DL Cheat Sheet for working on HPC environment

Note: The following Instruction are installed with support of NCCL2(build from source), Openmpi(4.0.3v) on RedHatEnterpriseServer(Owen cluster)

Checking server systems(OS), architecture, network, GPU, and all hardware specifications:

```
(tf_latest) [dong760@owens-login02 ~]$ lsb_release -a
LSB Version:
:core-4.1-amd64:core-4.1-noarch:cxx-4.1-amd64:cxx-4.1-noarch:desktop-4.1-amd64:deskto
p-4.1-noarch:languages-4.1-amd64:languages-4.1-noarch:printing-4.1-amd64:printing-4.1-n
oarch
Distributor ID: RedHatEnterpriseServer
Description:    Red Hat Enterprise Linux Server release 7.7 (Maipo)
Release:        7.7
Codename:       Maipo
(tf_latest) [dong760@o0801 ~]$ nvidia-smi
Fri Nov 13 22:59:56 2020
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 440.64.00    Driver Version: 440.64.00    CUDA Version: 10.2     |
|-------------------------------+----------------------+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|===============================+======================+=============
========|
|   0  Tesla P100-PCIE...  On   | 00000000:04:00.0 Off |                    0 |
| N/A   25C    P0    25W / 250W |     10MiB / 16280MiB |      0%   E. Process |
+-------------------------------+----------------------+----------------------+

+-----------------------------------------------------------------------------+
| Processes:                                                       GPU Memory |
|  GPU       PID   Type   Process name                             Usage      |
|=============================================================================
========|
|  No running processes found                                                 |
+-----------------------------------------------------------------------------+
# Verify You Have a CUDA-Capable GPU
(https://docs.nvidia.com/cuda/cuda-installation-guide-linux/index.html)
$ lspci | grep -i nvidia
# Check CPU driver and usage
$ cat /proc/cpuinfo
```

# Check networking driver and usage

```
$ ifconfig
```

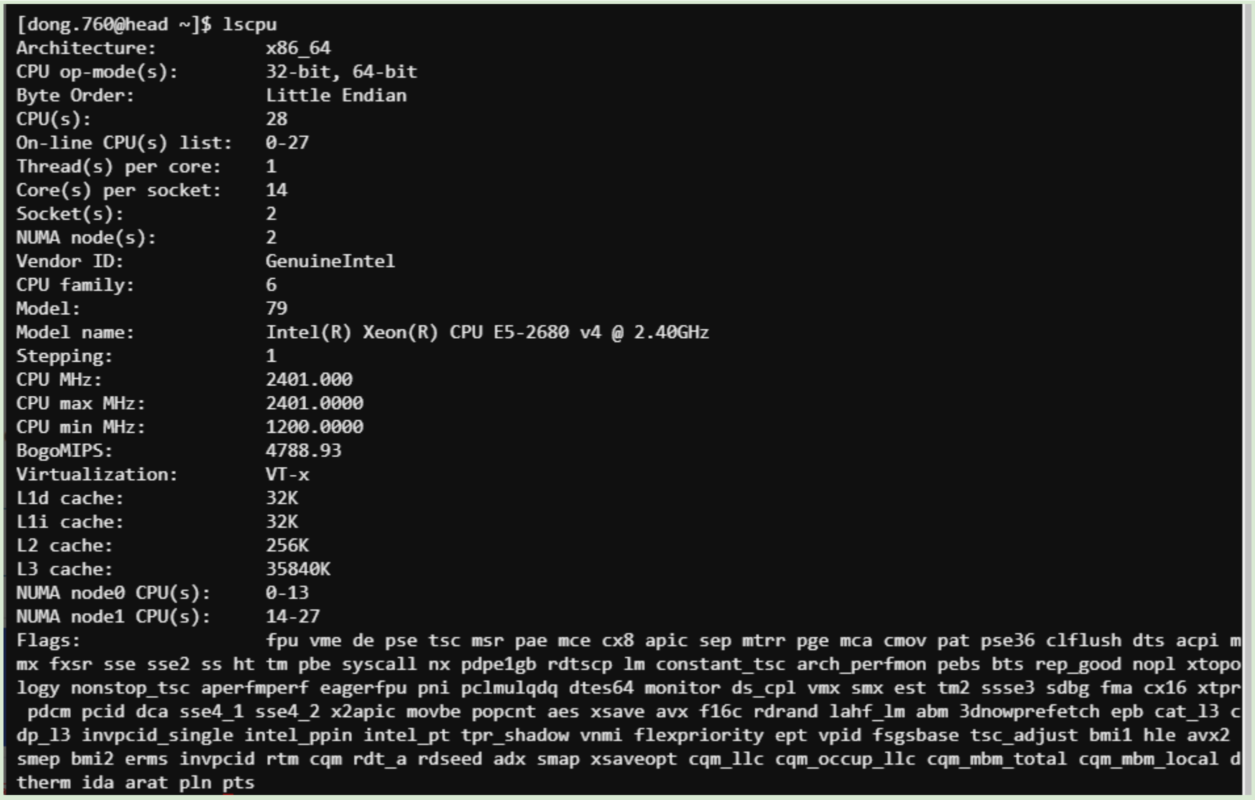
# Check the current ip address

```
(dask-tutorial) [dong760@p0002 dask]$ hostname -I | awk '{print $1}'
10.4.1.5
```

# Check the CPU information

```
$ lscpu
```

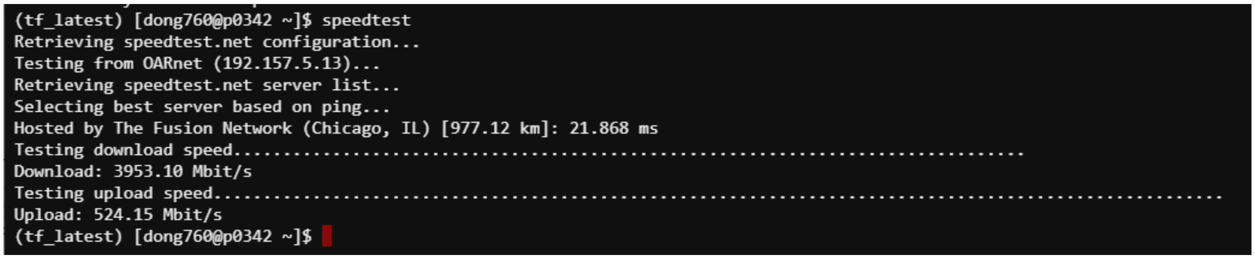
# Check the networking speed with python tool(Reference: https://pypi.org/project/speedtest-cli/):



```
[dong.760@head ~]$ lscpu
Architecture:          x86_64
CPU op-mode(s):        32-bit, 64-bit
Byte Order:            Little Endian
CPU(s):                28
On-line CPU(s) list:   0-27
Thread(s) per core:    1
Core(s) per socket:    14
Socket(s):             2
NUMA node(s):          2
Vendor ID:             GenuineIntel
CPU family:            6
Model:                 79
Model name:            Intel(R) Xeon(R) CPU E5-2680 v4 @ 2.40GHz
Stepping:              1
CPU MHz:               2401.000
CPU max MHz:           2401.0000
CPU min MHz:           1200.0000
BogoMIPS:              4788.93
Virtualization:        VT-x
L1d cache:             32K
L1i cache:             32K
L2 cache:              256K
L3 cache:              35840K
NUMA node0 CPU(s):     0-13
NUMA node1 CPU(s):     14-27
Flags:                 fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush dts acpi m
mx fxsr sse sse2 ss ht tm pbe syscall nx pdpe1gb rdtscp lm constant_tsc arch_perfmon pebs bts rep_good nopl xtopo
logy nonstop_tsc aperfmperf eagerfpu pni pclmulqdq dtes64 monitor ds_cpl vmx smx est tm2 ssse3 sdbg fma cx16 xtpr
 pdcm pcid dca sse4_1 sse4_2 x2apic movbe popcnt aes xsave avx f16c rdrand lahf_lm abm 3dnowprefetch epb cat_l3 c
dp_l3 invpcid_single intel_ppin intel_pt tpr_shadow vnmi flexpriority ept vpid fsgsbase tsc_adjust bmi1 hle avx2
smep bmi2 erms invpcid rtm cqm rdt_a rdseed adx smap xsaveopt cqm_llc cqm_occup_llc cqm_mbm_total cqm_mbm_local d
therm ida arat pln pts
```

```
pip install speedtest-cli
```

