SUMMIT

TORONTO | JUNE 14, 2023

CMP301

Train and deploy Stable Diffusion using AWS Trainium & AWS Inferentia

Vijay Niles
Senior Solutions Architect
Amazon Web Services

Steven Alyekhin
Senior Solutions Architect
Amazon Web Services

Agenda

Introduction to generative AI and stable diffusion

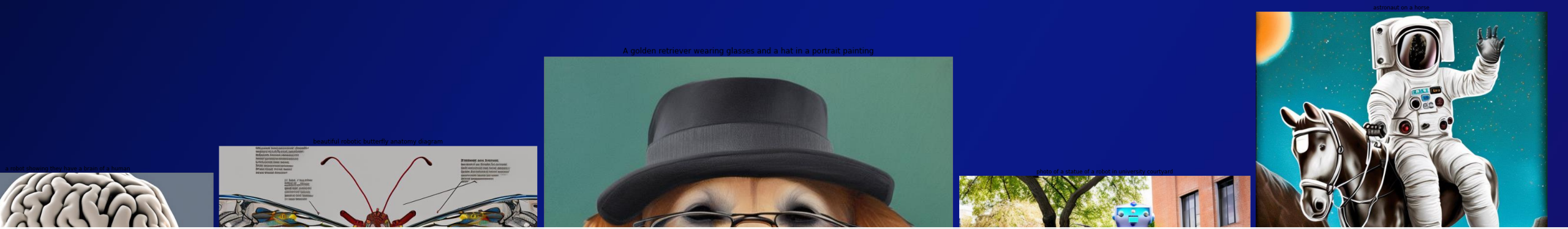
Why build generative AI on AWS?

Demo

AWS AI accelerators: AWS Trainium and AWS Inferentia2

Q&A

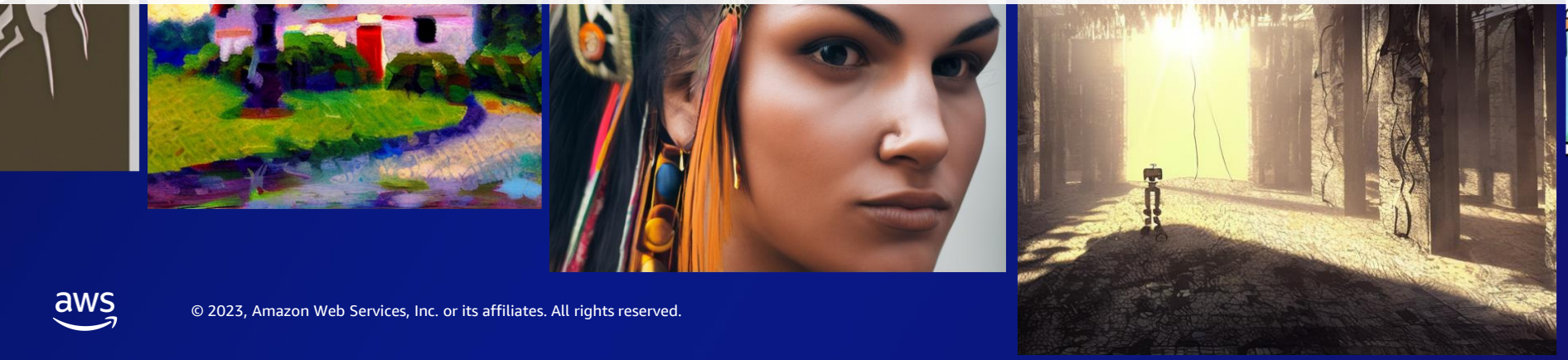
Generative AI is changing the way we work



> Question: What is generative AI?

> Generative AI is a branch of AI that focuses on creating new data. It is a subset of machine learning. The goal of generative AI is to create new data that is similar to the data that was used to train the model.

Generated with BLOOM



Stable Diffusion v2 fp16
on Amazon SageMaker

The road to generative AI

Word embeddings

Language models

Word2vec

Natural language processing (NLP)

Text summarization

Question answering

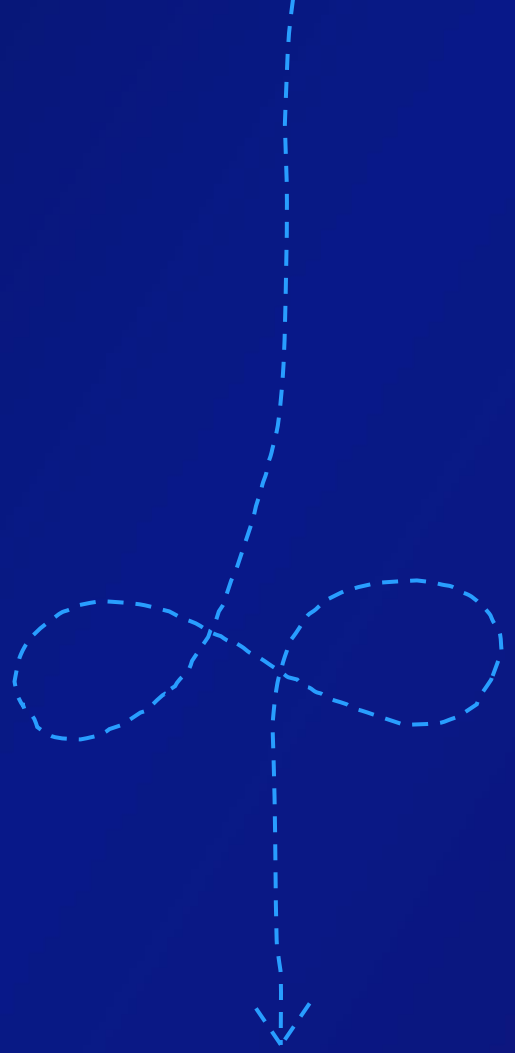
Sentiment analysis

Speech language understanding

Applications for generative AI

- Text generation (many types)
- Game design
- Industrial design
- Drug design research
- Image generation . . .

