faiss-gpu解析

原创 勤菜鸟 勤菜鸟 2018-08-14

环境准备：

双路CPU工作站

128G内存

英伟达1080Ti显卡两块

faiss源码包（2017年3月版）

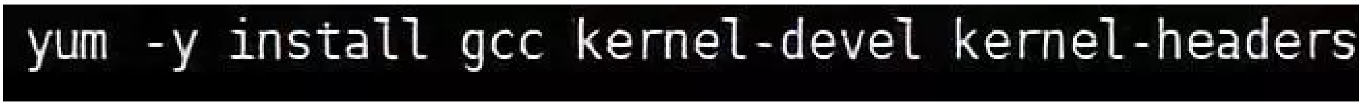
centos7.2+python2.7+anaconda2（环境变量与openblas安装同GPU版）

centos7.2+python3.6+anaconda3（conda安装faiss环境）

1 显卡驱动安装

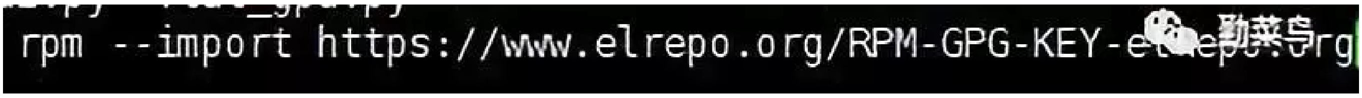
显卡安装有两种方式，一种是用yum从 ELRepo 源中安装，一种是从源码编译驱动。**我们选用第一种方式来安装（简单）。**

依赖环境安装



```
yum -y install gcc kernel-devel kernel-headers
```

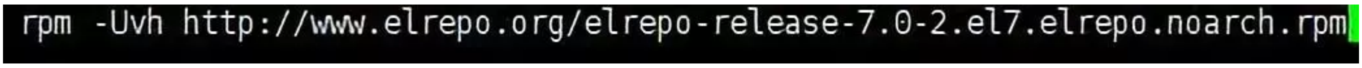
导入**ELRepo**公钥



```
rpm --import https://www.elrepo.org/RPM-GPG-KEY-elrepo.org
```

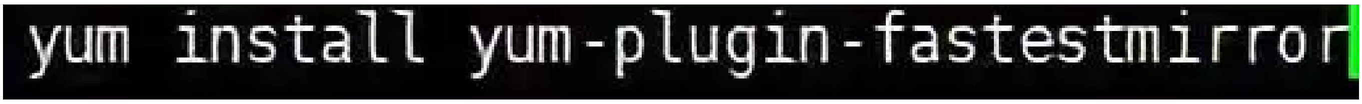
勤菜鸟

安装**ELRepo**



```
rpm -Uvh http://www.elrepo.org/elrepo-release-7.0-2.el7.elrepo.noarch.rpm
```

