Hewlett Packard Enterprise

TOWARDS TRAINING TRILLION PARAMETER MODELS ON HPE GPU SYSTEMS

Manjunath Sripadarao, Pierre Carrier

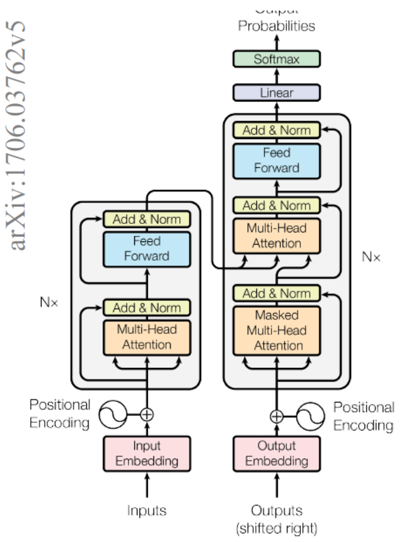
May 11, 2023

CUG paper: “Accelerating the Big Data Analytics Suite”
on AMD Tensile for **GEMM** operations

CONTENT

Attention Is All You Need

arXiv:1706.03762v5



GPT-2; GPT-3; …GPT-4

ZeRO: Memory Optimizations Toward Training Trillion Parameter Models

arXiv:1910.02054v3

- DeepSpeed
- ZeRO - Zero Redundancy Optimizer
- Megatron-DeepSpeed

- **Experimental Results**
- **Our Findings**
- **Conclusion**

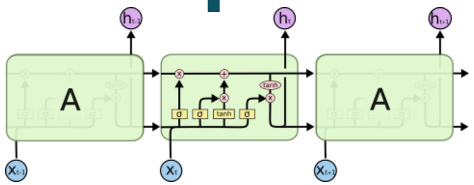
2017

2020

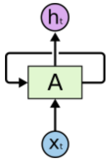
2023

CONTENT

Origin



https://colah.github.io/posts/2015-08-Understanding-LSTMs/



Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation

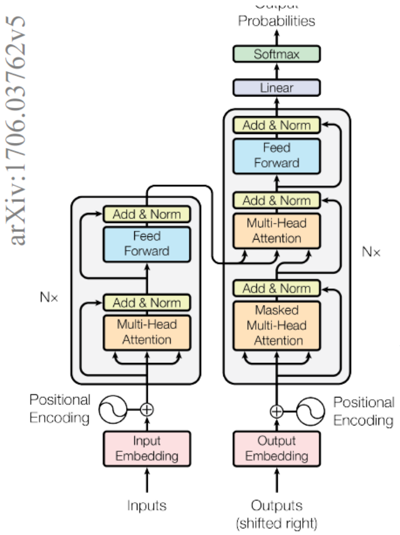
arXiv:1406.1078v3

RNN

...2012-2014

Attention Is All You Need

arXiv:1706.03762v5



2017

GPT-2; GPT-3; ...GPT-4

- DeepSpeed
- ZeRO - Zero Redundancy Optimizer
- Megatron-DeepSpeed

ZeRO: Memory Optimizations Toward Training Trillion Parameter Models

arXiv:1910.02054v3

2020

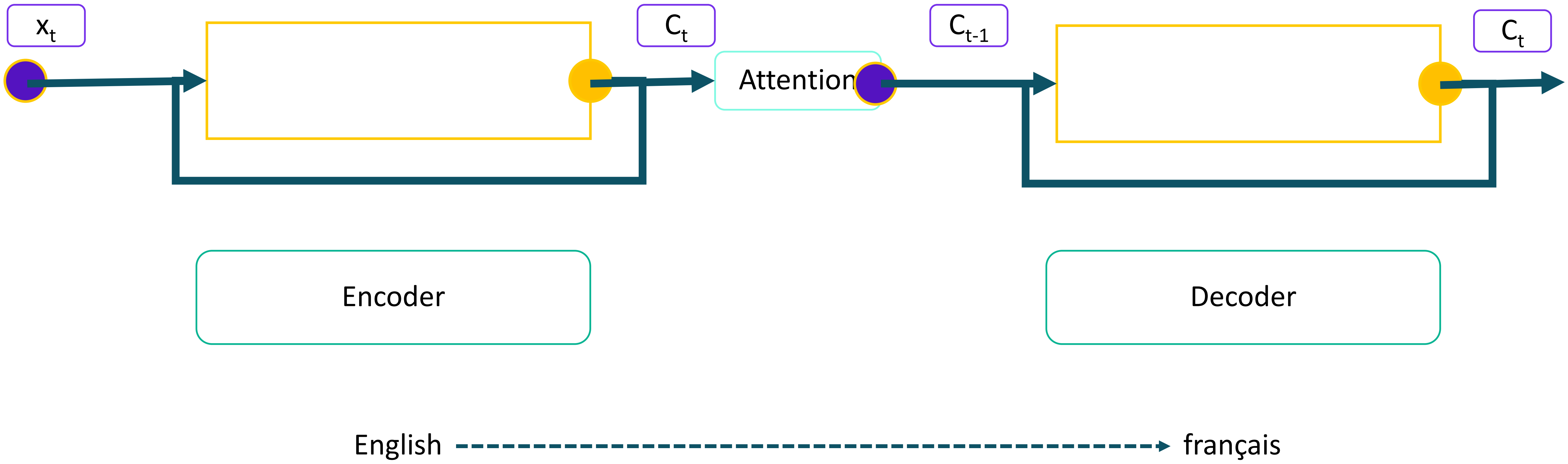
- **Experimental Results**
- **Our Findings**
- **Conclusion**

