

1. Anaconda 下载与安装

1.1. Anaconda 介绍

Anaconda 是一个 Python 环境管理工具，就是可以便捷获取包且对包能够进行管理，同时对环境可以统一管理的发行版本。Anaconda 包含了 conda、Python 在内的超过 180 个科学包及其依赖项。

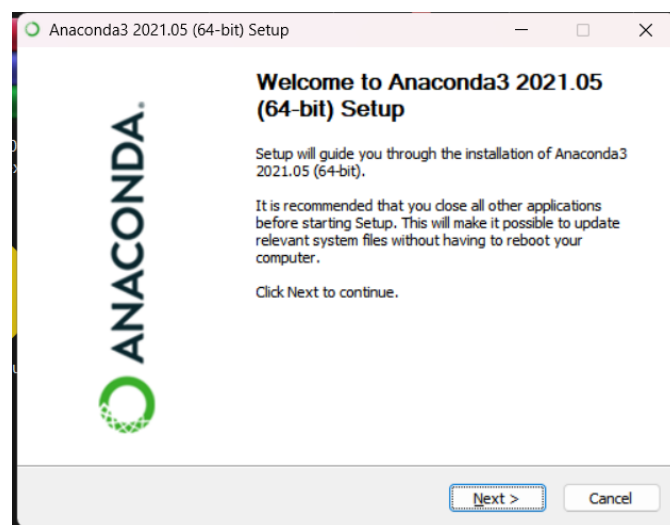
为什么要使用 Anaconda?如果我在本地只有一个 Python 环境。那我所有程序用到的各种包都只能放到同一个环境中，容易导致环境混乱,另外当我将写好的程序放到另一电脑上运行时又会遇到缺少相关包,需要自己手动一个个下载的情况,要是能每个程序开发都选用不同的环境,而开发好之后又能将该程序需要的环境(第三方包)都独立打包出来就好了。

1.2. Anaconda 下载与安装

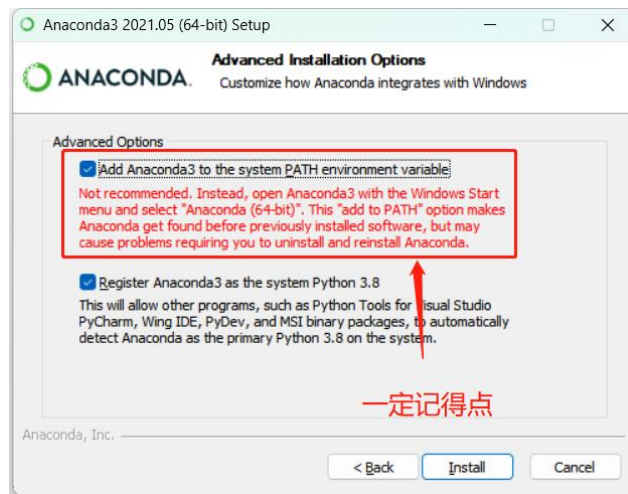
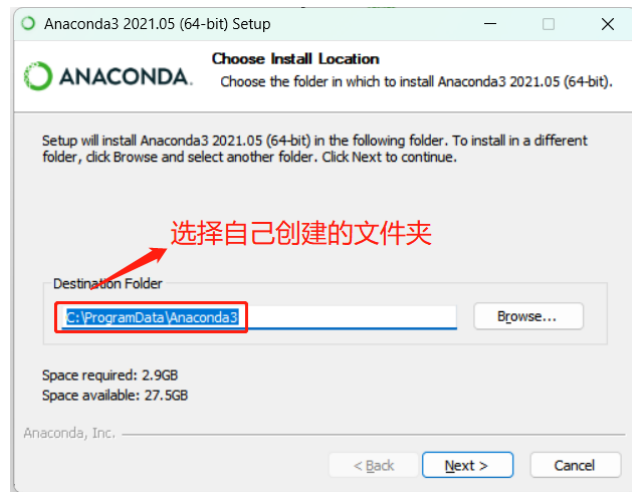
官网下载: <https://www.anaconda.com/> （不推荐）