安装**fastestmirror**插件**来加快下载速度**



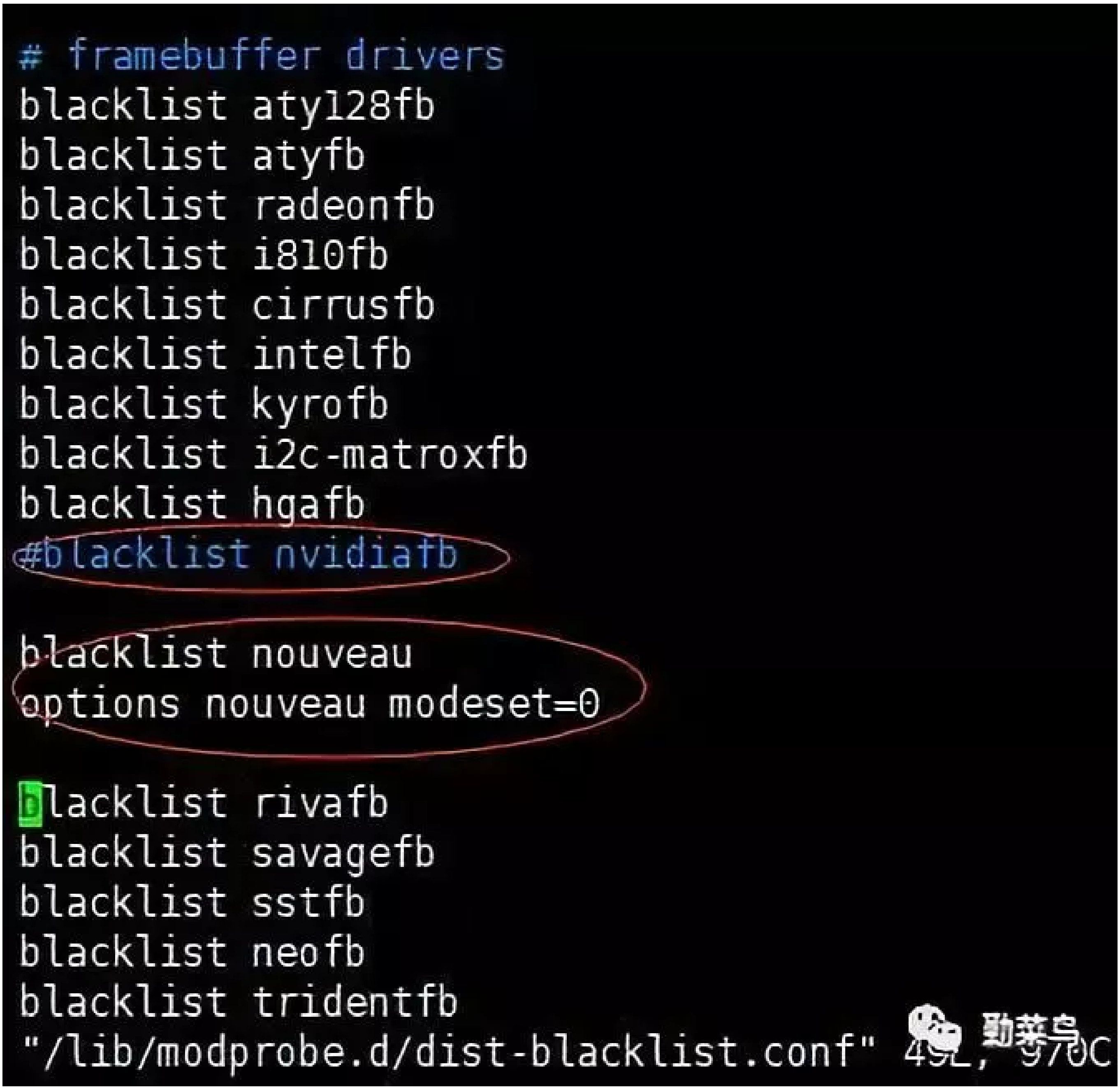
```
yum install yum-plugin-fastestmirror
```

屏蔽自带的显卡驱动

打开/lib/modprobe.d/dist-blacklist.conf，将nvidiafb注释掉。然后添加以下语句：

blacklist nouveau

options nouveau modeset=0

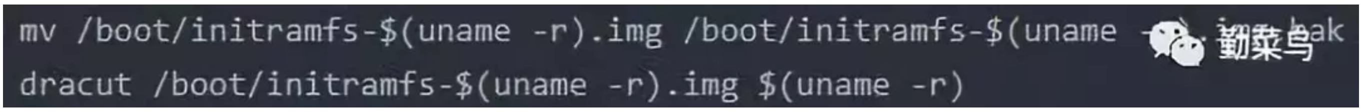


```
# framebuffer drivers
blacklist aty128fb
blacklist atyfb
blacklist radeonfb
blacklist i810fb
blacklist cirrusfb
blacklist intelfb
blacklist kyrofb
blacklist i2c-matroxfb
blacklist hgafb
#blacklist nvidiafb

blacklist nouveau
options nouveau modeset=0

blacklist rivafb
blacklist savagefb
blacklist sstfb
blacklist neofb
blacklist tridentfb
"/lib/modprobe.d/dist-blacklist.conf" 49L, 970C
```

