

Windows10 系统下深度学习环境配置

作者：刘子健 时间：2019/2/20

总结来说分为三个步骤：

- CUDA 和 cudnn 安装；
- Anaconda 安装；
- Tensorflow、Pytorch 等深度学习框架的安装。

*要说明的是，nvidia driver、CUDA、cudnn、框架都是要版本对应的。

CUDA 和 cudnn 安装

1、查看英伟达驱动的版本号

可通过在命令行中拖入 “C:\Program Files\NVIDIA Corporation\NVSMI” 路径下的 nvidia-smi 查看。nvidia-smi 可以查看驱动的版本号和 GPU 的使用率等信息。本机的显卡为 GTX 1050 Ti (388.13)，其对应的 CUDA 版本为 9.0。

Table 1. CUDA Toolkit and Compatible Driver Versions

CUDA Toolkit	Linux x86_64 Driver Version	Windows x86_64 Driver Version
CUDA 10.0.130	>= 410.48	>= 411.31
CUDA 9.2 (9.2.148 Update 1)	>= 396.37	>= 398.26
CUDA 9.2 (9.2.88)	>= 396.26	>= 397.44
CUDA 9.1 (9.1.85)	>= 390.46	>= 391.29
CUDA 9.0 (9.0.76)	>= 384.81	>= 385.54
CUDA 8.0 (8.0.61 GA2)	>= 375.26	>= 376.51
CUDA 8.0 (8.0.44)	>= 367.48	>= 369.30
CUDA 7.5 (7.5.16)	>= 352.31	>= 353.66
CUDA 7.0 (7.0.28)	>= 346.46	>= 347.62

(<https://docs.nvidia.com/cuda/cuda-toolkit-release-notes/index.html>)

2、安装对应版本的 CUDA 和 cudnn

CUDA 的版本链接：<https://docs.nvidia.com/cuda/cuda-toolkit-release-notes/index.html>

cudnn 的版本链接：<https://developer.nvidia.com/rdp/cudnn-archive>

其中，cudnn 下载时需要注册登录账号，国内的同学可以用学校的 EDU 邮箱账号注册，之前用 sina, QQ 等邮箱注册没成功。

CUDA 选择对应的机型信息进行下载（本机为 Cuda 9.0- Windows 10 exe(local)），在安装时，首先需要下载到一个保存路径（任意就好），在安装的界面，选择自定义安装，并且仅勾选 CUDA 和 CUDA 下的前四项即可（如下图）。



选择各模块的安装路径，可以选择默认：

C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v9.0

C:\ProgramData\NVIDIA Corporation\CUDA Samples\v9.0

接下来，需要配置环境变量（电脑右键->高级系统设置->环境变量，记得是修改系统用环境变量），在系统变量中添加以下变量（与 Path 同级位置）

CUDA_PATH = C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v9.0

CUDA_SDK_PATH = C:\ProgramData\NVIDIA Corporation\CUDA Samples\v9.0

CUDA_LIB_PATH = %CUDA_PATH%\lib\x64

CUDA_BIN_PATH = %CUDA_PATH%\bin

CUDA_SDK_BIN_PATH = %CUDA_SDK_PATH%\bin\win64

CUDA_SDK_LIB_PATH = %CUDA_SDK_PATH%\common\lib\x64

并且在 Path 中添加若干路径：

```
C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v9.0\lib\x64
C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v9.0\bin
C:\ProgramData\NVIDIA Corporation\CUDA Samples\v9.0\common\lib\x64
C:\ProgramData\NVIDIA Corporation\CUDA Samples\v9.0\bin\win64
C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v9.0\lib\x64
C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v9.0\bin
C:\ProgramData\NVIDIA Corporation\CUDA Samples\v9.0\common\lib\x64
C:\ProgramData\NVIDIA Corporation\CUDA Samples\v9.0\bin\win64
```

接着，将对应版本的 cudnn 下载，解压，并将其目录下的四个文件（三个文件夹和一个 txt）Copy 到 C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v9.0 下（可能需要管理员权限），这样 CUDA 和 cudnn 就安装好了。