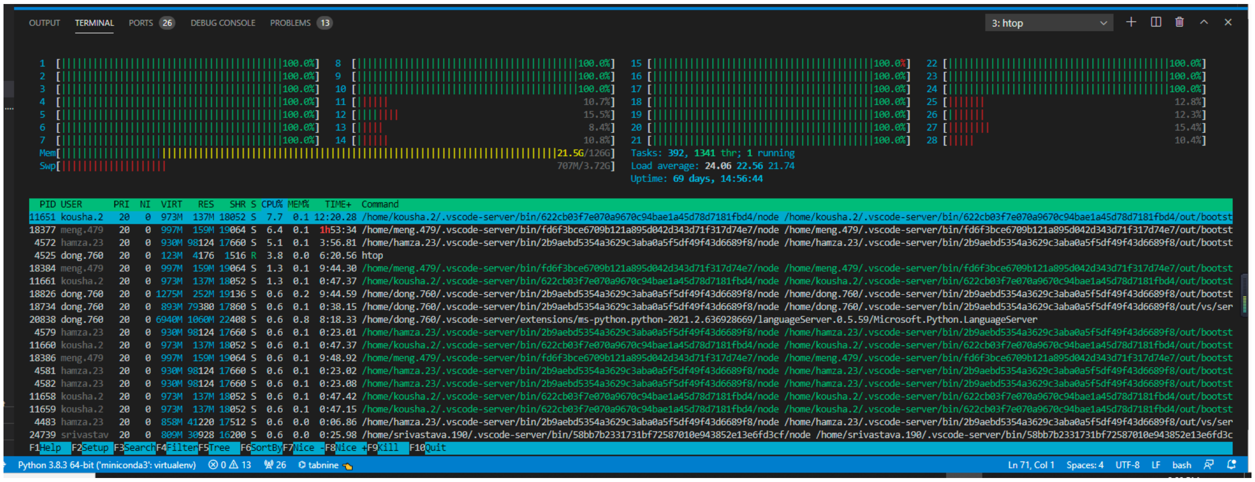
# Check current CPU/Memory usage:



```
(tf_latest) [dong760@p0342 ~]$ speedtest
Retrieving speedtest.net configuration...
Testing from OARnet (192.157.5.13)...
Retrieving speedtest.net server list...
Selecting best server based on ping...
Hosted by The Fusion Network (Chicago, IL) [977.12 km]: 21.868 ms
Testing download speed................................................................................
Download: 3953.10 Mbit/s
Testing upload speed......................................................................................................
Upload: 524.15 Mbit/s
(tf_latest) [dong760@p0342 ~]$
```

```
$ htop
```

(Or top without visualization)



```
  1 [||||||||||||||||||||||||||||||||||||||||100.0%]   8 [||||||||||||||||||||||||||||||||||||||||100.0%]   15 [||||||||||||||||||||||||||||||||||||||||100.0%]   22 [||||||||||||||||||||||||||||||||||||||||100.0%]
  2 [||||||||||||||||||||||||||||||||||||||||100.0%]   9 [||||||||||||||||||||||||||||||||||||||||100.0%]   16 [||||||||||||||||||||||||||||||||||||||||100.0%]   23 [||||||||||||||||||||||||||||||||||||||||100.0%]
  3 [||||||||||||||||||||||||||||||||||||||||100.0%]  10 [||||||||||||||||||||||||||||||||||||||||100.0%]   17 [||||||||||||||||||||||||||||||||||||||||100.0%]   24 [||||||||||||||||||||||||||||||||||||||||100.0%]
  4 [||||||||||||||||||||||||||||||||||||||||100.0%]  11 [|||||                                    10.7%]   18 [||||||||||||||||||||||||||||||||||||||||100.0%]   25 [|||||||                                  12.8%]
  5 [||||||||||||||||||||||||||||||||||||||||100.0%]  12 [|||||||                                  15.5%]   19 [||||||||||||||||||||||||||||||||||||||||100.0%]   26 [|||||||                                  12.3%]
  6 [||||||||||||||||||||||||||||||||||||||||100.0%]  13 [||||                                      8.4%]   20 [||||||||||||||||||||||||||||||||||||||||100.0%]   27 [||||||||                                 15.4%]
  7 [||||||||||||||||||||||||||||||||||||||||100.0%]  14 [|||||                                    10.8%]   21 [||||||||||||||||||||||||||||||||||||||||100.0%]   28 [|||||                                    10.4%]
  Mem[||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||21.5G/126G]   Tasks: 392, 1341 thr; 1 running
  Swp[|||||||||||||||||||                                                             707M/3.72G]   Load average: 24.06 22.56 21.74
                                                                                                    Uptime: 69 days, 14:56:44

  PID USER      PRI  NI  VIRT   RES   SHR S CPU% MEM%   TIME+  Command
11651 kousha.2   20   0  973M  137M 18052 S  7.7  0.1 12:20.28 /home/kousha.2/.vscode-server/bin/622cb03f7e070a9670c94bae1a45d78d7181fbd4/node /home/kousha.2/.vscode-server/bin/622cb03f7e070a9670c94bae1a45d78d7181fbd4/out/bootst
18377 meng.479   20   0  997M  159M 19064 S  6.4  0.1  1h53:34 /home/meng.479/.vscode-server/bin/fd6f3bce6709b121a895d042d343d71f317d74e7/node /home/meng.479/.vscode-server/bin/fd6f3bce6709b121a895d042d343d71f317d74e7/out/bootst
 4572 hamza.23   20   0  930M 98124 17660 S  5.1  0.1  3:56.81 /home/hamza.23/.vscode-server/bin/2b9aebd5354a3629c3aba0a5f5df49f43d6689f8/node /home/hamza.23/.vscode-server/bin/2b9aebd5354a3629c3aba0a5f5df49f43d6689f8/out/bootst
 4525 dong.760   20   0  123M  4176  1516 R  3.8  0.0  6:20.56 htop
18384 meng.479   20   0  997M  159M 19064 S  1.3  0.1  9:44.30 /home/meng.479/.vscode-server/bin/fd6f3bce6709b121a895d042d343d71f317d74e7/node /home/meng.479/.vscode-server/bin/fd6f3bce6709b121a895d042d343d71f317d74e7/out/bootst
11661 kousha.2   20   0  973M  137M 18052 S  1.3  0.1  0:47.37 /home/kousha.2/.vscode-server/bin/622cb03f7e070a9670c94bae1a45d78d7181fbd4/node /home/kousha.2/.vscode-server/bin/622cb03f7e070a9670c94bae1a45d78d7181fbd4/out/bootst
18826 dong.760   20   0 1275M  252M 19136 S  0.6  0.2  9:44.59 /home/dong.760/.vscode-server/bin/2b9aebd5354a3629c3aba0a5f5df49f43d6689f8/node /home/dong.760/.vscode-server/bin/2b9aebd5354a3629c3aba0a5f5df49f43d6689f8/out/bootst
18734 dong.760   20   0  893M 79380 17860 S  0.6  0.1  0:38.15 /home/dong.760/.vscode-server/bin/2b9aebd5354a3629c3aba0a5f5df49f43d6689f8/node /home/dong.760/.vscode-server/bin/2b9aebd5354a3629c3aba0a5f5df49f43d6689f8/out/vs/ser
20838 dong.760   20   0 6940M 1060M 22408 S  0.6  0.8  8:18.33 /home/dong.760/.vscode-server/extensions/ms-python.python-2021.2.636928669/languageServer.0.5.59/Microsoft.Python.LanguageServer
 4579 hamza.23   20   0  930M 98124 17660 S  0.6  0.1  0:23.01 /home/hamza.23/.vscode-server/bin/2b9aebd5354a3629c3aba0a5f5df49f43d6689f8/node /home/hamza.23/.vscode-server/bin/2b9aebd5354a3629c3aba0a5f5df49f43d6689f8/out/bootst
11660 kousha.2   20   0  973M  137M 18052 S  0.6  0.1  0:47.37 /home/kousha.2/.vscode-server/bin/622cb03f7e070a9670c94bae1a45d78d7181fbd4/node /home/kousha.2/.vscode-server/bin/622cb03f7e070a9670c94bae1a45d78d7181fbd4/out/bootst
18386 meng.479   20   0  997M  159M 19064 S  0.6  0.1  9:48.92 /home/meng.479/.vscode-server/bin/fd6f3bce6709b121a895d042d343d71f317d74e7/node /home/meng.479/.vscode-server/bin/fd6f3bce6709b121a895d042d343d71f317d74e7/out/bootst
 4581 hamza.23   20   0  930M 98124 17660 S  0.6  0.1  0:23.02 /home/hamza.23/.vscode-server/bin/2b9aebd5354a3629c3aba0a5f5df49f43d6689f8/node /home/hamza.23/.vscode-server/bin/2b9aebd5354a3629c3aba0a5f5df49f43d6689f8/out/bootst
 4582 hamza.23   20   0  930M 98124 17660 S  0.6  0.1  0:23.08 /home/hamza.23/.vscode-server/bin/2b9aebd5354a3629c3aba0a5f5df49f43d6689f8/node /home/hamza.23/.vscode-server/bin/2b9aebd5354a3629c3aba0a5f5df49f43d6689f8/out/bootst
11658 kousha.2   20   0  973M  137M 18052 S  0.6  0.1  0:47.42 /home/kousha.2/.vscode-server/bin/622cb03f7e070a9670c94bae1a45d78d7181fbd4/node /home/kousha.2/.vscode-server/bin/622cb03f7e070a9670c94bae1a45d78d7181fbd4/out/bootst
11659 kousha.2   20   0  973M  137M 18052 S  0.6  0.1  0:47.15 /home/kousha.2/.vscode-server/bin/622cb03f7e070a9670c94bae1a45d78d7181fbd4/node /home/kousha.2/.vscode-server/bin/622cb03f7e070a9670c94bae1a45d78d7181fbd4/out/bootst
 4483 hamza.23   20   0  858M 41220 17512 S  0.6  0.0  0:06.86 /home/hamza.23/.vscode-server/bin/2b9aebd5354a3629c3aba0a5f5df49f43d6689f8/node /home/hamza.23/.vscode-server/bin/2b9aebd5354a3629c3aba0a5f5df49f43d6689f8/out/vs/ser
24739 srivastav  20   0  809M 30928 16200 S  0.6  0.0  0:25.98 /home/srivastava.190/.vscode-server/bin/58bb7b2331731bf72587010e943852e13e6fd3cf/node /home/srivastava.190/.vscode-server/bin/58bb7b2331731bf72587010e943852e13e6fd3c
F1Help  F2Setup F3SearchF4FilterF5Tree  F6SortByF7Nice -F8Nice +F9Kill  F10Quit
```

