安装 Python 环境

Xiao-Xin Li

Zhejiang University of Technology

Revision: *2023/04/18*

---

1. 下载 Python

- Download Python

- Python Releases for Windows

目前（截止2023/04/18）最好是安装 Python 3.10 (?)，最新版的 Python 还不够稳定。

2. ~~在 Windows Subsystem for Linux（WSL）下安装 Python~~ ~~（不好用）~~

WSL 是一个在Windows 10\11 上能够运行原生 Linux 二进制可执行文件的兼容虚拟系统。

```
1  uname -r   # 查看 wsl 的版本
2
```

```
 3  ## ??
 4  sudo add-apt-repository ppa:deadsnakes/ppa
 5  sudo apt update
 6  sudo apt install software-properties-common
 7
 8  # 安装 Python
 9  sudo apt install python3.11
10  # 删除 Python
11  sudo apt-get remove --auto-remove python3.8
12  sudo apt-get remove python3.8
13
14  # 卸载老的虚拟环境
15  sudo apt remove python3-venv
16  sudo apt autoremove
17  # 安装虚拟环境
18  sudo apt install python3.11-venv
19
20  # 安装 pip
21  sudo apt install python3-pip
22  # 更新 pip
23  pip install --upgrade pip
24
25  # 创建虚拟环境
26  python3.11 -m venv pytorch2.0
27
28  # 安装 PyTorch
29  pip3 install torch torchvision torchaudio --index-url https://download.pytorch.o
```

3. Python 包管理

3.1 Python 资源

1. Python Package Index (PyPI): https://pypi.org/, Find, install and publish Python packages with the Python Package Index

3.2 Python 虚拟环境

在开发过程中，我们可能有多个项目，不同的项目依赖于不同的 Python 包，甚至所依赖的 Python 版本都是不同的，需要创建不同的 Python 环境。

1. Python 虚拟环境的创建，具体操作如下（详见：https://docs.python.org/zh-cn/3/library/venv.html）：

$$python\ -m\ venv\ <虚拟环境的路径>$$

2. Python 虚拟环境的激活（这里，虚拟环境的名字是：`python3.9`）：

- Mac下: `$ source /Users/xxli/work/PyEnvs/python3.9/bin/activate`
- Windows下: `C:\> <venv>\Scripts\activate.bat`

3. 如果不想每次都这么麻烦，可以为上面的命令创建"别名"，当再次需要激活虚拟环境时，直接输入别名即可：

```
1  # 在 Linux / Mac 下：
2  # 打开 .zshrc
3  # .bashrc, .zshrc 等是当打开 bash shell 或 zshell 时，自动执行的脚本。
4  #   这里，rc stands for 'RunCom' – Run Commands
5  $ subl .zshrc  # vim .zshrc
6
7  # 在 .zshrc 中，为上面的激活 Python 环境的命令添加 "别名"
8  alias mypython="source /Users/xxli/work/PyEnvs/python3.9/bin/activate"
```

```
1  # 在 Windows 下：
2  # ref: https://blog.csdn.net/YiRanZhiLiPoSui/article/details/83116819
3  # 1. 创建 .bat 文件
4  # 2. 使cmd启动时自动执行该 .bat 文件：
5  #    – 在 win10/win11 下，配置注册表：HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Com
6  #    – 新建字符串值，随便起个名字，比如 AutoRun;
7  #    – 双击编辑该值，数值数据里填刚才新建的 .bat 文件的路径，如：C:\cmd_auto.bat
8
9  @echo off
10 doskey ls=dir /b $*
11 :: doskey act=activate tensorflow-gpu $*
12 :: doskey tb=tensorboard --logdir $*
13 doskey pi=pip install $*
14 doskey basenv="D:\Program Files\Python\Python310-Envs\Basic\Scripts\activate.
```

3.3 安装 Python 包

1. pqi: Fast switching PyPi mirror image source. 由于国内通过pip下载python包的速度很慢，很容易因为超时而失败，而pqi可以把PyPI源迅速切换为国内源tuna, douban, aliyun, ustc从而加快python包的安装速度。