清华镜像源下载: <https://mirrors.tuna.tsinghua.edu.cn/anaconda/archive/> （推荐，国内地址，下载速度快）

这里推荐下载 2021.05 版本，下载下来 anaconda 的 exe 安装包后，在需要安装的目录下新建文件夹 Anaconda3，便于后期 Anaconda 的维护管理（找个空间大的盘）；双击 Anaconda3-2021.05-Windows-x86_64.exe 进行安装。



点击 Next-->I agree -->All Users，然后安装目录选择自己新建的 Anaconda3 文件夹。一定记得勾选 Add Anaconda3 to the system PATH environment variable，将 Anaconda 添加环境变量，最后点击 Install 安装即可。



安装完成后，按 Win+R，在运行窗口中输入 cmd,进入 cmd 命令窗口，然后输入 conda,查看结果，出现以下结果证明安装成功。

```
C:\Users\牛文涛>conda
usage: conda-script.py [-h] [-V] command ...

conda is a tool for managing and deploying applications, environments and packages.

Options:
  -h, --help            show this help message and exit
  -V, --version          show program's version number and exit

positional arguments:
  command
    clean                Remove unused packages and caches.
    compare              Compare packages between conda environments.
    config               Modify configuration values in .condarc. This is modeled after the git config command. Writes to the
                        user .condarc file (C:\Users\牛文涛\.condarc) by default.
    create               Create a new conda environment from a list of specified packages.
    info                 Display information about current conda install.
    init                 Initialize conda for shell interaction. [Experimental]
    install              Install a list of packages into a specified conda environment.
    list                 List linked packages in a conda environment.
    package              Low-level conda package utility. (EXPERIMENTAL)
    remove               Remove a list of packages from a specified conda environment.
    uninstall            Alias for conda remove.
    run                  Run an executable in a conda environment.
    search               Search for packages and display associated information. The input is a MatchSpec, a query language
                        for conda packages. See examples below.
    update               Updates conda packages to the latest compatible version.
    upgrade              Alias for conda update.
```