重建initramfs image



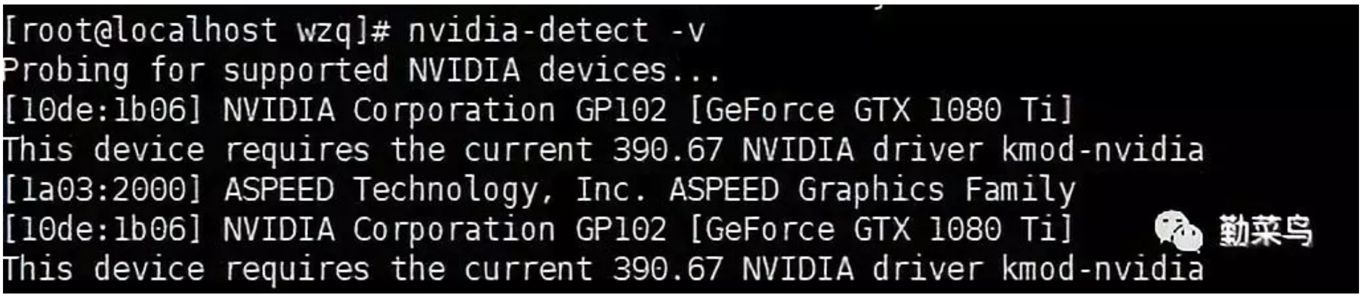
```
mv /boot/initramfs-$(uname -r).img /boot/initramfs-$(uname       bak
dracut /boot/initramfs-$(uname -r).img $(uname -r)
```

重启电脑，ls mod | grep nouveau查看nouveau是否已经禁用，若没有任何输出表示禁用成功。

安装检查显卡程序

sudo yum install nvidia-detect

检查显卡情况



```
[root@localhost wzq]# nvidia-detect -v
Probing for supported NVIDIA devices...
[10de:1b06] NVIDIA Corporation GP102 [GeForce GTX 1080 Ti]
This device requires the current 390.67 NVIDIA driver kmod-nvidia
[1a03:2000] ASPEED Technology, Inc. ASPEED Graphics Family
[10de:1b06] NVIDIA Corporation GP102 [GeForce GTX 1080 Ti]
This device requires the current 390.67 NVIDIA driver kmod-nvidia
```

根据 nvidia-detect 的输出信息，可以知道显卡的型号，以及要使用的驱动版本 390.67

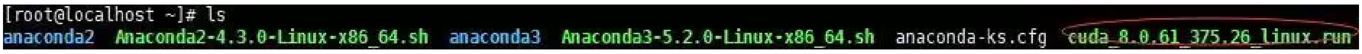
自动匹配驱动安装

yum -y install kmod-nvidia

重启电脑安装就完成啦！

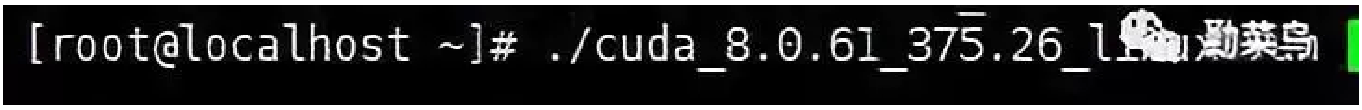
2 cuda-8.0安装

cuda安装比较简单，只需要从英伟达官网下载对应cuda安装包，并运行即可，对应安装包如下：



```
[root@localhost ~]# ls
anaconda2  Anaconda2-4.3.0-Linux-x86_64.sh  anaconda3  Anaconda3-5.2.0-Linux-x86_64.sh  anaconda-ks.cfg  cuda_8.0.61_375.26_linux.run
```