2023

ORIGIN: ENCODER-DECODER BASED LSTM MODEL

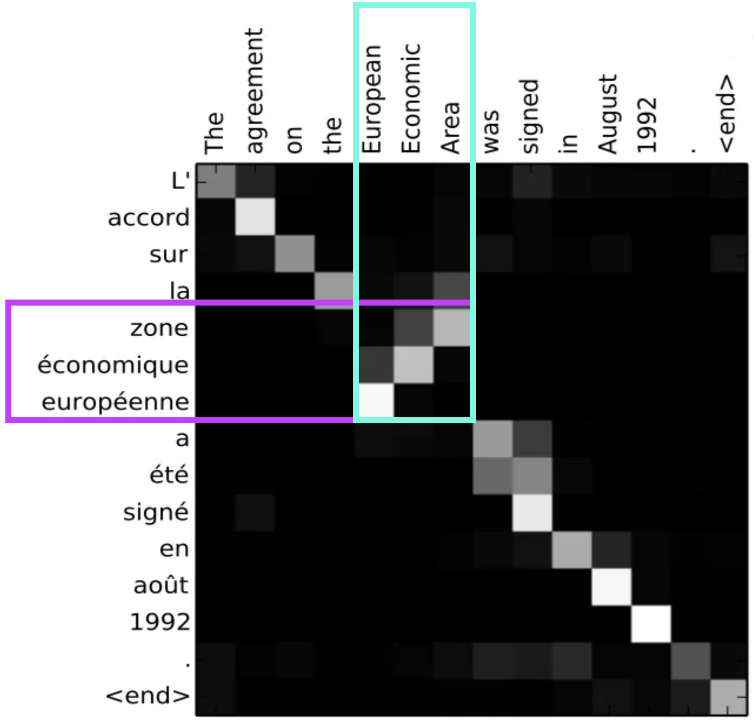
Long Short-Term Memory (LSTM)

- The precursor to the Transformer was an auto-regressive LSTM (where the output of previous step is fed back as input of next step). This looping caused the model to become slow to train.
- If the number of tokens is n then at least n steps were needed to train. **Transformer can be trained in 2 steps.**

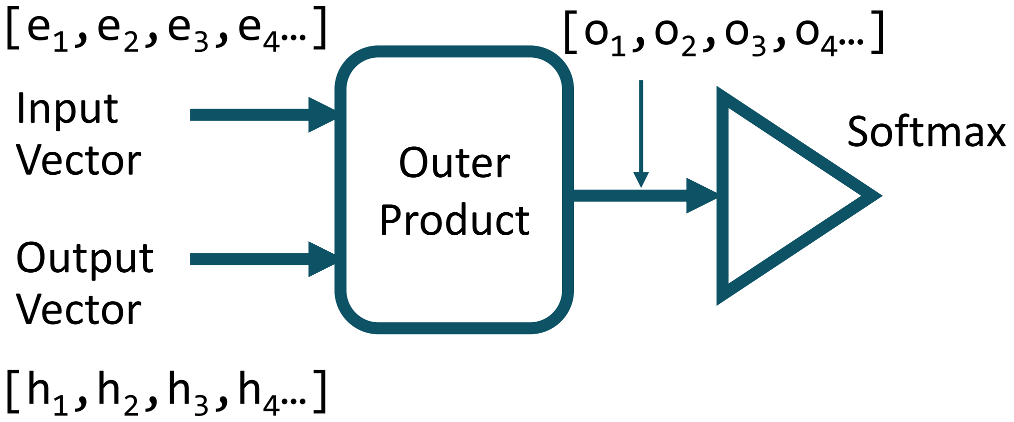


ORIGIN: ATTENTION

- LSTMs based Encoder-Decoder architecture force input sequences to a *fixed length context vector*.
- This puts a limitation on the length on the sentence that they can predict. To overcome this attention mechanism was introduced.
- Originally, no explicit modeling of long- and short-range dependencies.
- To solve some of these problems, researchers created a technique for paying **attention** to specific words.



$$\frac{\sum_{i=0}^{n} o_n}{n}$$



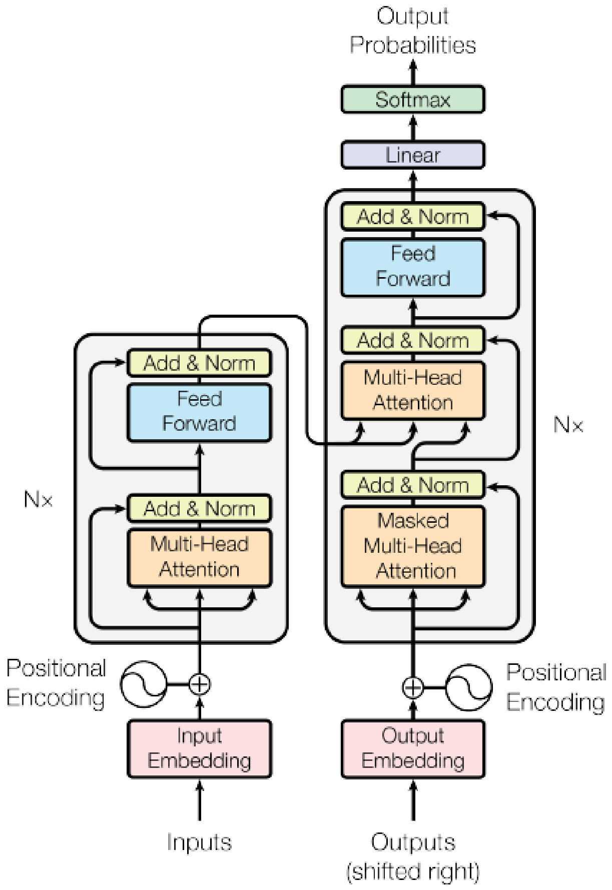
TRANSFORMER EMBEDDINGS AND TOKENIZATION

- **Transformer** is sequence to sequence neural network architecture.
- Input text is encoded with tokenizers to sequence of integers called input tokens.
- Input tokens are mapped to sequence of vectors (word embeddings) via embeddings layer.
- Output vectors (embeddings) can be classified to a sequence of tokens.
- Output tokens can then be decoded back to a text.

