MVAPICH
MPI, PGAS and Hybrid MPI+PGAS Library

Accelerating Deep Learning Training with Hybrid Parallelism and MVAPICH2-GDR

Presentation at MUG '21

Arpan Jain

Network Based Computing Laboratory (NBCL)

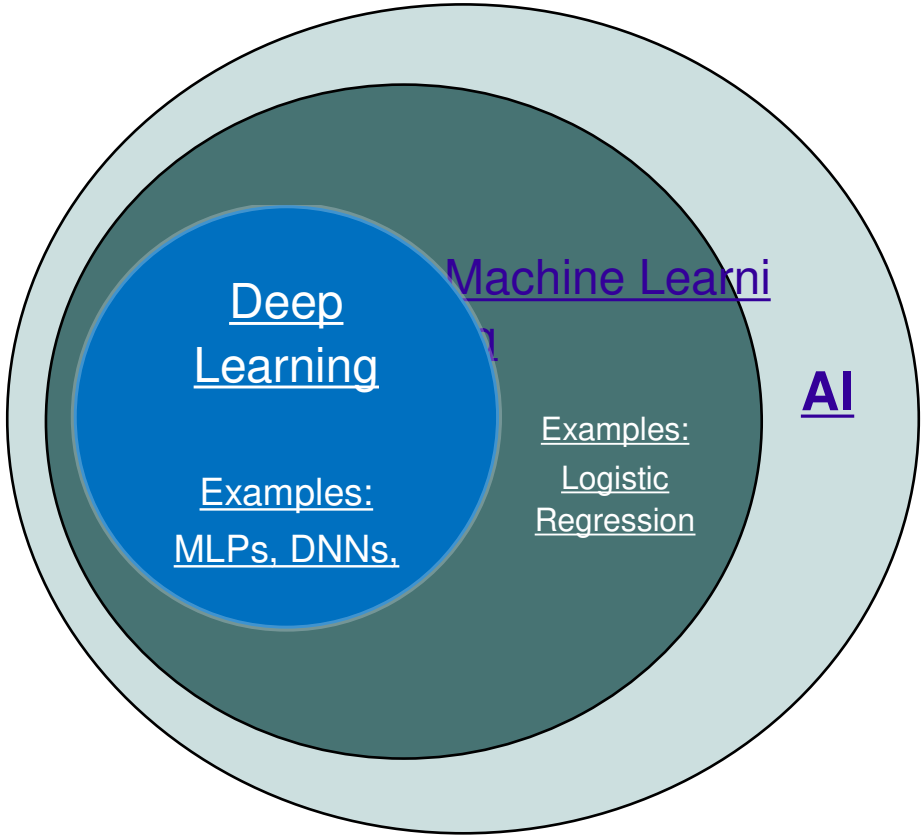
Dept. of Computer Science and Engineering , The Ohio State University

Jain.575@osu.edu

Understanding the Deep Learning Resurgence

- Deep Learning (DL) is a sub-set of Machine Learning (ML)
  - Perhaps, the most revolutionary subset!
  - **Feature extraction** vs. **hand-crafted features**

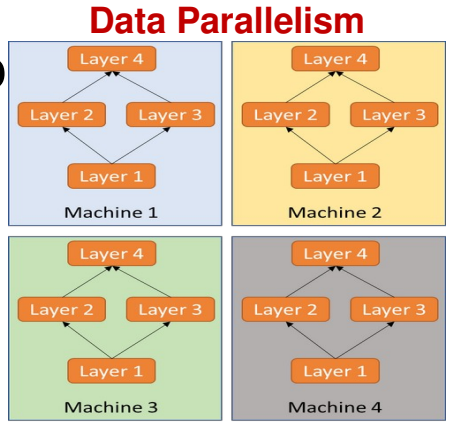
- Deep Learning
  - A renewed interest and a lot of hype!
  - Key success: Deep Neural Networks (DNNs)
  - Everything was there since the late 80s except the **"computability of DNNs"**