Text-to-image



Language models
(word embeddings)

Image models
(image embeddings)

Text-to-image models
(e.g., Stable Diffusion)

Stable Diffusion examples

Diffusion models

"Marigold the puppy"

Stable Diffusion training

"a photo of Marigold the puppy as 8bit art"

Why build generative AI on AWS?

The AI/ML flywheel

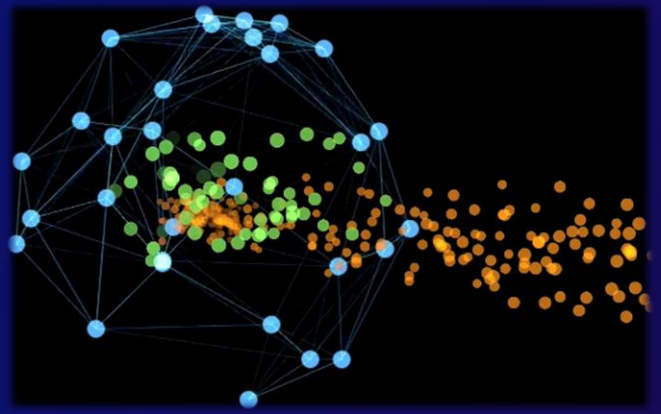
4 Investment
(research, optimization)