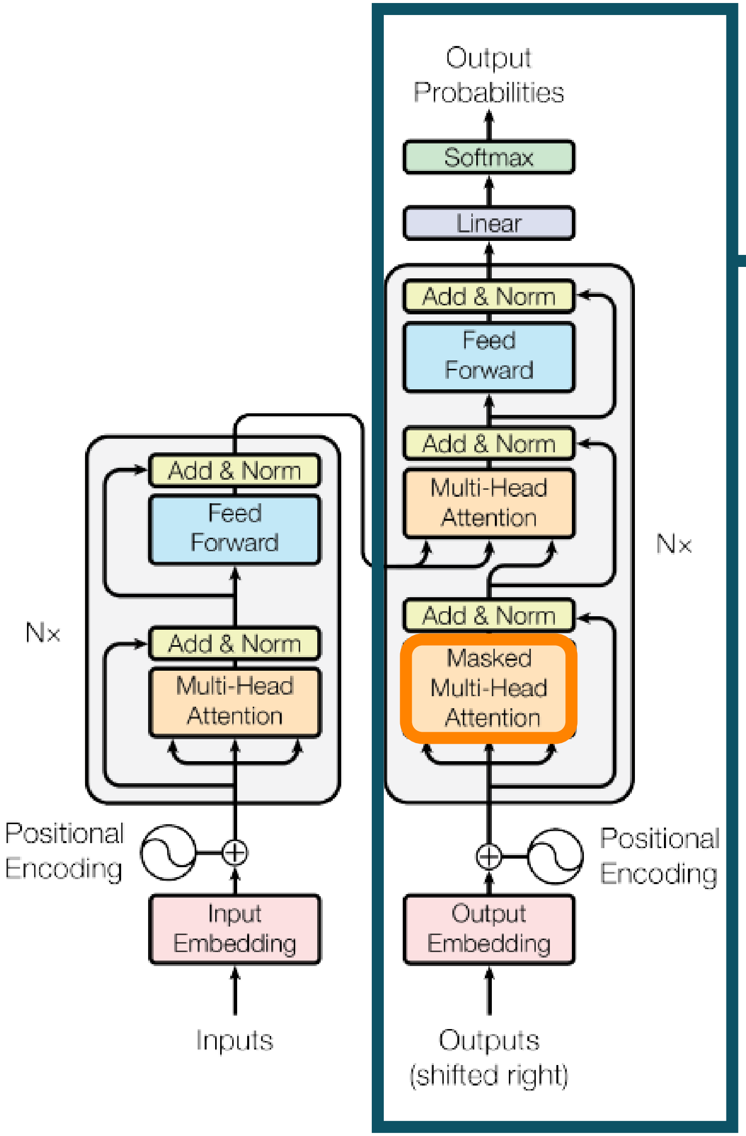
Tokenization is cutting input data into meaningful parts
that can be embedded into a vector space.

Embedding layers map tokens to word vectors
(sequence of numbers) called word embeddings.

https://arxiv.org/pdf/1706.03762.pdf



GEMM OPERATIONS

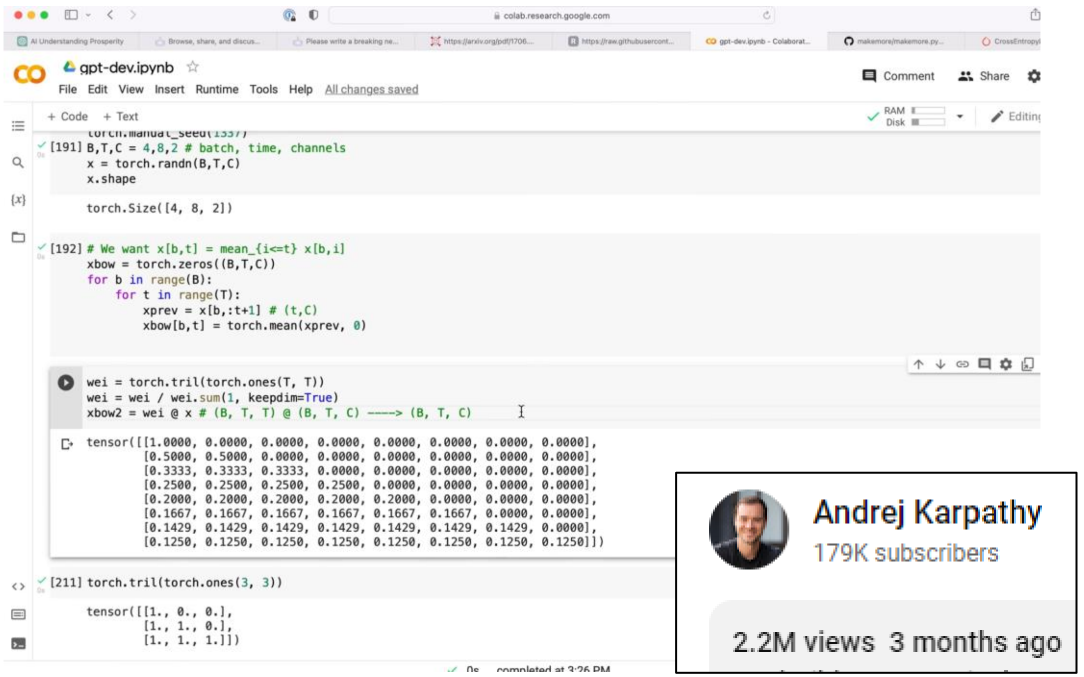


Nice intro

On decoder

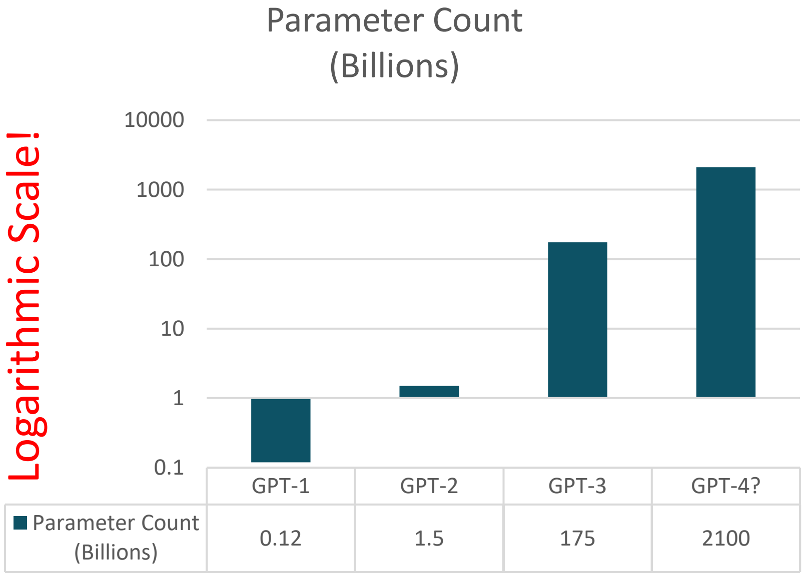
“Let's build GPT: from scratch, in code, spelled out”

https://www.youtube.com/watch?v=kCc8FmEb1nY



Nice discussion on GEMM (1h mark on masked/filtered attention)

GPT: THE MOST FAMOUS TRANSFORMER MODELS



	Architecture	Parameter count	Training data
GPT-1	12-level, 12-headed Transformer decoder (no encoder), followed by linear-softmax.	0.12 billion	BookCorpus:[19] **4.5 GB** of text, from 7000 unpublished books of various genres.
GPT-2	GPT-1, but with modified normalization	1.5 billion	WebText: **40 GB** of text, 8 million documents, from 45 million webpages upvoted on Reddit.
GPT-3	GPT-2, but with modification to allow larger scaling.	175 billion	**570 GB** plaintext, 0.4 trillion tokens. Mostly CommonCrawl, WebText, English Wikipedia, and two books corpora (Books1 and Books2)

…OpenAI: ChatGPT “3.5”; chatGPTplus: GPT-4

GPT-3: LANGUAGE MODELS ARE FEW-SHOT LEARNERS

- Massive language model of size 175 Billion parameters, released in June, 2020. It requires 800GB to store on the disk.
- Trained with 300 billion tokens.
- $3.64*10^3$ PF-days to train.
- On 1024 * 8-GPU A100s nodes researchers estimate it would take **34 days** to train the model.
- It was trained on V100s using Microsoft's infrastructure.

