deepspeed

Do More with Less:

Large Model Training and Inference with DeepSpeed

https://github.com/microsoft/DeepSpeed

LLMs in Production Part II | June 2023

MLOps Community

Samyam Rajbhandari

Co-founder and Architect for DeepSpeed

Microsoft

Model Scale
- 10+ Trillion parameters

Speed
- Fast & scalable training

Democratize AI
- Bigger & faster for all

Compressed Training
- Boosted efficiency

Accelerated inference
- Up to 12x faster & cheaper

Usability
- Few lines of code changes

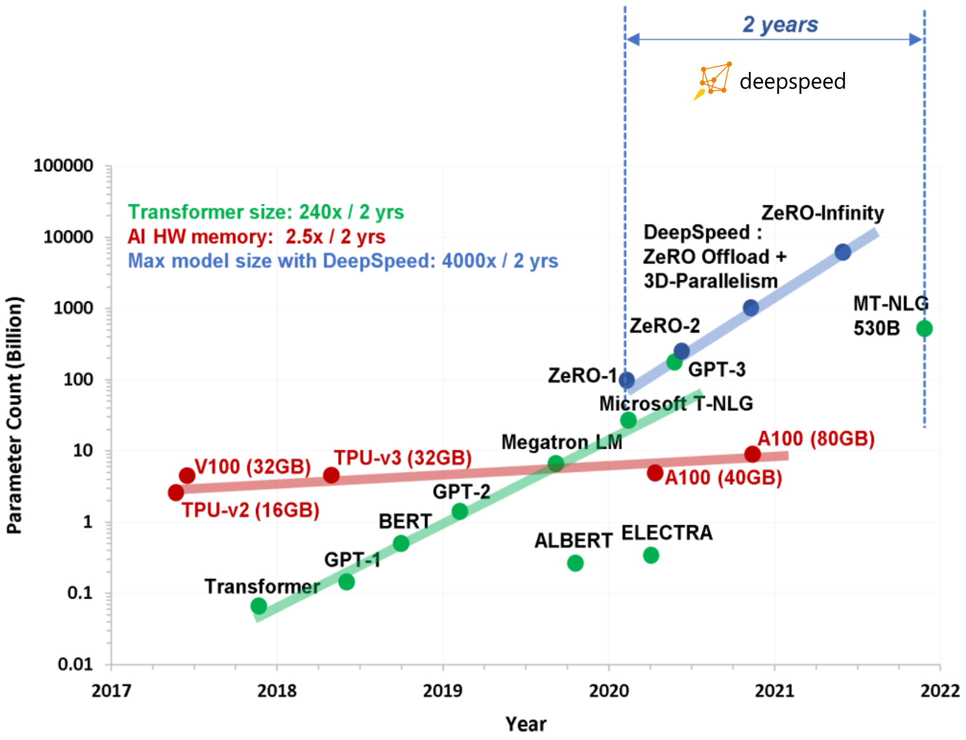
DeepSpeed: Reshaping the Large Model Training Landscape

DeepSpeed Powered Massive Models:

- METRO-LM **(5.4B)**
- Microsoft-Turing NLG (**17B**)
- GPT Neo-X **(20B)**
- AlexaTM **(20B)**
- YaLM **(100B)** Yandex
- GLM **(130B)**
- BLOOM: Big Science (**176B**)
- Jurrasic-1 (**178B**) **AI21labs**
- Megatron-Turing NLG (**530B**)
- ...

Key training technologies:

- ❑ Zero Redundancy Optimizer (ZeRO)
- ❑ ZeRO-Infinity
- ❑ 3D parallelism
- ❑ Memory and compute efficient MoE training
- ❑ Optimized CUDA/ROCm/CPU kernels
- ❑ Gradient compression 1-bit Adam/LAMB, 0/1 Adam
- ❑ Sparse Attention
- ❑ Mixture of quantization
- ❑ Progressive layer dropping
- ❑ Curriculum learning
- ❑ ...