3 Demand
(wider adoption)

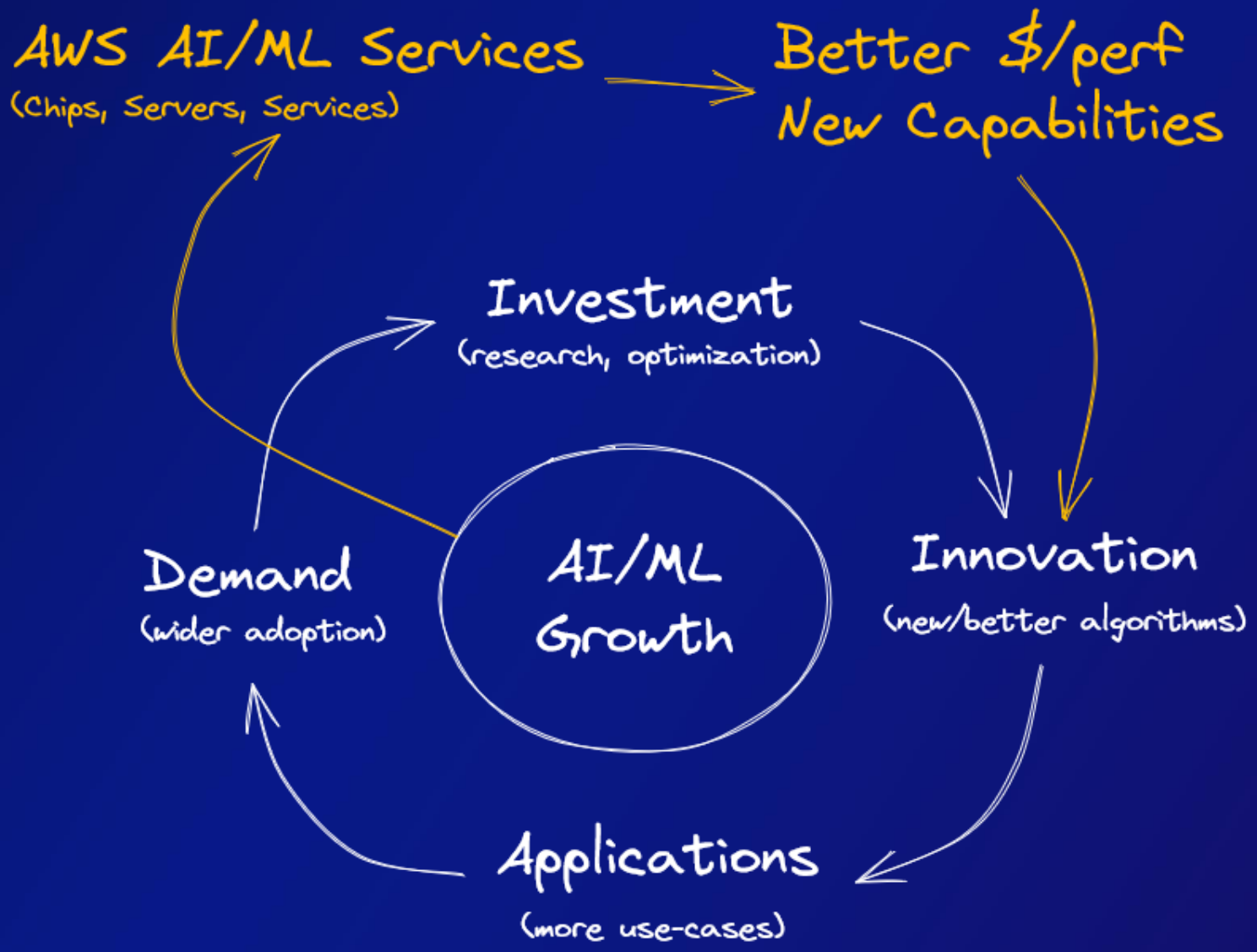
AI @AWS

1 Innovation
(new/better algorithms)

2 Applications
(more use-cases)

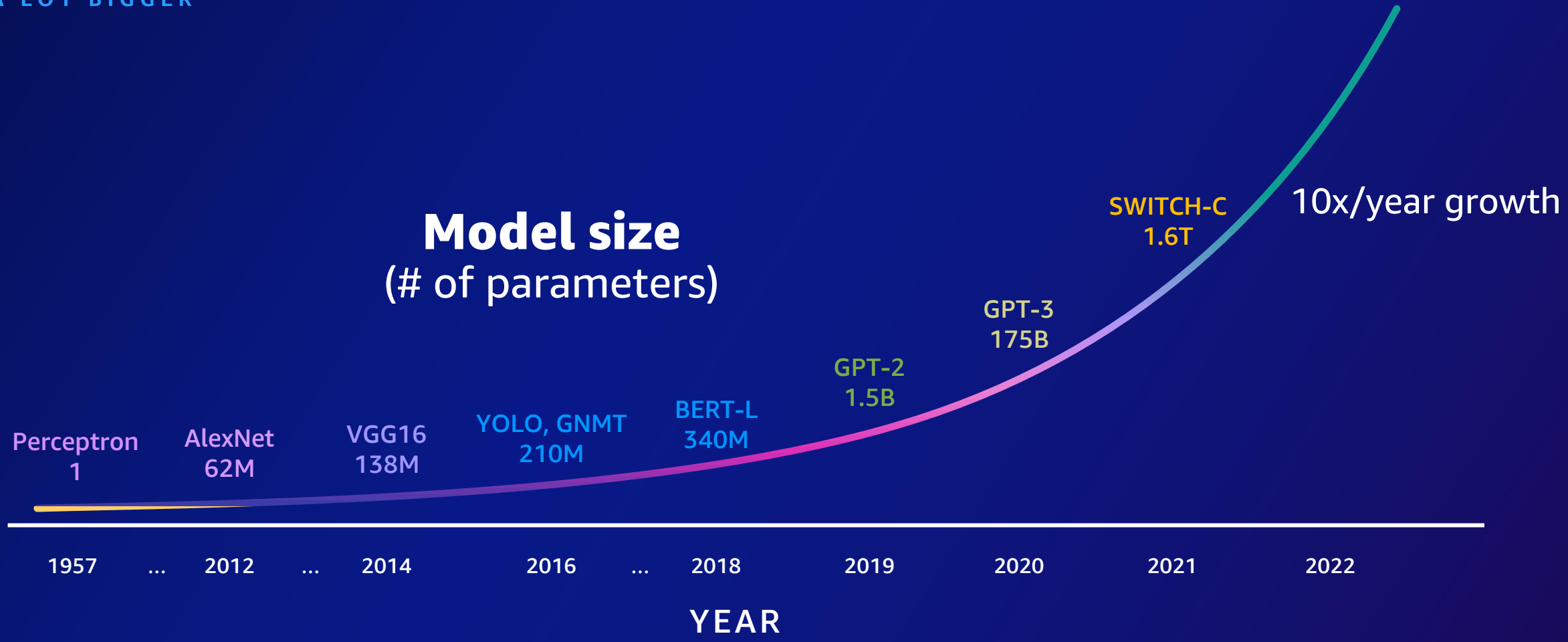


The AI/ML flywheel



AI models are getting bigger

. . . A LOT BIGGER



What's not going to change?

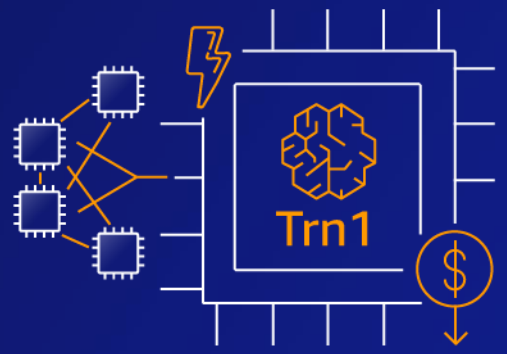
High performance

Cost effectiveness

Ease of use

AWS purpose-built accelerators for generative AI

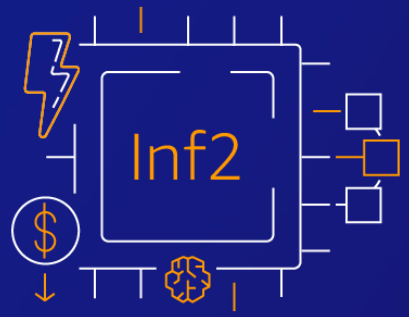
AWS Trainium



Cost-efficient, high-performance training of LLMs and diffusion models

Up to 50% cost-to-train savings

AWS Inferentia2



High performance at the lowest cost per inference for LLMs and diffusion models

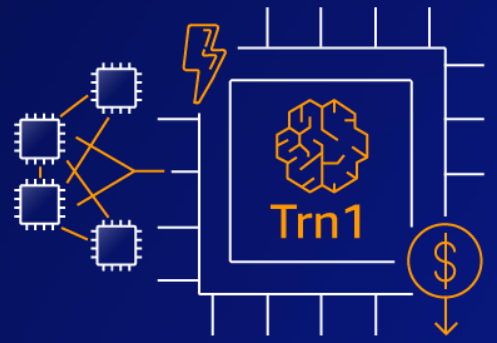
Up to 40% better price performance for generative AI

