1. **Install the following Environments in your HPCE account**

Channels: anaconda, pytorch,nvidia

Dependencies: python 3.6, pip, numpy, cudatoolkit, pytorch, torchvision,nltk,tensorboardX, tensorflow, git-lfs,tensorboard,pyyaml, conditional

1. **Create your own environment in the scratch.** For example, ‘myenv’ is created. Activate it using the command **conda activate myenv**
2. I have used the code base from https://github.com/07Agarg/HAF. Clone it in the scratch using the commands:

cd scratch

git clone https://github.com/07Agarg/HAF.git

cd HAF

1. **To Create and modify/edit any file**

nano <filename.extension>

ex: nano example.py

Then save the file by entering the required alphabets shown at the end of the file.

Here we use ctrl+x and y.

1. **To Read the existing file**

cat <filename.extension>

ex: cat example.py

1. **Run the code file**
2. If code doesn’t use GPU, then simply use the command: python3 <filename.py>
3. Else: We need to create a script file and run it.
4. **Creating a script file:**

nano <scriptfilename.sh>

Enter the below details in the script file

#!/bin/bash

#PBS -e <errorfilename.err>

#PBS -o <logfilename.log>

#PBS -l select=1:ncpus=1:ngpus=<n.o of gpus required>

#PBS -q gpuq

cd /lfs/usrhome/mtech/<RollNo>/scratch/<filefolder>

export PYTHONPATH=/lfs/usrhome/mtech/<RollNo>/scratch/<filefolder> && export LD\_LIBRARY\_PATH=/lfs/sware/cuda-10.1 && export CUDA\_VISIBLE\_DEVICES=0,1

<command to either run the file or download the data or some other command which makes use of GPU>

**Then save the file using the commands – ‘ctrl+x’ followed by ‘y’**

Below is the Sample script file to run a python file which is in the folder HAF of scratch

#!/bin/bash

#PBS -e errorfile.err

#PBS -o logfile.log

#PBS -l select=1:ncpus=1:ngpus=2

#PBS -q gpuq

cd /lfs/usrhome/mtech/cs21m053/scratch/HAF

export PYTHONPATH=/lfs/usrhome/mtech/cs21m053/scratch/HAF && export LD\_LIBRARY\_PATH=/lfs/sware/cuda-10.1 && export CUDA\_VISIBLE\_DEVICES=0,1

/lfs/usrhome/mtech/cs21m011/.conda/myenv/tabvcr/bin/python3 main.py --start testing --arch resnet50 --loss cross-entropy --optimizer adam --data inaturalist19-224 --workers 16 --output out/inat/cross-entropy

Here errorfile.err shows the errors generated while running the code.

logfile.log show the results generated which are accuracy values or loss values.

Make sure you change the names of these files for each script else rewrites the existing file and we will not be able to see the previous values.

To view the errorfile.err / logfile.log. Enter the command cat errorfile.err / cat logfile.log respectively.

1. **Run the script file**

qsub <scriptfile.sh>

the command qsub submits the job in the hpce. After submitting a job is created.

1. **To check the job status**

qstat

this command shows the number of jobs you have submitted and the time taken till then.

1. **Delete the job**

qdel <jobname.hn1>