# Nvidia Cudatoolkit vs Conda Cudatoolkit

Asked 2 years, 6 months ago  Modified 15 days ago  Viewed 42k times

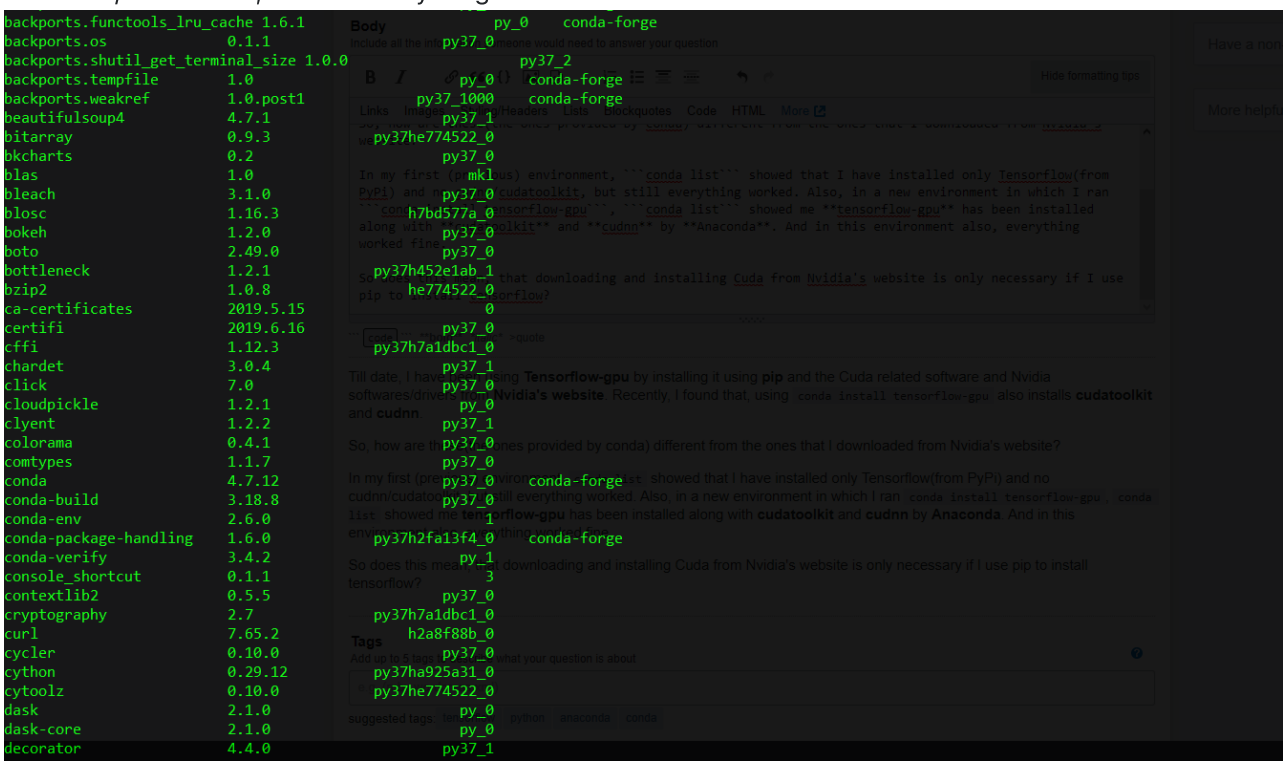
38

24

Till date, I have been using **Tensorflow-GPU** by installing it using **pip** and the Cuda related software and Nvidia softwares/drivers from **Nvidia's website**. Recently, I found that using `conda install tensorflow-gpu` also installs **cudatoolkit** and **cudnn**.