2. CUDA 与 CUDNN 安装

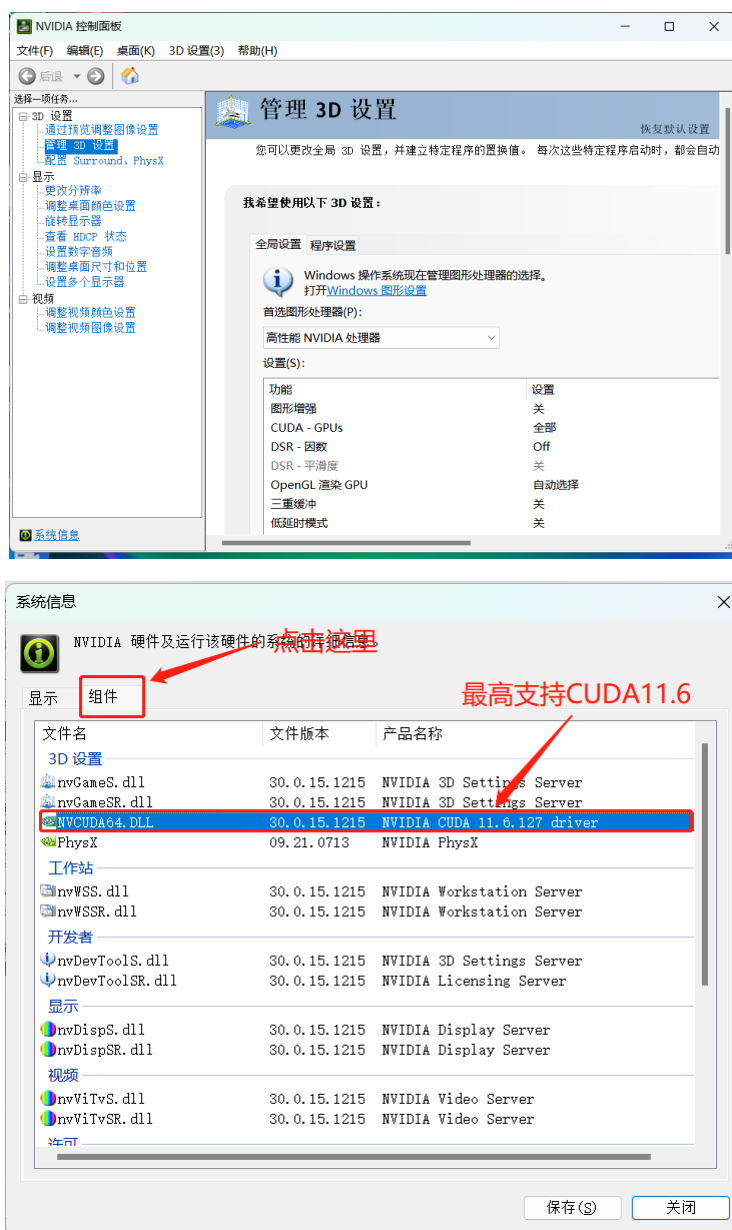
注意：如果你的电脑没有英伟达的显卡，这一部分可以跳过去了。

2.1. CUDA 介绍

CUDA 是 Nvidia 开发的一种并行计算平台和编程模型,用于在其自己的 GPU (图形处理单元)上进行常规计算。CUDA 使开发人员能够利用 GPU 的能力来实现计算的可并行化部分,从而加快计算密集型应用程序的速度。

2.2. CUDA 下载及安装

安装 CUDA 之前,需要先确定下自己电脑支持的 CUDA 最高版本,操作如下:桌面右击——打开 NVIDIA 控制面板——帮助——系统信息——组件,这里显示该版本的显卡驱动下,最高支持的 CUDA 版本为 11.6。

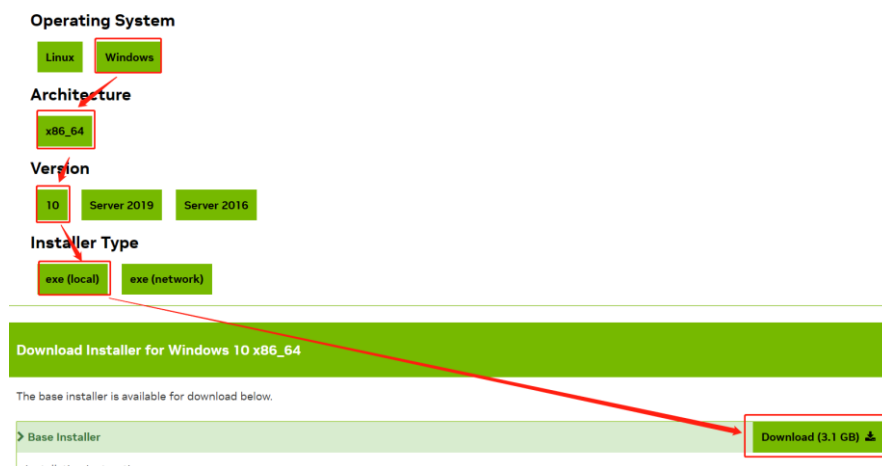


然后下载 CUDA 安装包

CUDA 下载地址: <https://developer.nvidia.com/cuda-toolkit-archive> (网址打开很慢...), 选择 CUDA 版本 (不要安装最新的或者过高的 CUDA 版本, 很多深度学习框架不支持高版本的 CUDA, 同时 30 系的显卡只支持 CUDA 11.0 以上版本的, 这里推荐 11.1 或者 11.0 版本的), 如果 CUDA 支持的最高版本较低, 可以升级显卡驱动。