Demo – Train and deploy Stable Diffusion on Trn1 and Inf2

AWS Trainium

Amazon EC2 Trn1 and Trn1n instances powered by AWS Trainium

COST-EFFICIENT, HIGH-PERFORMANCE DL TRAINING INSTANCES



High performance on training of popular NLP models on AWS

Up to 50% cost-to-train savings

Up to 4x network bandwidth

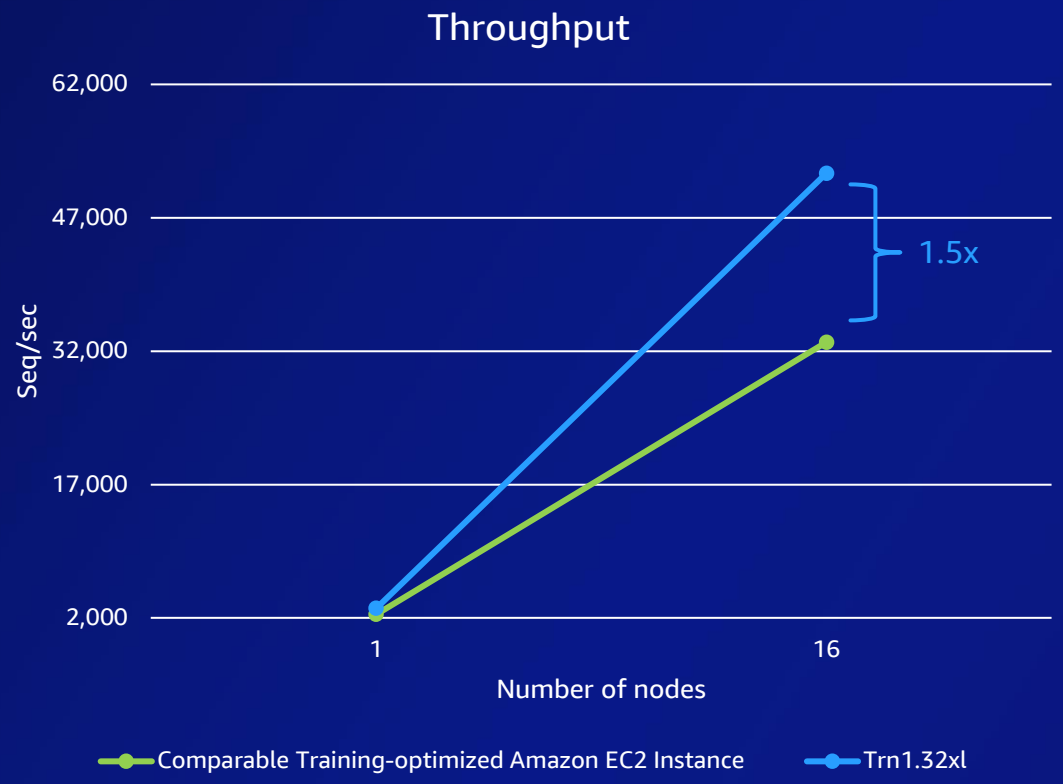
Instance size	vCPUs	Instance memory	Trainium chips	Accelerator memory	NeuronLink	Instance networking	On-demand price
Trn1.2xlarge	8	32 GB	1	32 GB	N/A	Up to 10 Gbps	$1.34/hr
Trn1.32xlarge	128	512 GB	16	512 GB	Yes	800 Gbps	$21.5/hr
Trn1n.32xlarge	128	512 GB	16	512 GB	Yes	1600 Gbps	$24.78/hr

Trn1 available now in US-East-1 (N. Virginia) and US-West-2 (Oregon)

Trn1 delivers high performance

BEST-IN-CLASS THROUGHPUT AND COST

Trn1 single node: 1.2x faster
Trn1 cluster: 1.5x faster



Trn1 single node: 1.8x lower cost
Trn1 cluster: 2.3x lower cost

Hugging Face BERT-Large Ph1 Pretrain; 128 Seqlen
trn1 – BF16(SR), p4d – Mixed Precision (BF16/FP32)