System capability to efficiently train models with ***trillions of parameters***

Model Scale
- 10+ Trillion parameters

Speed
- Fast & scalable training

Democratize AI
- Bigger & faster for all

Compressed Training
- Boosted efficiency

Accelerated inference
- Up to 12x faster & cheaper

Usability
- Few lines of code changes

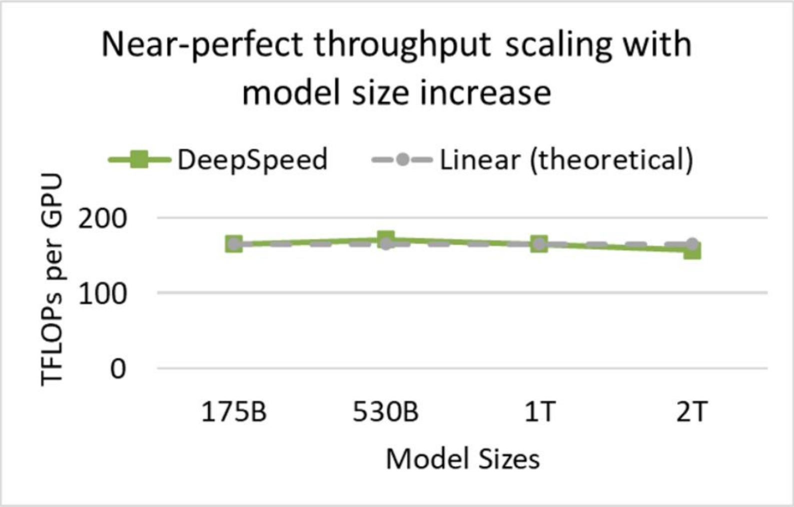
Fastest Transformer Kernels

⇩

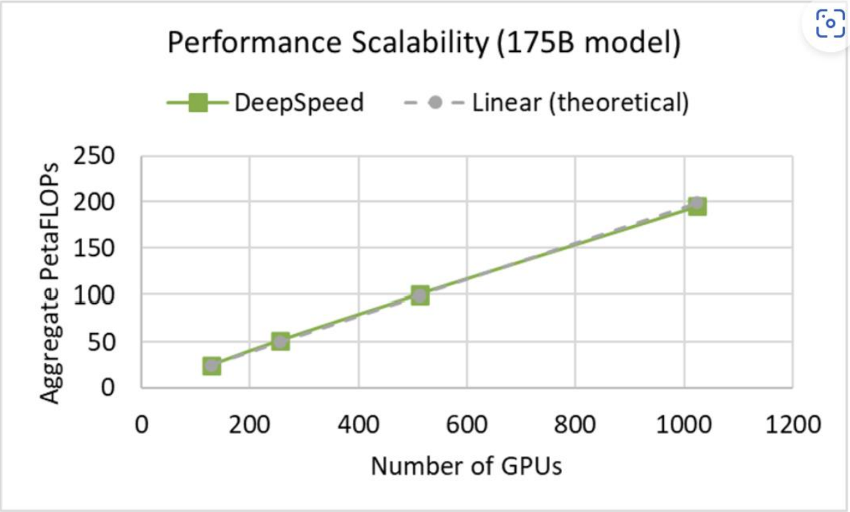
World Fastest BERT Training

#Devices	Source	Training Time
256 V100 GPUs	Nvidia	236 mins
256 V100 GPUs	DeepSpeed	144 mins
1024 TPU3 chips	Google	76 mins
1024 V100 GPUs	Nvidia	67 mins
1024 V100 GPUs	DeepSpeed	44 mins

Throughput Scaling on 1024 A100 Azure Cluster



(a)



(b)

[Azure empowers easy-to-use, high-performance, and hyperscale model training using DeepSpeed - DeepSpeed]

- Efficiency: ZeRO, ultra-fast GPU kernels, IO/compute/communication overlapping
- Effectiveness: Advance HP tuning, large-batch scaling

Model Scale
- 10+ Trillion parameters

Speed
- Fast & scalable training

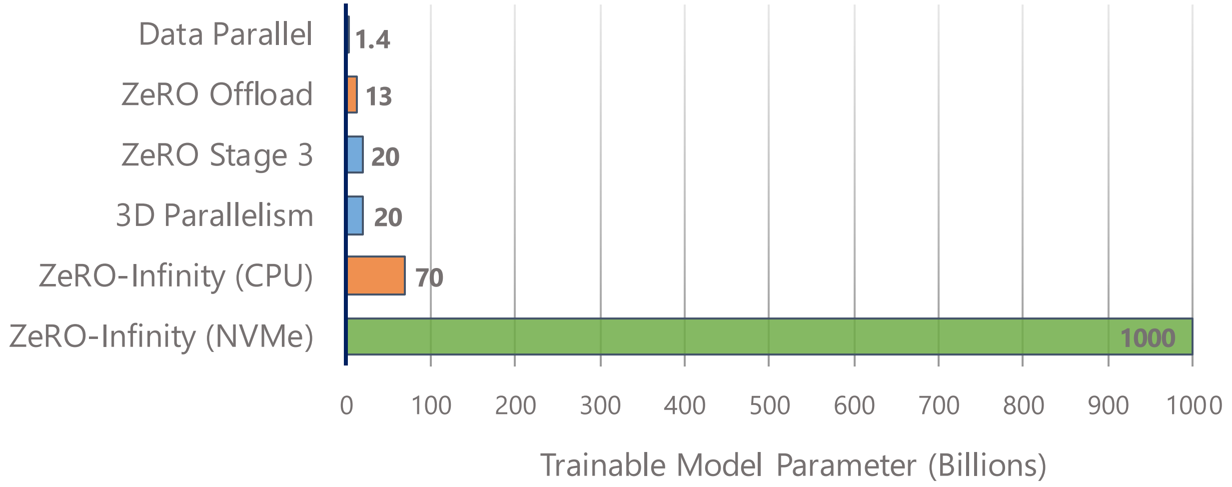
Democratize AI
- Bigger & faster for all

Compressed Training
- Boosted efficiency

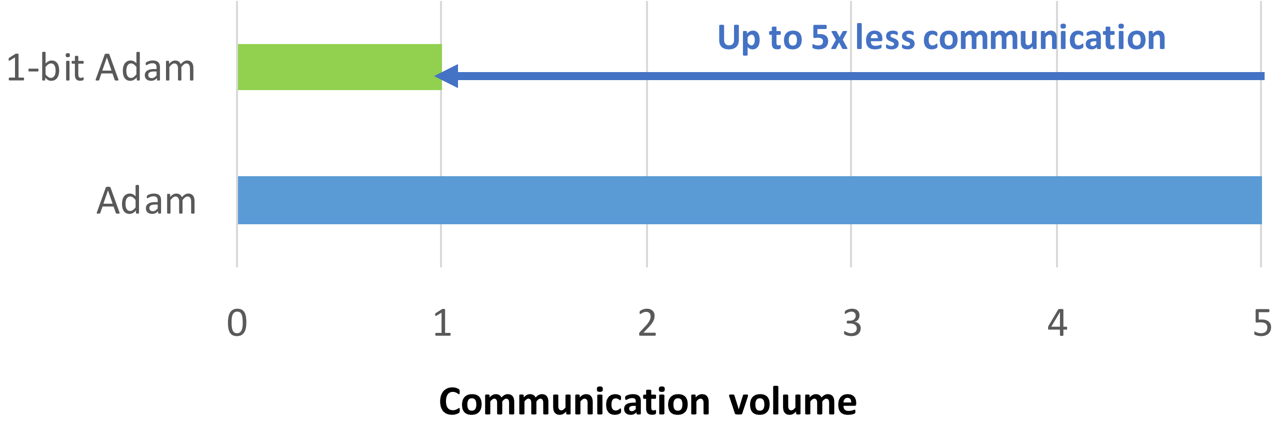
Accelerated inference
- Up to 12x faster & cheaper

Usability
- Few lines of code changes

ZeRO-Infinity: 1 Trillion model on a single GPU, 700x bigger



1-bit Adam: 5x less communication, 3.5x faster training



Model Scale
- 10+ Trillion parameters

Speed
- Fast & scalable training

Democratize AI
- Bigger & faster for all

Compressed Training
- Boosted efficiency

Accelerated inference
- Up to 12x faster & cheaper

Usability
- Few lines of code changes

DeepSpeed Journey

Let’s Train Bert Large! Early 2019, Microsoft

- Distributed Data Parallel Training
  - NVIDIA Apex
- Compute:
  - 64x V100 16GB
- Network:
  - **4 Gbps** Ethernet
  - For comparision DGX-2 SuperPod: 1600 Gbps
  - 400x slower
- Max Batch Size: 4 per GPU limited by memory
- **DeepScale:**
  - Smart Gradient Accumulation
    - Global Batch Size: 4K
    - Micro Batch: 4, Gradient Accumulation: 16
  - Bert in 8-days