# Killing a process: (Note: you need args “-9” to actually kill it. Don’t ask why, just do it)

$ kill -9 59623

Reference:

1. Horovod with MVAPICH2: http://hidl.cse.ohio-state.edu/userguide/horovod/
2. Horovod Installation Guide: https://github.com/horovod/horovod/blob/master/docs/install.rst
3. Horovod with MPI(The right way to run horovodrun): https://github.com/horovod/horovod/blob/master/docs/mpi.rst
4. NCCL Github(To build from source, with make): https://github.com/NVIDIA/nccl
5. NVIDIA Collective Communication Library (NCCL) Installation Guide, https://docs.nvidia.com/deeplearning/nccl/install-guide/index.html
6. How to Install NVIDIA Collective Communications Library (NCCL) 2 for TensorFlow on Ubuntu 16.04, https://tech.amikelive.com/node-735/how-to-install-nvidia-collective-communications-library-nccl-2-for-tensorflow-on-ubuntu-16-04/
7. How can I cope with 'Fail to determine if Gloo support has been built', https://github.com/horovod/horovod/issues/1831
8. 安装horovod（环境）的正确姿势https://zhuanlan.zhihu.com/p/108984987
9. mpi_lib.so: undefined symbol: _ZNK10tensorflow8OpKernel4nameEv on CPU nodes, https://github.com/horovod/horovod/issues/431
10.

Handy System related command: Here are some commands that might come to handy when working on some HPC environment (Here we use RI2 as example--think of as a computing center at OSU):

- Things to do, everytime you open the ri2:

export LANGUAGE=en_US.UTF-8

```
export LANG=en_US.UTF-8
export LC_ALL=en_US.UTF-8
```

- What is node: think of node as a machine, or running devices on remote server
- what is ri2: A cluster that have many nodes, you can find the available node with command:

```
 $ sinfo
```

- How to get the hostname of current allocated nodes?

```
$ hostname
gpu01.cluster
```

- How to get the hostname of all allocated notes? (Without their domain name)

```
$ srun hostname -a
gpu01
gpu02
```

(This will return all the hostname of the nodes that you allocated)

- How to check the current scheduling queue:

```
$ squeue
```

(Note: (PartitionTimeLimit): If you see these things, you will never get the node, because you cannot allocate an interactive node for more than 4 hours.)

- How to check the job that you requested:

```
$ squeue -u dong.760
```

- How to allocate a node:

```
$ salloc -N 2 -p sky-k80 -t 4:0:0
```

(Allocate 2 nodes from partition sky-k80 for 4 hours)

- How to get all the modules/packages that are available in ri2? (remember to use gcc/7.3.0, cuda/10.1 for our project)

```
$ module avail
```

- How to check the current loaded libraries?

```
$ module list
```

- How to load and unload modules?

```
$ module load cuda/10.1
$ module unload cuda/10.1
```

Instructions for installing Horovod with NCCL and Openmpi (You need to make sure there is MPI avail on system):

-- Check cluster traffic ---

https://ondemand.osc.edu/pun/sys/systemstatus/clusters

-- Grab a node --

Note: user command $sinfo first, to check the partition that is available to use (for e.g., has status idle)

```
Owens: qsub -I -l nodes=1:ppn=28:gpus=1 -l walltime=1:00:00 -A PAS1777
```

```
Pitzer: qsub -I -l nodes=1:ppn=40:gpus=2 -l walltime=1:00:00 -A PAS1777
OR
Pitzer: salloc --nodes 1 -p gpuserial --ntasks-per-node=40 --gpus-per-node=2 --time 1:00:00 --account=PAS1777
srun --pty --jobid <JOBID> /bin/bash
```

-- Load modules --

```
module load cuda/10.0.130 openmpi/4.0.3
or
module load cuda/10.1.168 # --> This is for running horovod and gloo, openmpi (maybe), and it can pass the command, $horovodrun --check-build
module load cuda/10.2.89 openmpi/4.0.3 # --> This is for mvapich, because your rpm was build with cuda10.2 (Be cautious of the cuda version you used to build rpm, NCCL, and TF)
```

