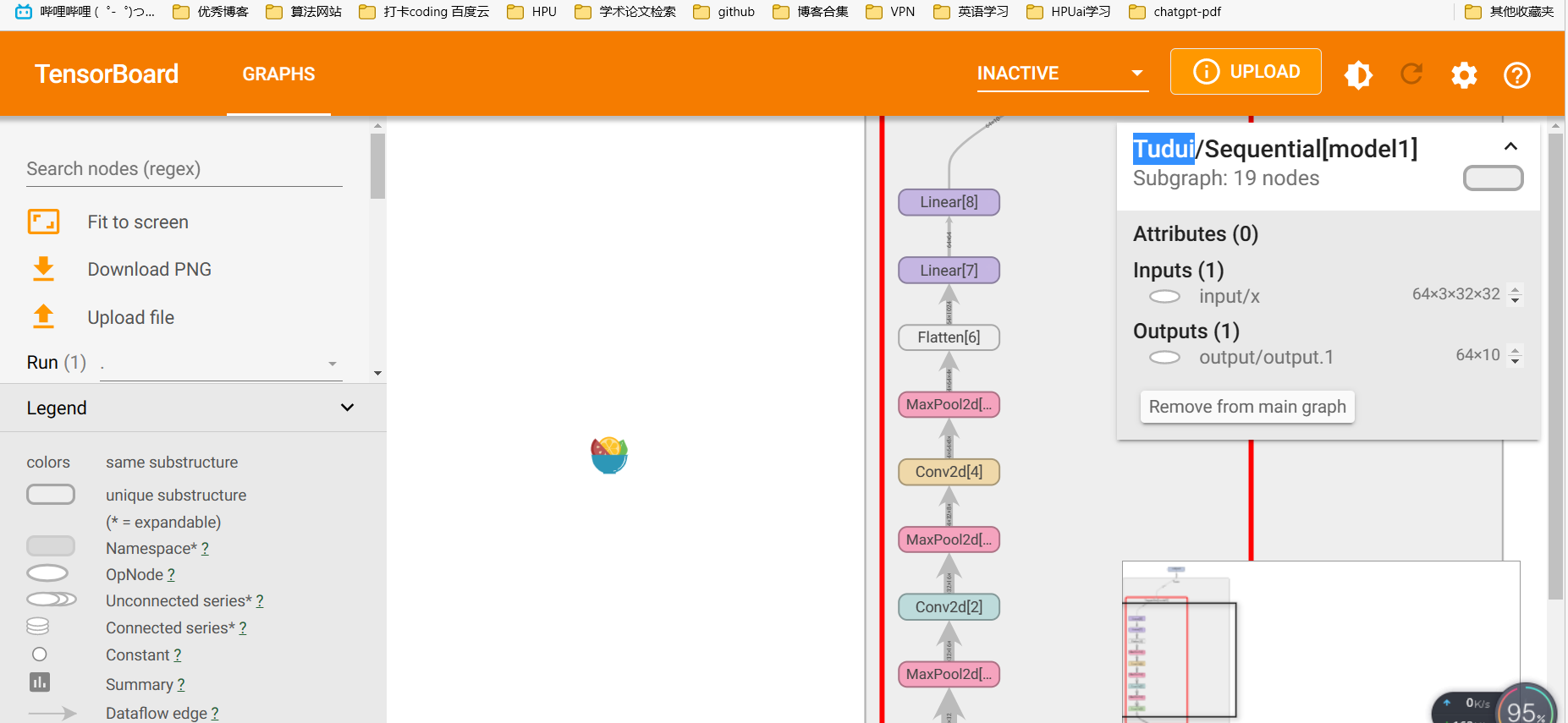
writer = SummaryWriter("./logs\_seq") # 新建文件夹名称 存放tensorboare日志

writer.add\_graph(tudui, input)

表示出来：

writer.add\_scalar("train\_loss", loss.item(), total\_train\_step)

writer.close()