GPT-3 training data		
Dataset	# tokens	Proportion within training
Common Crawl	410 billion	60%
WebText2	19 billion	22%
Books1	12 billion	8%
Books2	55 billion	8%
Wikipedia	3 billion	3%

GPT-2

In a shocking finding, scientist discovered a herd of unicorns living in a remote, previously unexplored valley, in the Andes Mountains. Even more surprising to the researchers was the fact that the unicorns spoke perfect English.

The scientist named the population, after their distinctive horn, Ovid's Unicorn. These four-horned, silver-white unicorns were previously unknown to science. Now, after almost two centuries, the mystery of what sparked this odd phenomenon is finally solved.

Dr. Jorge Pérez, an evolutionary biologist from the University of La Paz, and several companions, were exploring the Andes Mountains when they found a small valley, with no other animals or humans. Pérez noticed that the valley had what appeared to be a natural fountain, surrounded by two peaks of rock and silver snow.

Pérez and the others then ventured further into the valley. "By the time we reached the top of one peak, the water looked blue, with some crystals on top," said Pérez.

Pérez and his friends were astonished to see the unicorn herd. These creatures could be seen from the air without having to move too much to see them – they were so close they could touch their horns.

While examining these bizarre creatures the scientists discovered that the creatures also spoke some fairly regular English. Pérez stated, "We can see, for example, that they have a common 'language,' something like a dialect or dialectic."

Dr. Pérez believes that the unicorns may have originated in Argentina, where the animals were believed to be descendants of a lost race of people who lived there before the arrival of humans in those parts of South America.

While their origins are still unclear, some believe that perhaps the creatures were created when a human and a unicorn met each other in a time before human civilization. According to Pérez, "In South America, such incidents seem to be quite common."

However, Pérez also pointed out that it is likely that the only way of knowing for sure if unicorns are indeed the descendants of a lost alien race is through DNA. "But they seem to be able to communicate in English quite well, which I believe is a sign of evolution, or at least a change in social organization," said the scientist.

- For the prompt given it could generate coherent paragraphs of text. This was **revolutionary**.
- There were *no significant algorithmic changes except scaling the number of parameters* (size of the model) and the amount of training data.
- Cemented the concept of *multi-task learning*.

  →A Large Language Model can easily generalize to unseen dataset from another domain, like legal, or biology, with a *minimal training* (fine-tuning).

GPT-3.5; CHATGPT

complete this fictitious story: In a shocking finding, scientist discovered a herd of unicorns living in a remote, previously unexplored valley, in the Andes Mountains. Even more surprising to the researchers was the fact that the unicorns spoke perfect English.

The news of the discovery of the unicorn herd spread like wildfire, and the scientific community was abuzz with excitement. A team of researchers was quickly assembled to investigate this remarkable phenomenon further. The team set up a research station at the entrance of the valley and began their study of the unicorns.

As the researchers spent more time with the unicorns, they discovered that the creatures were not only capable of speaking perfect English but also had a sophisticated society with a complex system of governance and hierarchy. They observed that the unicorns had a communal way of life, and each member of the herd had a specific role to play in maintaining the group's overall well-being.

The researchers were amazed at how much they could learn from the unicorns, and they soon realized that this discovery could revolutionize our understanding of not only mythical creatures but also the natural world in general. However, they were also aware that they needed to be careful to protect the unicorns and their habitat from outside interference, so they kept their discovery under wraps for the time being.

Over the years, the researchers continued to study the unicorns and their society, and they made many groundbreaking discoveries. They eventually published their findings in a scientific journal, and the world was stunned by the revelation that unicorns, previously thought to be nothing more than a legend, had been living in the Andes Mountains all along.

Regenerate response

Send a message.

Free Research Preview. ChatGPT may produce inaccurate information about people, places, or facts. ChatGPT Mar 23 Version

Safety:

Add "Complete this *fictitious* story"

GPT-4

Sparks of AGI: early experiments with GPT-4

By Sébastien Bubeck

Also…

Intelligence is a very general mental capability that, among other things, involves the ability to:

- Reason ✓
- Plan ✗
- Solve problems ✓
- Think abstractly ✓
- Comprehend complex ideas ✓
- Learning quickly and learn from experience ✓

https://www.youtube.com/watch?v=qbIk7-JPB2c

CONTENT

Origin

https://colah.github.io/posts/2015-08-Understanding-LSTMs/

Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation

arXiv:1406.1078v3

RNN

...2012-2014

Attention Is All You Need

arXiv:1706.03762v5

2017

GPT-2; GPT-3; ...GPT-4

- DeepSpeed
- ZeRO - Zero Redundancy Optimizer
- Megatron-DeepSpeed

ZeRO: Memory Optimizations Toward Training Trillion Parameter Models

arXiv:1910.02054v3

2020

15-20 min.