[CUDA Toolkit 11.5.0](#) (October 2021), [Versioned Online Documentation](#)
[CUDA Toolkit 11.4.4](#) (February 2022), [Versioned Online Documentation](#)
[CUDA Toolkit 11.4.3](#) (November 2021), [Versioned Online Documentation](#)
[CUDA Toolkit 11.4.2](#) (September 2021), [Versioned Online Documentation](#)
[CUDA Toolkit 11.4.1](#) (August 2021), [Versioned Online Documentation](#)
[CUDA Toolkit 11.4.0](#) (June 2021), [Versioned Online Documentation](#)
[CUDA Toolkit 11.3.1](#) (May 2021), [Versioned Online Documentation](#)
[CUDA Toolkit 11.3.0](#) (April 2021), [Versioned Online Documentation](#)
[CUDA Toolkit 11.2.2](#) (March 2021), [Versioned Online Documentation](#)
[CUDA Toolkit 11.2.1](#) (February 2021), [Versioned Online Documentation](#)
[CUDA Toolkit 11.2.0](#) (December 2020), [Versioned Online Documentation](#)
[CUDA Toolkit 11.1.1](#) (October 2020), [Versioned Online Documentation](#)
[CUDA Toolkit 11.1.0](#) (September 2020), [Versioned Online Documentation](#)
[CUDA Toolkit 11.0.3](#) (August 2020), [Versioned Online Documentation](#)
[CUDA Toolkit 11.0.2](#) (July 2020), [Versioned Online Documentation](#)
[CUDA Toolkit 11.0.1](#) (June 2020), [Versioned Online Documentation](#)
[CUDA Toolkit 10.2](#) (Nov 2019), [Versioned Online Documentation](#)
[CUDA Toolkit 10.1 update2](#) (Aug 2019), [Versioned Online Documentation](#)

分别选取 Windows——x86_64——10——exe(local)——Download



下载完成后，双击 cuda_11.1.0_456.43_win10.exe 开始进行 CUDA 的安装：
OK——同意——自定义——下一步——选择安装位置（这里可以选择自己新建的文件夹）

注意：选择安装位置时，尽量不要使用默认的安装位置（默认安装为 C 盘，CUDA 安装后比较大，建议换个磁盘新建个文件夹单独安装 CUDA），按照下面的 CUDA 的文件结构新建文件夹

CUDA

|- NVIDIA Corporation

|----- CUDA Samples

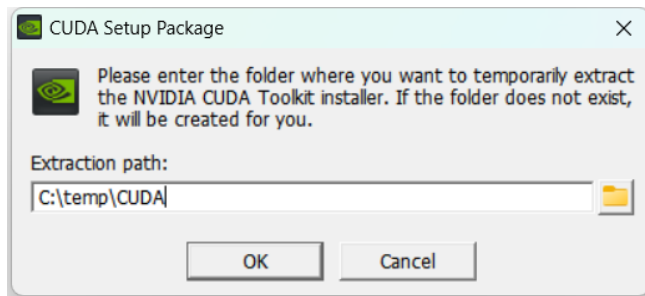
|----- v11.1

|

|- NVIDIA GPU Computing Toolkit

|----- CUDA

|----- v11.1





到这里 CUDA 安装完成，下面进行 CUDNN 的安装。

2.3. CUDNN 下载及安装

CUDNN 下载地址: <https://link.csdn.net/?target=https%3A%2F%2Fdeveloper.nvidia.com%2Frdp%2Fcudnn-download> (下载 CUDNN 需要 NVIDIA 账号,需要提前注册号, 此外 NVIDIA 网址打开很慢, 有点耐心.....), 这里 CUDNN 下载选择 CUDA 11.x 版本的 Windows 平台的。

cuDNN Download

NVIDIA cuDNN is a GPU-accelerated library of primitives for deep neural networks.

☒ I Agree To the Terms of the [cuDNN Software License Agreement](#)

Note: Please refer to the [Installation Guide](#) for release prerequisites, including supported GPU architectures and compute capabilities, before downloading.