DeepScale was the precursor to DeepSpeed

NVIDIA Apex

```
for i in range(iterations):

    for j in range(gas):

        loss = model.forward( get_batch() )
        local_gradients = backward_gradients(loss/gas)
        average_gradients += distributed.reduce(local_gradients/gpus)

    optimizer.step(average_gradients)
```

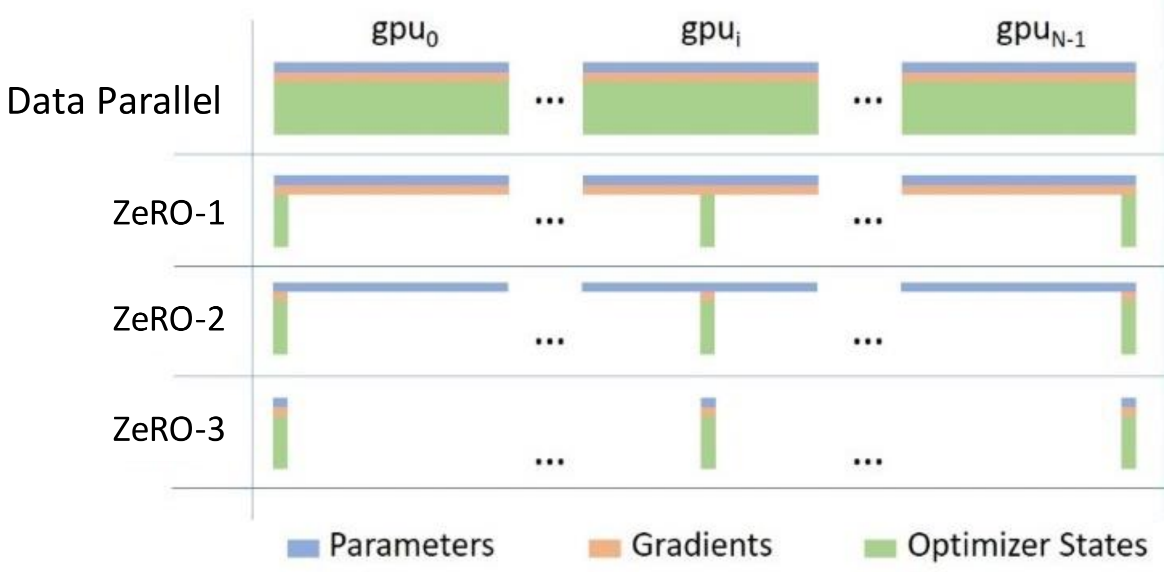
DeepScale

```
for i in range(iterations):

    for j in range(gas):

        loss = model.forward( get_batch() )
        local_gradients += backward_gradients(loss/gas)

    average_gradients = distributed.reduce(local_gradients/gpus)
    optimizer.step(average_gradients)
```

ZeRO and DeepSpeed

- Multi-billion parameter models in 2019
  - 1.5B GPT-2, 8.3B Megatron
- Data Parallel replicates model states
  - Limited by Single GPU memory
- Model Parallel incurs high communication
  - Limited within a single Node
- Zero Redundancy Optimizer (ZeRO)
  - Data Parallel without Replication
  - Partitions Optimizer States, Gradients, and Parameters
- Microsoft Turing-NLG 17B
  - Largest LLM at the time
  - ZeRO + MP
- **DeepScale → DeepSpeed**
  - And it's a Palindrome

	Max Parameter (in billions)	Max Parallelism	Compute Efficiency	Usability (Model Rewrite)
Data Parallel (DP)	Approx. 1.2	>1000	Very Good	Great
Model Parallel (MP)	Approx. 20	Approx. 16	Good	Needs Model Rewrite
MP + DP	Approx. 20	> 1000	Good	Needs Model Rewrite
ZeRO	*> 1000*	*> 1000*	Very Good	Great



ZeRO was open-sourced and released with the DeepSpeed Library, 2020

Models Trained: TNLG-17B, Bloom-176B, GPT-NeoX, MPT, Alexa-TM, Metro-LM, etc

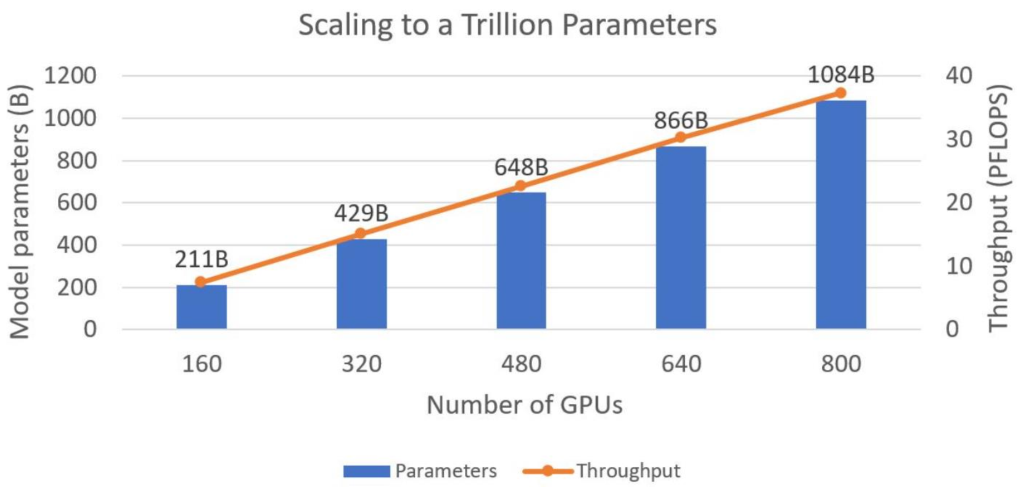
A Trillion Parameters!