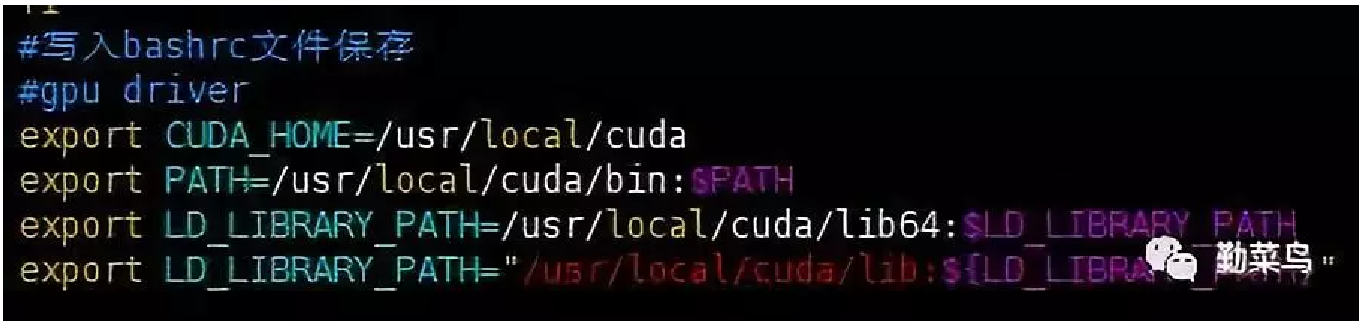
运行安装



```
[root@localhost ~]# ./cuda_8.0.61_375.26_li
```

配置环境变量

vim ~/.bashrc

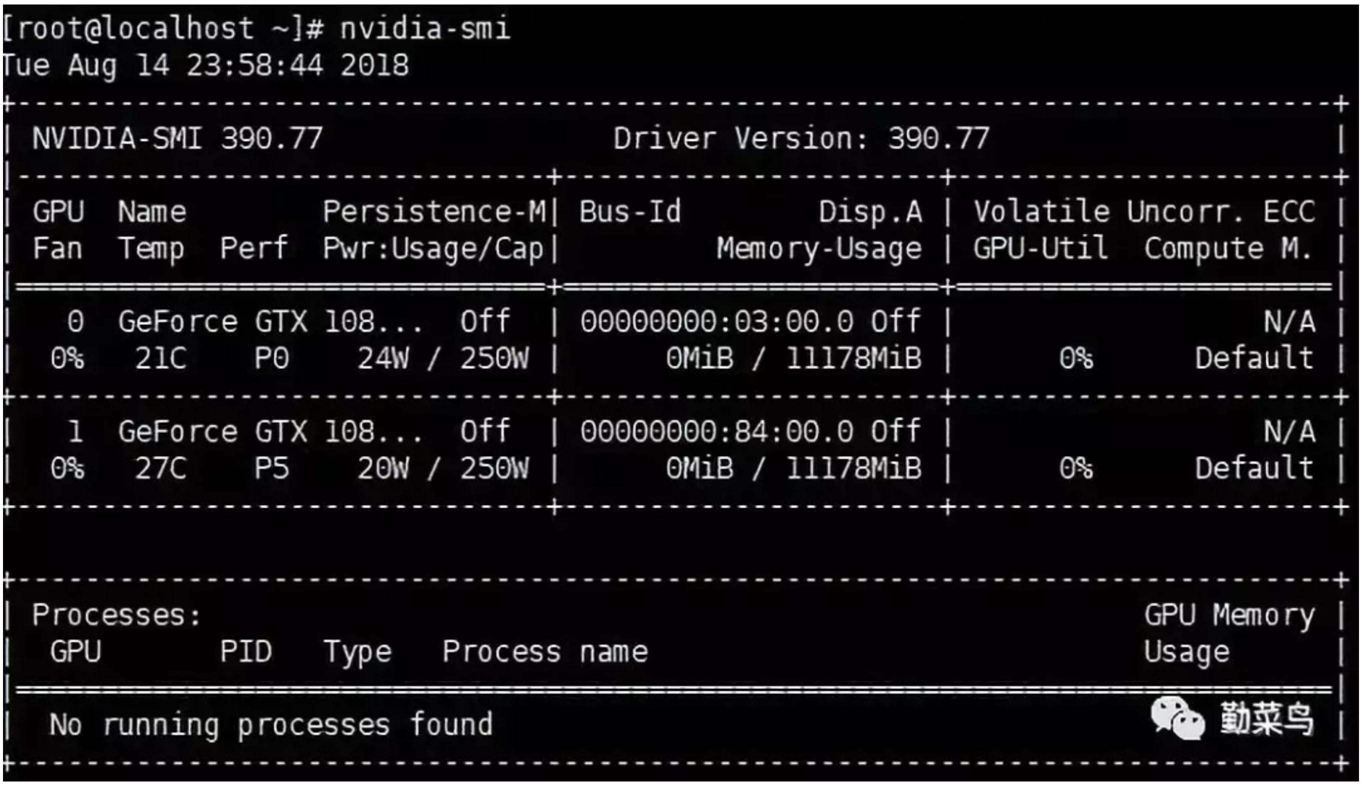


```
#写入bashrc文件保存
#gpu driver
export CUDA_HOME=/usr/local/cuda
export PATH=/usr/local/cuda/bin:$PATH
export LD_LIBRARY_PATH=/usr/local/cuda/lib64:$LD_LIBRARY_PATH
export LD_LIBRARY_PATH="/usr/local/cuda/lib:${LD_LIBRARY_PATH}"
```