Ease of use

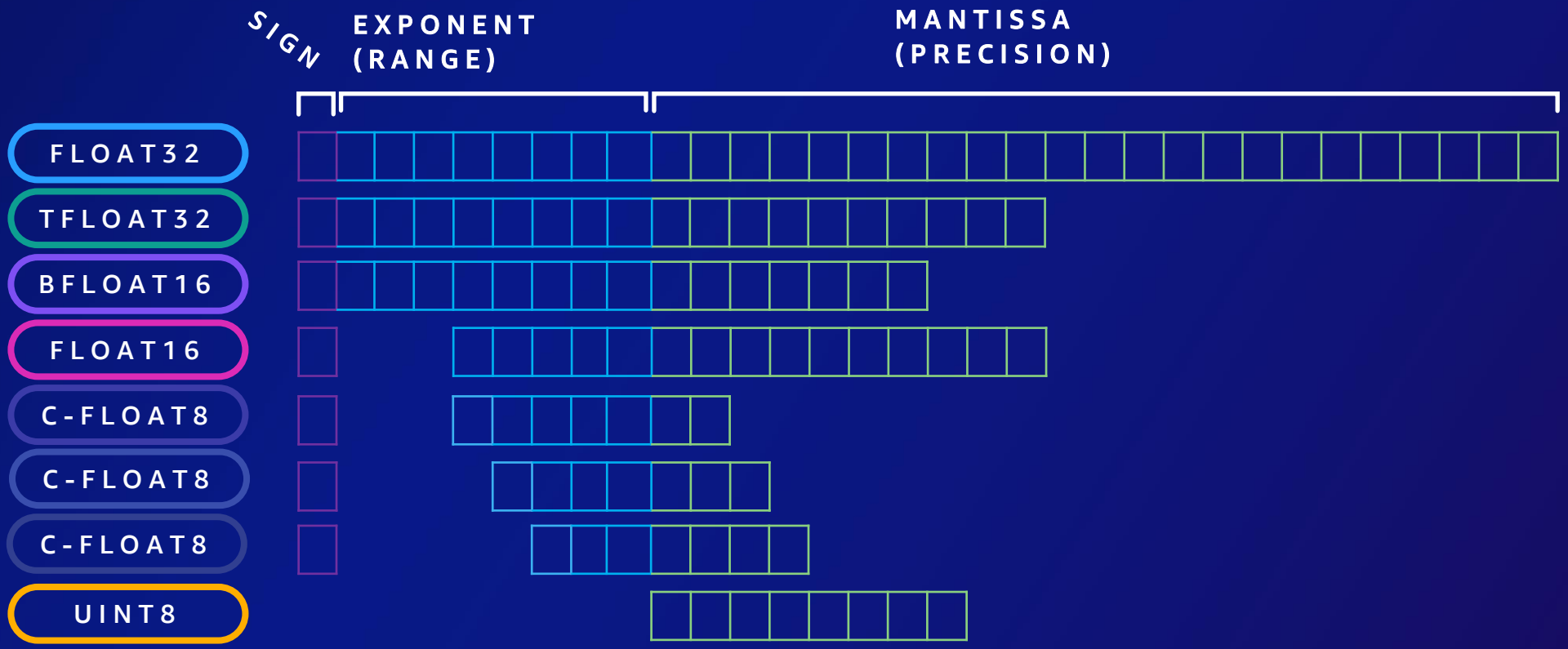
Bring your own model

JIT compile to Trainium

```python
import os
...
import torch
import torch_xla
import torch_xla.core.xla_model as xm
...
from transformers import BertForPreTraining

model = BertForPreTraining.from_pretrained('bert-large-uncased')

def train_loop_fn(model, optimizer, train_loader, device, epoch, global_step, training_ustep, running_loss):
    max_grad_norm = 1.0
    for i, data in enumerate(train_loader):
        training_ustep += 1
        input_ids, segment_ids, input_mask, masked_lm_labels, next_sentence_labels = data
        outputs = model(input_ids=input_ids,
                        attention_mask=input_mask,
                        token_type_ids=segment_ids,
                        labels=masked_lm_labels,
                        next_sentence_label=next_sentence_labels)
        loss = outputs.loss / flags.grad_accum_usteps
        loss.backward()
        running_loss += loss.detach()

        if (training_ustep + 1) % flags.grad_accum_usteps == 0:
            xm.mark_step()
            running_loss_cpu = running_loss.detach().cpu().item()
            running_loss.zero_()
            torch.nn.utils.clip_grad_norm_(model.parameters(), max_grad_norm)
            xm.optimizer_step(optimizer)
            optimizer.zero_grad()
            scheduler.step()
            global_step += 1
            if global_step >= flags.steps_this_run:
                break

    return global_step, training_ustep, running_loss
```