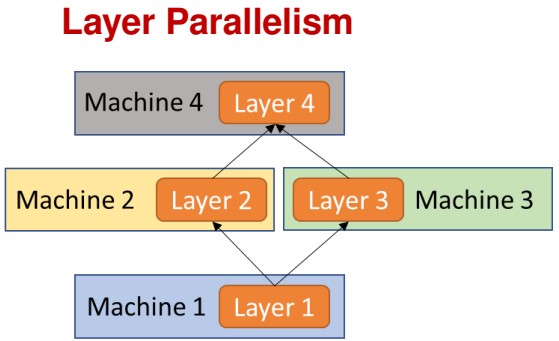


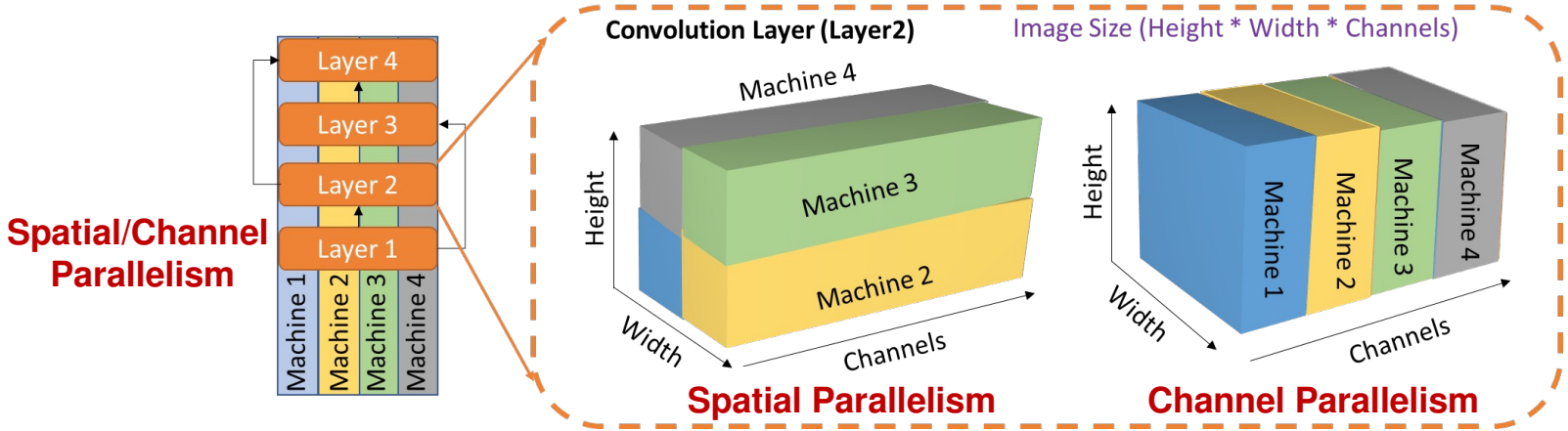
Adopted from: http://www.deeplearningbook.org/contents/intro.html

Distributed/Parallel Training Strategies for DNNs

- Data Parallelism (most commo
- Layer Parallelism
- Domain Parallelism
  - Spatial/Channel Parallelism
  - **Sub-graph Parallelism**