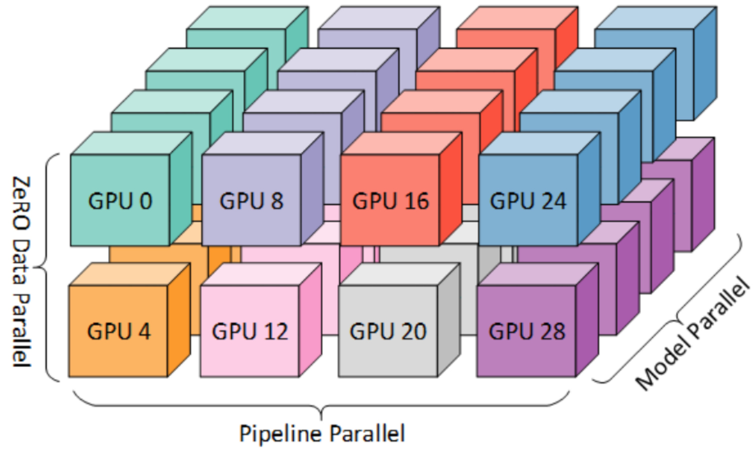
- June 2020: Open-AI GPT-3 175B
- What would it take to get to a trillion parameters?
- Two possible approaches
  - ZeRO-3
  - **3D Parallelism**

- **3D Parallelism**
  - Pipeline, Model and ZeRO Parallelism
  - Extremely good Compute Efficiency

- Megatron-Turing 530B
  - 2months+ and 2K A100 GPUs
  - 270B tokens
  - Microsoft NVIDIA collaboration

- Bloom-176B
  - 3.5 months and 384 GPUs
  - 350B Tokens
  - Collaborations across dozens of organizations

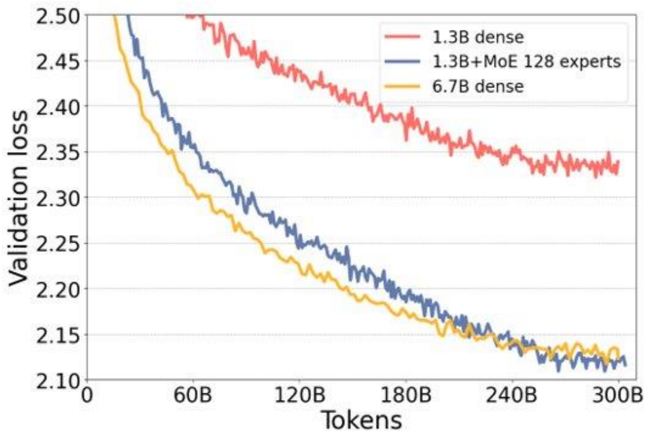




	Max Parameter (in billions)	Max Parallelism	Compute Efficiency	Usability (Model Rewrite)
Data Parallel (DP)	Approx. 1.2	>1000	Very Good	Great
Model Parallel (MP)	Approx. 20	Approx. 16	Good	Needs Model Rewrite
MP + PP + DP	> 1000	> 1000	Excellent	Needs Significant Model Rewrite
ZeRO	*> 1000*	*> 1000*	Very Good	Great

End of dense scaling?

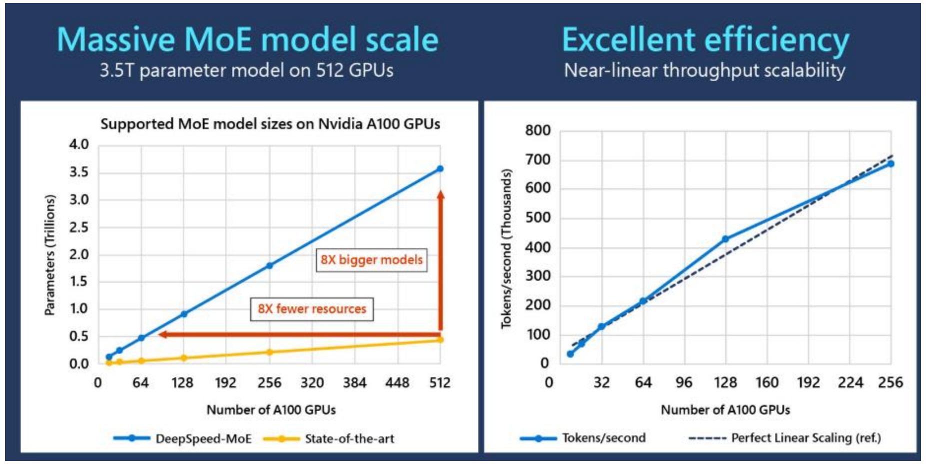
- Megatron-Turing 530B
  - Over 2 months on 2K A100 GPUs
  - < 300B Tokens
  - Under-trained
- Training tokens on recent models
  - LAMMA 65B → 1.2 Trillion Tokens
  - MPT 7B → 1T Trillion Tokens
- 500B-1T model on a trillion tokens
  - ***6 months – 1 year*** on 2K GPUs
  - ***10T → 10 years***
- **Scale with Sparsity**
  - Mixture of Experts
  - 5x reduction in training cost



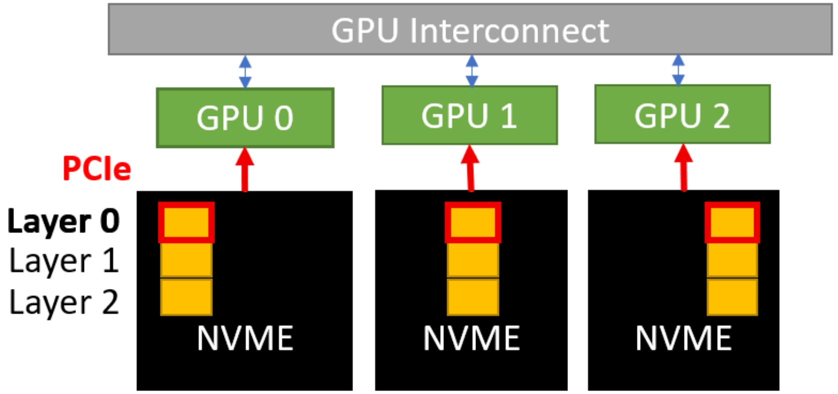
Training Throughput on 128 A100

	Training samples per sec	Throughput gain/ Cost Reduction
6.7B dense	70	1x
1.3B+MoE-128 (52B Total)	372	5x