可通过在命令行中输入 `nvcc -V` 查看对应的 CUDA 版本号，并可以通过拖入 C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v9.0\extras\demo_suite 目录下的 `bandwidthTest.exe` 和 `deviceQuery.exe` 进行是否安装成功的检查（如下图）。

```
C:\Users\Asus>"C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v9.0\extras\demo_suite\bandwidthTest.exe"
[CUDA Bandwidth Test] - Starting...
Running on...

Device 0: GeForce GTX 1050 Ti
Quick Mode

Host to Device Bandwidth, 1 Device(s)
PINNED Memory Transfers
  Transfer Size (Bytes)      Bandwidth(MB/s)
  33554432                  12543.6

Device to Host Bandwidth, 1 Device(s)
PINNED Memory Transfers
  Transfer Size (Bytes)      Bandwidth(MB/s)
  33554432                  12968.1

Device to Device Bandwidth, 1 Device(s)
PINNED Memory Transfers
  Transfer Size (Bytes)      Bandwidth(MB/s)
  33554432                  95209.8

Result = PASS

NOTE: The CUDA Samples are not meant for performance measurements. Results may vary when GPU Boost is enabled.
```

Anaconda 安装

Anaconda 指的是一个开源的 Python 发行版本，其包含了 conda、Python 等 180 多个科学包及其依赖项。使用 Anaconda 安装 python 中的库，可以省掉很多各个包之间的冲突等问题（这也是 python 使用中很烦的一点）。可在 Anaconda 的官网或者国内的镜像源网站下载。

在安装时，仅需注意勾选添加到环境变量即可（默认是未勾选的）。

同时我们可以为 conda 添加国内源，让下载库时速度更快些吧

`conda config --add channels https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkg/free/`

conda config --set show_channel_urls yes

而 Pip 源，我们只需要在安装时，添加 `-i URL` 即可

`pip install -i https://pypi.tuna.tsinghua.edu.cn/simple (pkgs name)`

这样为我们在后面下载各个库函数时节约了不少时间，不然好多包用官网源下载还是很慢的（当然搭梯子或许也可以解决这个问题）。

深度学习框架的安装

本文以 Tensorflow 和 Pytorch 为例进行讲解：

1、Tensorflow-gpu 安装

本文选择的是用 Conda 命令安装，之前用 pip 安装好像会出现和 numpy 冲突的问题。

这里也要注意与 CUDA 版本的对应关系（如下图），不然装错了又没办法用喽。

Version	Python version	Compiler	Build tools	cuDNN	CUDA
tensorflow_gpu-1.11.0	3.5-3.6	MSVC 2015 update 3	Bazel 0.15.0	7	9
tensorflow_gpu-1.10.0	3.5-3.6	MSVC 2015 update 3	Cmake v3.6.3	7	9
tensorflow_gpu-1.9.0	3.5-3.6	MSVC 2015 update 3	Cmake v3.6.3	7	9
tensorflow_gpu-1.8.0	3.5-3.6	MSVC 2015 update 3	Cmake v3.6.3	7	9
tensorflow_gpu-1.7.0	3.5-3.6	MSVC 2015 update 3	Cmake v3.6.3	7	9
tensorflow_gpu-1.6.0	3.5-3.6	MSVC 2015 update 3	Cmake v3.6.3	7	9
tensorflow_gpu-1.5.0	3.5-3.6	MSVC 2015 update 3	Cmake v3.6.3	7	9
tensorflow_gpu-1.4.0	3.5-3.6	MSVC 2015 update 3	Cmake v3.6.3	6	8
tensorflow_gpu-1.3.0	3.5-3.6	MSVC 2015 update 3	Cmake v3.6.3	6	8
tensorflow_gpu-1.2.0	3.5-3.6	MSVC 2015 update 3	Cmake v3.6.3	5.1	8
tensorflow_gpu-1.1.0	3.5	MSVC 2015 update 3	Cmake v3.6.3	5.1	8
tensorflow_gpu-1.0.0	3.5	MSVC 2015 update 3	Cmake v3.6.3	5.1	8

