

Installing PyTorch Framework

When you first login to Carey, a Python 3.6 virtual environment will be automatically generated and activated. This allows you to manage your own python packages without needing root or sudo access.

Installing PyTorch is as simple as running the following:

```
module load nvidia/cuda-11.0
module load nvidia/cudnn-v8.0.180-forcuda11.0

pip install --upgrade pip

SRC=https://download.pytorch.org/whl/torch_stable.html

pip install torch==1.7.1+cu110 torchvision==0.8.2+cu110 -f $SRC
```

We can then test that this has been installed by importing that package. If we don't get an error, PyTorch was installed successfully,

```
[pszaj@carey ~]$ python -c 'import torch'
[pszaj@carey ~]$
```

We can start an interactive Python shell and test whether CUDA and CuDNN are available:

```
[pszaj@carey ~]$ srun -q csug --gres gpu --pty python
srun: No --mem parameter used. Setting to 6GB
Python 3.6.8 (default, Aug 7 2019, 17:28:10)
[GCC 4.8.5 20150623 (Red Hat 4.8.5-39)] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import torch
>>> torch.cuda.is_available()
True
>>> torch.backends.cudnn.enabled
True
>>> torch.backends.cudnn.version()
7603
>>>
```

When calling srun, we explicitly requested a Generic Resource (GRes). We can request more GPUs than one, but we don't need that right now. As you can see, CUDA is available to Torch.

Let's begin by running a PyTorch MNIST example:

```
mkdir mnist
cd mnist
wget https://raw.githubusercontent.com/pytorch/examples/master/mnist/main.py
```

It may be necessary to downgrade Pillow as the latest version doesn't work with PyTorch.

```
pip3 install pillow==6.1
```

We also need to produce a job submission file, such 'train.sh' with the following contents:

```
#!/bin/bash

#SBATCH -p cs -A cs -q csug
#SBATCH -c4 --mem=4g
#SBATCH --gres gpu:1

module load nvidia/cuda-11.0
module load nvidia/cudnn-v8.0.180-forcuda11.0

python main.py --batch-size 256 --epochs 5
```

This script can then be submitted with 'sbatch train.sh'. The job will redirect all its output to a file called 'slurm-X.out', where X is the Job ID, and you can show the output from this file using 'cat slurm-X.out'.

We are using a large batch size of 256 and have limited the number of epochs to 5 to avoid congestion on the cluster during the lab session. The script should finish running after a couple of minutes.

After just a few training epochs, the accuracy should be at least 99%.

Document History

Date	Author	Description
31 March 2021 (current)	Aaron Jackson	Updated install guide to PyTorch 1.7.1 using CUDA 11.0 since the GPU drivers have now been updated.
19 June 2020	Aaron Jackson	Referenced specific module versions using “module load” and referenced a specific version of PyTorch. This should allow the installation guide to be relevant for much longer, in case new versions are released.
24 January 2020	Aaron Jackson	Initial version