环境变量生效

source ~/.bashrc

查看安装结果



```
[root@localhost ~]# nvidia-smi
Tue Aug 14 23:58:44 2018
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 390.77                 Driver Version: 390.77                    |
|-------------------------------+----------------------+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|===============================+======================+======================|
|   0  GeForce GTX 108...  Off  | 00000000:03:00.0 Off |                  N/A |
|  0%   21C    P0    24W / 250W |      0MiB / 11178MiB |      0%      Default |
+-------------------------------+----------------------+----------------------+
|   1  GeForce GTX 108...  Off  | 00000000:84:00.0 Off |                  N/A |
|  0%   27C    P5    20W / 250W |      0MiB / 11178MiB |      0%      Default |
+-------------------------------+----------------------+----------------------+

+-----------------------------------------------------------------------------+
| Processes:                                                       GPU Memory |
|  GPU       PID   Type   Process name                             Usage      |
|=============================================================================|
|  No running processes found                                                 |
+-----------------------------------------------------------------------------+
```

勤菜鸟

3 faiss-gpu源码包编译

faiss-gpu源码包编译前必须保证openblas已经安装并且软链接已经创建。

GPU源码编译

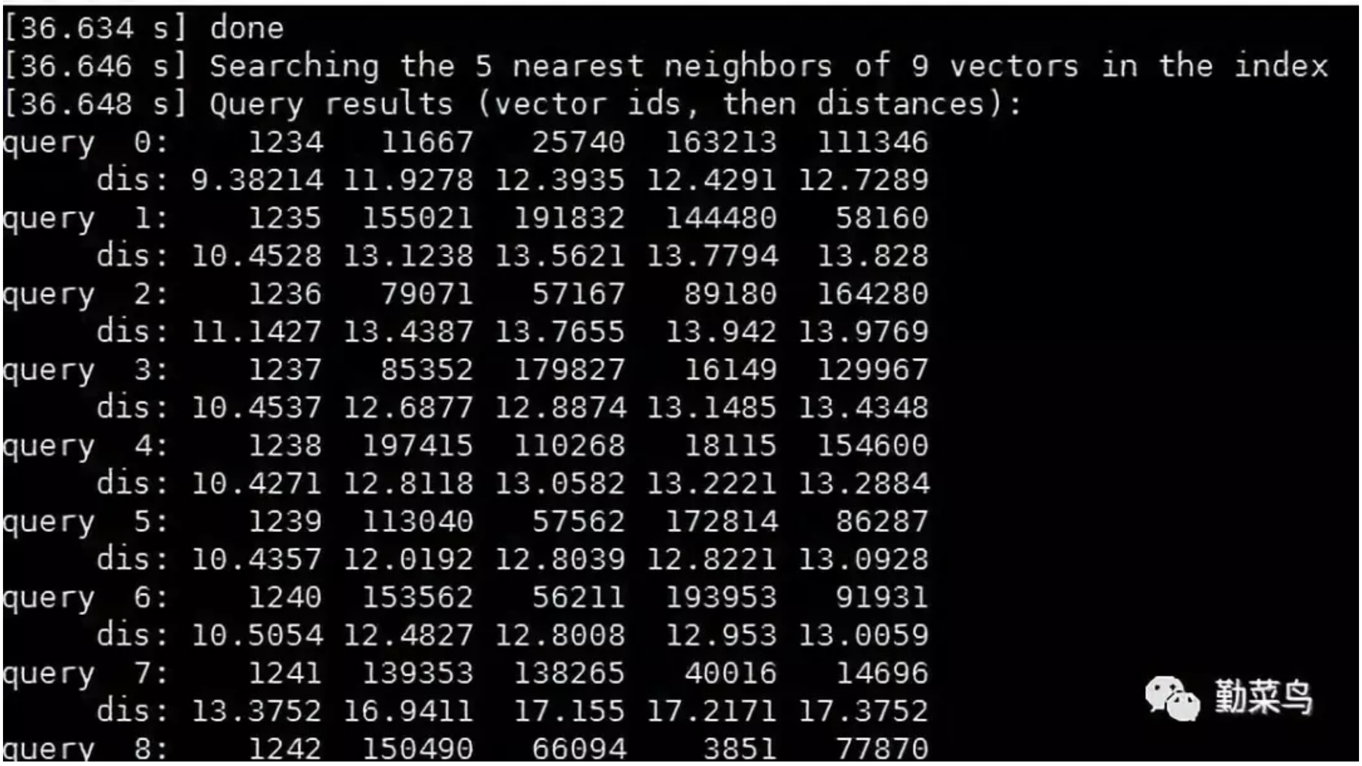
进入faiss目录，并运行 cd gpu

make

编译完成后，进行示例代码测试：

make test/demo_ivfpq_indexing_gpu

./test/demo_ivfpq_indexing_gpu



```
[36.634 s] done
[36.646 s] Searching the 5 nearest neighbors of 9 vectors in the index
[36.648 s] Query results (vector ids, then distances):
query  0:    1234   11667   25740  163213  111346
     dis: 9.38214 11.9278 12.3935 12.4291 12.7289
query  1:    1235  155021  191832  144480   58160
     dis: 10.4528 13.1238 13.5621 13.7794  13.828
query  2:    1236   79071   57167   89180  164280
     dis: 11.1427 13.4387 13.7655  13.942 13.9769
query  3:    1237   85352  179827   16149  129967
     dis: 10.4537 12.6877 12.8874 13.1485 13.4348
query  4:    1238  197415  110268   18115  154600
     dis: 10.4271 12.8118 13.0582 13.2221 13.2884
query  5:    1239  113040   57562  172814   86287
     dis: 10.4357 12.0192 12.8039 12.8221 13.0928
query  6:    1240  153562   56211  193953   91931
     dis: 10.5054 12.4827 12.8008  12.953 13.0059
query  7:    1241  139353  138265   40016   14696
     dis: 13.3752 16.9411  17.155 17.2171 17.3752
query  8:    1242  150490   66094    3851   77870
```

faiss-gpu python接口编译

cd gpu

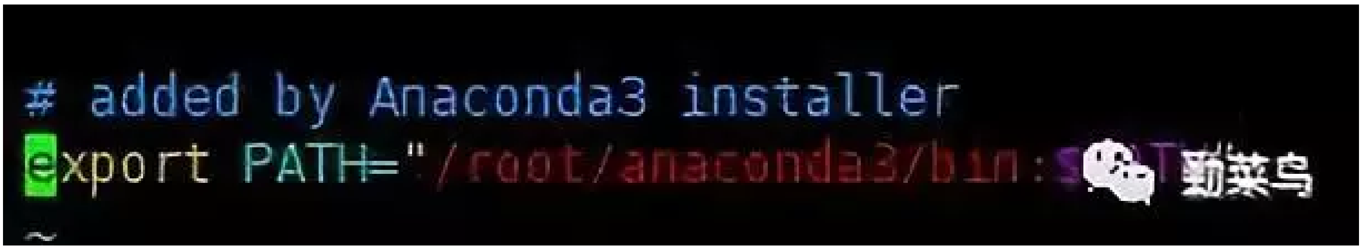
make py

测试

python –c “import faiss”