效果图

有了损失才能求导 优化梯度

终端：tensorboard --logdir="logs\_seq" # 和之前的日志文件夹一一对应

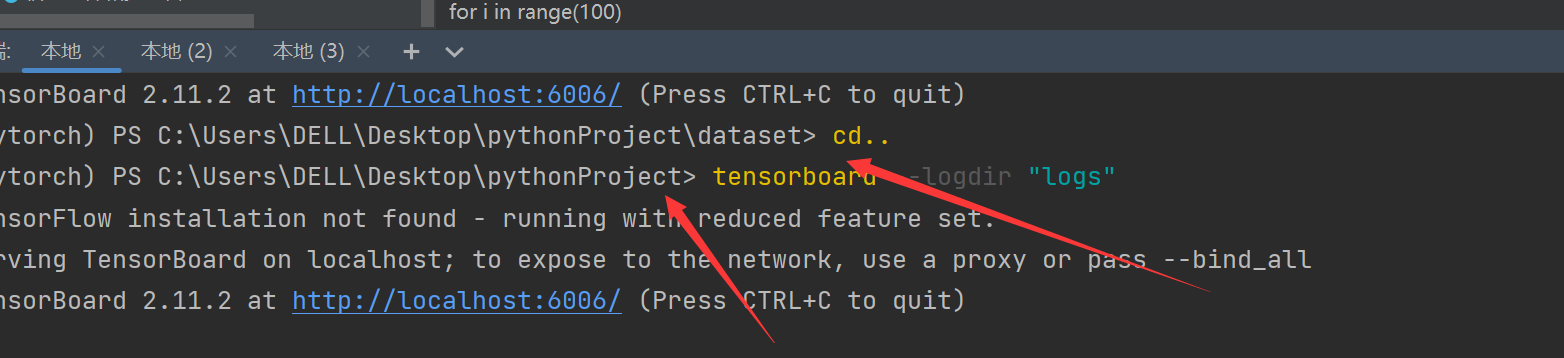
小土堆：

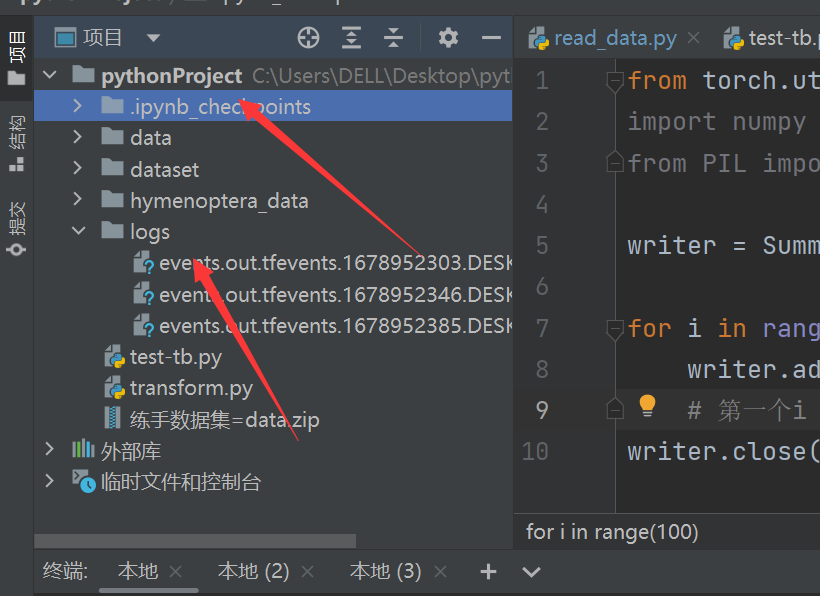
tensorboard --logdir=logs

tensorboard --logdir=logs --port=6007