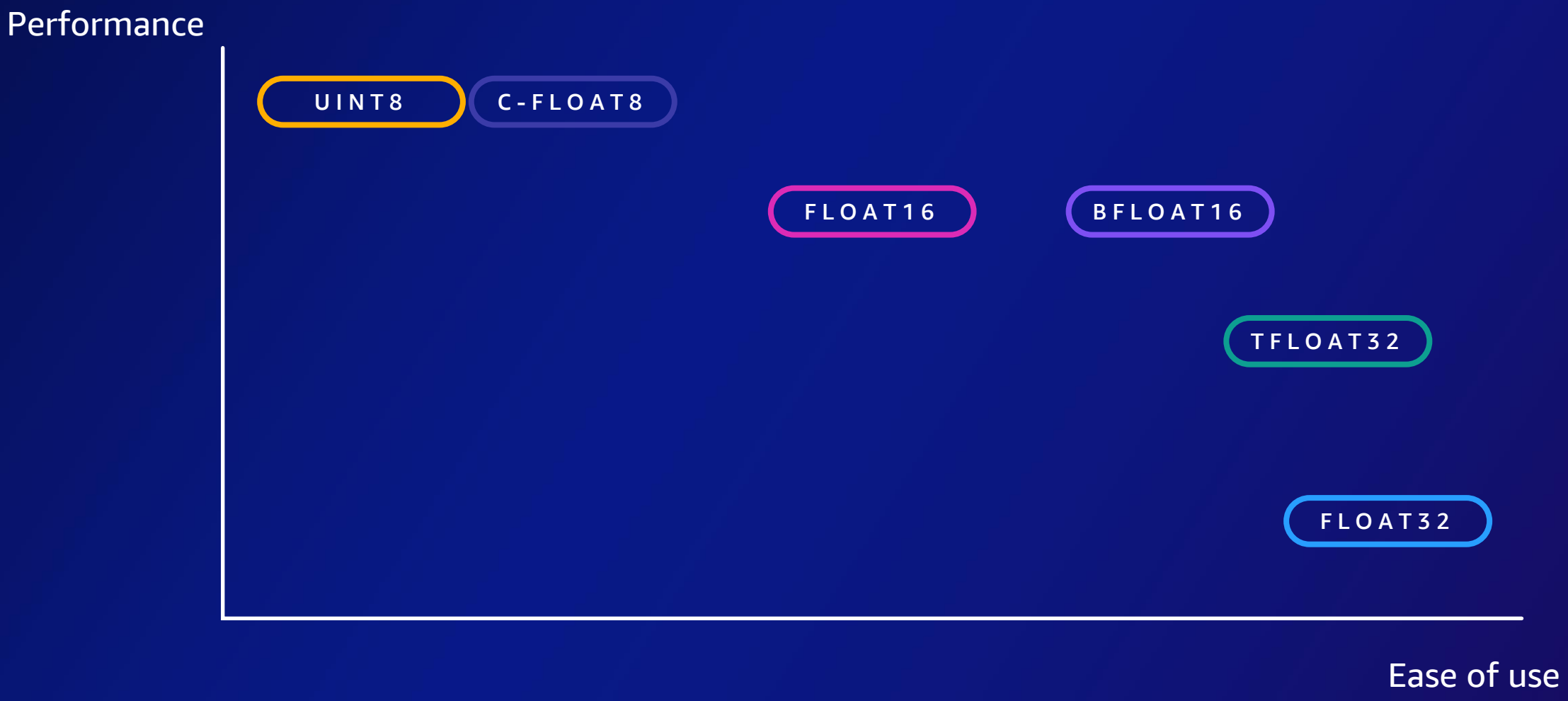
Rich data type selection

CHOOSE THE RIGHT DATA TYPE FOR YOUR WORKLOAD



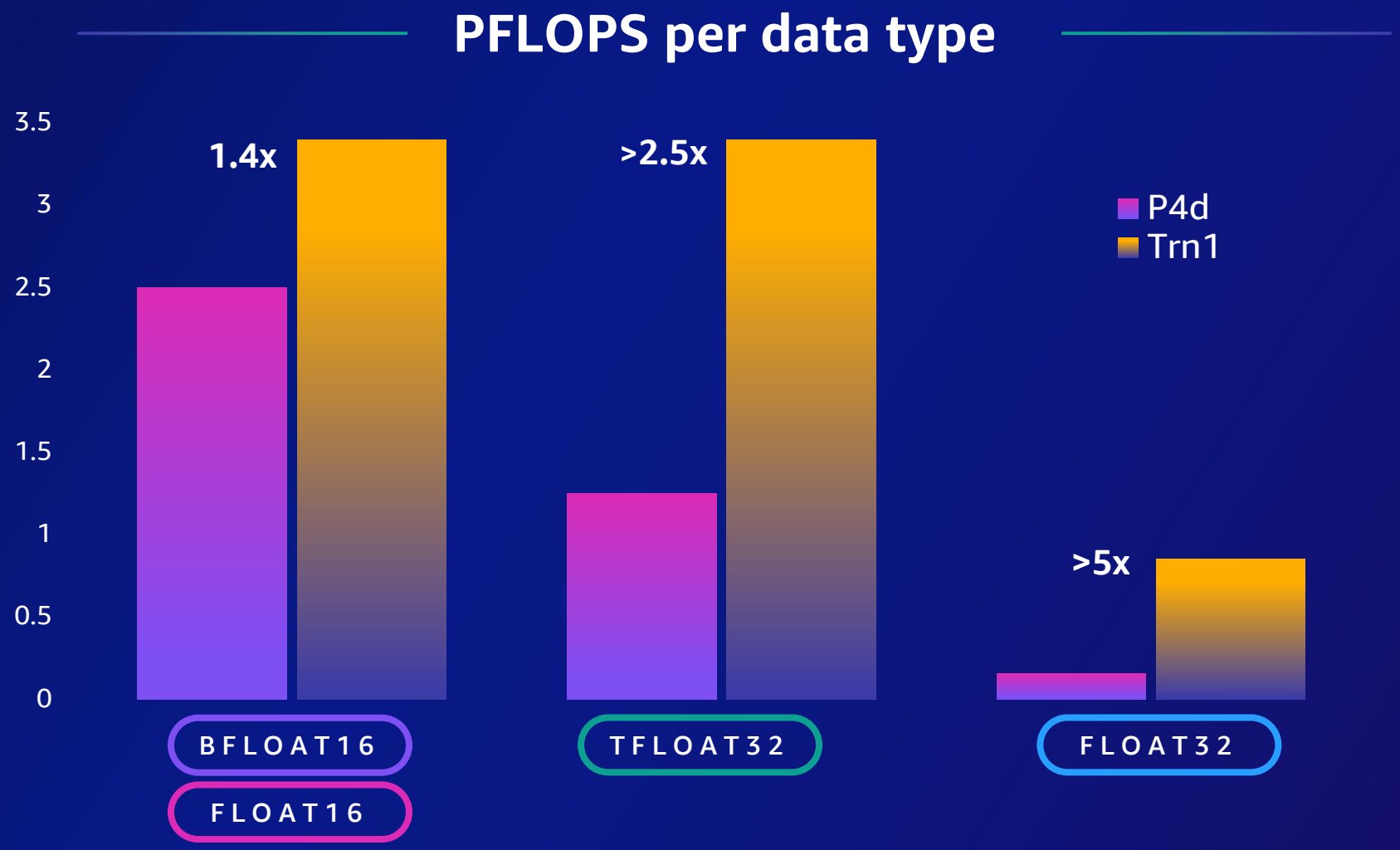
Rich data type selection

CHOOSE THE RIGHT DATA TYPE FOR YOUR WORKLOAD



Rich data type selection

CHOOSE THE RIGHT DATA TYPE FOR YOUR WORKLOAD

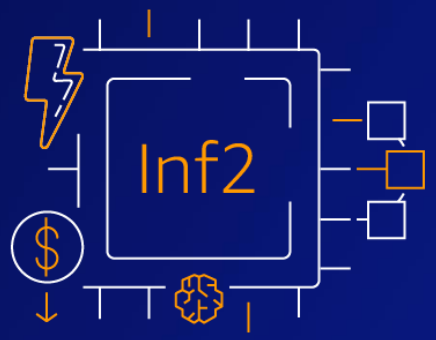


AWS Inferentia2

Amazon EC2 Inf2 instances powered by AWS Inferentia2

NEW

HIGH PERFORMANCE AT THE LOWEST COST FOR GENERATIVE AI MODELS



Up to 4x higher throughput and 10x lower latency