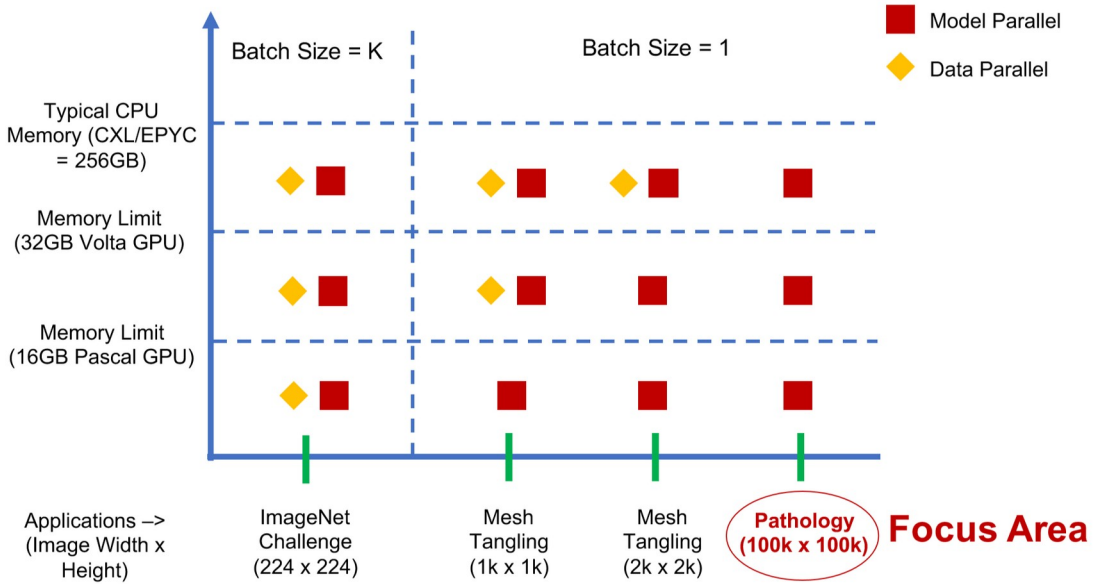




Data Parallelism is not enough!!

Why do we need model parallelism?

- Data Parallel training has a major limitation
- Current state-of-art models cannot be trained on single GPU
- Areas such as Pathology have very large images that makes training impossible using Data Parallelism



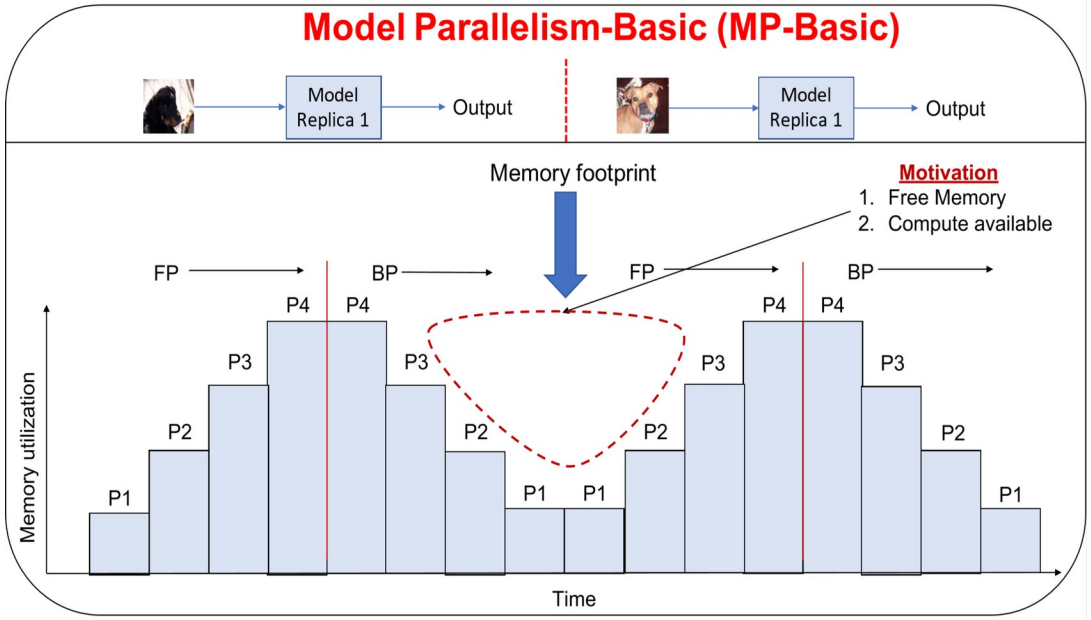
Memory requirement increases with the increase in image size!

A. Jain, A. Awan, A. Aljuhani, J. Hashmi, Q. Anthony, H. Subramoni, D. Panda, R. Machiraju, A. Parwani, "GEMS: GPU Enabled Memory Aware Model Parallelism System for Distributed DNN", SC '20

Need for Memory Aware Designs

Why do we need Memory aware designs?