- **Experimental Results**
- **Our Findings**
- **Conclusion**

2023

DeepSpeed

- Framework from Microsoft https://www.microsoft.com/en-us/research/blog/**deepspeed-extreme-scale-model-training-for-everyone/**
- OpenSource

  https://github.com/microsoft/DeepSpeed
- DeepSpeed is an easy-to-use **deep learning optimization software suite** that enables unprecedented scale and speed for Deep Learning Training and Inference.
- It is a Lightweight wrapper around PyTorch.
- **Supports :**
  - Distributed training
  - Gradient Accumulation… reference…
  - Mixed Precision
  - Checkpointing distributed models
  - 1-bit and 0/1 Adaptive Moment Estimation (Adam) optimizer

DEEPSPEED, ZERO AND ZERO-OFFLOAD

- DeepSpeed supports the Zero Redundancy Optimizer (**ZeRO),** a staged optimizer, where optimizations in earlier stages are available in the later stages.
  - **Stage 1:** The optimizer states are partitioned across the processes, so that each process updates only its partition.
  - **Stage 2:** The reduced 32-bit gradients are partitioned, so that each process retains only the gradients corresponding to its portion of the optimizer states.
  - **Stage 3:** The mixed-precision 16-bit model parameters are also partitioned, across both the forward and backward passes of the processes.
- **ZeRO** reduces the memory consumption of each GPU by partitioning the various model training states (weights, gradients, and optimizer states) across the available devices (GPUs and CPUs) in the distributed training hardware.
- **ZeRO-Offload** is a system for offloading optimizer and gradient states to CPU memory within ZeRO-Stage1 and ZeRO-Stage2.

ZeRO: Memory Optimizations Toward Training Trillion Parameter Models

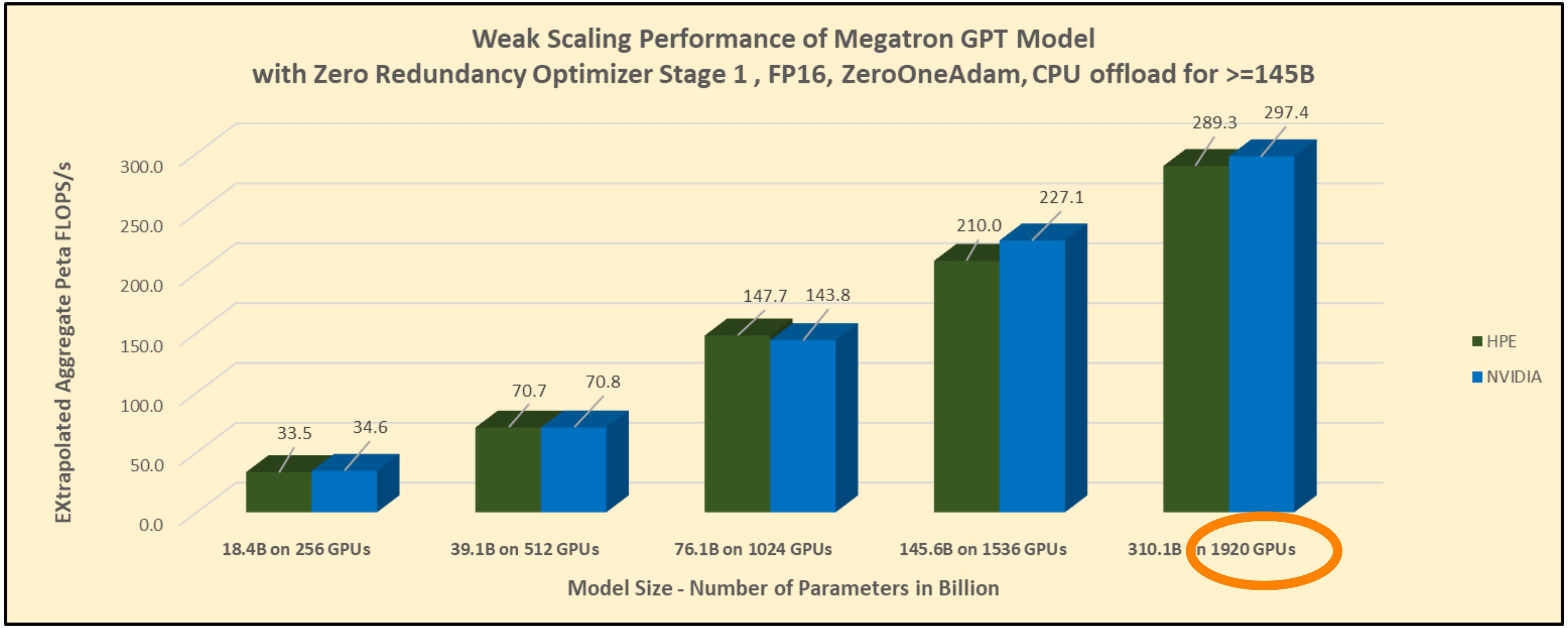
arXiv:1910.02054v3

MEGATRON-DEEPSPEED

- Megatron is a large, powerful transformer developed by the Applied Deep Learning Research team at NVIDIA.
- Megatron-DeepSpeed is the DeepSpeed version of NVIDIA's Megatron-LM.
- **DataParallel (DP)** - the same setup is replicated multiple times, and each being fed a slice of the data. The processing is done in parallel and all setups are synchronized at the end of each training step.
- **TensorParallel (TP)** – a tensor is split into shards (parameters+activations) and a single shard of the tensor resides on its designated GPU. Each shard gets processed separately and in parallel and the results are synced at the end of the step.
- **PipelineParallel (PP)** - the model layers are split up across multiple GPUs, so that one or more layers are placed on each GPU.
- **Zero Redundancy Optimizer (ZeRO)** - sharding is similar to TensorParallel, except the whole tensor gets reconstructed in time for a forward or backward computation. It supports CPU and NVMe offloading techniques to compensate for limited GPU memory.

