Using Tensor Swapping and NVLink to Overcome GPU Memory Limits with TensorFlow

Sam Matzek
Deep learning is memory constrained

- GPUs have limited memory
- Neural networks are growing deeper and wider
- Amount and size of data to process is always growing
GPU Memory Usage

Input data

Kernels

Tensors (Layer outputs)

GPU memory

loss
Model Training in GPU Memory

GPU memory

Tensor 1

Tensor 2

Tensor 3

loss
Model Training with Tensor Swapping

GPU memory

Tensor 2

Tensor 4

loss

System memory
TensorFlow Large Model Support Graph Modifications

Enabling TensorFlow Large Model Support

Keras API

```python
from tensorflow_large_model_support import LMS
lms = LMS()
lms.batch_size = 1
# ...
model.fit_generator(generator=training_gen,
                      callbacks=[lms])
```

Estimator API

```python
from tensorflow_large_model_support import LMS
lms = LMS()
# ...
mnist_classifier.train(input_fn=train_input_fn, steps=20000,
                        hooks=[logging_hook, lms])
```

*Examples for TFLMS v2.0.0*
What’s possible with Large Model Support?

• 10x image resolution - Keras ResNet50
• 10x image resolution - DeepLabV3 2D image segmentation
• 5x MRI resolution - 3D U-Net 3D image segmentation

Measured with TFLMS v2.0.0 on TensorFlow 1.13, CUDA 10.1, cuDNN 7.5
3D U-Net image segmentation

• 3D U-Net generally has high memory usage requirements
• International Multimodal Brain Tumor Segmentation Challenge (BraTS)
• Existing Keras model with TensorFlow backend
Effect of 2x resolution on Dice Coefficients
(higher is better)
“Swapping makes everything slow”
Typical GPU connectivity

System Memory

CPU

System memory bus: 76.8 GB/s

PCIe: 32 GB/s

GPU

GPU

NVLIn
POWER9 CPU to GPU connectivity

System Memory

System memory bus: 170 GB/s
NVLink 2.0: 150 GB/s

CPU

GPU

GPU

NVLink 2.0
Effects of NVLink 2.0 on Large Model Support

PCLe connected GPU training one high res 3D MRI with large model support

NVLink 2.0 connected GPU training one high res 3D MRI with large model support
Effects of NVLink 2.0 on epoch times

Epoch times at high resolution with swapping

<table>
<thead>
<tr>
<th></th>
<th>Seconds</th>
</tr>
</thead>
<tbody>
<tr>
<td>POWER9 server with NVLink 2.0</td>
<td></td>
</tr>
<tr>
<td>GPU server with PCI</td>
<td></td>
</tr>
<tr>
<td>GPU server with PCI contention</td>
<td></td>
</tr>
</tbody>
</table>
Effects of NVLink 2.0 on GPU Utilization

- POWER9 server with NVLink 2.0: 100% GPU utilization
- GPU server with PCI: 20% GPU utilization
- GPU server with PCI contention: 20% GPU utilization

Memory Copy Throughput:
- 70 GB/s
- 60 GB/s
- 50 GB/s
- 40 GB/s
- 30 GB/s
- 20 GB/s
- 10 GB/s
- 0 GB/s
Multi-GPU model training with NVLink 2.0

2.1x faster with HALF the number of GPUs!

http://ibm.biz/3dunet-tflms-multigpu
Patches versus whole image

Epoch time (seconds)

- Patch based: 2000 seconds
- Whole image with large model support: 600 seconds

3x faster

Total training time (hours)

- Patch based: 120 hours
- Whole image with large model support: 20 hours

3.5x faster

https://arxiv.org/abs/1812.07816
Overhead of Large Model Support with NVLink 2.0

Measured with TFLMS v2.0.0 on TensorFlow 1.13, CUDA 10.1, cuDNN 7.5
Overhead of Large Model Support with NVLink 2.0

ResNet50 on POWER9 with 32GB NVIDIA Volta V100

Megapixels / second

Image resolution (squared)
DeepLabV3 on POWER9 with 32GB NVIDIA Volta V100

- 1.2 GB transferred to GPU, GPU utilization 81%
- LMS enabled
- 148 GB transferred to GPU, GPU utilization 90%
- 438 GB transferred to GPU, GPU utilization 89%
- 826 GB transferred to GPU, GPU utilization 84%
- 1.4 TB transferred to GPU, GPU utilization 64%

Using bs=16, fine_tune_batch_norm=true, measured on 32GB GPU with TensorFlow 1.13, CUDA 10.1, cuDNN 7.5
Large Model Support with NVLink 2.0

- Tensor swapping can be used to overcome GPU memory limits
- Allows training of:
  - deeper models
  - higher resolution data
  - larger batch sizes
- NVLink 2.0 between CPU and GPU allow tensor swapping with minimal overhead
More information

TensorFlow Large Model Support
https://github.com/IBM/tensorflow-large-model-support

TFLMS: Large Model Support in TensorFlow by Graph Rewriting

TensorFlow Large Model Support Case Study

Performance of 3DUnet Multi GPU Model for Medical Image Segmentation using TensorFlow Large Model Support
http://ibm.biz/3dunet-tflms-multigpu

Fast and Accurate 3D Medical Image Segmentation with Data-swapping Method
https://arxiv.org/abs/1812.07816

Data-parallel distributed training of very large models beyond GPU capacity
https://arxiv.org/abs/1811.12174

POWER9 server with NVLink 2.0 connections between CPU and GPU (IBM AC922):