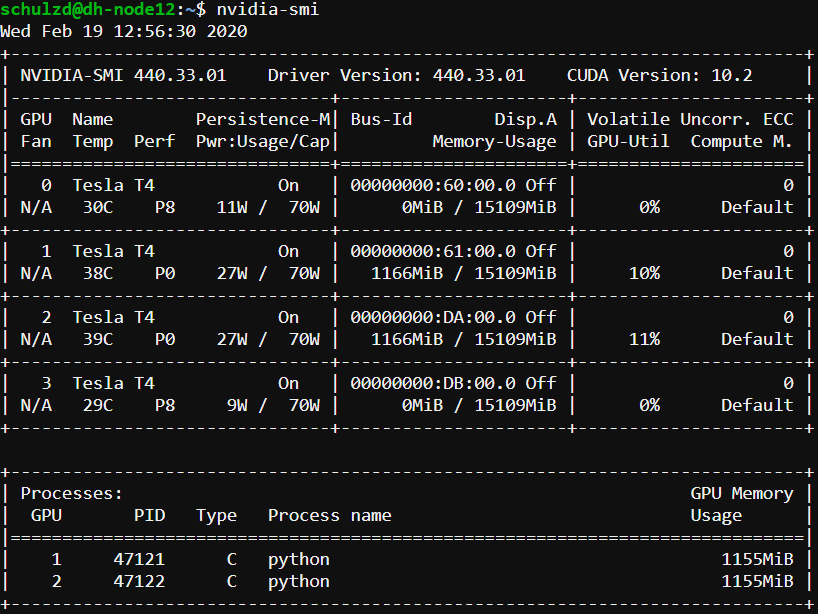
**Lab 10: Multi-GPU Scaling**

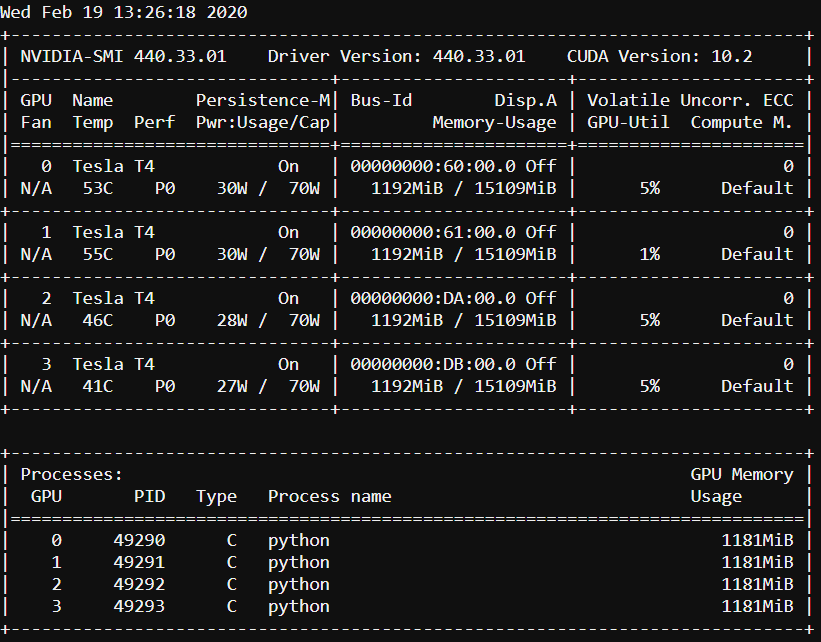
**David Schulz**

1. 2 T4 Experiment
   1. All of these experiments were trained with a batch size of 32 and for 10 epochs.



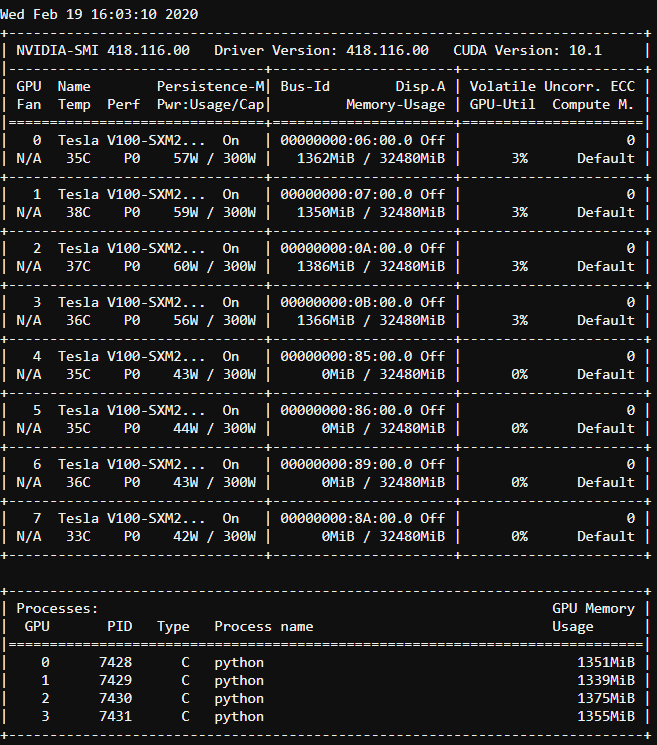
* 1. Turnaround time: 9 min 31 sec
  2. Loss value: 0.0173
  3. Observations: The utilization percentage shown in the image above was about as high as it got. It was more frequently around 7 or 8 percent. This experiment will be used as the control to compare to the remaining experiments.