4 conda安装faiss

Anaconda安装faiss的cpu和gpu版本比较简单，但是值得注意的是，经小编实测，在anconda仓库中安装的最新代码只能在anconda3上正常运行，在anconda2上面经测试不能正常运行。也就是说，最新的anconda上的faiss包对python2支持的并不好，所以这里我们安装anconda3并配置如下anconda环境变量。



```
# added by Anaconda3 installer
export PATH="/root/anaconda3/bin:$PATH"
~
```

Openblas的安装和软链接创建同CPU版本，这里不再赘述。

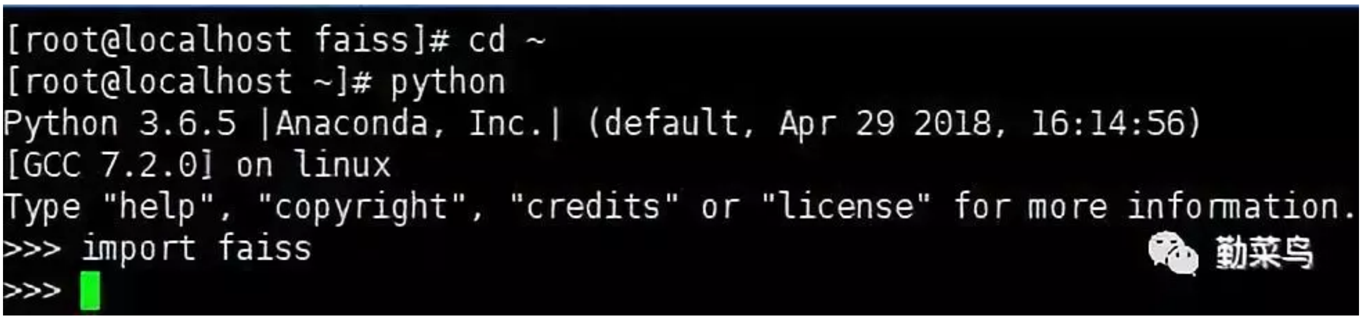
faiss-cpu版本安装

conda install faiss-cpu -c pytorch

faiss-gpu版本安装（默认cuda-8.0版本）

conda install faiss-gpu -c pytorch

测试

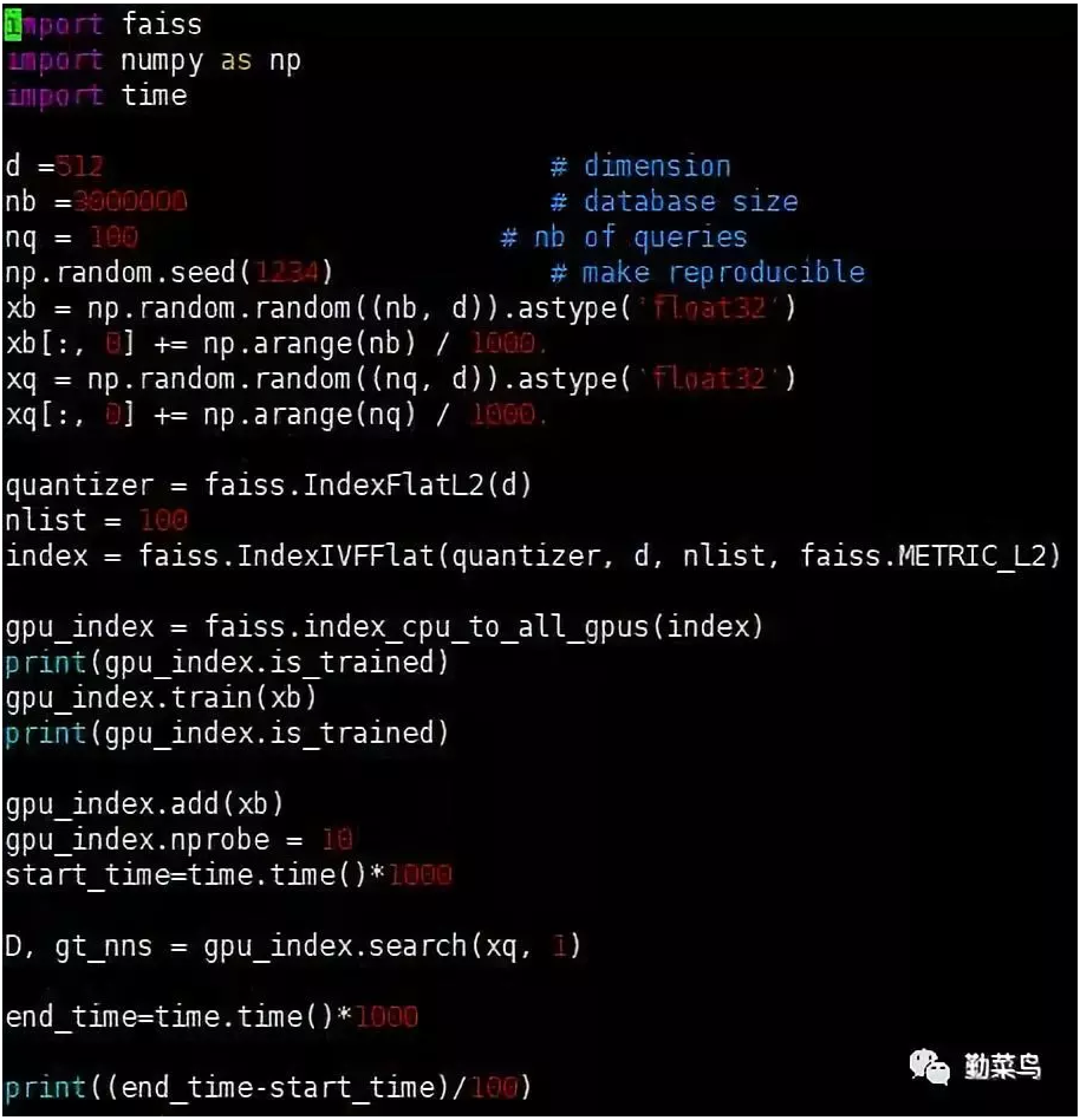


```
[root@localhost faiss]# cd ~
[root@localhost ~]# python
Python 3.6.5 |Anaconda, Inc.| (default, Apr 29 2018, 16:14:56)
[GCC 7.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import faiss
>>> 
```

经anconda安装后faiss可直接在python中像其他库一样使用，而不必像源码包编译完成后一样只能在faiss的编译文件目录下运行。

5 faiss-gpu入门实例

小编用如下代码测试faiss-gpu代码。



```
import faiss
import numpy as np
import time

d =512                                  # dimension
nb =3000000                             # database size