DeepSpeed-MoE for training multi-trillion parameter MoE models with excellent efficiency



Democratization of LLMs



- Accessibility to large model training
  - 256 V100 GPUs to fine-tune GPT-3 175B model
  - Limited access to such resources

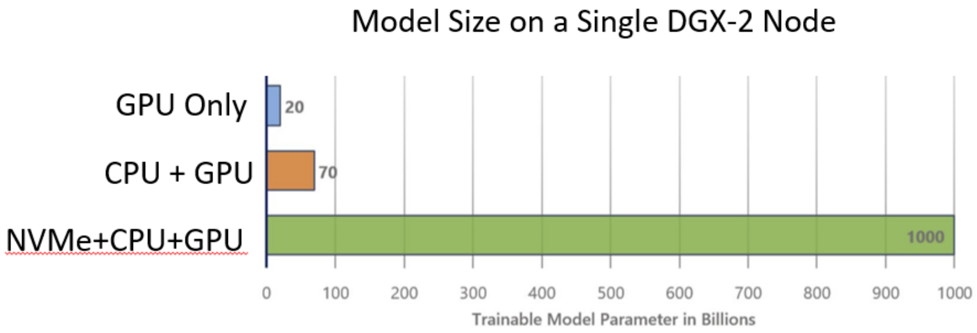
- Can we leverage GPU/CPU/NVMe memory
  - 32T params on 32 nodes
  - 1T params on a single node

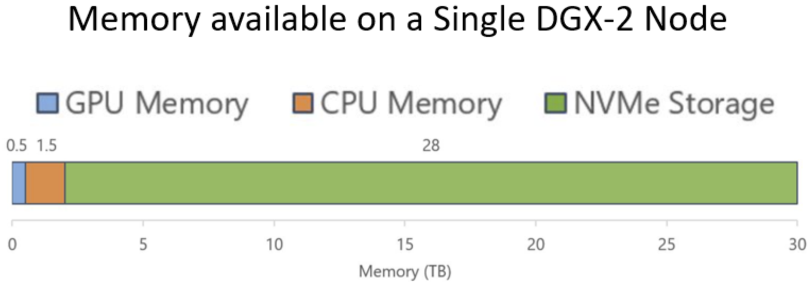
- Bandwidth: PCIe < NVME < CPU << GPU
  - PCIe → 16GB/s 1x
  - NVMe → 30 GB/s 2x
  - CPU → 200 GB/s 12x
  - GPU → 2TB /s 120x

ZeRO-Infinity

- Partition each parameter across GPUs
- Send from NVMe to GPU in parallel
- Bandwidth Increases linearly with devices
  - 8 Node Cluster: 30 → 240 GB/s

- *Finetune 100B+ parameter models on a single GPU*
- *Eliminate barrier to entry*





deepspeed

Do More with Less:

Large Model Training and Inference with DeepSpeed

https://github.com/microsoft/DeepSpeed

LLMs in Production Part II | June 2023

Samyam Rajbhandari

Co-founder and Architect for DeepSpeed

Microsoft

Model Scale

- 10+ Trillion parameters

Speed

- Fast & scalable training

Democratize AI

- Bigger & faster for all

Compressed Training

- Boosted efficiency

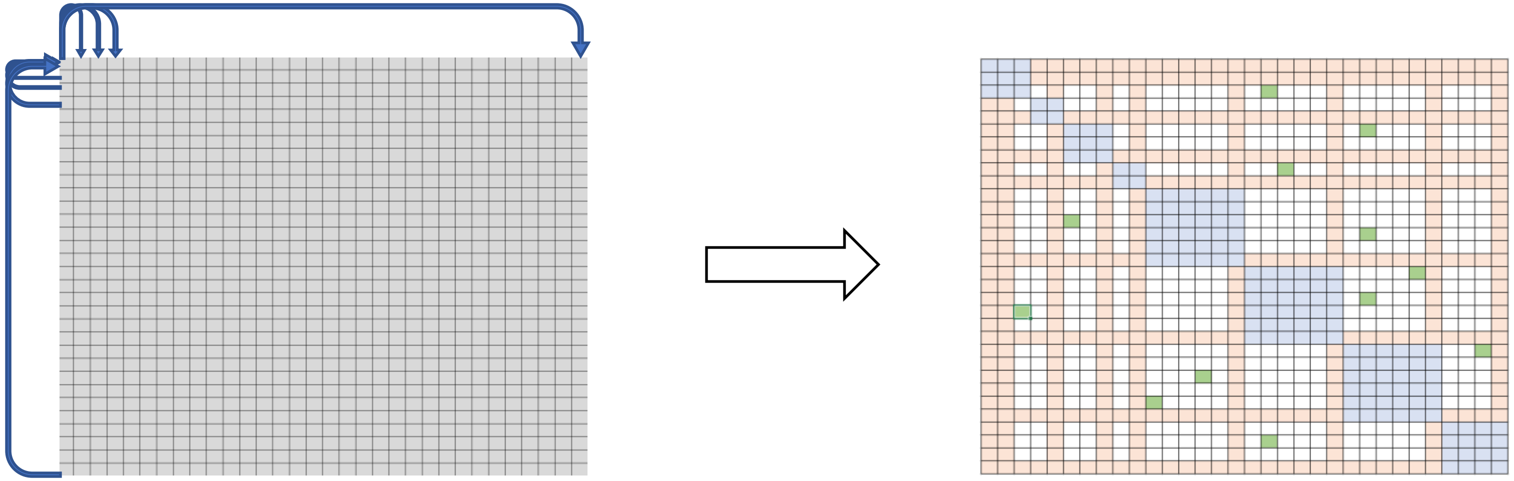
Accelerated inference

- Up to 12x faster & cheaper

Usability

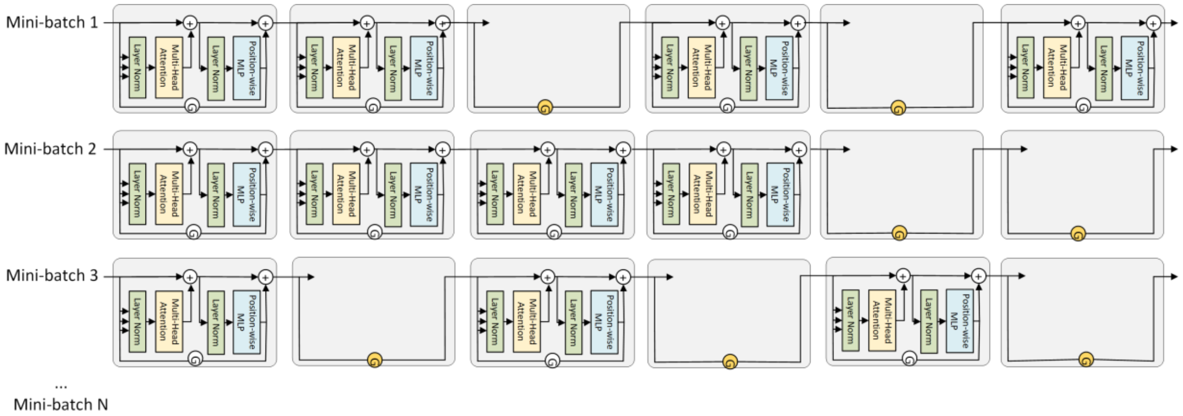
- Few lines of code changes

- **Sparse attention**: 10x longer seq, up to 6x faster



- **Progressive Layer Drop**: Compressed robust training

  - 24% faster when training the same number of samples
  - 2.5X faster to get similar accuracy on downstream tasks



Model Scale
- 10+ Trillion parameters

Speed
- Fast & scalable training

Democratize AI
- Bigger & faster for all

Compressed Training
- Boosted efficiency

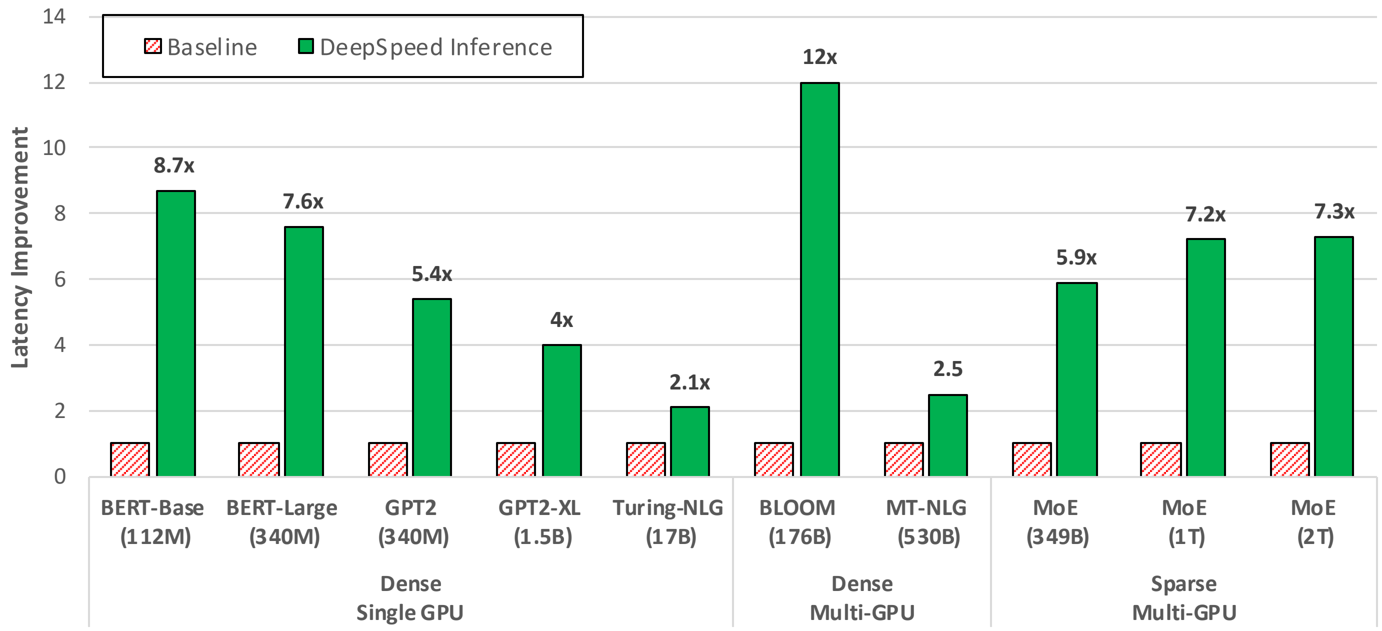
Accelerated inference
- Up to 12x faster & cheaper

Usability
- Few lines of code changes

DeepSpeed Accelerated inference for large-scale transformer models

A systematic composition of diverse set of optimizations

- ❑ Many-GPU Dense transformer optimizations – *powering large and very large models like Megatron-Turing 530B*
- ❑ Massive Scale Sparse Model Inference – *a trillion parameter MoE model inference under 25ms*
- ❑ ZeRO-Inference –> *40x bigger model inference on single-GPU device*



DeepSpeed Inference: SoTA latency and throughput across the large model inference landscape

Model Scale
- 10+ Trillion parameters

Speed
- Fast & scalable training

Democratize AI
- Bigger & faster for all

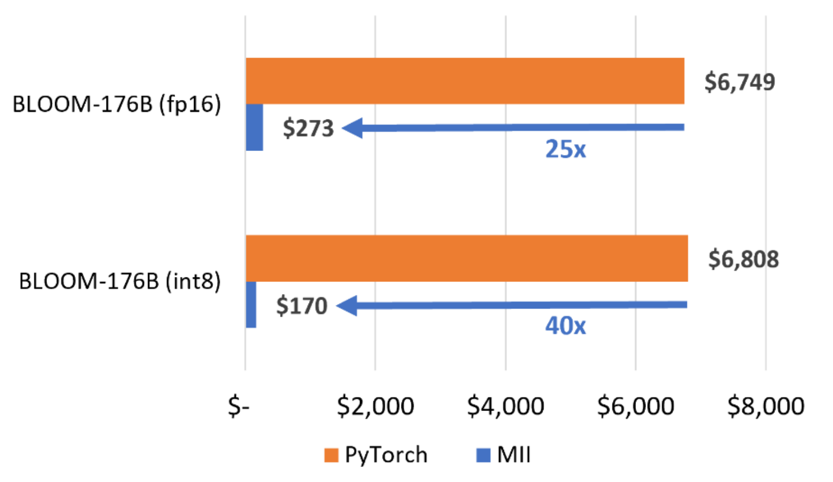
Compressed Training
- Boosted efficiency