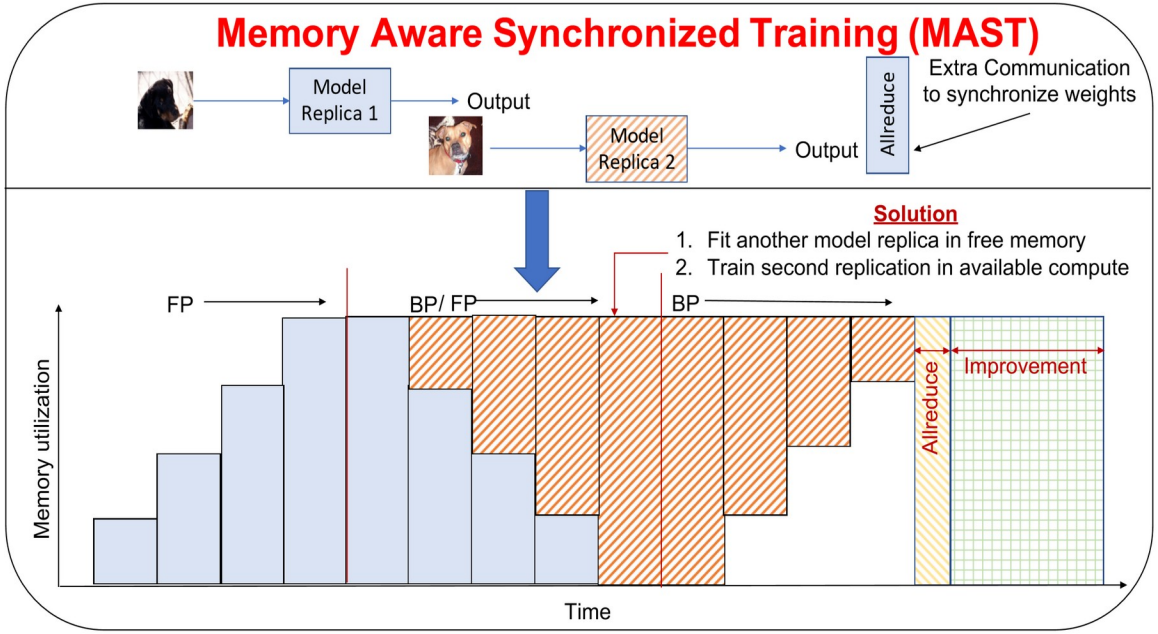
- Maximum Batch Size (BS) depends on the memory.
- For large models, maximum BS is 1 on multiple GPUs and it can affect the accuracy
- Basic Model Parallelism suffers from underutilization of memory and compute



A. Jain, A. Awan, A. Aljuhani, J. Hashmi, Q. Anthony, H. Subramoni, D. Panda, R. Machiraju, A. Parwani, "GEMS: GPU Enabled Memory Aware Model Parallelism System for Distributed DNN", SC '20

GEMS: GPU Enabled Memory Aware Model Parallelism Systems

- We propose GEMS to overcome the limitations
  - Memory Aware Synchronized Training (MAST)
  - Memory Aware Synchronized Training with Enhanced Replications (MASTER)
  - Hybrid Designs (Memory Aware designs + Data Parallelism)



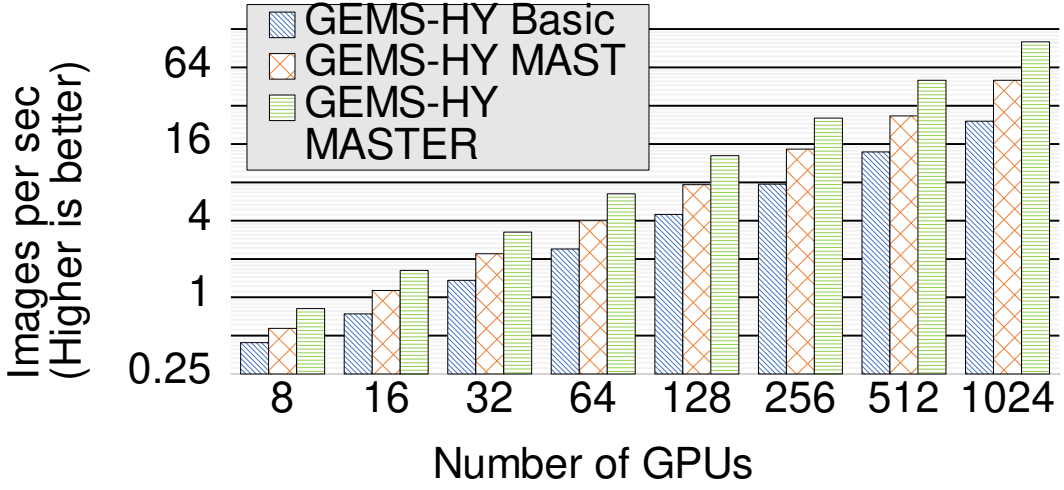
Enables the training of out-of-core batch sizes