For more information, refer to the cuDNN Developer Guide, Installation Guide and Release Notes on the [Deep Learning SDK Documentation](#) web page.

Download cuDNN v8.9.3 (July 11th, 2023), for CUDA 12.x

Download cuDNN v8.9.3 (July 11th, 2023), for CUDA 11.x

Local Installers for Windows and Linux, Ubuntu(x86_64, armsbsa)

[Local Installer for Windows \(Zip\)](#)

[Local Installer for Linux x86_64 \(Tar\)](#)

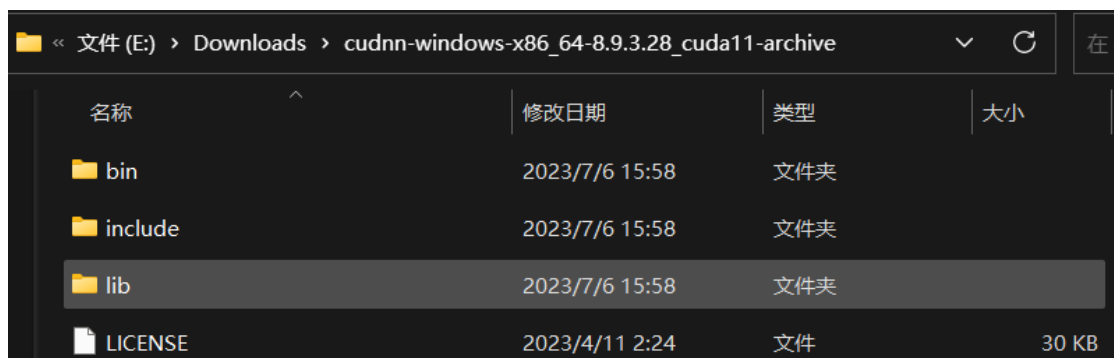
[Local Installer for Linux PPC \(Tar\)](#)

[Local Installer for Linux SBSA \(Tar\)](#)

[Local Installer for Debian 11 \(Deb\)](#)

[Local Installer for Ubuntu 18.04 x86_64 \(Deb\)](#)

