

SCHOOL OF INFORMATION SCIENCE AND ENGINEERING

EAST UNIVERSITY OF SCIENCE AND TECHNOLOGY

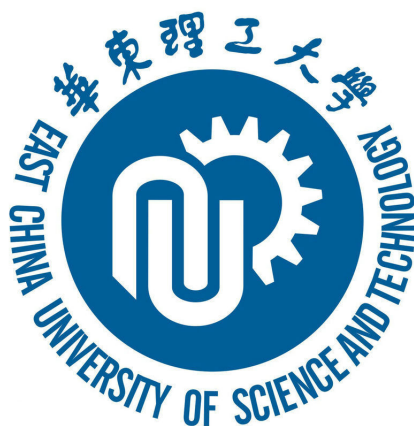
---

# Tutorial for Innovation Program 2019 1.0

---

*Written By :*

Wei Zhang

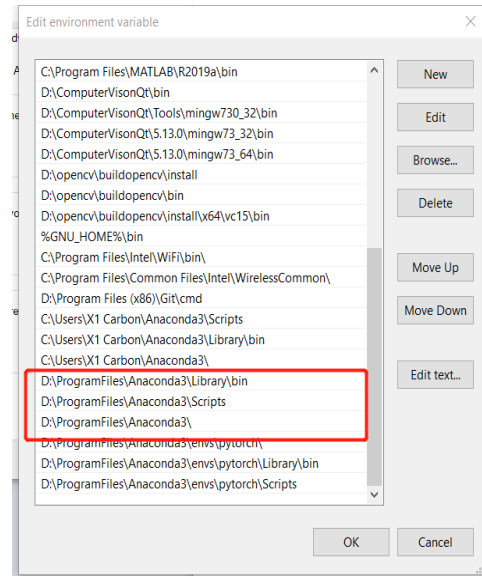


## 1 前言

由于人工智能正在几乎所有自动化领域得到渗透。深度学习是用于开发和最大化类人计算机效率的最棘手的架构之一。为了帮助产品开发人员，Google, Facebook 和其他大型科技公司已经发布了各种 Python 环境框架，可以学习，构建和培训多样化的神经网络。谷歌的 TensorFlow 是深度学习的开源框架，多年来受到欢迎。使用新框架，PyTorch 因其易于编写的代码而受到初学者的关注。PyTorch 是基于 Python, C++ 和 CUDA 后端开发的，可用于 Linux, macOS 和 Windows。这两个广泛使用的框架之间存在一些差异，可能是因为它们实现代码，可视化技术以及静态和动态编程的方式。当下学术研究领域，Pytorch 水生火热，大部分 CVPR 中论文都使用了 Pytorch，因此本教程仅提供了 Win10 下 Pytorch(五 GPU) 的环境配置教程，稍后会更新有 GPU 的教程，若对 tensorflow 有兴趣的同学可[点击此处浏览深度学习之 anaconda 以及 Tensorflow 的安装](#)

## 2 配置 conda 环境

1. 下载 anaconda3.7:[Anaconda 官网](#) 不要忘记添加相关的环境变量 (以自己的安装路径为准)



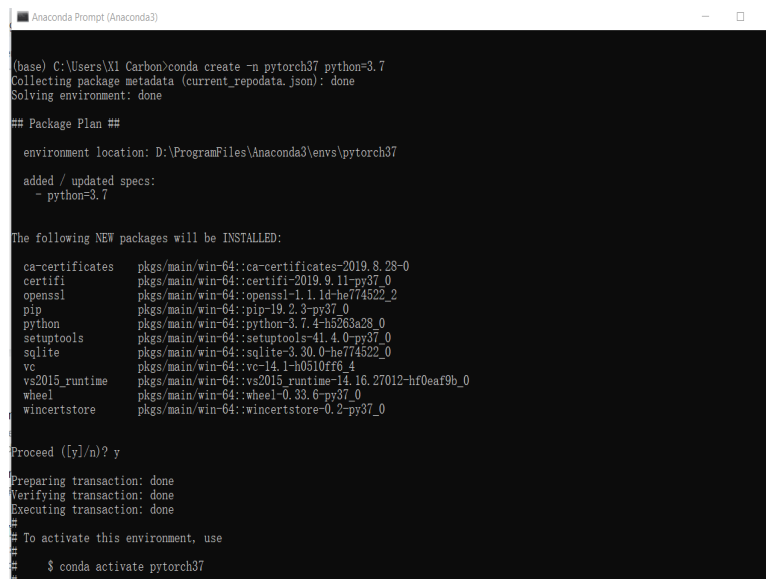
2. 我们在安装 PyTorch 之前最好先创建一个 pytorch 的虚拟环境。之所以创建虚拟环境是因为 Python 为不同的项目需求创建不同的虚拟环境非常常见。在实际项目开发中，我们通常会根据自己的需求去下载各种相应的框架库，但是可能每个项目使用的框架库不一样，或使用框架的版本不一样，这样需要我们根据需求不断的更新或卸载相应的库，