A. Jain, A. Awan, A. Aljuhani, J. Hashmi, Q. Anthony, H. Subramoni, D. Panda, R. Machiraju, A. Parwani, "GEMS: GPU Enabled Memory Aware Model Parallelism System for Distributed DNN", SC '20

GEMS at Scale (1,024 V100 GPUs on LLNL Lassen)

- Hybrid Designs
  - GEMS-HY Basic (Model Parallelism + Data Parallelism)
  - GEMS-HY MAST (MAST + Data Parallelism)
  - GEMS-HY MASTER (MASTER + Data Parallelism)
- Setup
  - **ResNet-1k** on 512 X 512 images
  - 128 Replications on 1024 GPUs
- Scaling efficiency
  - **97.32%** on 1024 nodes using **MVAPICH2-GDR**

97.32% scaling efficiency on 1024 V100 GPUs using MVAPICH2-GDR

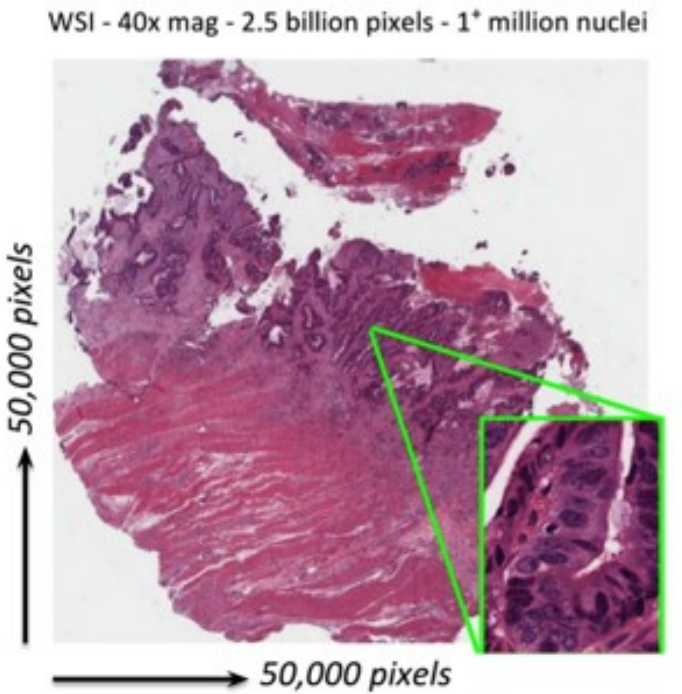


A. Jain, A. Awan, A. Aljuhani, J. Hashmi, Q. Anthony, H. Subramoni, D. Panda, R. Machiraju, A. Parwani, "GEMS: GPU Enabled Memory Aware Model Parallelism System for Distributed DNN", SC '20

Exploiting GEMS in AI-Driven Digital Pathology

- Pathology whole slide image (WSI)
  - Each WSI = 100,000 x 100,000 pixels
  - Can not fit in a single GPU memory
  - Tiles are extracted to make training possible
- Two main problems with tiles
  - Restricted tile size because of GPU memory limitation
  - Smaller tiles loose structural information
- Can we use GEMS to train on larger tiles to get better accuracy and diagnosis?
- Reduced training time significantly using **MVAPICH2-GDR**
  - **32 hours (1 node, 1 GPU) -> 7.25 hours (1 node, 4 GPUs) ->**