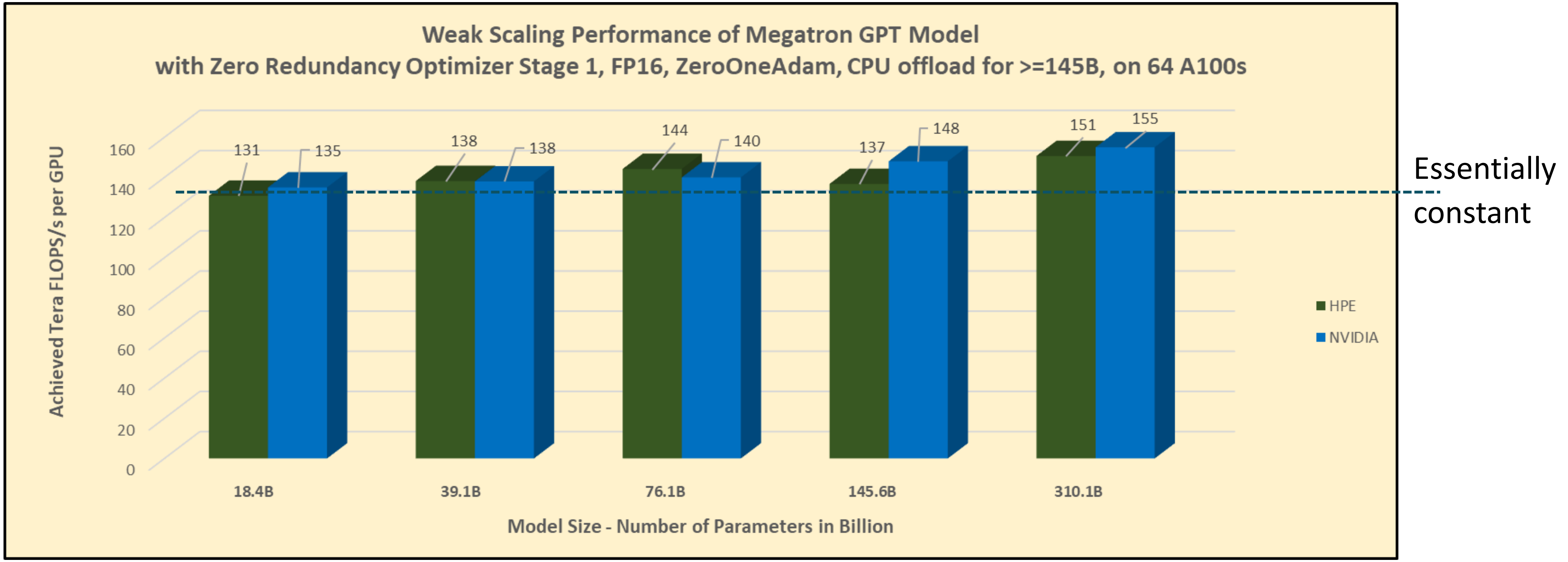
EXTRAPOLATED AGGREGATE PETA FLOPS OF GPT-3 MODELS

- Scaling results with Megatron-DeepSpeed GPT models run on 64 A100 GPUs.
- **HPE Extrapolated Aggregate Peta FLOPs** to larger GPU counts and comparison with NVIDIA's Megatron-LM results.
- We couldn't run the 500 B and 1 Trillion models due to resource constraint.



ACHIEVED TFLOPS OF GPT-3 MODELS

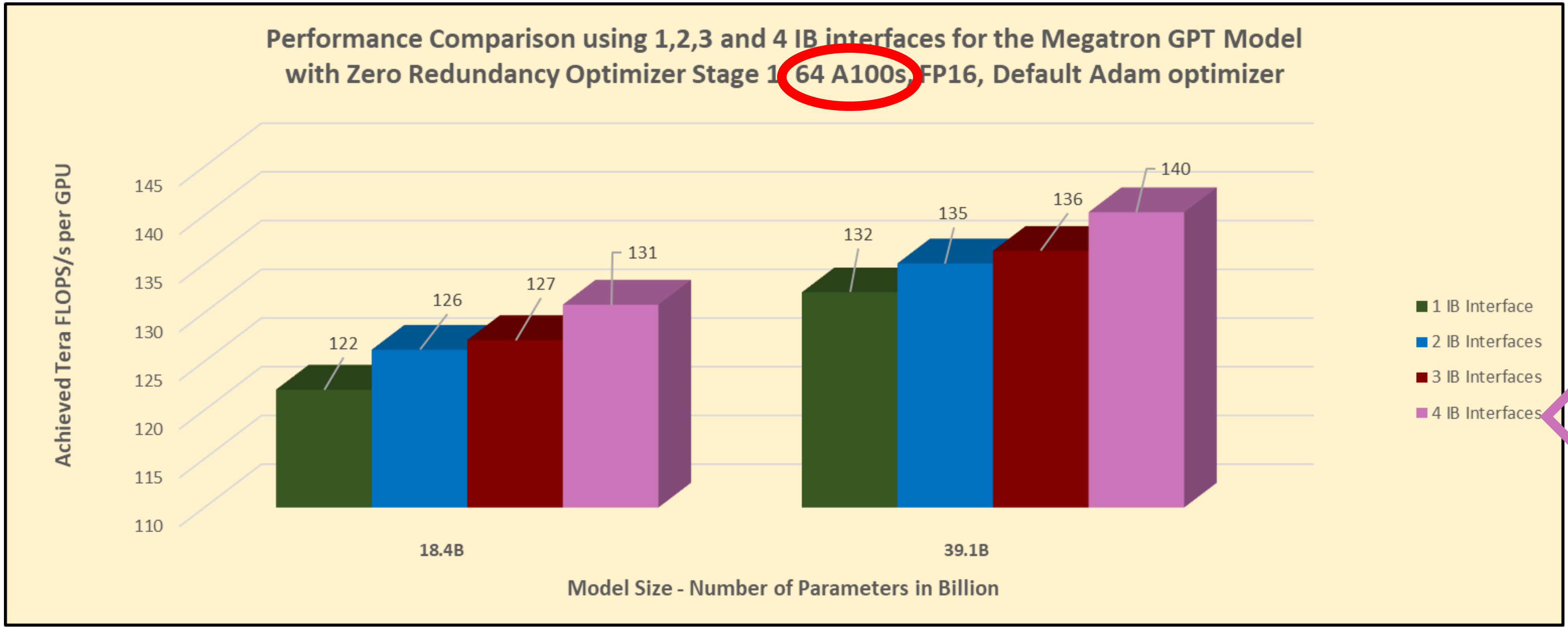
- **Achieved** TFLOPs with Megatron-DeepSpeed model compared to NVIDIA's Megatron-LM numbers.