tensorboard --logdir="p10" 指定日志所在的文件夹 这里是 p10



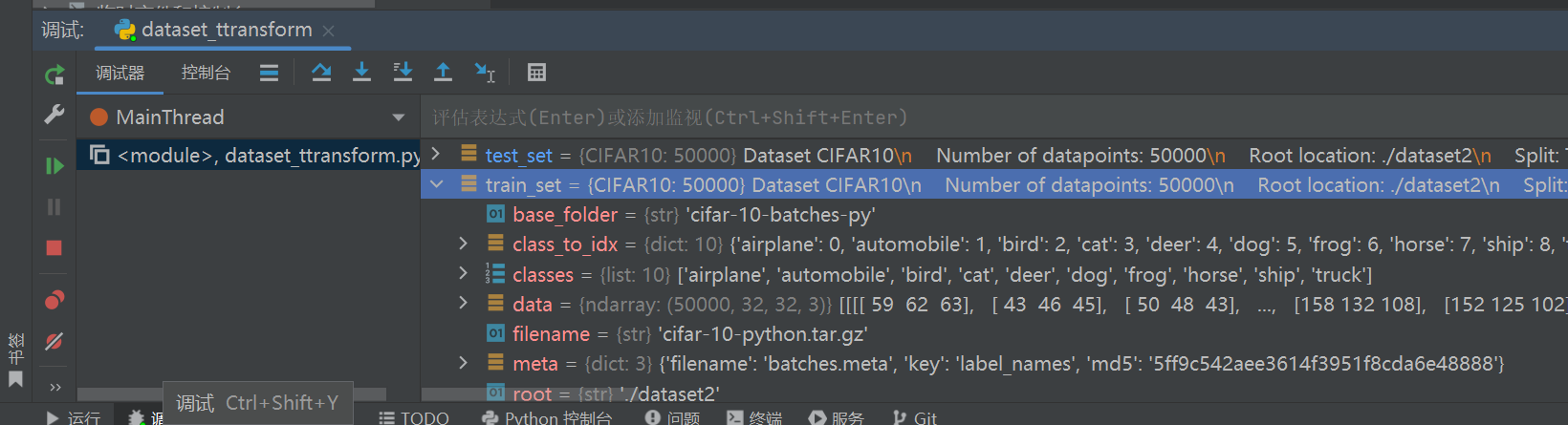




这里的目录要在logs所在目录下才可以

==============================================s

如何调试debug 右键设置断点 右键 点击虫子 出现class类 也叫做调试器



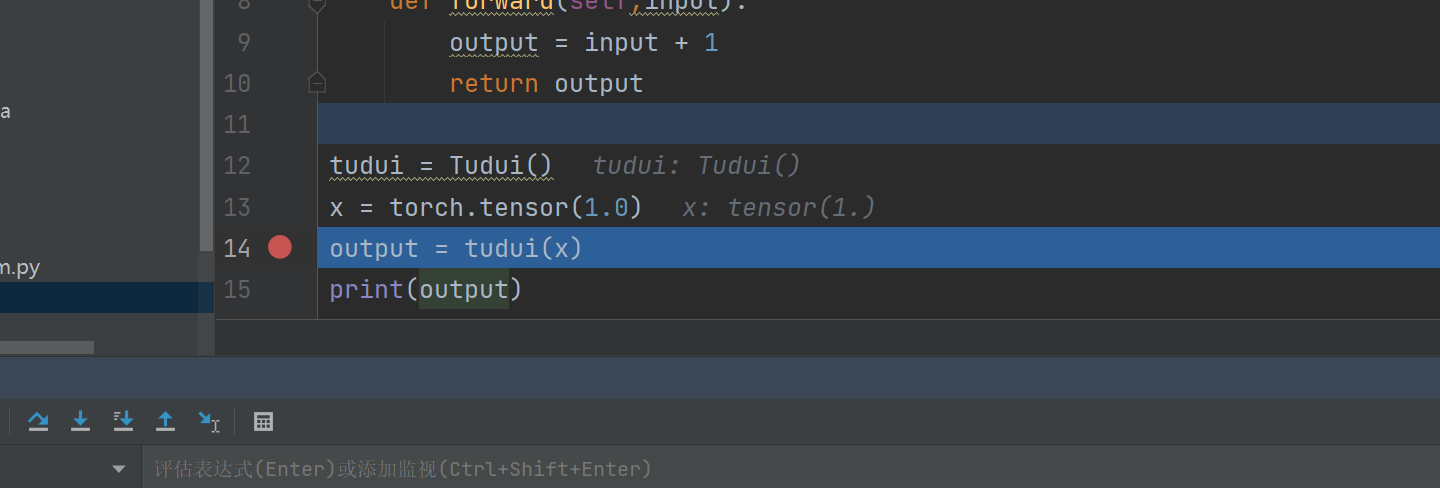
root="./dataset2"