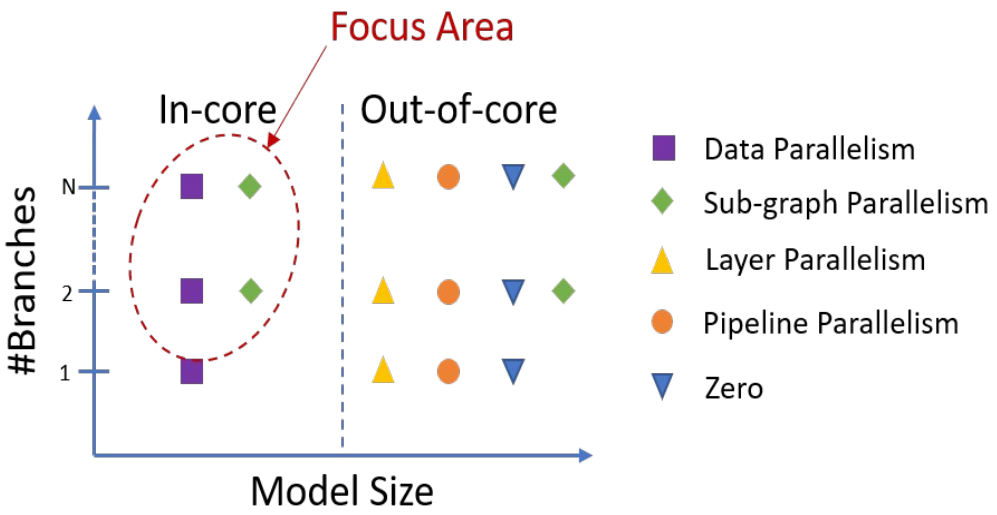
27 mins (32 nodes, 128 GPUs)



https://blog.kitware.com/digital-slide-archive-large-image-and-histomicstk-open-source-informatics-tools-for-management-visualization-and-analysis-of-digital-histopathology-data/

Strategies for Distributed Training of Transformers

- Data-Parallelism– only for models that fit the memory
- Out-of-core models
  - Cannot be fit inside single processing element memory
- Sub-graph parallelism can work for in-core and out-of-core models!
- Care is needed for good performance

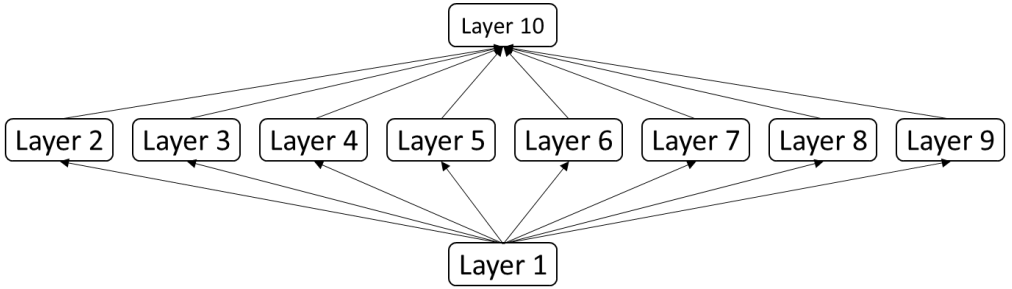


Multi-branch Deep Neural Networks

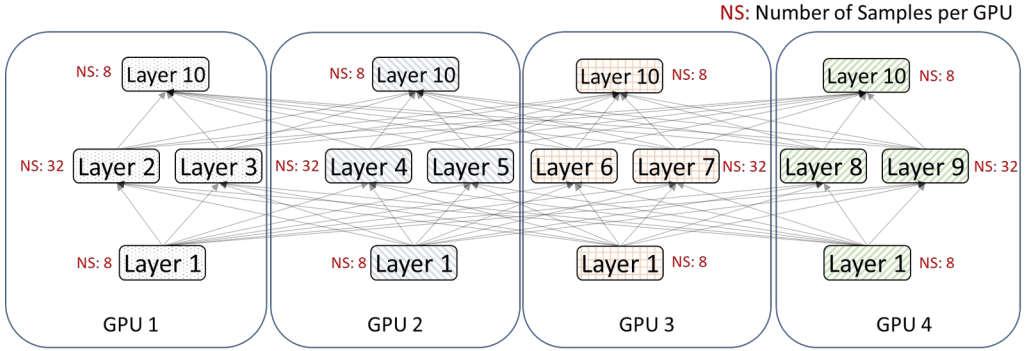
SUPER: SUb-Graph Parallelism for TransformERs