9.8 TB/s aggregated accelerator memory bandwidth

Support for ultra-large generative AI models

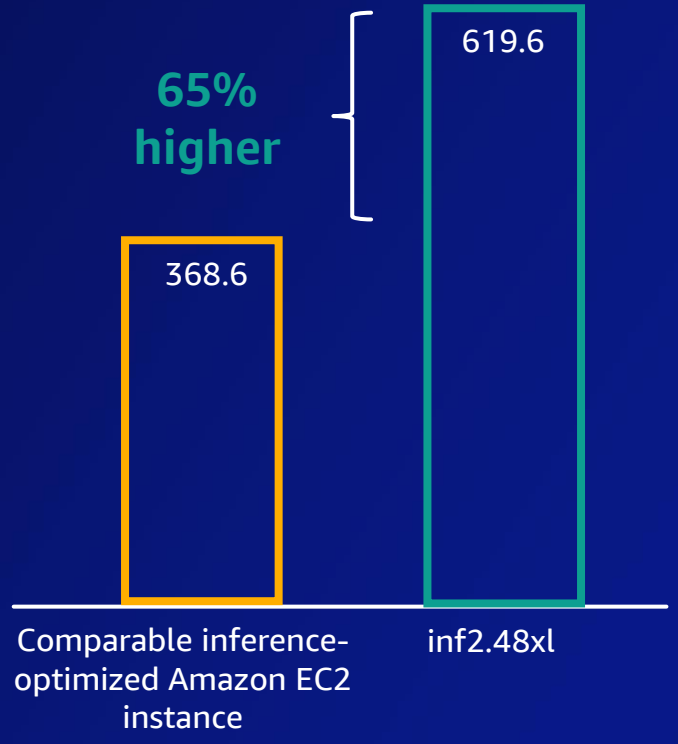
Instance size	vCPUs	Instance memory	Inferentia2 chips	Accelerator memory	NeuronLink	Instance networking	On-demand price
Inf2.xlarge	4	16 GB	1	32 GB	N/A	Up to 15 Gbps	$0.76/hr
Inf2.8xlarge	32	128 GB	1	32 GB	N/A	Up to 25 Gbps	$1.97/hr
Inf2.24xlarge	96	384 GB	6	192 GB	Yes	50 Gbps	$6.49/hr
Inf2.48xlarge	192	768 GB	12	384 GB	Yes	100 Gbps	$12.98/hr

Inf2 available now in US-East-1 (N. Virginia) and US-East-2 (Ohio)