将下载下来的 CUDNN 压缩包解压，得到下面的文件。



« 软件 (D:) > CUDA > NVIDIA GPU Computing Toolkit > CUDA > v11.1.1 >

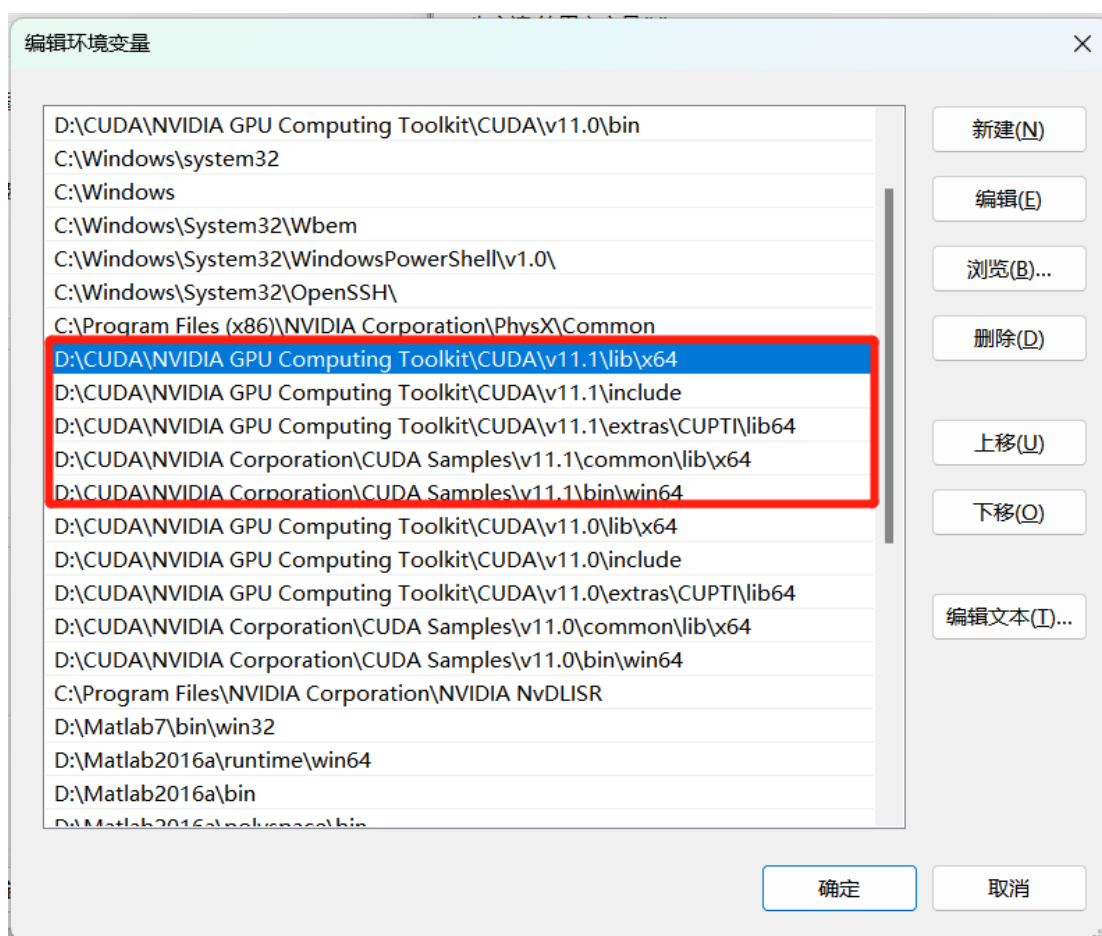
名称	修改日期	类型	大小
bin	2023/3/28 9:42	文件夹	
compute-sanitizer	2022/2/7 15:03	文件夹	
extras	2022/2/7 15:03	文件夹	
include	2023/3/28 9:42	文件夹	
lib	2022/2/7 15:03	文件夹	
libnvvp	2022/2/7 15:03	文件夹	
nvml	2022/2/7 15:03	文件夹	
nvvm	2022/2/7 15:03	文件夹	
src	2022/2/7 15:03	文件夹	
tools	2022/2/7 15:03	文件夹	
CUDA_Toolkit_Release_Notes.txt	2020/9/16 13:05	文本文档	16 KB
DOCS	2020/9/16 13:05	文件	1 KB

D:\CUDA\NVIDIA GPU Computing Toolkit\CUDA\v11.1\lib\x64
D:\CUDA\NVIDIA GPU Computing Toolkit\CUDA\v11.1\include

D:\CUDA\NVIDIA GPU Computing Toolkit\CUDA\v11.1\extras\CUPTI\lib64

D:\CUDA\NVIDIA Corporation\CUDA Samples\v11.1\common\lib\x64

D:\CUDA\NVIDIA Corporation\CUDA Samples\v11.1\bin\win64



到此，CUDNN 的安装结束，下面验证 CUDA 和 CUDNN 是否正确安装，打开 cmd 命令界面，输入 **nvcc -V** 命令，若出现以下回复，证明安装成功。