# Expected Result:

```
(tf_latest) [dong760@o0674 ~]$ module list

Currently Loaded Modules:
  1) xalt/latest              3) intel/19.0.5     5) openmpi/4.0.3
  2) gcc-compatibility/8.4.0   4) modules/sp2020   6) cuda/10.0.130
```

-- Install Miniconda --

```
wget https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh
sh Miniconda3-latest-Linux-x86_64.sh -b -p $PWD/miniconda3
cd miniconda3
OR
wget http://mvapich.cse.ohio-state.edu/download/mvapich/gdr/2.3.4/mofed4.7/mvapich2-gdr-mcast.cuda10.2.mofed4.7.gnu8.4.0-2.3.4-1.el7.x86_64.rpm
echo 'environment set up'
export PATH=$PWD/miniconda3/bin:$PATH
export LD_LIBRARY_PATH=$PWD/lib:$LD_LIBRARY_PATH
export CPATH=$PWD/include:$CPATH
conda activate tf_latest
which conda
```

# Expected Result:

```
(tf_latest) [dong760@o0674 ~]$ which conda
~/miniconda3/condabin/conda
```

# Expected Result:

```
# Verify pip points to correct library.
$ which pip
```

```
(tf_latest) [dong760@o0674 ~]$ which pip
~/miniconda3/envs/tf_latest/bin/pip
```

-- Install tensorflow --

```
conda create -n tf_latest python=3.6.5
conda activate tf_latest
conda install tensorflow
pip install tensorflow-gpu==2.3.0
```

# Expected Result:

```
ipython
import tensorflow as tf
print(tf.__version__)
tf.test.gpu_device_name()
import horovod.tensorflow as hvd
```

```
In [2]: import tensorflow as tf
   ...: print(tf.__version__)
   ...:
2.3.0
In [3]: tf.test.gpu_device_name()
...
2020-11-13 20:52:39.701150: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1716]
Found device 0 with properties:
pciBusID: 0000:04:00.0 name: Tesla P100-PCIE-16GB computeCapability: 6.0
coreClock: 1.3285GHz coreCount: 56 deviceMemorySize: 15.90GiB
deviceMemoryBandwidth: 681.88GiB/s
...
```

-- Install PyTorch (for CUDA 10.2) [FOR PYTORCH DISTRIBUTED, YOU MUST BUILD FROM SOURCE] --

```
pip install torch torchvision
```

-- Get MPI and setup environment --

```
wget http://mvapich.cse.ohio-state.edu/download/mvapich/gdr/2.3.4/mofed4.5/mvapich2-gdr-mcast.cuda10.0.mofed4.5.gnu4.8.5-2.3.4-1.el7.x86_64.rpm
rpm2cpio mvapich2-gdr-mcast.cuda10.0.mofed4.5.gnu4.8.5-2.3.4-1.el7.x86_64.rpm | cpio -id
```

```
export MV2_PATH=/users/PAA0023/dong760/opt/mvapich2/gdr/2.3.4/mcast/no-openacc/cuda10.0/mofed4.5/mpirun/gnu4.8.5
OR
export MV2_PATH=/users/PAA0023/dong760/opt/mvapich2/gdr/2.3.4/mcast/no-openacc/cuda10.2/mofed4.7/mpirun/gnu8.4.0
export PATH=$MV2_PATH/bin:$PATH
export LD_LIBRARY_PATH=$MV2_PATH/lib64:$LD_LIBRARY_PATH
export CPATH=$MV2_PATH/include:$CPATH
export MV2_USE_CUDA=1
export MV2_SUPPORT_DL=1
```

--- Change the mpicc and mpicxx file

```
cd $MV2_PATH/bin
# Modify the file as follow:
prefix=/users/PAA0023/dong760/opt/mvapich2/gdr/2.3.4/mcast/no-openacc/cuda10.0/mofed4.5/mpirun/gnu4.8.5
exec_prefix=/users/PAA0023/dong760/opt/mvapich2/gdr/2.3.4/mcast/no-openacc/cuda10.0/mofed4.5/mpirun/gnu4.8.5
sysconfdir=/users/PAA0023/dong760/opt/mvapich2/gdr/2.3.4/mcast/no-openacc/cuda10.0/mofed4.5/mpirun/gnu4.8.5/etc
includedir=/users/PAA0023/dong760/opt/mvapich2/gdr/2.3.4/mcast/no-openacc/cuda10.0/mofed4.5/mpirun/gnu4.8.5/include
libdir=/users/PAA0023/dong760/opt/mvapich2/gdr/2.3.4/mcast/no-openacc/cuda10.0/mofed4.5/mpirun/gnu4.8.5/lib64
```

# Expected Result:

```
which mpicc
which mpicxx
(tf_latest) [dong760@o0674 tensorflow2]$ which mpicc
~/opt/mvapich2/gdr/2.3.4/mcast/no-openacc/cuda10.0/mofed4.5/mpirun/gnu4.8.5/bin/mpicc
(tf_latest) [dong760@o0674 tensorflow2]$ which mpicxx
~/opt/mvapich2/gdr/2.3.4/mcast/no-openacc/cuda10.0/mofed4.5/mpirun/gnu4.8.5/bin/mpicxx
```

-- Test your MPI installation --

```
### First create a hostfile for GPUs.
Pitzer: cat $PBS_NODEFILE | uniq | sed 'p;p' > hostfile
Owens: cat $PBS_NODEFILE | uniq > hostfile
```

```
### Test allreduce (DL collective) with 2 GPUs
$MV2_PATH/bin/mpirun_rsh -n 2 --hostfile=hostfile MV2_USE_CUDA=1
MV2_SUPPORT_DL=1
$MV2_PATH/libexec/osu-micro-benchmarks/mpi/collective/osu_allreduce
### Test point-to-point functionality with 2 GPUs
$MV2_PATH/bin/mpirun_rsh -n 2 --hostfile=hostfile MV2_USE_CUDA=1
MV2_SUPPORT_DL=1 $MV2_PATH/libexec/osu-micro-benchmarks/mpi/pt2pt/osu_latency
```

# Expected Result:

```
(tf_latest) [dong760@o0674 ~]$ $MV2_PATH/bin/mpirun_rsh -n 2 --hostfile=hostfile
MV2_USE_CUDA=1 MV2_SUPPORT_DL=1
$MV2_PATH/libexec/osu-micro-benchmarks/mpi/collective/osu_allreduce

# OSU MPI Allreduce Latency Test v5.6.3
# Size       Avg Latency(us)
4                       1.19
8                       1.03
16                      1.00
32                      1.05
64                      1.10
128                     1.08
256                     1.14
512                     1.27
1024                    1.44
2048                    1.88
4096                    3.99
8192                    8.09
16384                  13.10
32768                  17.15
65536                  28.21
131072                 45.02
262144                 86.18
524288                169.54
1048576               381.30
```

-- Install Horovod with NCCL support ---

```
# Explicitly define the NCCL path
export NCCL_HOME=/users/PAA0023/dong760/nccl-2.7.8-1/build
export NCCL_LIBRARY=/users/PAA0023/dong760/nccl-2.7.8-1/build/lib
export NCCL_INCLUDE_DIR=/users/PAA0023/dong760/nccl-2.7.8-1/build/include
export CPATH=$NCCL_HOME/build/include:$CPATH
export LD_LIBRARY_PATH=$NCCL_HOME/build/lib:$LD_LIBRARY_PATH
```

```
# Explicitly define the CUDA library path
export PATH=$PATH:$CUDA_HOME/bin
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$CUDA_HOME/lib64

# Explicitly define the CUDA and NCCL path for Horovod
export HOROVOD_CUDA_HOME=$CUDA_HOME
export HOROVOD_CUDA_INCLUDE=$CUDA_HOME/include
export HOROVOD_NCCL_HOME=$NCCL_HOME
export HOROVOD_NCCL_LIB=$NCCL_LIBRARY
export HOROVOD_NCCL_INCLUDE=$NCCL_INCLUDE_DIR


HOROVOD_GPU_OPERATIONS=NCCL HOROVOD_WITH_TENSORFLOW=1 pip install
--no-cache-dir horovod[tensorflow]
OR

HOROVOD_NCCL_HOME=$NCCL_HOME HOROVOD_CUDA_HOME=$CUDA_HOME
HOROVOD_CUDA_INCLUDE=$CUDA_HOME/include
HOROVOD_CUDA_LIB=$CUDA_HOME/lib64 HOROVOD_GPU_OPERATIONS=NCCL
HOROVOD_WITH_TENSORFLOW=1 pip install --no-cache-dir horovod[tensorflow]
```