根目录下的数据集保存地址

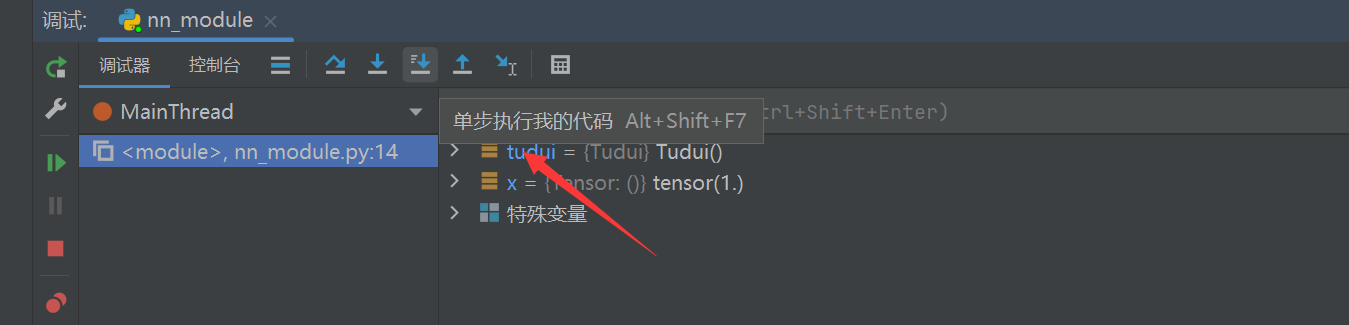
. / 根目录 . 表示当前目录

../ 向上两个根目录

设置断点



设置断点的意思是整个程序调试运行到这个红点就停止，

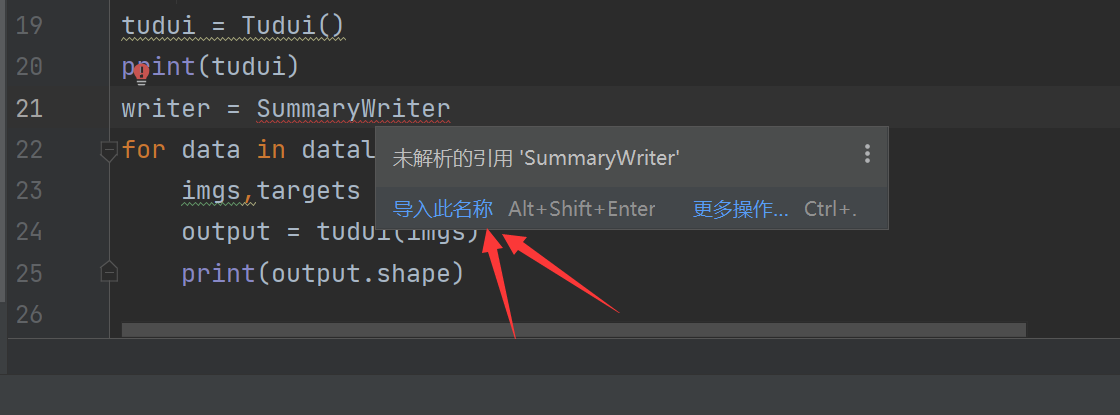


可以单步执行代码 然后跳转到即将执行的代码

from torch import nn  
import torch  
这里对class Tudui(nn.Module):

内部函数调用可以直接点代码-生成-重写方法直接module即可  
class Tudui(nn.Module):  
 def \_\_init\_\_(self) :  
 super().\_\_init\_\_()  
  
 def forward(self,input):  
 output = input + 1  
 return output  
  
tudui = Tudui()  
x = torch.tensor(1.0)  
output = tudui(x)  
print(output)

导入引用



李沐动手学深度学习