nq = 100                            # nb of queries
np.random.seed(1234)                    # make reproducible
xb = np.random.random((nb, d)).astype('float32')
xb[:, 0] += np.arange(nb) / 1000.
xq = np.random.random((nq, d)).astype('float32')
xq[:, 0] += np.arange(nq) / 1000.

quantizer = faiss.IndexFlatL2(d)
nlist = 100
index = faiss.IndexIVFFlat(quantizer, d, nlist, faiss.METRIC_L2)

gpu_index = faiss.index_cpu_to_all_gpus(index)
print(gpu_index.is_trained)
gpu_index.train(xb)
print(gpu_index.is_trained)

gpu_index.add(xb)
gpu_index.nprobe = 10
start_time=time.time()*1000

D, gt_nns = gpu_index.search(xq, 1)

end_time=time.time()*1000

print((end_time-start_time)/100)
```

在小编双显卡环境下，用IVFFlat检索图中所示数据，相似最近邻单次检索的平均时长仅为0.197ms，小编不得不感慨GPU强大的计算能力。

Reference

1. https://blog.csdn.net/u010641294/article/details/72820772
2. https://github.com/facebookresearch/faiss/blob/master/INSTALL.md
3. https://blog.csdn.net/u013378306/article/details/69229919
4. https://github.com/facebookresearch/faiss/wiki/Running-on-GPUs