1. 4 T4 Experiment



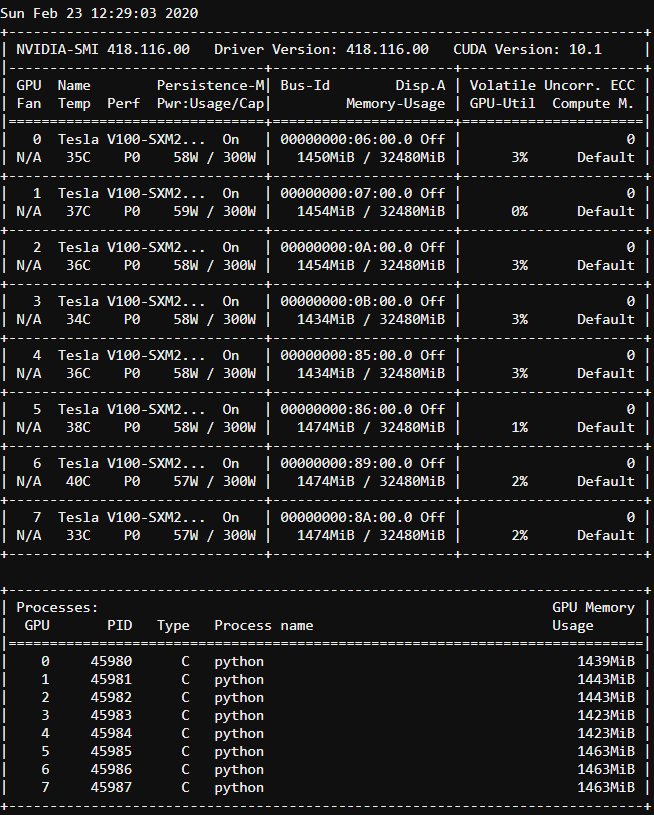
* 1. Turnaround time: 8 min 46 sec
  2. Loss value: 0.0183
  3. Observations: After using 2 more T4 GPUs, the turnaround time was a little less than a minute faster. The utilization of each GPU was most frequently around 5%, about half the utilization of each GPU when using 2. The final loss value was exactly 0.01 more than the first experiment, which isn’t much of a difference.

1. 4 V100 Experiment

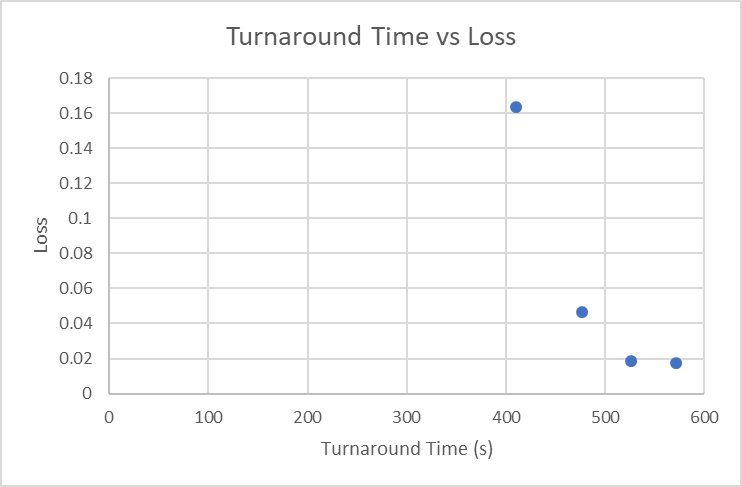
****

* 1. Turnaround time: 7 min 57 sec
  2. Loss value: 0.0463
  3. Observations: After using 4 V100 GPUs from one of the DGX-1s, the utilization of each GPU usually only got up to 3%. The turnaround time decreased by 49 seconds from part 2. However, the loss value increased by about 0.03. This isn’t a big problem yet, but it may indicate that less training time with the same hyperparameters doesn’t necessarily result in the same loss value.

1. 8 V100 Experiment



* 1. Turnaround time: 6 min 50 sec
  2. Loss value: 0.1638
  3. Observations: After using all 8 V100s in one DGX-1, the utilization of each GPU still only got up to 3%. The turnaround time decreased from part 3 by over a minute. However, the loss value increased by 0.1175!



* 1. This graph confirms my hypothesis in part 3. Although running the training with multiple GPUs at a time makes the training finish faster, the loss isn’t able to get as low because of the lack of communication between certain layers of the network in each GPU. My guess is loss could improve when using multiple GPUs by increasing the batch size and, therefore, the number of epochs, meaning more utilization of each GPU and less reliance on the communication between GPUs.