Profiling Deep Learning

Denis Boyda

Section overview

Zhen Xie

Murali Emali

Denis Boyda

TensorFlow profiler

Nvidia Nsight profiler

Intel Vtune profiler

MPItrace profiler

Horovod timeline

Outline

- What is profiling
- Motivation to profile
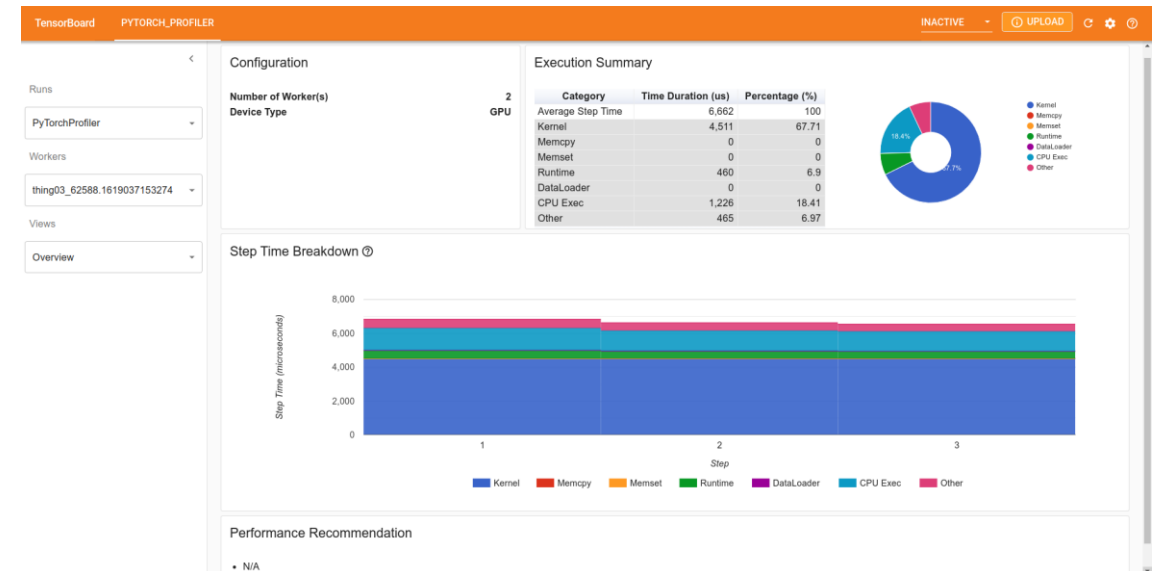- Some frequent problems
- Common Profiler APIs

What is profiling?

- Measuring time/memory cost

- Identifying bottleneck

- Tracking call stack

- Measuring communications

Argonne Leadership Computing Facility

Motivation to profile

What is the purpose of profiling?
- finding performance issues
- optimizations and tuning
- fine-tuning

Do you need to deeply profile/optimize your script?
- training vs evaluation
- Are you developing your model or you already know the solution?
- Is your accuracy increasing with scaling?

- Is speed up worth your affords?
  - Is GPU utilization high?
  - Do you expect high improvements?
  - Are you planning to scale?

"The real problem is that programmers have spent far too much time worrying about efficiency in the wrong places and at the wrong times; ***premature optimization is the root of all evil (or at least most of it) in programming.***"

@Donald Knuth

Argonne
NATIONAL LABORATORY

Some frequent problems

- slow IO
  - Use designated PT/TF data loaders
  - Multithreading

- (extra) copy CPU ↔ GPU
  - Move all operations to device
    - Rewrite your code with using only PT/TF tensors
    - Use Numba / CuPy
  - Overlap copy and computation
  - Use asynchronous copy

- small batch size
  - Increase batch size until you totally load GPU

- high precision
  - Can you decrease precision to float/half?
  - Autotuning mixed precision in PT

Argonne
NATIONAL LABORATORY

Common profiler APIs

- Context manager

```python
with profiler.profile() as prof:
    model(inputs)

with torch.profiler.profile(
        activities=[torch.profiler.ProfilerActivity.CPU,
                    torch.profiler.ProfilerActivity.CUDA],
        schedule=torch.profiler.schedule(wait=0, warmup=1, active=3),
        on_trace_ready=torch.profiler.tensorboard_trace_handler(dir_name)
        ) as prof:

    for _ in range(8):
        model(inputs)
```

- Decorator

```python
@profile
def train_loop(batch_size, n_training_iterations, models, opts, global_size):

    logger = logging.getLogger()

    rank = hvd.rank()
    for i in range(n_training_iterations):
```

- Python module

```
python -m line_profiler train_GAN.py.lprof
```

```
python -m torch.utils.bottleneck example2/v3.py
```

Let's play with examples!