For fine tuning: only 16 nodes (4GPU/node) are needed

EFFECT OF IB BANDWIDTH ON PERFORMANCE OF GPT-3 MODELS

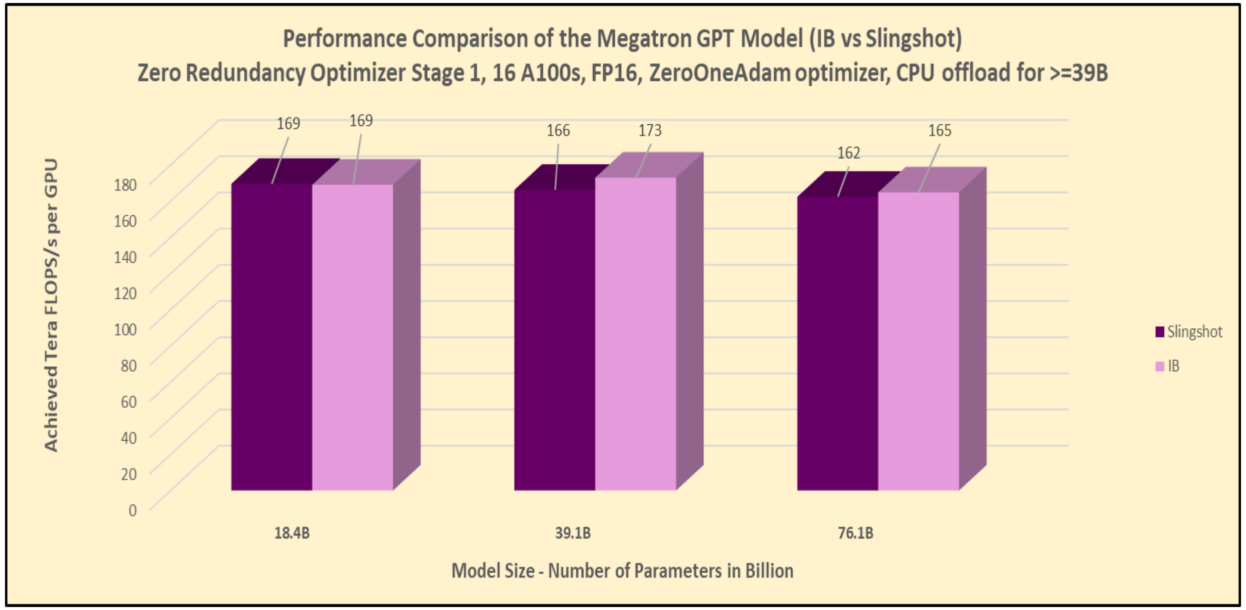
- Performance of 18.4B and 39.1B parameter model with 1,2,3 and 4 IB interfaces.

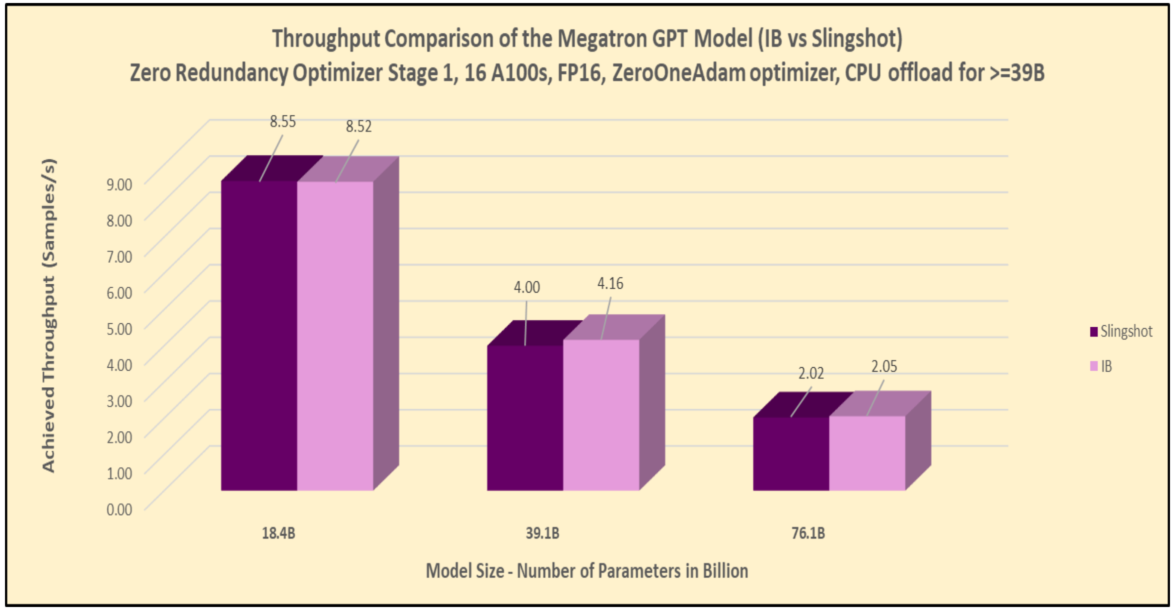


Gaining few FLOPS/s matters, when training can take months

PERFORMANCE COMPARISON OF SLINGSHOT VS INFINIBAND

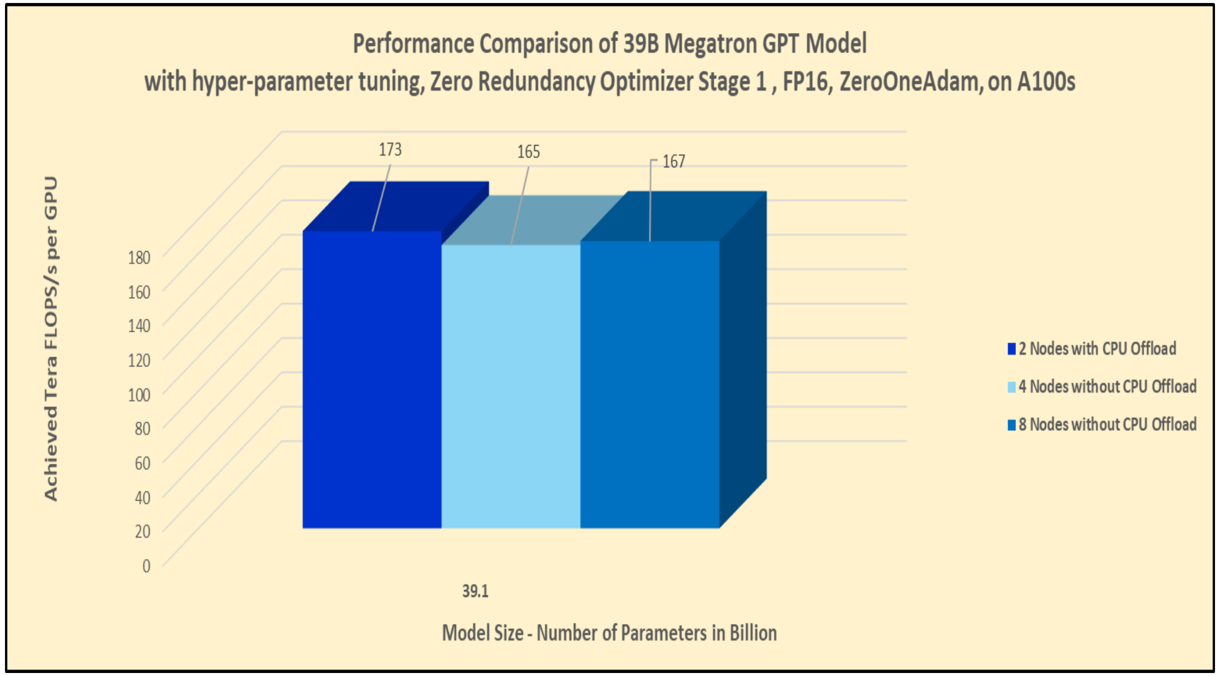
- Performance comparison of IB vs SlingShot on NVIDIA A100 GPUs.
- The obtained performance could be further improved by **fine tuning** the hyper-parameters for Slingshot.