管理起来相当麻烦。所以通过创建虚拟环境，相当于为不同的项目创建一块独立的空间，在这个空间里，你安装任何库和框架都是独立的，不会影响到外部环境。

打开 Anaconda Prompt 之后，在命令行输入：

```
1 conda create -n pytorch37 python=3.7
```

注意，这里的 pytorch37 是虚拟环境的名称，可随意取，因为本机已经安装了名为 pytorch 的 env，推荐取名为 pytorch。然后继续下列命令，进入 pytorch 虚拟环境：



```

Anaconda Prompt (Anaconda3)
(base) C:\Users\XI Carbon>conda create -n pytorch37 python=3.7
Collecting package metadata (current_repodata.json): done
Solving environment: done

## Package Plan ##

  environment location: D:\ProgramFiles\Anaconda3\envs\pytorch37

  added / updated specs:
    - python=3.7

The following NEW packages will be INSTALLED:

ca-certificates pkgs/main/win-64::ca-certificates-2019.8.28-0
certifi          pkgs/main/win-64::certifi-2019.9.11-py37_0
openssl          pkgs/main/win-64::openssl-1.1.1d-he774522_2
pip              pkgs/main/win-64::pip-19.2.3-py37_0
python           pkgs/main/win-64::python-3.7.4-h5263a28_0
setuptools       pkgs/main/win-64::setuptools-41.4.0-py37_0
sqlite           pkgs/main/win-64::sqlite-3.30.0-he774522_0
vc               pkgs/main/win-64::vc-14.1-h0510ff6_4
vs2015_runtime  pkgs/main/win-64::vs2015_runtime-14.16.27012-hf0eaf9b_0
wheel            pkgs/main/win-64::wheel-0.33.6-py37_0
wincertstore     pkgs/main/win-64::wincertstore-0.2-py37_0

Proceed ([y]/n)? y
Preparing transaction: done
Verifying transaction: done
Executing transaction: done

#
# To activate this environment, use
#
#     $ conda activate pytorch37
#

```

```
1 conda activate pytorch
```

### 3. 在新环境中安装 python

在命令行输入：

```
1 conda install python
```

```

# $ conda activate pytorch37
# To deactivate an active environment, use
# $ conda deactivate

(base) C:\Users\Xl Carbon>conda activate pytorch37

(pytorch37) C:\Users\Xl Carbon>conda install python
Collecting package metadata (current_repodata.json): done
Solving environment: done

# All requested packages already installed.

(pytorch37) C:\Users\Xl Carbon>python
Python 3.7.4 (default, Aug 9 2019, 18:34:13) [MSC v.1915 64 bit (AMD64)] :: Anaconda, Inc. on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> import torch
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ModuleNotFoundError: No module named 'torch'
>>>

```

此时我们在命令行中输入如上图所示的 `python`，进入新环境中的 `python`。输入 `import torch` 我们可以看到没有此库，等下可以用此方法检测 PyTorch 是否安装成功，省去了打开 IDE 的麻烦。好了，PyTorch 的虚拟环境创建好了。接下来就是安装 PyTorch 了。不要关闭 Prompt，我们还要继续输入命令。

### 3 安装 Pytorch

PyTorch 的官网提供了简单的安装方法，只需简单的命令即可。

首先，打开 PyTorch 官网安装页面 <https://pytorch.org/get-started/locally/> 可以看到页面里

Please select the PyTorch version and environment details for your system.

PyTorch Build	Stable (1.2)		Preview (Nightly)	
Your OS	Linux	Mac	Windows	
Package	Conda	Pip	LibTorch	Source
Language	Python 2.7	Python 3.5	Python 3.6	Python 3.7 C++
CUDA	9.2	10.0	None	
Run this Command:	conda install pytorch torchvision cpuonly -c pytorch			

