Tutorial. Nov 18, 2024 1:30-5 pm

Distributed Deep Learning on GPU-based Clusters

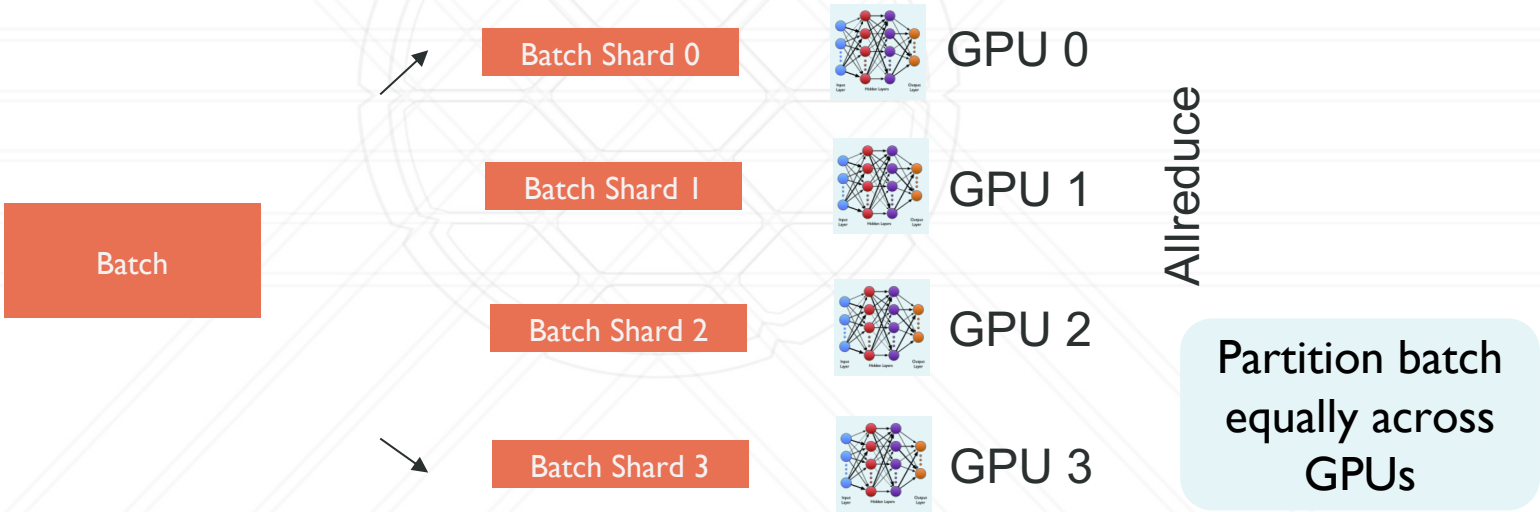
Abhinav Bhatele, Siddharth Singh, Prajwal Singhania
Department of Computer Science

Data parallelism

- Work on different parts of the data in parallel on different GPUs
- Example: PyTorch's DDP, FSDP, and DeepSpeed's ZeRO

Work distribution in data parallelism

- Each worker has a full copy of the entire NN and processes different mini-batches
- All reduce operation to synchronize gradients



DDP Strategy in Lightning

```
from lightning.fabric.strategies import DDPStrategy

pl_strategy = DDPStrategy()

fabric = Fabric(
        strategy=pl_strategy,
                . . .)
```

Running the code

- Code location - train.py

```
CONFIG_FILE=configs/ddp.json
sbatch --ntasks-per-node=4  train.sh
```

FSDP

- Using DDP is limited to smaller model sizes
- FSDP implements memory optimizations to fit larger models on a GPU
    - Shard parameters of each Decoder block across GPUs

FSDPStrategy in Lightning

```
from lightning.fabric.strategies import
FSDPStrategy

pl_strategy =FSDPStrategy(
                auto_wrap_policy={Block}
)

fabric = Fabric(strategy=pl_strategy, . . .)
```

⇦ Specify what modules to shard

Running the code

```
CONFIG_FILE=configs/fsdp.json
sbatch --ntasks-per-node=4  train.sh
```

Abhinav Bhatele and Siddharth Singh

Department of Computer Science

bhatele@umd.edu, ssingh37@umd.edu