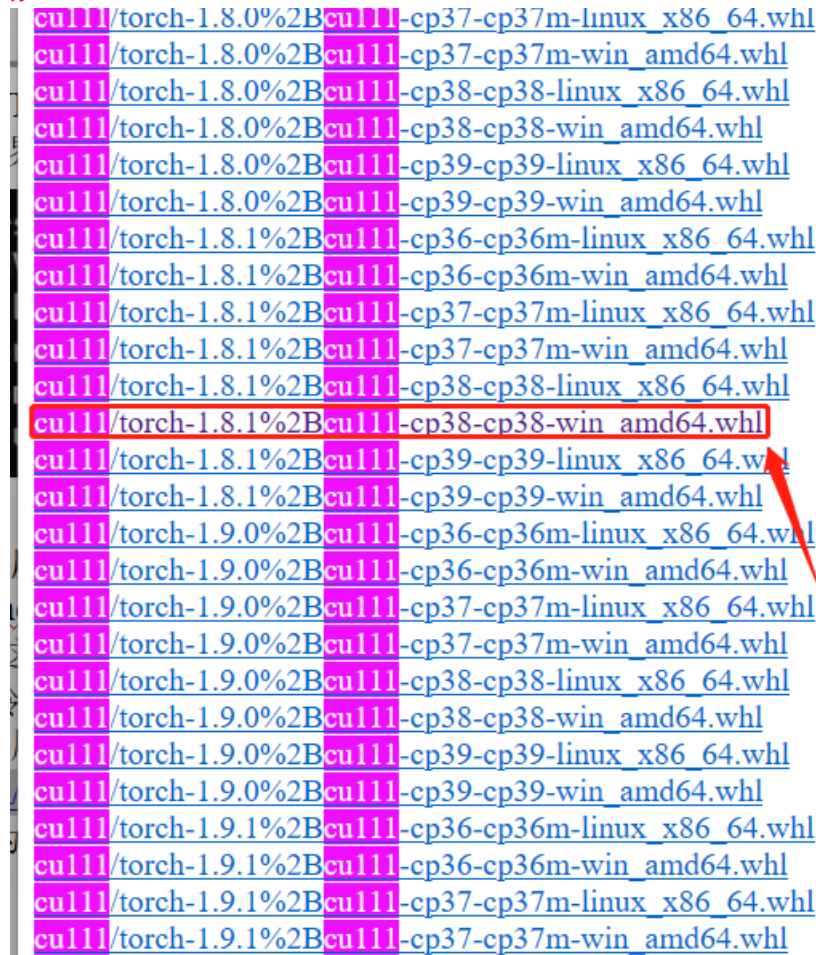
```
C:\Users\...>nvcc -V
nvcc: NVIDIA (R) Cuda compiler driver
Copyright (c) 2005-2020 NVIDIA Corporation
Built on Tue_Sep_15_19:12:04 Pacific Daylight Time_2020
Cuda compilation tools, release 11.1, V11.1.74
Build cuda_11.1.relgpu_drvr455TC455_06.29069683_0
```

3. Pytorch 深度学习框架的安装

首先在 cmd 界面中使用 conda 命令: **conda create -n torch python=3.8** 新建一个环境，这里环境命名为 torch，python 版本选择 3.8；然后使用 **conda activate torch** 命令激活环境。

Pytorch 使用离线安装的方式，Pytorch 离线安装包网址为: https://download.pytorch.org/whl/torch_stable.html，在网页中按 Ctrl+F 搜索 **cu111**(cu 即为 CUDA 的简写，111 为 11.1 版本，这里以自己实际安装的版本为准),搜索结果见下图。

选择合适的 torch 安装包，**cp** 为 **python** 缩写，**cp38** 为 **python3.8** 版本，**win** 代表 **Windows** 平台，这里以自己的实际情况为准，python 版本要与环境中的 python 版本相同！！！选好之后下载 torch 安装包，然后选择相应的 torchvision 版本，注意：**torch** 和 **torchvision** 版本有对应关系，不能随便下载，具体对应关系例如：**torch 1.8.0——torchvision 0.9.0 torch 1.6.0——torchvision 0.7.0**