# Expected Result:

```
(tf_latest) [dong760@o0675 ~]$  horovodrun --check-build
...
2020-11-14 11:54:14.184027: I tensorflow/stream_executor/cuda/cudart_stub.cc:29] Ignore
above cudart dlerror if you do not have a GPU set up on your machine.
Horovod v0.20.3:

Available Frameworks:
    [X] TensorFlow
    [ ] PyTorch
    [ ] MXNet

Available Controllers:
    [X] MPI
    [X] Gloo

Available Tensor Operations:
    [X] NCCL
    [ ] DDL
    [ ] CCL
    [X] MPI
    [X] Gloo
```

-- Test your horovod installation ---

```
import tensorflow as tf
print(tf.__version__)
tf.test.gpu_device_name()
import horovod.tensorflow as hvd
```

```
git clone https://github.com/horovod/horovod.git
# With GPU
$ horovodrun --gloo -np 4 python tensorflow2_mnist.py
$ horovodrun --gloo -np 4 python tensorflow2_synthetic_benchmark.py
```

# Expected Result:

```
(tf_latest) [dong760@o0674 tensorflow2]$ horovodrun --gloo -np 4 python tensorflow2_synthetic_benchmark.py
...
Fri Nov 13 21:11:43 2020[1]<stderr>:pciBusID: 0000:04:00.0 name: Tesla P100-PCIE-16GB computeCapability: 6.0
Fri Nov 13 21:11:43 2020[1]<stderr>:coreClock: 1.3285GHz coreCount: 56 deviceMemorySize: 15.90GiB deviceMemoryBandwidth: 681.88GiB/s
Fri Nov 13 21:11:43 2020[3]<stderr>:2020-11-13 21:11:43.235213: I tensorflow/core/common_runtime/gpu/gpu_device.cc:1716] Found device 0 with properties:
Fri Nov 13 21:11:43 2020[3]<stderr>:pciBusID: 0000:04:00.0 name: Tesla P100-PCIE-16GB computeCapability: 6.0
Fri Nov 13 21:11:43 2020[3]<stderr>:coreClock: 1.3285GHz coreCount: 56 deviceMemorySize: 15.90GiB deviceMemoryBandwidth: 681.88GiB/s
...
Fri Nov 13 21:11:44 2020[0]<stdout>:Model: ResNet50
Fri Nov 13 21:11:44 2020[0]<stdout>:Batch size: 32
Fri Nov 13 21:11:44 2020[0]<stdout>:Number of GPUs: 4
Fri Nov 13 21:11:44 2020[0]<stdout>:Running warmup...
Fri Nov 13 21:11:44 2020[0]<stderr>:WARNING:tensorflow:AutoGraph could not transform <function benchmark_step at 0x2b6da2b25f28> and will run it as-is.
Fri Nov 13 21:11:44 2020[0]<stderr>:Please report this to the TensorFlow team. When filing the bug, set the verbosity to 10 (on Linux, `export AUTOGRAPH_VERBOSITY=10`) and attach the full output.
Fri Nov 13 21:11:44 2020[0]<stderr>:Cause: 'arguments' object has no attribute 'posonlyargs'
Fri Nov 13 21:11:44 2020[0]<stderr>:To silence this warning, decorate the function with @tf.autograph.experimental.do_not_convert
Fri Nov 13 21:13:00 2020[0]<stdout>:Running benchmark...
Fri Nov 13 21:13:56 2020[0]<stdout>:Iter #0: 5.6 img/sec per GPU
Fri Nov 13 21:14:53 2020[0]<stdout>:Iter #1: 5.6 img/sec per GPU
Fri Nov 13 21:15:50 2020[0]<stdout>:Iter #2: 5.6 img/sec per GPU
Fri Nov 13 21:16:46 2020[0]<stdout>:Iter #3: 5.7 img/sec per GPU
Fri Nov 13 21:17:43 2020[0]<stdout>:Iter #4: 5.7 img/sec per GPU
Fri Nov 13 21:18:39 2020[0]<stdout>:Iter #5: 5.7 img/sec per GPU
Fri Nov 13 21:19:36 2020[0]<stdout>:Iter #6: 5.7 img/sec per GPU
=>> PBS: job killed: walltime 3617 exceeded limit 3600
Process 0 exit with status code 137.
Process 1 exit with status code 137.
Process 2 exit with status code 137.
Process 3 exit with status code 137.
...
```

Other Debuging Error:

- **Error1: ‘dense_features’:**

```
# if you hit error: ImportError: cannot import name 'dense_features' from 'tensorflow.python.feature_column'
# ==> SOLU:
pip uninstall tensorflow-estimator
pip install tensorflow-estimator==2.3.0
```

```
# If you tried to run the command, $horovodrun --check-build, you might also see the following errors:
(tf_latest) [dong760@p0237 ~]$ horovodrun --check-build
2020-11-27 15:30:05.007540: I tensorflow/stream_executor/platform/default/dso_loader.cc:48] Successfully opened dynamic library libcudart.so.10.1
2020-11-27 15:30:19.973225: I tensorflow/stream_executor/platform/default/dso_loader.cc:48] Successfully opened dynamic library libcudart.so.10.1
2020-11-27 15:30:31.062035: I tensorflow/stream_executor/platform/default/dso_loader.cc:48] Successfully opened dynamic library libcudart.so.10.1
Traceback (most recent call last):
  File "/users/PAA0023/dong760/miniconda3/envs/tf_latest/bin/horovodrun", line 8, in <module>
    sys.exit(run_commandline())
  File "/users/PAA0023/dong760/miniconda3/envs/tf_latest/lib/python3.6/site-packages/horovod/runner/launch.py", line 717, in run_commandline
    args = parse_args()
  File "/users/PAA0023/dong760/miniconda3/envs/tf_latest/lib/python3.6/site-packages/horovod/runner/launch.py", line 479, in parse_args
    check_build(args.verbose)
  File "/users/PAA0023/dong760/miniconda3/envs/tf_latest/lib/python3.6/site-packages/horovod/runner/launch.py", line 134, in check_build
```

```
    gloo=get_check(gloo_built(verbose=verbose)),
  File
"/users/PAA0023/dong760/miniconda3/envs/tf_latest/lib/python3.6/site-packages/horovod/comm
on/util.py", line 119, in wrapper
    retval = f(*args, **kwargs)
  File
"/users/PAA0023/dong760/miniconda3/envs/tf_latest/lib/python3.6/site-packages/horovod/comm
on/util.py", line 152, in gloo_built
    raise RuntimeError('Failed to determine if Gloo support has been built. '
RuntimeError: Failed to determine if Gloo support has been built. Run again with --verbose for
more details.
```

