

Ubuntu下深度学习环境配置

一.装双系统

[参考一](#) [参考二](#)

查看硬盘分区模式

我的电脑——管理——磁盘管理——磁盘右击选择属性——卷，我的磁盘是GPT分区（注：在GPT硬盘格式下其实是没有什么主分区和逻辑分区区别的，都是主分区，所以我把分区类型都设为主分区。）

查看启动方式：

按Win+R打开运行，输入msinfo32，回车查看系统信息。在BIOS模式中如果显示“传统”，表示系统启动方式为Legacy BIOS;如果为UEFI，则显示UEFI。

至此，明确了我的电脑是UEFI+GPT

之前做过启动盘的U盘恢复成非启动盘：在磁盘管理中删除卷、格式化即可。

关闭快速启动（电源设置）和安全启动（BIOS）

注意：要安装系统的磁盘要处于空闲状态，即被删除的状态

二.安装NVIDIA驱动

NVIDIA驱动即CUDA driver version驱动版本(与之相对应的是CUDA runtime version运行时版本)，也就是说NVIDIA不能随便装，要根据选用的CUDA版本去安装，下图是版本对应关系

Table 1. CUDA Toolkit and Compatible Driver Versions

CUDA Toolkit	Linux x86_64 Driver Version	Windows x86_64 Driver Version
CUDA 10.1.105	>= 418.39	>= 418.96
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

CUDA我

准备装10.0版本，所以NVIDIDA驱动我要安装大于410.48版的，这里我选择官网的430.64版。一开始是现将驱动包下载下来（网速很慢，下了半天），然后再安装，不过发现安装失败，出现下面的情况，所以放弃。

```
Verifying archive integrity... Error in check sums xxxxxxxx xxxxx
```

最后准备使用**PPA源下载**的方式安装：

```
sudo add-apt-repository ppa:graphics-drivers/ppa
sudo apt-get update
sudo apt-get install nvidia-430 #此处要根据上面查询到的版本适当更改
sudo apt-get install mesa-common-dev
sudo apt-get install freeglut3-dev
```

上面的430是通过[NVIDIA驱动查找](#)找到的 重启

```
nvidia-smi #若出现电脑GPU列表，即安装成功
```

[CUDA、显卡驱动和Tensorflow版本之间的对应关系](#)

三.CUDA安装

[参考链接](#) 官网下的CUDA包中自带NVIDIA驱动，上面已经安装好了，所以在安装CUDA的时候需要注意是**不带驱动安装**

```
sudo sh cuda_xxxxx_linux.run#注意在driver处选择n
```

在home下打开.bashrc文件（ctrl+H显示隐藏文件），在最下面添加如下

```
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/usr/local/cuda-10.0/lib64
export PATH=$PATH:/usr/local/cuda-10.0/bin
export CUDA_HOME=$CUDA_HOME:/usr/local/cuda-10.0
```

```
gong@gong-ThinkPad-T470p:~$ nvcc --version
nvcc: NVIDIA (R) Cuda compiler driver
Copyright (c) 2005-2018 NVIDIA Corporation
Built on Sat_Aug_25_21:08:01_CDT_2018
Cuda compilation tools, release 10.0, V10.0.130
```

四.cuDNN安装

[参考链接](#) cuDNN下载是Linux那个版本，不是Ubuntu的版本

```
1 tar -zxvf cudnn-10.0-linux-x64-v7.6.5.32.solitairetheme8.tgz
2 sudo cp cuda/include/cudnn.h /usr/local/cuda/include/
3 sudo cp cuda/lib64/libcudnn* /usr/local/cuda/lib64/ -d
4 sudo chmod a+r /usr/local/cuda/include/cudnn.h
5 sudo chmod a+r /usr/local/cuda/lib64/libcudnn*
```

五.Anaconda安装

安装完Anaconda之后每次打开终端，开头都会有base，表明每次都自动进入了虚拟环境，使用下面的指令可以关闭自动进入。

```
conda config --set auto_activate_base false
```