Sub-Graph Parallelism

- Exploits inherent parallelism in modern DNN architectures
- Improves the Performance of multi-branch DNN architectures
- Can be used to accelerate the training of state-of-the-art Transformer models
- Provides better than Data-Parallelism for in-core models



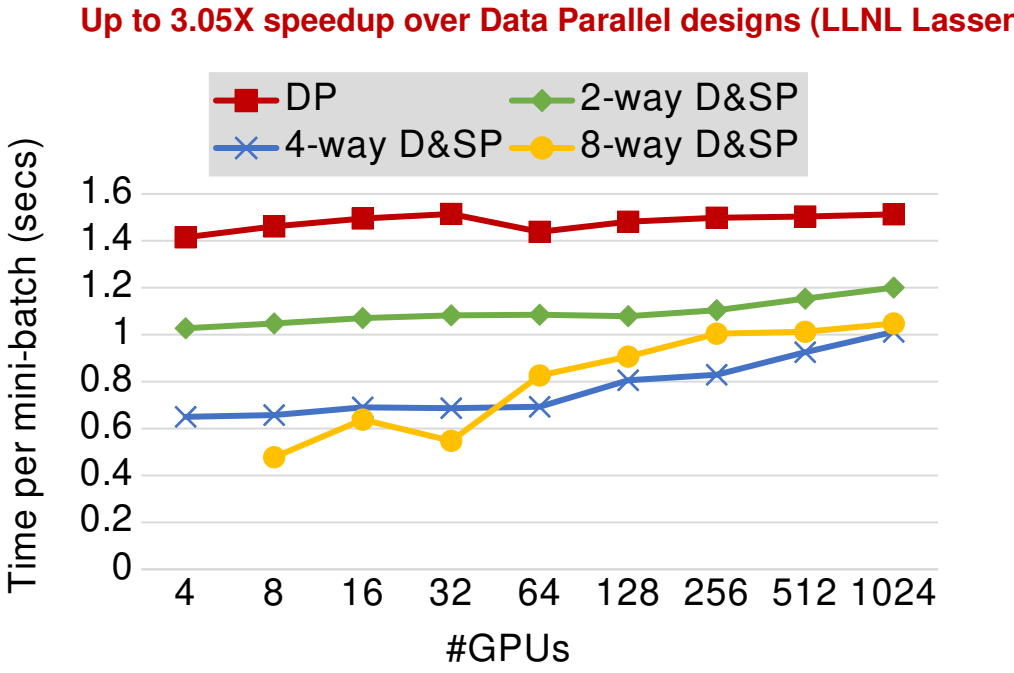
Simple example of a multi-branch DNN architecture



4-way Sub-Graph Parallelism combined with Data-Parallelism (D&S

Accelerating Transformers using SUPER

- We propose sub-graph parallelism integrated with data parallelism to accelerate the training of Transformers.
- Approach
  - Data and Sub-Graph Parallelism (D&SP)
    - #-way D&SP (#: number of sub-graphs)
- Setup
  - T5-Large-Mod on WMT Dataset
  - 1024 NVIDIA V100 GPUs
- Speedup
  - Up to **3.05X** over Data Parallelism (DP)



Conclusion

- Model Parallelism is needed to enable the DNN training on very large image sizes.
  - Basic model parallelism suffers from underutilization of compute and memory
  - Memory aware designs are needed to optimize the performance of model parallelism
- Proposed GEMS and SUPER
  - Memory aware designs to overcome the limitations of model parallelism
  - Hybrid designs to scale proposed designs to large number of nodes
- Achieved **97.32%** scaling efficiency on 1024 V100 GPUs using MVAPICH2-GDR for out-of-core CNNs and **3.05X** speedup over Data Parallelism for Transformers
- Future plan
  - Explore new dimensions of parallelism to further accelerate the DNN training on

Thank You!

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

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

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

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