Accelerating DNN training on BlueField DPUs

Click to add text

Presentation at MUG ‘21

Arpan Jain

Network Based Computing Laboratory (NBCL)
Dept. of Computer Science and Engineering, The Ohio State University

Jain.575@osu.edu
BlueField DPU / Smart NIC Architecture

- BlueField includes the ConnectX6 network adapter and data processing cores
- System-on-chip containing 64-bit ARMv8 A72
- Why BlueField DPU for Deep Learning?
  - State-of-the-art DPUs bring more compute power to network
  - Deep Learning training needs all the available compute power it can get
Exploiting DPUs for Deep Neural Network Training

- There are several phases in Deep Neural Network Training:
  - Fetching Training Data
  - Data Augmentation
  - Forward Pass
  - Backward Pass
  - Weight Update
  - Model Validation
- Different phases can be offloaded to DPUs to accelerate the training.
Offload Naive (O-N): Offloading DL Training using Data Parallelism

Data parallelism can be used to train DNN on DPUs.

Accelerating DNN Training using Offload-Naive

- Time per iteration can be used to distribute the work (batch size) between CPU and DPU
- Speedup:
  - We report up to 1.03X speedup
  - Maximum speedup possible: 1.04X
- Offload-Naive does not give significant speedup as forward and backward pass are compute-intensive tasks and DPUs are not as powerful as CPUs

Design 1: Offload Data Augmentation (O-DA)

- Offloads the reading of training data from memory and data augmentation on input data to DPUs.
- Creates two types of processes
  - Training processes (on CPU)
  - Data Augmentation processes (On DPU)
- Initializes two buffers to enable asynchronous communication
- Each training processes has one data augmentation processes on DPU.

Design 2: Offload Model Validation (O-MV)

- Offloads validation of model after each epoch to DPUs.
- Model validation is a less compute-intensive task as it has only forward pass.
- Creates two types of processes
  - Training processes (on CPU)
  - Testing processes (On DPU)
- One communication operation per epoch.
- Validation data is equally divided among testing processes.

Design 3: Offload Hybrid (O-Hy)

- Offloads data augmentation and model validation to DPUs.
- Creates three types of processes
  - Training processes (on CPU)
  - Data Augmentation processes (On DPU)
  - Testing processes (On DPU)
- Each Data Augmentation process on DPU supports multiple training processes.
- Data Augmentation processes does asynchronous communication and Testing processes does synchronous communication.
Training ResNet-20 on CIFAR-10 Dataset

- Speedup
  - Single node: O-DA (13.8%) and O-MV (3.1%)
  - Multi-node: Achieves average 13.9% speedup on 1-16 nodes
Training ResNet-56 on SVHN Dataset

- Speedup
  - Single node: O-DA (7%), O-MV (5.5%), and O-Hy (10.1%)
  - Multi-node: 9.3% speedup on 16 nodes

Training ShuffleNet on Tiny ImageNet Dataset

- Speedup
  - Single node: O-DA (12.5%), O-MV (1.2%), and O-Hy (8.9%)
  - Multi-node: 10.2% speedup on 16 nodes
Conclusion

• Proposed novel offloading designs for DPUs
  – Offload Naive
  – Offload Data Augmentation
  – Offload Model Validation
  – Offload Hybrid
• Reported up to 15%, 12.5%, and 11.2% speedup for CIFAR-10, SVHN, and Tiny ImageNet datasets
• Demonstrated consistent performance gain on multiple nodes.
• Uses Torchvision, PyTorch, Horovod, and MPI for flexibility and scalability
• Future Work
  – Use DPUs to accelerate DNN training on GPUs
  – Evaluate TransFormer models
Thank You!

Jain.575@osu.edu

Network-Based Computing Laboratory
http://nowlab.cse.ohio-state.edu/

High Performance Deep Learning
http://hidl.cse.ohio-state.edu/

High-Performance MPI/PGAS Project
http://mvapich.cse.ohio-state.edu/