在命令行中输入：`conda install tensorflow-gpu==1.12.0`（注意要选择对应的版本号）

等待片刻，安装结束。在命令行中，我们可以输入 `python test_gpu.py` 运行测试文件来判断是否安装成功（`test_gpu.py` 为这个文件同目录下的 py 文件）。

```
C:\Users\Asus\python .\Desktop\test_gpu.py
E:\Anaconda\lib\site-packages\b5py\__init__.py:36: FutureWarning: Conversion of the second argument of issubdtype from `float` to `np.float64` is deprecated. In the future, it will be `np.floating`, with automatic casting among the floats now implemented via UPSTC casting rules.
  from .conv import register_converters as _register_converters
2019-02-20 10:24:41.828136: I tensorflow/core/platform/cpu_feature_guard.cc:141] Your CPU supports instructions that this TensorFlow binary
2019-02-20 10:24:43.018625: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1432] Found device 0 with properties:
  name: GeForce GTX 1050 Ti major: 6 minor: 1 memoryClockRate(GHz): 1.455
  pciBusID: 0000:01:00.0
  totalMemory: 4.00GiB freeMemory: 3.31GiB
2019-02-20 10:24:43.052392: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1511] Adding visible gpu devices: 0
2019-02-20 10:24:50.830696: I tensorflow/core/common_runtime/gpu/gpu_device.cc:982] Device interconnect StreamExecutor with strength 1 edge
2019-02-20 10:24:50.844398: I tensorflow/core/common_runtime/gpu/gpu_device.cc:988] 0
2019-02-20 10:24:50.854884: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1001] 0: N
2019-02-20 10:24:50.918163: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1115] Created TensorFlow device (/job:localhost/replica:0/task:0/device:GPU:0 with name: GeForce GTX 1050 Ti, pci bus id: 0000:01:00.0, compute capability: 6.1)
Device mapping:
/job:localhost/replica:0/task:0/device:GPU:0 -> device: 0, name: GeForce GTX 1050 Ti, pci bus id: 0000:01:00.0, compute capability: 6.1
2019-02-20 10:24:50.983584: I tensorflow/core/common_runtime/direct_session.cc:307] Device mapping:
/job:localhost/replica:0/task:0/device:GPU:0 -> device: 0, name: GeForce GTX 1050 Ti, pci bus id: 0000:01:00.0, compute capability: 6.1
add: (Add): /job:localhost/replica:0/task:0/device:GPU:0
2019-02-20 10:24:51.100367: I tensorflow/core/common_runtime/placer.cc:927] add: (Add)/job:localhost/replica:0/task:0/device:GPU:0
init: (NoOp): /job:localhost/replica:0/task:0/device:GPU:0
2019-02-20 10:24:51.114338: I tensorflow/core/common_runtime/placer.cc:927] init: (NoOp)/job:localhost/replica:0/task:0/device:GPU:0
a: (Const): /job:localhost/replica:0/task:0/device:GPU:0
2019-02-20 10:24:51.127339: I tensorflow/core/common_runtime/placer.cc:927] a: (Const)/job:localhost/replica:0/task:0/device:GPU:0
b: (Const): /job:localhost/replica:0/task:0/device:GPU:0
2019-02-20 10:24:51.144291: I tensorflow/core/common_runtime/placer.cc:927] b: (Const)/job:localhost/replica:0/task:0/device:GPU:0
[2. 4. 6.]
```

2、Pytorch-gpu 安装

说起 pytorch 的安装，也是一把辛酸泪呀。

在 pytorch 的官网（<https://pytorch.org/>）我们可以查询到具体的安装命令

PyTorch Build	Stable (1.0)		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	8.0	9.0	10.0	None
Run this Command:	conda install pytorch torchvision cudatoolkit=9.0 -c pytorch			

***也要注意版本号对应哦**

开始本来想的是用 Conda 安装（`conda install pytorch torchvision cudatoolkit=9.0 -c pytorch`），省的出冲突的问题嘛，但是每次下载包都会中断，试了一下午还是不行，后来想的要么就离线把包给下载下来，然后用 conda 安装离线包（`conda install pkgs`），但是命令行下还是迟迟没有反应，最终决定还是采用 pip 安装吧。

pip 安装（`pip install https://download.pytorch.org/whl/cu90/torch-1.0.1-cp36-cp36m-win_amd64.whl`）时也是出现下载包中断的问题，于是也是离线下载的安装包（别问我为啥离线下载没中断，本机电脑用的是 wifi 连接的，学校的网络也不太稳定，因此一次又一次的中断，于是我/偷偷/在服务器上命令下载离线安装包

`wget https://download.pytorch.org/whl/cu90/torch-1.0.1-cp36-cp36m-win_amd64.whl`），又回传到本机电脑，然后使用的 `pip install torch-1.0.1-cp36-cp36m-win_amd64` 命令安装。最后再输入 `pip install torchvision`，这样 pytorch 及其对应的包也就安装好了，当然也可以通过一些命令进行成功测试。

在 python 环境下输入：

```
>>> import torch as t
```

```
>>> device = t.device('cuda:0' if t.cuda.is_available() else 'cpu')
```

```
>>> print(device)
```

若最终显示的结果是 `cuda:0` 也就说明 gpu 版本已经安装好啦。

好了，不说了，我要去搬砖了...