cu111/torch-1.8.0%2Bcu111-cp37-cp37m-linux_x86_64.whl
cu111/torch-1.8.0%2Bcu111-cp37-cp37m-win_amd64.whl
cu111/torch-1.8.0%2Bcu111-cp38-cp38-linux_x86_64.whl
cu111/torch-1.8.0%2Bcu111-cp38-cp38-win_amd64.whl
cu111/torch-1.8.0%2Bcu111-cp39-cp39-linux_x86_64.whl
cu111/torch-1.8.0%2Bcu111-cp39-cp39-win_amd64.whl
cu111/torch-1.8.1%2Bcu111-cp36-cp36m-linux_x86_64.whl
cu111/torch-1.8.1%2Bcu111-cp36-cp36m-win_amd64.whl
cu111/torch-1.8.1%2Bcu111-cp37-cp37m-linux_x86_64.whl
cu111/torch-1.8.1%2Bcu111-cp37-cp37m-win_amd64.whl
cu111/torch-1.8.1%2Bcu111-cp38-cp38-linux_x86_64.whl
cu111/torch-1.8.1%2Bcu111-cp38-cp38-win_amd64.whl
cu111/torch-1.8.1%2Bcu111-cp39-cp39-linux_x86_64.whl
cu111/torch-1.8.1%2Bcu111-cp39-cp39-win_amd64.whl
cu111/torch-1.9.0%2Bcu111-cp36-cp36m-linux_x86_64.whl
cu111/torch-1.9.0%2Bcu111-cp36-cp36m-win_amd64.whl
cu111/torch-1.9.0%2Bcu111-cp37-cp37m-linux_x86_64.whl
cu111/torch-1.9.0%2Bcu111-cp37-cp37m-win_amd64.whl
cu111/torch-1.9.0%2Bcu111-cp38-cp38-linux_x86_64.whl
cu111/torch-1.9.0%2Bcu111-cp38-cp38-win_amd64.whl
cu111/torch-1.9.0%2Bcu111-cp39-cp39-linux_x86_64.whl
cu111/torch-1.9.0%2Bcu111-cp39-cp39-win_amd64.whl
cu111/torch-1.9.1%2Bcu111-cp36-cp36m-linux_x86_64.whl
cu111/torch-1.9.1%2Bcu111-cp36-cp36m-win_amd64.whl
cu111/torch-1.9.1%2Bcu111-cp37-cp37m-linux_x86_64.whl
cu111/torch-1.9.1%2Bcu111-cp37-cp37m-win_amd64.whl

torch 和 torchvision 都下载好之后，在 cmd 界面中激活相应的环境，然后通过 cd 命令将文件路径跳转到下载的 torch 和 torchvision 目录下，通过以下命令进行离线安装：**pip install torch-1.8.1+cu101-cp38-cp38-win_amd64.whl -i http://pypi.douban.com/simple/ --trusted-host pypi.douban.com**，这里 torch-1.8.1+cu101-cp38-cp38-win_amd64.whl 以自己文件夹下的实际文件名为准，同理将 torch 的文件名换为 torchvision 进行安装。

在 cmd 界面中使用 pip list 命令查看安装的包列表，查看 torch 是否安装成功。下图为安装成功的界面，可以看到版本号都带有 cu111，证明是 CUDA 11.1 的版本。

至此，深度学习的 Pytorch 安装结束！

同时如果需要安装其他包，项目文件中有 requirements.txt 文件的，可以使用 **pip install -r requirements.txt** 命令进行批量安装，注意：如果 requirements.txt 有 torch、torchvision 的，记得用#注释掉或者删除掉，防止重复安装。

scipy	1.4.1
seaborn	0.11.2
setuptools	59.6.0
six	1.16.0
soupsieve	2.3.2.post1
tabulate	0.8.10
tb-nightly	2.10.0a20220710
tensorboard-data-server	0.6.1
tensorboard-plugin-wit	1.8.1
terminaltables	3.1.10
thop	0.1.1.post2209072238
threadpoolctl	3.1.0
tifffile	2020.9.3
torch	1.8.1+cu111
torchsummary	1.5.1
torchvision	0.9.1+cu111
tqdm	4.59.0
typing_extensions	4.1.1
urllib3	1.26.8
warmup-scheduler	0.3
Werkzeug	2.0.3
wheel	0.29.0
win32-setctime	1.1.0
wincertstore	0.2
yacs	0.1.8
yapf	0.32.0
yolox	0.3.0
zipp	3.6.0