Accelerated inference
- Up to 12x faster & cheaper

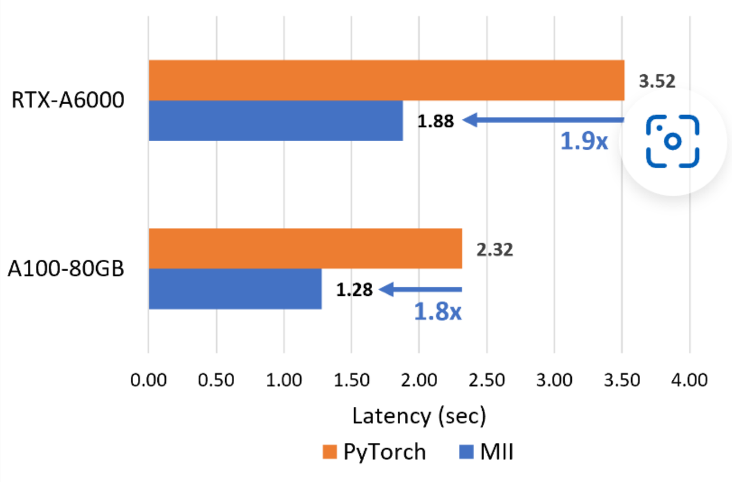
Usability
- Few lines of code changes

DeepSpeed-MII powered by DeepSpeed-Inference

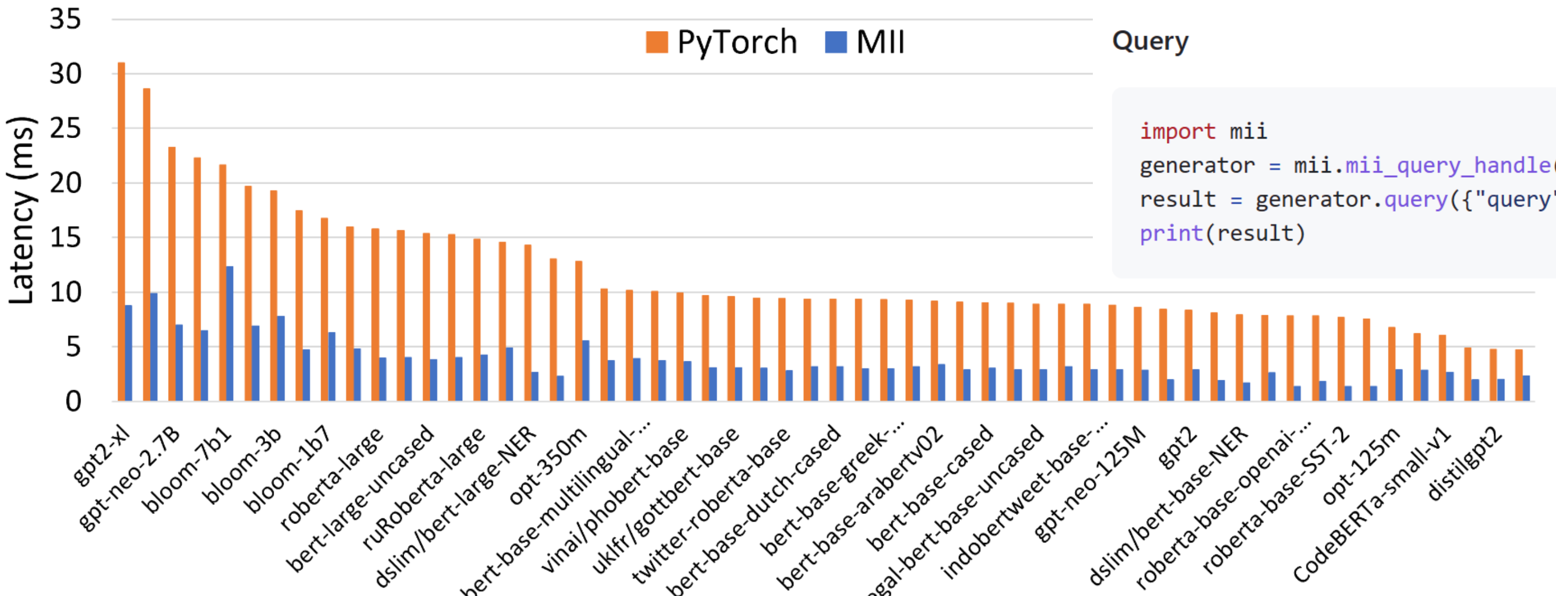
40x cost reduction to generate 1M tokens on Azure



State-of-the-art Stable Diffusion latency



DeepSpeed-MII accelerates 24,000+ different models



Deployment

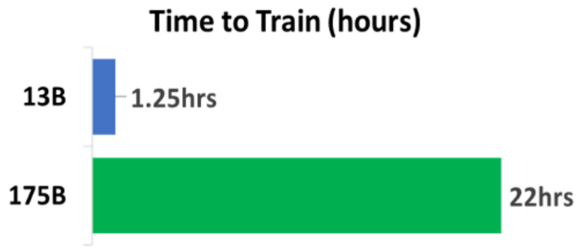
```python
import mii
mii_configs = {"tensor_parallel": 1, "dtype": "fp16"}
mii.deploy(task="text-generation",
           model="bigscience/bloom-560m",
           deployment_name="bloom560m_deployment",
           mii_config=mii_configs)
```

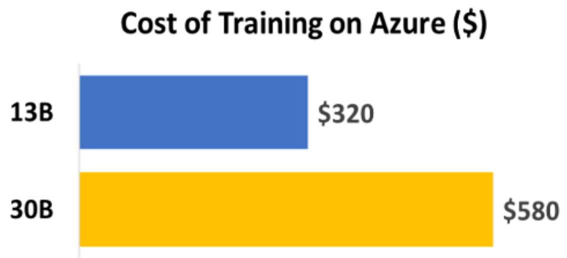
Query

```python
import mii
generator = mii.mii_query_handle("bloom560m_deployment")
result = generator.query({"query": ["DeepSpeed is", "Seattle is"]}, do_sample=True, max_new_tokens=30)
print(result)
```