安装完后发现终端输入

```
whereis python
```

找不到anaconda中的python，说明环境变量没有配好 输入

```
sudo gedit ~/.bashrc
```

在最后输入

```
export PATH="/home/gong/anaconda3/bin:$PATH"
```

然后

```
source ~/.bashrc
```

完成Anaconda环境变量的配置

六.TensorFlow安装

创建虚拟环境

```
conda create -n tf
```

激活环境

```
conda activate tf
```

关闭环境为

```
conda deactivate
```

安装指定版本

```
conda install tensorflow-gpu=1.15
```

漫长的等待~ 可以去做个饭 吃完回来, 继续等待 在下载链表中发现它居然还下载了cudo-10和对应的cudnn。下载完成, 在python环境下输入

```
import tensorflow as tf
```

不报错说明安装成功。 在pycharm中查看CUDA和GPU是否可用：

```
tf.test.is_built_with_cuda()  
tf.test.is_gpu_available()
```

参考：[1.ubuntu16.04下安装&配置anaconda+tensorflow新手教程](#) [2.Ubuntu16.04环境下从Anaconda3安装到Tensorflow\(gpu\)及pycharm安装和测试mnist完整教程](#)

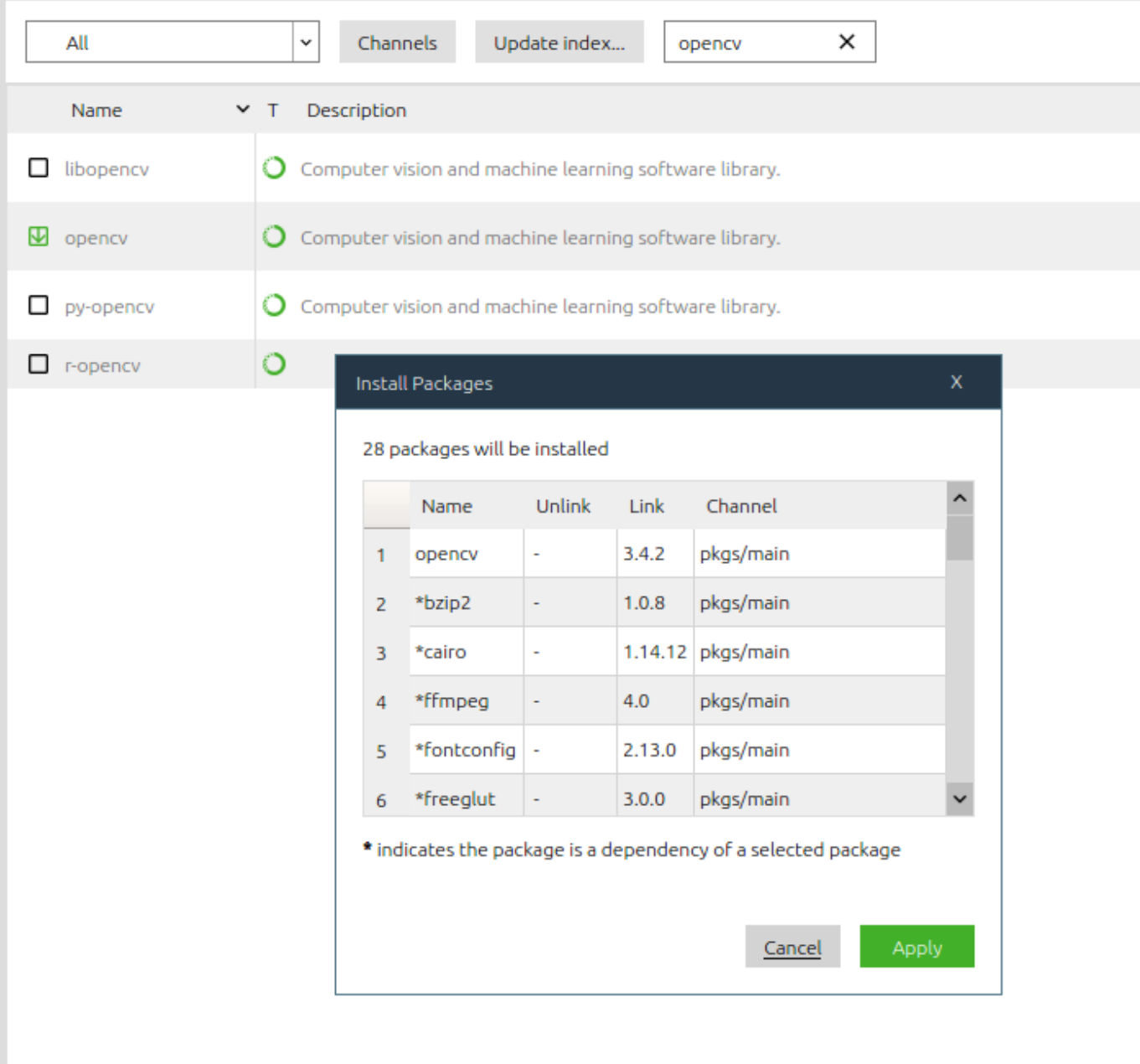
七.PyCharm安装

ctrl+shift+F10可以为当前py自动配置config并且运行

八.Opencv安装

失败一

直接打开Anacon



运行代码时出现

```
Rebuild the library with Windows, GTK+ 2.x or Carbon support
```

需要先执行

```
sudo apt-get install libgtk2.0-dev
```

再安装opencv,最后发现还是没解决，猜测Anaconda下载的opencv本身是阉割版本，需要自己手动编译opencv

失败二

讲两个opencv包从github下下来后，建立build文件夹，终端进入tf虚拟环境，输入

```
cmake -D WITH_TBB=ON -D BUILD_NEW_PYTHON_SUPPORT=ON -D
CMAKE_INSTALL_PREFIX=/usr/local -D WITH_V4L=ON -D WITH_OPENGL=ON -D
WITH_VTK=ON .. -D CMAKE_BUILD_TYPE=RELEASE -D
OPENCV_EXTRA_MODULES_PATH=/home/gong/opencv3/opencv_contrib/modules -D
WITH_IPP=OFF -D
PYTHON_DEFAULT_EXECUTABLE=/home/gong/anaconda3/envs/tf/lib/python3.7 -D