   `pip install pqi`

2. 下面具体演示下 `pqi` 的用法：

```
1  $ pqi -h
2  PQI
3  Usage:
```

```
 4    pqi ls
 5    pqi use <name>
 6    pqi show
 7    pqi add <name> <url>
 8    pqi remove <name>
 9    pqi (-h | --help)
10    pqi (-v | --version)
11 Options:
12   -h --help       Show this screen.
13   -v --version    Show version.
14
15 $ pqi show
16 Current source is pypi.
17
18 $ pqi ls
19 pypi     https://pypi.python.org/simple/
20 tuna     https://pypi.tuna.tsinghua.edu.cn/simple
21 douban   http://pypi.douban.com/simple/
22 aliyun   https://mirrors.aliyun.com/pypi/simple/
23 ustc     https://mirrors.ustc.edu.cn/pypi/web/simple
24
25 $ pqi use aliyun
26 Source is changed to aliyun(https://mirrors.aliyun.com/pypi/simple/).
27
```

3.4 `pip install`

```
$ pip install matplotlib
Looking in indexes: https://pypi.tuna.tsinghua.edu.cn/simple, https://pypi.mirrors.ust
c.edu.cn/simple/, https://pypi.douban.com/simple/, https://pypi.python.org/simple/
Collecting matplotlib
  Downloading matplotlib-3.7.1-cp39-cp39-macosx_10_12_x86_64.whl (7.4 MB)
                                      ━━━━━ 7.4/7.4 MB 8.4 MB/s eta 0:00:00
Requirement already satisfied: numpy>=1.20 in ./work/PyEnvs/python3.9/lib/python3.9/si
te-packages (from matplotlib) (1.24.2)
Collecting importlib-resources>=3.2.0
  Downloading importlib_resources-5.12.0-py3-none-any.whl (36 kB)
Collecting contourpy>=1.0.1
  Downloading contourpy-1.0.7-cp39-cp39-macosx_10_9_x86_64.whl (244 kB)
                                      ━━━━━ 244.3/244.3 kB 2.3 MB/s eta 0:00:00
Requirement already satisfied: pillow>=6.2.0 in ./work/PyEnvs/python3.9/lib/python3.9/
site-packages (from matplotlib) (9.4.0)
Collecting python-dateutil>=2.7
  Downloading python_dateutil-2.8.2-py2.py3-none-any.whl (247 kB)
                                      ━━━━━ 247.7/247.7 kB 5.5 MB/s eta 0:00:00
Requirement already satisfied: pyparsing>=2.3.1 in ./work/PyEnvs/python3.9/lib/python3
.9/site-packages (from matplotlib) (3.0.8)
Collecting cycler>=0.10
  Downloading cycler-0.11.0-py3-none-any.whl (6.4 kB)
Requirement already satisfied: packaging>=20.0 in ./work/PyEnvs/python3.9/lib/python3.
9/site-packages (from matplotlib) (21.3)
Collecting fonttools>=4.22.0
  Downloading fonttools-4.39.0-py3-none-any.whl (1.0 MB)
                                      ━━━━━ 1.0/1.0 MB 8.9 MB/s eta 0:00:00
Collecting kiwisolver>=1.0.1
  Downloading kiwisolver-1.4.4-cp39-cp39-macosx_10_9_x86_64.whl (65 kB)
                                      ━━━━━ 65.5/65.5 kB 2.1 MB/s eta 0:00:00
Requirement already satisfied: zipp>=3.1.0 in ./work/PyEnvs/python3.9/lib/python3.9/si
te-packages (from importlib-resources>=3.2.0->matplotlib) (3.8.0)
Collecting six>=1.5
  Downloading six-1.16.0-py2.py3-none-any.whl (11 kB)
Installing collected packages: six, kiwisolver, importlib-resources, fonttools, cycler
, contourpy, python-dateutil, matplotlib
Successfully installed contourpy-1.0.7 cycler-0.11.0 fonttools-4.39.0 importlib-resour
ces-5.12.0 kiwisolver-1.4.4 matplotlib-3.7.1 python-dateutil-2.8.2 six-1.16.0
(python3.9)
```