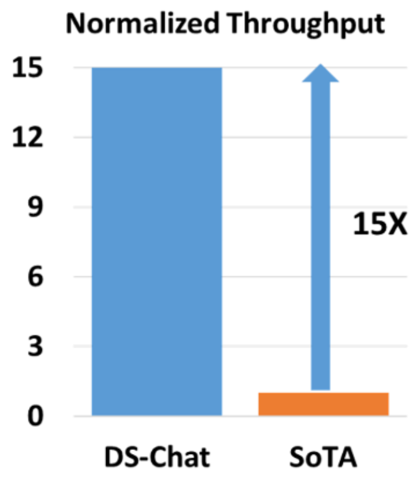
DEEPSPEED CHAT

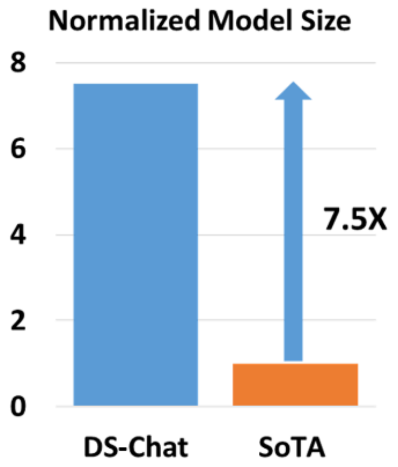
Fast Training with Affordable Cost





Train 15X Faster and Scale to 5x Bigger Models than SOTA RLHFs





Easy-Breezy Training

A complete end-to-end RLHF training experience with a single click

High Performance System

Hybrid Engine achieves 15X training speedup over SOTA RLHF systems with unprecedented cost reduction at all scales

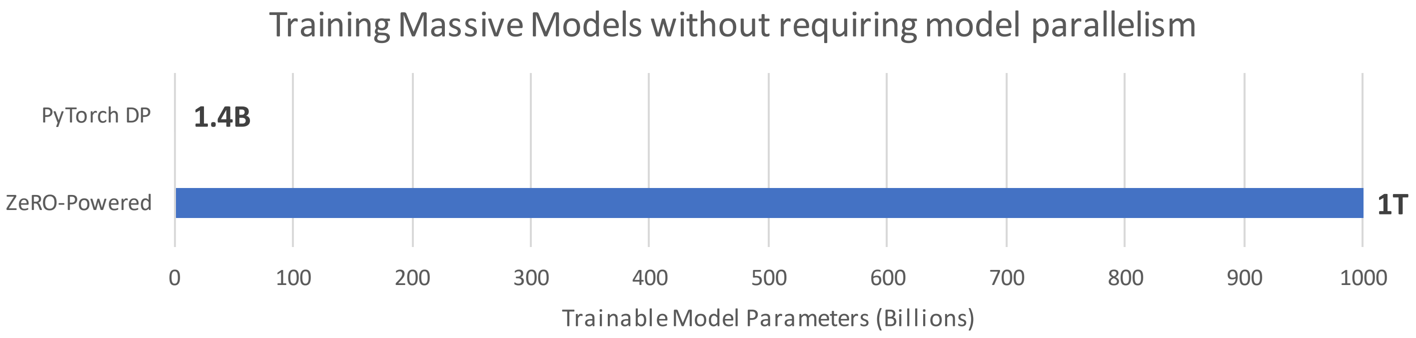
Accessible Large Model Support

Training ChatGPT-Style models with tens to hundreds of billions parameters on a single or multi-GPUs through ZeRO and LoRA

A Universal Acceleration Backend for RLHF

Support InstructGPT pipeline and large-model finetuning for various models and scenarios

- Only few lines of code changes to enable DeepSpeed on PyTorch models
- Scalable and convenient data parallelism



- HuggingFace and PyTorch Lightning integrate DeepSpeed as a performance-optimized backend

```
deepspeed examples/pytorch/translation/run_translation.py \
--deepspeed tests/deepspeed/ds_config_zero3.json \
--model_name_or_path t5-small --per_device_train_batch_size 1 \
--output_dir output_dir --overwrite_output_dir --fp16 \
```

```
trainer = Trainer(gpus=4, plugins='deepspeed', precision=16)
```

deepspeed.py hosted with ❤ by GitHub   view raw

- Infrastructure agnostic, supporting AzureML, Azure VMs, local-nodes

Model Scale
- 10+ Trillion parameters

Speed
- Fast & scalable training

Democratize AI
- Bigger & faster for all

Compressed Training
- Boosted efficiency

Accelerated inference
- Up to 10x faster & cheaper

Usability
- Few lines of code changes

DeepSpeed/ZeRO Usability

```python
# construct torch.nn.Module
model = MyModel()

# wrap w. DeepSpeed engine
engine, *_ = deepspeed.initialize(
    model=model,
    config=ds_config)

# training-loop w.r.t. engine
for batch in data_loader:
  loss = engine(batch)
  engine.backward(loss)
  engine.step()
```

```python
ds_config = {
  "optimizer": {
    "type": "Adam",
    "params": {"lr": 0.001}
  },
  "zero": {
    "stage": 3,
    "offload_optimizer": {
        "device": "[cpu|nvme]"
    },
    "offload_param": {
        "device": "[cpu|nvme]"
    }
  }
}
```

DeepSpeed OSS Community

- 4 Million+ installs since release in 2020
- 200 unique contributors
- 1000 public packages have hard dependencies on DeepSpeed
  - Open-source frameworks
    - Hugging Face, PyTorch-Lightning, EleutherAI, MosaicML, etc.
  - External companies
    - Meta AI (FAIR), AstraZeneca, Fidelity, Salesforce, Intel, Bloomberg, Tencent, SAP, etc.
  - National Labs
    - Oak Ridge, Argonne, Lawrence Livermore, etc.

Thank You!

deepspeed

We welcome contributions! Make your first pull request ☺

Please star our repo if you enjoyed this talk!

https://github.com/microsoft/DeepSpeed

www.deepspeed.ai

Follow us on twitter: @MSFTDeepSpeed

ZeRO-Infinity in Action