BUILD_opencv_python3=ON -D BUILD_opencv_python2=OFF -D
PYTHON3_EXECUTABLE=/home/gong/anaconda3/envs/tf/lib/python3.7 -D
PYTHON3_INCLUDE_DIR=/home/gong/anaconda3/envs/tf/include/python3.7m -D
PYTHON3_LIBRARY=/home/gong/anaconda3/envs/tf/lib/libpython3.7m.so.1.0 -D
PYTHON_NUMPY_PATH=/home/gong/anaconda3/envs/tf/lib/python3.7/site-packages
-D BUILD_EXAMPLES=ON -D
PYTHON3_PACKAGESS_PATH=/home/gong/anaconda3/envs/tf/lib/python3.7/site-
packages -D BUILD_LIBPROTOBUF_FROM_SOURCES=ON -D BUILD_opencv_hdf=OFF ..
```

中间出现了很多的包下载失败，原因是外网访问被拒绝，虽然最后出现了下图的完成。

```
-- Configuring done
-- Generating done
-- Build files have been written to: /home/gong/opencv3/opencv/build
```

然后输入

```
make -j8
```

到15%的时候，失败：

```
Makefile:160: recipe for target 'all' failed
```

成功

```
conda remove opencv
conda install -c menpo opencv
pip install --upgrade pip
pip install opencv-contrib-python
```

这里用了手机热点，网速快，不然用公寓的wifi下不下来 虽然下载成功，同时代码运行也没有问题，但是这里

opencv 3.4.2 3.4.2

的 **opencv-contrib-python 4.2.0.32** 版本不一致，估计以后还会爆坑

九.额外收货

1.修改默认使用的python版本

Ubuntu默认有两个python版本，2.7.12和3.5.2，默认使用的2.7.12，通过

```
echo alias python=python3 >> ~/.bashrc  
source ~/.bashrc  
python --version
```

可将python2改为python3

2.安装pip和pip3

```
sudo apt-get install python-pip  
pip -V  
  
sudo apt-get install python3-pip  
pip3 -V
```

3.一些必备

[Chrome无VPN安装](#)

4.知识点

Anaconda是python的一个发行版

十.结尾

至此，通过两个下午加两个晚上的努力，最终实现了在Ubuntu系统中成功安装NVIDIA驱动、CUDA、cuDNN、Anaconda、TensorFlow、Pycharm、Opencv的安装，为深度学习打下了环境基础。