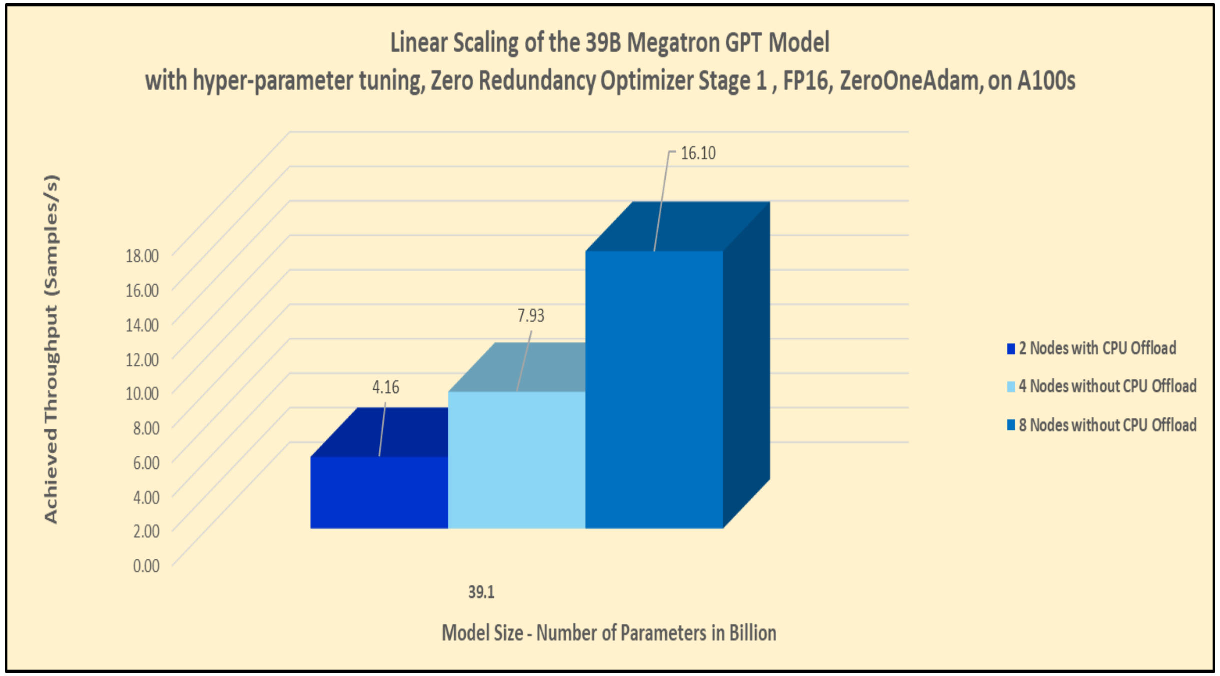




EXPERIMENTAL RESULTS OF MEGATRON-DEEPSPEED GPT MODELS

- The below graphs depict the nearly linear scaling performance of the 39B GPT model.
- With right hyper-parameter tuning, we were able to achieve better throughput as seen below.
- Our experiments confirm that CPU offload not only enables us to finetune massive models on nearly **half** the number of GPUs, but it also helps in achieving the same or slightly better training throughput / TFLOPS per GPU as evident from the below numbers.





OUR FINDINGS

- Leveraging the DeepSpeed's **CPU offloading** technique enables us to finetune and train massive GPT models with limited GPU resources, without having much of an impact on the model's throughput.
- From our experiments, it is evident that all 4 IB interfaces are required to achieve highest training throughput.
- With our experiments, we have been able to reproduce the weak scaling performance using the ZeRO-offload method.
- Hyper-parameter tuning of micro batch, data parallel, tensor parallel and pipeline parallel sizes is very important to maximize the throughput.
- With the obtained results, it is seen that **SlingShot** interconnect with NVIDIA's A100s could be proposed as an effective alternate to the default Mellanox InfiniBand with NVIDIA's A100s.

THANK YOU!

manjunath.sripadarao@hpe.com

pierre.carrier@hpe.com

CONFIDENTIAL DISCLOSURE AGREEMENT

- The information contained in this presentation is proprietary to Hewlett Packard enterprise and is offered in confidence, subject to the terms and conditions of a binding Confidential Disclosure Agreement (CDA)
- HPE requires customers and partners to have signed a CDA in order to view this training
- The information contained in this training is HPE confidential
- This presentation is NOT to be used as a ‘leave behind’ for customers and information may only be shared verbally with HPE external customers under NDA
- This presentation may be shared with Partners under NDA in hard-copy or electronic format for internal training purposes only
- Do not remove any classification labels, warnings or disclaimers on any slide or modify this presentation to change the classification level
- Do not remove this slide from the presentation
- HPE does not warrant or represent that it will introduce any product to which the information relates
- The information contained herein is subject to change without notice
- HPE makes no warranties regarding the accuracy of this information
- The only warranties for HPE products and services are set forth in the express warranty statements accompanying such products and services
- Nothing herein should be construed as constituting an additional warranty
- HPE shall not be liable for technical or editorial errors or omissions contained herein
- Strict adherence to the HPE Standards of Business Conduct regarding this classification level is critical

CALCULATING THE FLOPS/GPU

$a = 96Bslh^2$

$b = s/6h$

$c = v/16lh$

$f_i = a * (1 + b + c)$

$$flops\_per\_gpu \sim \frac{f_i}{10^{12} * num\_gpus * elapsed\_time}$$

- B = Global Batch Size
- s = Sequence Length
- l = Number of Layers
- h = Hidden Size
- v = Vocabulary Size
- f_i = FLOP/iteration