- Error2:(Unsolved): I am trying to run the code with mvapich

```
(tf_latest) [dong760@p0237 tensorflow2]$ mpirun_rsh --export-all -np 4 --hostfile hostfile python
tensorflow2_synthetic_benchmark.py
2020-11-27 23:16:10.219071: I tensorflow/stream_executor/platform/default/dso_loader.cc:48]
Successfully opened dynamic library libcudart.so.10.1
2020-11-27 23:16:10.219072: I tensorflow/stream_executor/platform/default/dso_loader.cc:48]
Successfully opened dynamic library libcudart.so.10.1
2020-11-27 23:16:10.219073: I tensorflow/stream_executor/platform/default/dso_loader.cc:48]
Successfully opened dynamic library libcudart.so.10.1
2020-11-27 23:16:10.219072: I tensorflow/stream_executor/platform/default/dso_loader.cc:48]
Successfully opened dynamic library libcudart.so.10.1
[cli_0]: write_line error; fd=10 buf=:cmd=init pmi_version=1 pmi_subversion=1
:
system msg for write_line failure : Bad file descriptor
[cli_0]: Unable to write to PMI_fd
[cli_0]: write_line error; fd=10 buf=:cmd=get_appnum
:
system msg for write_line failure : Bad file descriptor
[][error_sighandler] Caught error: Segmentation fault (signal 11)
[cli_0]: write_line error; fd=10 buf=:cmd=init pmi_version=1 pmi_subversion=1
:
system msg for write_line failure : Bad file descriptor
[cli_0]: Unable to write to PMI_fd
[cli_0]: write_line error; fd=10 buf=:cmd=get_appnum
:
system msg for write_line failure : Bad file descriptor
[cli_0]: write_line error; fd=10 buf=:cmd=init pmi_version=1 pmi_subversion=1
:
system msg for write_line failure : Bad file descriptor
[cli_0]: Unable to write to PMI_fd
[cli_0]: write_line error; fd=10 buf=:cmd=get_appnum
:
system msg for write_line failure : Bad file descriptor
```

```
[][error_sighandler] Caught error: Segmentation fault (signal 11)
[cli_0]: write_line error; fd=10 buf=:cmd=init pmi_version=1 pmi_subversion=1
:
system msg for write_line failure : Bad file descriptor
[cli_0]: Unable to write to PMI_fd
[cli_0]: write_line error; fd=10 buf=:cmd=get_appnum
:
system msg for write_line failure : Bad file descriptor
[][error_sighandler] Caught error: Segmentation fault (signal 11)
[p0237.ten.osc.edu:mpispawn_0][child_handler] MPI process (rank: 1, pid: 122955) terminated with signal 11 -> abort job
[p0237.ten.osc.edu:mpirun_rsh][process_mpispawn_connection] mpispawn_0 from node p0237 aborted: MPI process error (1)
```

Creating conda env from environment.yml

- Reference, https://docs.conda.io/projects/conda/en/latest/user-guide/tasks/manage-environments.html#creating-an-environment-from-an-environment-yml-file

```
# The first line of the yml file sets the new environment's name
conda env create -f environment.yml
# Activate the new environment:
conda activate myenv
# Verify that the new environment was installed correctly: (You can also use conda info --envs)
conda env list
```

Attachment:

File1: The conda environment that I used in OSC server (Work for TF only, but with Horovod)

```
(tf_latest) [dong760@owens-login02 ~]$ conda env export > experiment.yml
name: tf_latest
channels:
  - conda-forge
  - defaults
dependencies:
  - _libgcc_mutex=0.1=main
  - _tflow_select=2.3.0=mkl
  - argon2-cffi=20.1.0=py36h8c4c3a4_2
  - astor=0.8.1=py36_0
  - async_generator=1.10=py_0
  - attrs=20.2.0=pyh9f0ad1d_0
  - backports=1.0=py_2
  - backports.functools_lru_cache=1.6.1=py_0
  - blas=1.0=mkl
  - bleach=3.2.1=pyh9f0ad1d_0
  - c-ares=1.16.1=h7b6447c_0
  - ca-certificates=2020.10.14=0
  - certifi=2020.6.20=pyhd3eb1b0_3
  - cffi=1.14.3=py36h829019c_1
  - decorator=4.4.2=py_0
  - defusedxml=0.6.0=py_0
  - entrypoints=0.3=py36h9f0ad1d_1002
  - google-pasta=0.2.0=py_0
  - h5py=2.10.0=py36hd6299e0_1
  - hdf5=1.10.6=hb1b8bf9_0
  - importlib-metadata=2.0.0=py_1
  - importlib_metadata=2.0.0=1
  - intel-openmp=2020.2=254
  - ipykernel=5.3.4=py36hac0dd68_1
  - ipython=5.8.0=py36_1
  - ipython_genutils=0.2.0=py_1
  - jinja2=2.11.2=pyh9f0ad1d_0
  - json5=0.9.5=pyh9f0ad1d_0
  - jsonschema=3.2.0=py_2
  - jupyter_client=6.1.7=py_0
  - jupyter_core=4.6.3=py36h9f0ad1d_2
  - jupyterlab=2.2.9=py_0
  - jupyterlab_pygments=0.1.2=pyh9f0ad1d_0
  - jupyterlab_server=1.2.0=py_0
  - keras=2.3.1=0
  - keras-applications=1.0.8=py_1
  - keras-base=2.3.1=py36_0
  - libedit=3.1.20191231=h14c3975_1
  - libffi=3.2.1=hf484d3e_1007
  - libgcc-ng=9.1.0=hdf63c60_0
  - libgfortran-ng=7.3.0=hdf63c60_0
  - libprotobuf=3.13.0.1=hd408876_0
  - libsodium=1.0.18=h516909a_1
  - libstdcxx-ng=9.1.0=hdf63c60_0
  - markupsafe=1.1.1=py36he6145b8_2
  - mistune=0.8.4=py36h8c4c3a4_1002
  - mkl=2020.2=256
  - mkl-service=2.3.0=py36he904b0f_0
  - mkl_fft=1.2.0=py36h23d657b_0
  - mkl_random=1.1.1=py36h0573a6f_0
  - nbclient=0.5.1=py_0
  - nbconvert=6.0.7=py36h9f0ad1d_2
  - nbformat=5.0.8=py_0
  - ncurses=6.2=he6710b0_1
  - nest-asyncio=1.4.2=pyhd8ed1ab_0
  - notebook=6.1.4=py36h9f0ad1d_1
  - openssl=1.0.2u=h7b6447c_0
  - opt_einsum=3.1.0=py_0
  - packaging=20.4=pyh9f0ad1d_0
  - pandoc=2.11.0.4=hd18ef5c_0
  - pandocfilters=1.4.2=py_1
  - pexpect=4.8.0=pyh9f0ad1d_2
  - pickleshare=0.7.5=py_1003
  - pip=20.2.4=py36_0
  - prometheus_client=0.8.0=pyh9f0ad1d_0
  - prompt_toolkit=1.0.15=py_1
  - ptyprocess=0.6.0=py_1001
  - pycparser=2.20=pyh9f0ad1d_2
  - pygments=2.7.2=py_0
  - pyparsing=2.4.7=pyh9f0ad1d_0
  - pyrsistent=0.17.3=py36h8c4c3a4_1
  - python=3.6.5=hc3d631a_2
  - python-dateutil=2.8.1=py_0
  - python_abi=3.6=1_cp36m
  - pyzmq=19.0.2=py36h9947dbf_2
  - readline=7.0=h7b6447c_5
  - send2trash=1.5.0=py_0
  - setuptools=50.3.0=py36hb0f4dca_1
  - simplegeneric=0.8.1=py_1
  - six=1.15.0=py_0
  - sqlite=3.33.0=h62c20be_0
  - tensorflow=2.0.0=mkl_py36hef7ec59_0
  - tensorflow-base=2.0.0=mkl_py36h9204916_0
  - terminado=0.9.1=py36h9f0ad1d_1
  - testpath=0.4.4=py_0
  - tk=8.6.10=hbc83047_0
  - tornado=6.1=py36h1d69622_0
  - traitlets=4.3.3=py36h9f0ad1d_1
  - wcwidth=0.2.5=pyh9f0ad1d_2
  - webencodings=0.5.1=py_1
  - wheel=0.35.1=py_0
  - wrapt=1.12.1=py36h7b6447c_1
  - xz=5.2.5=h7b6447c_0
  - yaml=0.2.5=h7b6447c_0
  - zeromq=4.3.3=he1b5a44_2
```

```
  - zipp=3.4.0=py_0
  - zlib=1.2.11=h7b6447c_3
  - pip:
    - absl-py==0.10.0
    - astunparse==1.6.3
    - cachetools==4.1.1
    - chardet==3.0.4
    - cloudpickle==1.6.0
    - dataclasses==0.7
    - gast==0.2.2
    - google-auth==1.22.1
    - google-auth-oauthlib==0.4.1
    - grpcio==1.33.1
    - horovod==0.20.3
    - idna==2.10
    - keras-preprocessing==1.1.2
    - markdown==3.3.3
    - numpy==1.18.5
    - oauthlib==3.1.0
    - opt-einsum==3.3.0
    - protobuf==3.13.0
    - psutil==5.7.3
    - pyasn1==0.4.8
    - pyasn1-modules==0.2.8
    - pyyaml==5.3.1
    - requests==2.24.0
    - requests-oauthlib==1.3.0
    - rsa==4.6
    - scipy==1.4.1
    - tensorboard==2.0.2
    - tensorboard-plugin-wit==1.7.0
    - tensorflow-estimator==2.3.0
    - tensorflow-gpu==2.3.0
    - termcolor==1.1.0
    - urllib3==1.25.11
    - werkzeug==1.0.1
prefix: /users/PAA0023/dong760/miniconda3/envs/tf_latest
```