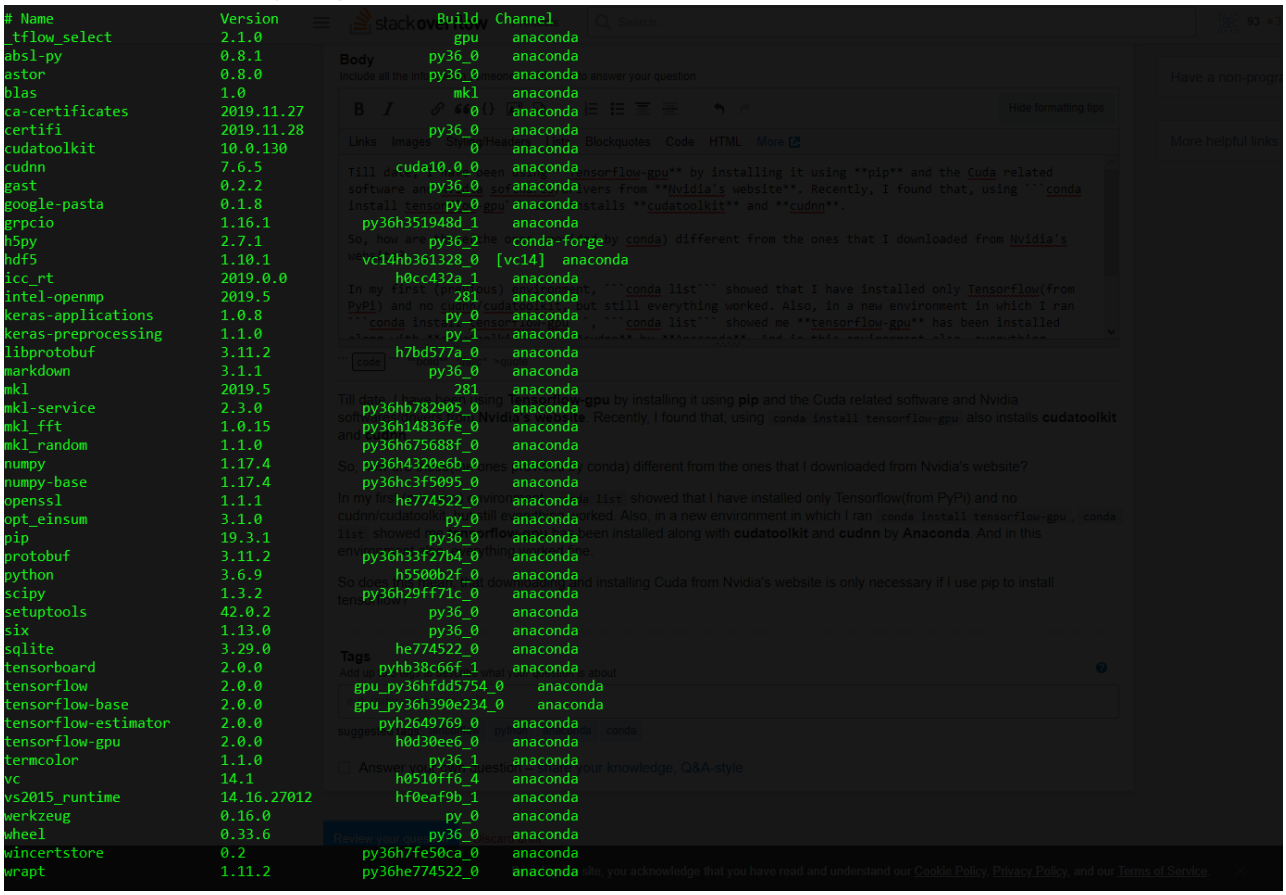
So, how are these(the ones provided by conda) different from the ones that I downloaded from Nvidia's website?

In my first (previous) environment, `conda list` showed that I have installed only TensorFlow(from PyPi) and no cudnn/cudatoolkit, but still everything worked.




Also, in a new environment in which I ran `conda install tensorflow-gpu`, `conda list` showed me **tensorflow-gpu** has been installed along with **cudatoolkit** and **cudnn** by **Anaconda**. And in this