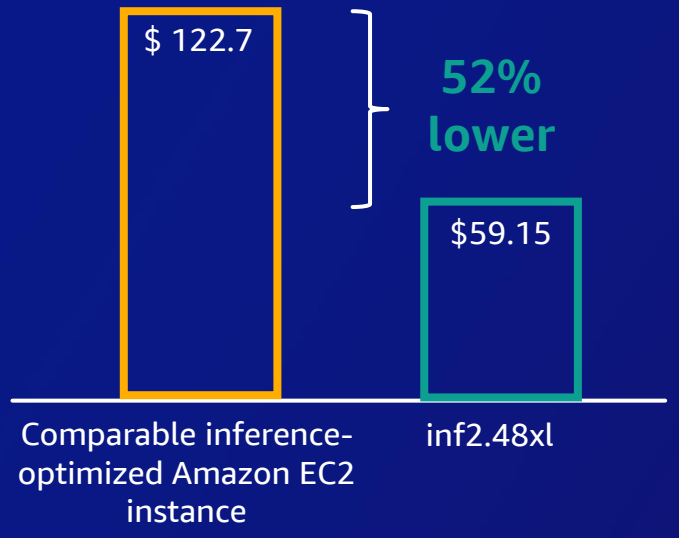
aws

AWS Inferentia2 LLM performance

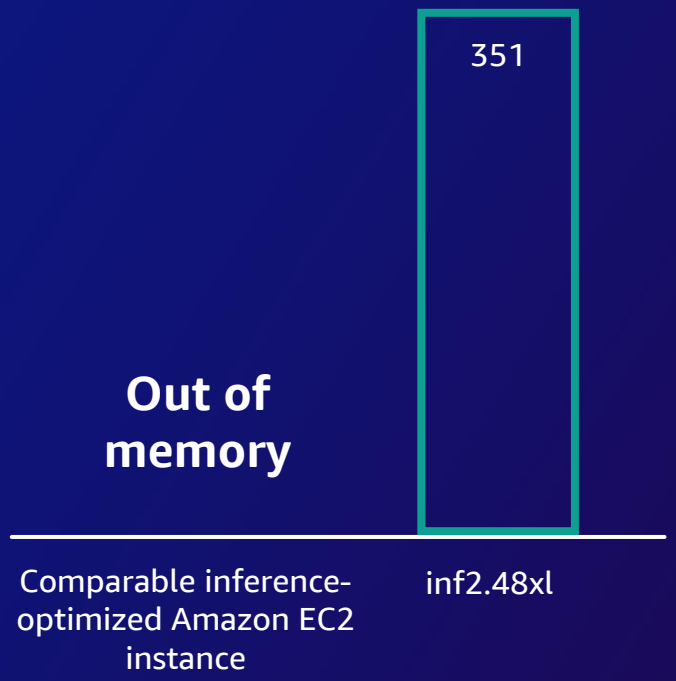
OPT-30B throughput
(tokens/sec)
FP16, Seqlen 2048, B16



OPT-30B cost per million
(USD)
FP16, Seqlen 2048, B16

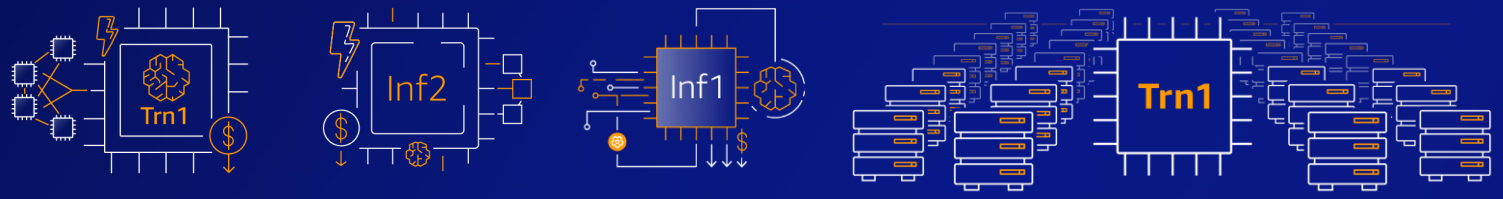


OPT-66B throughput
(tokens/sec)
FP16, Seqlen 2048,



Full stack AI/ML integration

Accelerated compute



Amazon EC2 Trn1/Trn1n, Inf2, Inf1

Amazon EC2 Trn1 UltraClusters

Storage & networking

- Amazon Simple Storage Service (Amazon S3)
- Amazon Elastic Block Store (Amazon EBS)
- Amazon FSx for Lustre
- Amazon Elastic File System (Amazon EFS)
- Elastic Fabric Adapter

Frameworks & services

- ML frameworks
- Amazon SageMaker
- AWS Deep Learning AMIs
- AWS Deep Learning Containers
- Amazon Elastic Kubernetes Service (Amazon EKS)
- Amazon Elastic Container Service (Amazon ECS)

Getting started with AWS Trainium and AWS Inferentia

Launch instances
(Trn1, Inf2, Inf1)

AWS Deep Learning AMIs

AWS Deep Learning Containers

Bring your own model

PyTorch

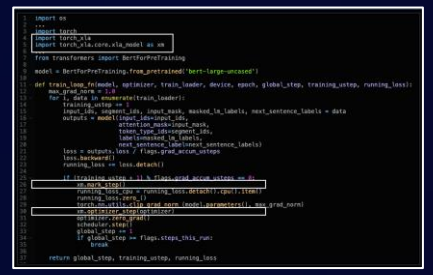
TensorFlow

Hugging Face

OpenXLA

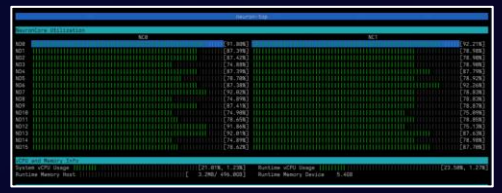
Modify a few lines of code → run

Neuron SDK



Monitor, tune, scale

Neuron SDK



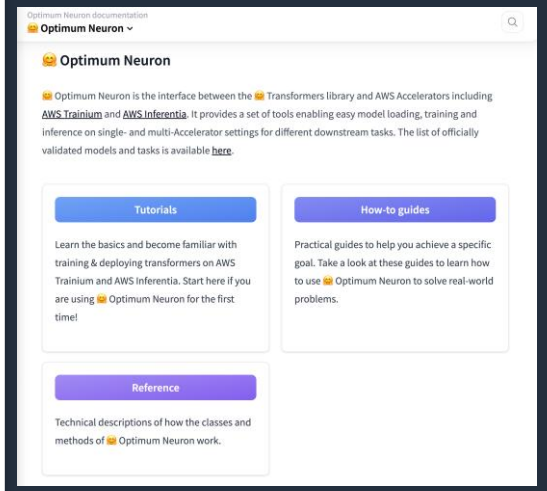
Getting started with Hugging Face Optimum Neuron

Hugging Face

+

Neuron SDK

Documentation & tutorials



Hugging Face documentation

Code examples

Access GitHub

skillbuilder.aws

Your time is now

Build in-demand cloud skills your way

Thank you!

Please complete the session survey in the mobile app

Vijay Niles
vnniles@amazon.com

Steven Alyekhin
salyekh@amazon.com