File 2:The conda environment that I used on RI2 server (Work for both TF and Pytorch, the horovod is tricky to install on some server sometime)

```
(tf_latest) [dong.760@head ~]$ conda env export
name: tf_latest
channels:
  - pytorch
  - defaults
dependencies:
  - _libgcc_mutex=0.1=main
  - blas=1.0=mkl
  - ca-certificates=2021.1.19=h06a4308_1
  - certifi=2020.12.5=py36h06a4308_0
  - cudatoolkit=10.1.243=h6bb024c_0
  - cycler=0.10.0=py36_0
  - dataclasses=0.7=py36_0
  - dbus=1.13.16=hb2f20db_0
  - expat=2.2.9=he6710b0_2
  - fontconfig=2.13.0=h9420a91_0
  - freetype=2.10.2=h5ab3b9f_0
  - glib=2.63.1=h5a9c865_0
  - gst-plugins-base=1.14.0=hbbd80ab_1
  - gstreamer=1.14.0=hb453b48_1
  - icu=58.2=he6710b0_3
  - intel-openmp=2020.2=254
  - jpeg=9b=h024ee3a_2
  - kiwisolver=1.2.0=py36hfd86e86_0
  - lcms2=2.11=h396b838_0
  - libedit=3.1.20191231=h14c3975_1
  - libffi=3.2.1=hf484d3e_1007
  - libgcc-ng=9.1.0=hdf63c60_0
  - libpng=1.6.37=hbc83047_0
  - libstdcxx-ng=9.1.0=hdf63c60_0
  - libtiff=4.1.0=h2733197_1
  - libuuid=1.0.3=h1bed415_2
  - libuv=1.40.0=h7b6447c_0
  - libxcb=1.14=h7b6447c_0
  - libxml2=2.9.10=he19cac6_1
  - lz4-c=1.9.2=heb0550a_3
  - matplotlib=3.3.1=0
  - matplotlib-base=3.3.1=py36h817c723_0
  - mkl=2020.2=256
  - mkl-service=2.3.0=py36he904b0f_0
  - mkl_fft=1.2.0=py36h23d657b_0
  - mkl_random=1.1.1=py36h0573a6f_0
  - ncurses=6.2=he6710b0_1
  - numpy-base=1.19.1=py36hfa32c7d_0
  - olefile=0.46=py36_0
  - openssl=1.0.2u=h7b6447c_0
  - pcre=8.44=he6710b0_0
  - pillow=7.2.0=py36hb39fc2d_0
  - pip=20.2.3=py36_0
  - pyparsing=2.4.7=py_0
  - pyqt=5.9.2=py36h05f1152_2
  - python=3.6.5=hc3d631a_2
  - python-dateutil=2.8.1=py_0
  - pytorch=1.7.1=py3.6_cuda10.1.243_cudnn7.6.3_0
  - qt=5.9.6=h8703b6f_2
  - readline=7.0=h7b6447c_5
  - setuptools=50.3.0=py36hb0f4dca_1
  - sip=4.19.8=py36hf484d3e_0
  - six=1.15.0=py_0
  - sqlite=3.33.0=h62c20be_0
  - tk=8.6.10=hbc83047_0
  - torchaudio=0.7.2=py36
  - torchvision=0.8.2=py36_cu101
  - tornado=6.0.4=py36h7b6447c_1
  - typing_extensions=3.7.4.3=pyha847dfd_0
  - wheel=0.35.1=py_0
  - xz=5.2.5=h7b6447c_0
  - zlib=1.2.11=h7b6447c_3
  - zstd=1.4.5=h9ceee32_0
  - pip:
    - absl-py==0.10.0
    - astunparse==1.6.3
    - backcall==0.2.0
    - cachetools==4.1.1
    - cffi==1.14.5
    - chardet==3.0.4
    - cloudpickle==1.6.0
    - decorator==4.4.2
    - deepspeed==0.3.12
    - future==0.18.2
    - gast==0.3.3
    - google-auth==1.22.1
    - google-auth-oauthlib==0.4.1
    - google-pasta==0.2.0
    - grpcio==1.32.0
    - h5py==2.10.0
    - horovod==0.21.2
    - idna==2.10
    - importlib-metadata==2.0.0
    - ipython==7.16.1
    - ipython-genutils==0.2.0
```

```
    - jedi==0.17.2
    - keras-preprocessing==1.1.2
    - markdown==3.3
    - mpi4py==3.0.3
    - ninja==1.10.0.post2
    - numpy==1.18.5
    - oauthlib==3.1.0
    - opt-einsum==3.3.0
    - parso==0.7.1
    - pexpect==4.8.0
    - pickleshare==0.7.5
    - prompt-toolkit==3.0.8
    - protobuf==3.13.0
    - psutil==5.8.0
    - ptyprocess==0.6.0
    - pyasn1==0.4.8
    - pyasn1-modules==0.2.8
    - pycparser==2.20
    - pygments==2.7.2
    - pyyaml==5.4.1
    - requests==2.24.0
    - requests-oauthlib==1.3.0
    - rsa==4.6
    - scipy==1.4.1
    - speedtest-cli==2.1.3
    - tensorboard==2.3.0
    - tensorboard-plugin-wit==1.7.0
    - tensorboardx==1.8
    - tensorflow-estimator==2.3.0
    - tensorflow-gpu==2.3.0
    - termcolor==1.1.0
    - tqdm==4.59.0
    - traitlets==4.3.3
    - urllib3==1.25.10
    - wcwidth==0.2.5
    - werkzeug==1.0.1
    - wrapt==1.12.1
    - zipp==3.3.0
prefix: /home/dong.760/miniconda3/envs/tf_latest
```