environment also, everything worked fine.



```
# Name                    Version           Build  Channel
_tflow_select             2.1.0                gpu  anaconda
absl-py                   0.8.1            py36_0  anaconda
astor                     0.8.0            py36_0  anaconda
blas                      1.0                 mkl  anaconda
ca-certificates           2019.11.27           0  anaconda
certifi                   2019.11.28       py36_0  anaconda
cudatoolkit               10.0.130             0  anaconda
cudnn                     7.6.5          cuda10.0_0  anaconda
gast                      0.2.2            py36_0  anaconda
google-pasta              0.1.8               py_0  anaconda
grpcio                    1.16.1       py36h351948d_1  anaconda
h5py                      2.7.1            py36_2  conda-forge
hdf5                      1.10.1          vc14hb361328_0  [vc14]  anaconda
icc_rt                    2019.0.0            h0cc432a_1  anaconda
intel-openmp              2019.5                 281  anaconda
keras-applications        1.0.8               py_0  anaconda
keras-preprocessing       1.1.0               py_1  anaconda
libprotobuf               3.11.2           h7bd577a_0  anaconda
markdown                  3.1.1            py36_0  anaconda
mkl                       2019.5                 281  anaconda
mkl-service               2.3.0       py36hb782905_0  anaconda
mkl_fft                   1.0.15      py36h14836fe_0  anaconda
mkl_random                1.1.0       py36h675688f_0  anaconda
numpy                     1.17.4      py36h4320e6b_0  anaconda
numpy-base                1.17.4      py36hc3f5095_0  anaconda
openssl                   1.1.1            he774522_0  anaconda
opt_einsum                3.1.0               py_0  anaconda
pip                       19.3.1           py36_0  anaconda
protobuf                  3.11.2      py36h33f27b4_0  anaconda
python                    3.6.9                h5500b2f_0  anaconda
scipy                     1.3.2       py36h29ff71c_0  anaconda
setuptools                42.0.2           py36_0  anaconda
six                       1.13.0           py36_0  anaconda
sqlite                    3.29.0           he774522_0  anaconda
tensorboard               2.0.0          pyhb38c66f_1  anaconda
tensorflow                2.0.0          gpu_py36hfdd5754_0  anaconda
tensorflow-base           2.0.0          gpu_py36h390e234_0  anaconda
tensorflow-estimator      2.0.0             pyh2649769_0  anaconda
tensorflow-gpu            2.0.0                h0d30ee6_0  anaconda
termcolor                 1.1.0            py36_1  anaconda
vc                        14.1                 h0510ff6_4  anaconda
vs2015_runtime            14.16.27012          hf0eaf9b_1  anaconda
werkzeug                  0.16.0              py_0  anaconda
wheel                     0.33.6           py36_0  anaconda
wincertstore              0.2         py36h7fe50ca_0  anaconda
wrapt                     1.11.2      py36he774522_0  anaconda
```

So does this mean, that downloading and installing Cuda from Nvidia's website is **only** necessary if I use pip to install TensorFlow?

python tensorflow anaconda conda

Share Follow

edited Dec 30, 2019 at 11:14

asked Dec 30, 2019 at 11:07

Rajdeep Dutta

646 ● 1 ● 6 ● 15

2 Answers

Sorted by:

Trending sort available

Highest score (default)

41

If using anaconda to install tensorflow-gpu, yes it will install cuda and cudnn for you in same conda environment as tensorflow-gpu. All you need to install yourself is the latest nvidia-driver (so that it works with the latest CUDA level and all older CUDA levels you use.)

This has many advantages over the pip install tensorflow-gpu method:

1. Anaconda will always install the CUDA and CuDNN version that the TensorFlow code was compiled to use.
2. You can have multiple conda environments with different levels of TensorFlow, CUDA, and CuDNN and just use conda activate to switch between them.
3. You don't have to deal with installing CUDA and cuDNN manaually at the system wide level.

The disadvantage when compared to pip install tensorflow-gpu, is the latest version of tensorflow is added to pypi weeks before Anaconda is able to update the conda recipe and publish their builds of the latest

TensorFlow version.

Share Follow

answered Jan 27, 2020 at 23:29

William D. Irons

2,024 ● 1 ● 17 ● 19

3 Do you have any idea what problems could occur if CUDA is installed on the host os? (outside of the conda env) – filip Sep 11, 2020 at 9:41

2 Anaconda should ignore any version of CUDA outside of the conda env. It shouldn't be able to find it. – William D. Irons Sep 11, 2020 at 16:45

I didn't have any errors. having both installed. I'm not sure if it gets ignored though. – Paul Totzke Oct 28, 2020 at 22:46

1 I never had errors installing CUDA outside conda. @william Between the two methods you suggested, which would offer better performance? – Govarthenan Rajadurai Jun 19, 2021 at 6:15

1 I wouldn't expect how CUDA was installed to affect performance. Whatever version offers the most recent version of CUDA I would expect to have the best performance. – William D. Irons Jun 19, 2021 at 17:47

0

Nvidia now has official channel for conda. the package name is nvidia/cuda. I prefer conda for easier managing different cuda enviroment.

What I found to be missing in `conda-forge/cudatoolkit` is nvcc and I guess some other utils for compiling but not running enviroment is also missing.

this post say `conda-forge/cudatoolkit-dev` will install nvcc, but I haven't tried.

Share Follow

answered Jun 22 at 3:41

tothedistance

31 ● 4