CSDS 451: Designing High Performant Systems for AI
Fall 2023
PA 3
Due: November 21, 2023, before 11:59 pm,
Total Points: 20

Submission: This assignment consists of two parts: **code** and your **write-up**.

Code: You should provide all source code/data files in a zip file. Please do not include PDFs in the zip so that they may be viewed using Canvas.

Write-up: All responses to questions that are not written code should be typed and compiled to PDF. There is no requirement for what text editing tool that you use, however we recommend using LaTeX or Markdown. **Hand-written submissions (including using a tablet), outside of drawn diagrams for certain problems, will not be graded.**

This assignment takes a step back from implementing the low-level details within algorithms, and instead focuses on a CNN as a whole. Designing models themselves can be very difficult, but so can designing the infrastructure that hosts them. Thankfully, some of this process can be automated to make our lives easier. One such tool that aims to abstract away this process (and that you will be using in this assignment) is PyTorch Lightning.

NOTE: Please comment your code appropriately. We will review and compile the code that you write and verify that it works ourselves. **If you did anything special to run your code outside of calling `python File.py`, then please also submit that with your code.**

You must use the GPU(s) for your CNN.

1) A single GPU: You will be using the file, `PA3-Template.py`, as a starting point for this assignment. This should be available on the same page as this document. This contains a CNN that was created using PyTorch, however we want you to modify this script such that the model runs with PyTorch Lightning. In addition to getting a GPU node, you should use Lmod to load the following modules (these are what I used, so just a suggestion):

- **gcc/6.3.0**
- **python/3.8.6**

After you have the version of Python that you want to use in the path, then I recommend creating a virtual environment in order to install the following packages (again, these are just what I used to get everything working, you can use other versions if you wish):

- **Torch v1.11.0**, that is compiled with CUDA 10.2.
  - **Note:** the install directive I'm giving below also installs torchvision too. You should be able to get CIFAR-10 from there.
  - `pip install torch==1.11.0+cu102 torchvision==0.12.0+cu102 torchaudio==0.11.0 --extra-index-url https://download.pytorch.org/whl/cu102`
- **PyTorch Lightning v1.8.0**
  - `pip install pytorch-lightning==1.8.0`

Now that everything is installed, you should be good to begin the assignment. **(20 points total.)**

a) Modify the provided file, `PA3-Template.py`, such that it trains on the CIFAR-10 dataset and utilizes PyTorch Lightning to instantiate and train the model using a data parallel strategy (**not distributed data parallel**). **(10 points.)**

- **Hint: It should not run immediately out of the box, maybe take a look at data dimensionality as it passes through the model.**

b) Find 4 batch sizes that work with your model and hardware on the HPC. What batch sizes have you chosen? **(0.5 points.)**

c) Train the model on all 4 batch sizes from **Problem b)** using 1 GPU and time how long it takes to train the CNN for 2 epochs. Make a plot comparing batch size to training time. **(3 points.)**

d) Now, repeat **Problem c)** for 2 GPUs (**Hint:** you will have to allocate a new worker node with the flag `--gpus=2`). You should show a plot comparing the batch size to training time (for 2 epochs). **(3 points.)**

e) Make sure that your model is actually training with 2 GPUs. You should do this by SSH-ing into your worker node from a separate instance and using the command `nvidia-smi`, take a screenshot of the output, and paste it here. **(0.5 points.)**

f) How does batch size affect the training speed? Is the trend the same between training on 1 GPU vs. 2 GPUs? **(1.5 points.)**

g) Compare and contrast the training speeds you observed with 1 vs. 2 GPUs. Which one was faster? Why do you think that is? **(1.5 points.)**