File 3: This is the first file I will run after allocated a node at RI2

```
[dong.760@head ~]$ source cse5194_Distributed_DNN-ri2_repo-/load_mv2.sh
```

```
Export the file path...
pwd
/home/dong.760/cudnn/cuda/lib64/:/home/dong.760/opt/mvapich2/gdr/2.3.4/mcast/no-openacc/cuda10.1/mofed4.5/mpirun/gnu7.3.0/lib64/:/opt/gcc/7.3.0/lib64/:/opt/cuda/10.1/lib64/
/usr/bin/which: no mpirun_rsh in
(/home/dong.760/opt/mvapich2/gdr/2.3.4/mcast/no-openacc/cuda10.1/mofed4.5/mpirun/gnu7.3.0/bin/:/opt/gcc/7.3.0/bin/:/opt/cuda/10.1/bin:/home/dong.760/miniconda3/envs/tf_latest/bin:/home/dong.760/miniconda3/condabin:/home/dong.760/miniconda3/bin:/usr/lib64/qt-3.3/bin:/home/dong.760/perl5/bin:/usr/local/bin:/usr/bin:/usr/local/sbin:/usr/sbin:/act/bin:/usr/local/cuda/bin:/opt/ibutils/bin:/act/bin:/home/dong.760/.local/bin:/home/dong.760/bin)
Running OSU Micro Benchmark on CPU...
Running OSU Micro Benchmark on GPU...
Testing MPI Hello World
srun: job 332400 queued and waiting for resources
srun: job 332400 has been allocated resources
gpu18.cluster
(tf_latest) [dong.760@head ~]$ cat cse5194_Distributed_DNN-ri2_repo-/load_mv2.sh
#!/bin/bash
# salloc -N 2 -p sky-k80 # bdw-k80, sky-k80, bdw-v100
# salloc -N 2 --ntasks-per-node=3 -p bdw-v100
# Activate the conda environment
source ./miniconda3/bin/activate
export PYTHONNOUSERSITE=true
conda activate tf_latest # Don't use pytorch_latest, because that one is not build from source, but installed from conda package
# Some language(Option)
export LANGUAGE=en_US.UTF-8
export LANG=en_US.UTF-8
export LC_ALL=en_US.UTF-8
# Load the required module for running Deep Learning model with MVAPIC
module load cuda/10.1 gcc/7.3.0
# ======== MVAPICH2-GDR Env set up ==================================
echo "Export the file path..."
 # Enable CUDA
export MV2_USE_CUDA=1
# MVAPICH2-GDR supports TensorFlow/PyTorch/MXNet with Horovod/MPI design but a special flag is needed to run the jobs properly.
export MV2_SUPPORT_DL=1
# MVAPICH2-GDR 2.3.4 boosts the performance by taking advantage of the new GDRCOPY module from NVIDIA. In order to take advantage, you need this:
export MV2_GPUDIRECT_GDRCOPY_LIB=/opt/gdrcopy2.0/lib64/libgdrapi.so

export CURR_DIR=$PWD
```

```
export
MV2_PATH=$CURR_DIR/opt/mvapich2/gdr/2.3.4/mcast/no-openacc/cuda10.1/mofed4.5/mpirun
/gnu7.3.0/
# MVAPICH2
export
PATH=/home/dong.760/opt/mvapich2/gdr/2.3.4/mcast/no-openacc/cuda10.1/mofed4.5/mpirun/g
nu7.3.0/bin/:$PATH
export
LD_LIBRARY_PATH=/home/dong.760/opt/mvapich2/gdr/2.3.4/mcast/no-openacc/cuda10.1/mof
ed4.5/mpirun/gnu7.3.0/lib64/:$LD_LIBRARY_PATH
# cuDNN
export CUDNN_ROOT=/home/dong.760/cudnn/cuda/
export CUDNN_INCLUDE_DIR=/home/dong.760/cudnn/cuda/include/
export CUDNN_LIBRARY=/home/dong.760/cudnn/cuda/lib64/
export LD_LIBRARY_PATH=/home/dong.760/cudnn/cuda/lib64/:$LD_LIBRARY_PATH

echo pwd
echo $LD_LIBRARY_PATH
# rpm2cpio mvapich2-gdr-mcast.cuda10.0.mofed4.5.gnu4.8.5-2.3.4-1.el7.x86_64.rpm | cpio -id
which mpirun_rsh

# export CURR_DIR=$PWD
# export
MV2_PATH=$CURR_DIR/opt/mvapich2/gdr/2.3.4/mcast/no-openacc/cuda10.1/mofed4.5/mpirun
/gnu7.3.0/
# export
PATH=$CURR_DIR/opt/mvapich2/gdr/2.3.4/mcast/no-openacc/cuda10.1/mofed4.5/mpirun/gnu7
.3.0/bin/:$PATH
# export
LD_LIBRARY_PATH=$CURR_DIR/opt/mvapich2/gdr/2.3.4/mcast/no-openacc/cuda10.1/mofed4
.5/mpirun/gnu7.3.0/lib64/:$LD_LIBRARY_PATH
# export
PATH=/home/dong.760/cse5194/opt/mvapich2/gdr/2.3.4/mcast/no-openacc/cuda10.1/mofed4.5/
mpirun/gnu7.3.0/bin:$PATH
# export
LD_LIBRARY_PATH=/home/dong.760/cse5194/opt/mvapich2/gdr/2.3.4/mcast/no-openacc/cuda
10.1/mofed4.5/mpirun/gnu7.3.0/lib64:$LD_LIBRARY_PATH
# mpirun_rsh -n 2 gpu24 gpu24 MV2_USE_CUDA=1 MV2_SUPPORT_DL=1
$MV2_PATH/libexec/osu-micro-benchmarks/mpi/collective/osu_allreduce
# mpirun_rsh -n 2 gpu01 gpu02
$MV2_PATH/libexec/osu-micro-benchmarks/mpi/collective/osu_allreduce -d cuda # With cuda
support

# ======== Running OSU Micro Benchmark  =================================
echo "Running OSU Micro Benchmark on CPU..."
# cd opt/mvapich2/gdr/2.3.4/mcast/no-openacc/cuda10.0/mofed4.5/mpirun/gnu4.8.5/
# $MV2_PATH/bin/mpirun_rsh -n 2 gpu23 gpu23 MV2_USE_CUDA=1 MV2_SUPPORT_DL=1
$MV2_PATH/libexec/osu-micro-benchmarks/mpi/collective/osu_allreduce
# mpirun_rsh -n 2 gpu29 gpu30 MV2_USE_CUDA=1 MV2_SUPPORT_DL=1
$MV2_PATH/libexec/osu-micro-benchmarks/mpi/collective/osu_allreduce

echo "Running OSU Micro Benchmark on GPU..."
# $MV2_PATH/bin/mpirun_rsh -n 2 gpu23 gpu23 MV2_USE_CUDA=1 MV2_SUPPORT_DL=1
$MV2_PATH/libexec/osu-micro-benchmarks/mpi/collective/osu_allreduce -d cuda
# mpirun_rsh -n 2 gpu23 gpu23 MV2_USE_CUDA=1 MV2_SUPPORT_DL=1
$MV2_PATH/libexec/osu-micro-benchmarks/mpi/collective/osu_allreduce -d cuda

# echo "Running Deep Learning Frameworks with Horovod and MVAPICH2-GDR on CPU..."
# $MV2_PATH/bin/mpirun_rsh -n 2 gpu29 gpu30 python tf_cnn_benchmarks.py
--model=resnet50 --variable_update=horovod


# ======== Running MPI Hello World Example ================
echo "Testing MPI Hello World"
export MPIRUN=$MV2_PATH/bin/mpirun_rsh
export MPI_HOSTS=/home/dong.760/host_file
srun hostname
# $MPIRUN -n 2 gpu29 gpu29 MV2_USE_CUDA=1 MV2_SUPPORT_DL=1
$CURR_DIR/mpitutorial/tutorials/mpi-hello-world/code/mpi_hello_world
# mpirun_rsh -n 2 gpu23 gpu24 MV2_USE_CUDA=1 MV2_SUPPORT_DL=1
$CURR_DIR/mpitutorial/tutorials/mpi-hello-world/code/mpi_hello_world
# mpiexec -n 2 python MV2_USE_CUDA=1 MV2_SUPPRESS_CUDA_USAGE_WARNING=1
-m mpi4py.bench helloworld
```