的这张图，非常简单。我们的环境是：Windows、Conda、Python 3.7、None CUDA(因为是 CPU，如果使用 GPU 的话需要安装 CUDA 并根据相应版本下载，对应的命令是：(记住先输入 `exit()`) 退出 python

```
>>> import torch
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ModuleNotFoundError: No module named 'torch'
>>> exit()

(pytorch37) C:\Users\Xl Carbon>conda install pytorch torchvision cpuonly -c pytorch
Collecting package metadata (current_repodata.json): done
Solving environment: done

## Package Plan ##

  environment location: D:\ProgramFiles\Anaconda3\envs\pytorch37

  added / updated specs:
    - cpuonly
    - pytorch
    - torchvision

The following NEW packages will be INSTALLED:

blas                pkgs/main/win-64::blas-1.0-mkl
effi                pkgs/main/win-64::effi-1.12.3-py37h7aldbcl_0
cpuonly            pytorch/noarch::cpuonly-1.0-0
freetype            pkgs/main/win-64::freetype-2.9.1-ha9979f8_1
icc_rt             pkgs/main/win-64::icc_rt-2019.0.0-h0cc432a_1
intel-openmp       pkgs/main/win-64::intel-openmp-2019.4-245
jpeg               pkgs/main/win-64::jpeg-9b-hb8344c_2
libpng             pkgs/main/win-64::libpng-1.6.37-h2a8f88b_0
libtiff            pkgs/main/win-64::libtiff-4.0.10-hb898794_2
mkl                pkgs/main/win-64::mkl-2019.4-245
mkl-service        pkgs/main/win-64::mkl-service-2.3.0-py37hb782905_0
mkl_fft            pkgs/main/win-64::mkl_fft-1.0.14-py37h14836fe_0
mkl_random         pkgs/main/win-64::mkl_random-1.1.0-py37h675688f_0
ninja              pkgs/main/win-64::ninja-1.9.0-py37h74a9793_0
numpy              pkgs/main/win-64::numpy-1.16.5-py37h19fblc0_0
numpy-base        pkgs/main/win-64::numpy-base-1.16.5-py37hc3f5095_0
olefile            pkgs/main/win-64::olefile-0.46-py37_0
```

到这里, PyTorch 已经安装完了。接着, 我们再次输入 python 命令进入 python 检验 PyTorch 是否安装成功

```
1 >>> import torch
2 >>> import torchvision
```

```
Anaconda Prompt (Anaconda3) - conda install pytorch torchvision cpuonly -c pytorch - python
blas                pkgs/main/win-64::blas-1.0-mkl
effi                pkgs/main/win-64::effi-1.12.3-py37h7aldbcl_0
cpuonly            pytorch/noarch::cpuonly-1.0-0
freetype            pkgs/main/win-64::freetype-2.9.1-ha9979f8_1
icc_rt             pkgs/main/win-64::icc_rt-2019.0.0-h0cc432a_1
intel-openmp       pkgs/main/win-64::intel-openmp-2019.4-245
jpeg               pkgs/main/win-64::jpeg-9b-hb8344c_2
libpng             pkgs/main/win-64::libpng-1.6.37-h2a8f88b_0
libtiff            pkgs/main/win-64::libtiff-4.0.10-hb898794_2
mkl                pkgs/main/win-64::mkl-2019.4-245
mkl-service        pkgs/main/win-64::mkl-service-2.3.0-py37hb782905_0
mkl_fft            pkgs/main/win-64::mkl_fft-1.0.14-py37h14836fe_0
mkl_random         pkgs/main/win-64::mkl_random-1.1.0-py37h675688f_0
ninja              pkgs/main/win-64::ninja-1.9.0-py37h74a9793_0
numpy              pkgs/main/win-64::numpy-1.16.5-py37h19fblc0_0
numpy-base        pkgs/main/win-64::numpy-base-1.16.5-py37hc3f5095_0
olefile            pkgs/main/win-64::olefile-0.46-py37_0
pillow             pkgs/main/win-64::pillow-6.2.0-py37hdc69c19_0
pyparser           pkgs/main/win-64::pyparser-2.19-py37_0
pytorch            pytorch/win-64::pytorch-1.2.0-py3.7_cpu_1
six                pkgs/main/win-64::six-1.12.0-py37_0
tk                 pkgs/main/win-64::tk-8.6.8-hd6ae2cd_0
torchvision        pytorch/win-64::torchvision-0.4.0-py37_cpu
xz                 pkgs/main/win-64::xz-5.2.4-h2fa13f4_4
zlib               pkgs/main/win-64::zlib-1.2.11-h62dc097_3
zstd               pkgs/main/win-64::zstd-1.3.7-h508b16e_0

Proceed ([y]/n)? y

Preparing transaction: done
Verifying transaction: done
Executing transaction: done

(pytorch37) C:\Users\Xl Carbon>python
Python 3.7.4 (default, Aug 9 2019, 18:34:13) [MSC v.1915 64 bit (AMD64)] :: Anaconda, Inc. on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> import torch
>>> import torchvision
>>>
```

注：torchvision 是独立于 pytorch 的关于图像操作的一些方便工具库。主要包一些常用视觉数据集如 MNIST，COCO 等等，模型如 AlexNet，VGG 等等，图像操作和生成图像网格

至此，PyTorch 便成功安装。以上就是第一个教程的全部内容，希望对大家的学习有所帮助，也希望大家钻研探索，有问题可以随时一起探讨。

下一小结的教程，我将加入选择合适的 IDE 进行编程，当然此时 anaconda 中的 IDE 已经可以运行 PyTorch，我比较喜欢 Pycharm，萝卜青菜各有所爱，这里没有好坏推荐哈！手写数字识别的源代码我已上传，[点击此处查看](#)，后续我也会上传各段代码说明的 Notebook 最后，欢迎大家对计算机视觉方向，如人体姿态识别（HPE），3D 重建，甚至是现在的图像识别分类，都可以和我探讨一起进步，多谢啦！