Optimizing PyTorch

Tom Rochette <tom.rochette@coreteks.org>

July 26, 2021 — 9ebcbd05

0.1 Context

0.2 Learned in this study

0.3 Things to explore

- Multi-GPU usage
- Training multiple models on a single GPU in multiple processes

1 Overview

- Use cases
  - Code is not using 100% of the CPU/RAM
    - Increase batch size
    - Parallelize data loading with GPU computation
  - Data does not fit in GPU RAM
    - Reduce batch size
  - GPU usage is 100% yet there is no progress
    - This might be due to using multithreading and having CPU/GPU trashing occurring
      - In my experiment I've seen 100% GPU usage, not completely sure about CPU usage

2 Checklist

- Data loader
  - Efficient
  - In a separate thread/non-blocking
- Data transfer between CPU and GPU
  - Minimal
- Batch size
  - Take as much GPU RAM as possible
- GPU usage is near 100% (GPU should be your bottleneck)
- Verify that GPU memory is freed

3 Notes

- Run your script with python's profiler to determine which part of your script is CPU expensive

```
python -m cProfile -o my_profile.prof train.py
```

- Run your script with `nvprof` to determine what is being done on the GPU

```
nvprof -o my_profile.nvvp python train.py
```

- Free up the memory you used with `del` (e.g., `del my_tensors`)
- If running PyTorch in multiple processes, make sure to configure `OMP_NUM_THREADS` to a low number as PyTorch uses multithreaded BLAS to do linear algebra on CPU. If this is not specified, the processes will likely attempt to use all cores, which will cause issues since each process will be effectively trying to use all the cores as well

4 See also

5 References

- https://sagivtech.com/2017/09/19/optimizing-pytorch-training-code/
- https://github.com/Kaixhin/grokking-pytorch
- https://pytorch.org/docs/stable/notes/multiprocessing.html
- https://www.dangdatascience.com/articles/writing-fast-pytorch/
- https://towardsdatascience.com/9-tips-for-training-lightning-fast-neural-networks-in-pytorch-8e63a502f565
- https://gist.github.com/sonots/5abc0bccec2010ac69ff74788b265086