August 1, 2022

Training Deep Learning Models on Habana Gaudi®

Milind S. Pandit, Senior Solutions Architect, mpandit@habana.ai

https://www.habana.ai

habana®
An Intel Company

A little about Habana

- Founded in 2016 to develop purpose-built AI processors
- Launched inference processor in 2018, training processor in 2019
- Acquired by Intel in late-2019
- Fully leveraging Intel’s scale, resources and infrastructure
- Accessing Intel ecosystem and customer partnerships
- Delivering aggressive roadmap optimized for AI data center performance and efficiency

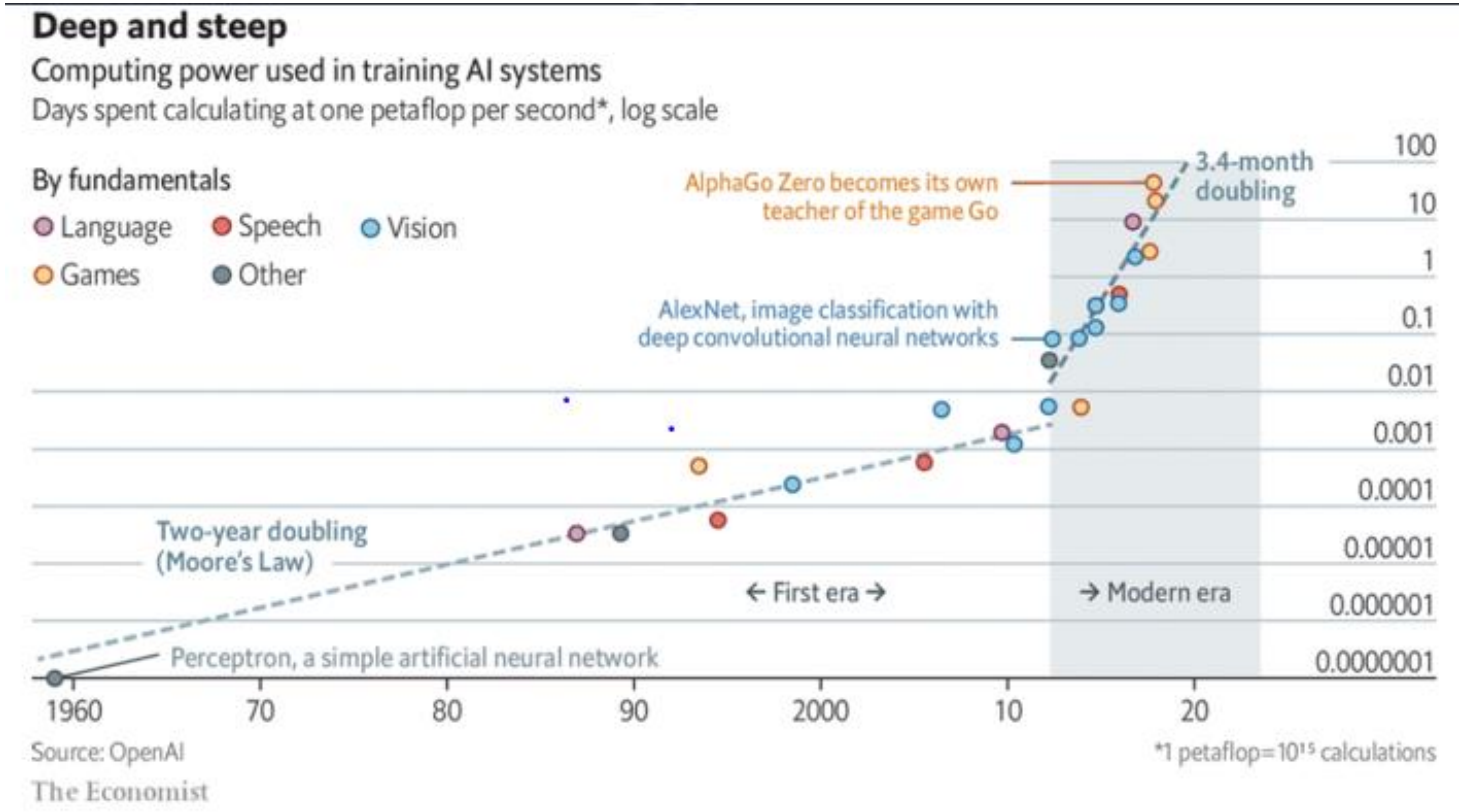
intel

&

habana®

Demand for compute for ML training doubles every 3.4 months

- Increasing Complexity
  - Businesses need higher precision in their model predictions
  - Results in larger and more complex models
  - Requires frequent retraining of models
- Increasing Costs
  - Increasing compute power required for frequent training of larger models drives up cost to train
  - Becomes a barrier for innovation and growth



Deep and steep

Computing power used in training AI systems

Days spent calculating at one petaflop per second*, log scale

Source: OpenAI

*1 petaflop=10^{15} calculations

The Economist

Need for dedicated AI processors to address the compute, memory and communication challenges

Today’s Cost to Train: Biggest Barrier to AI Implementation

“Cost is the most significant challenge to implementing AI/ML solutions.”

56% of AI/ML customers

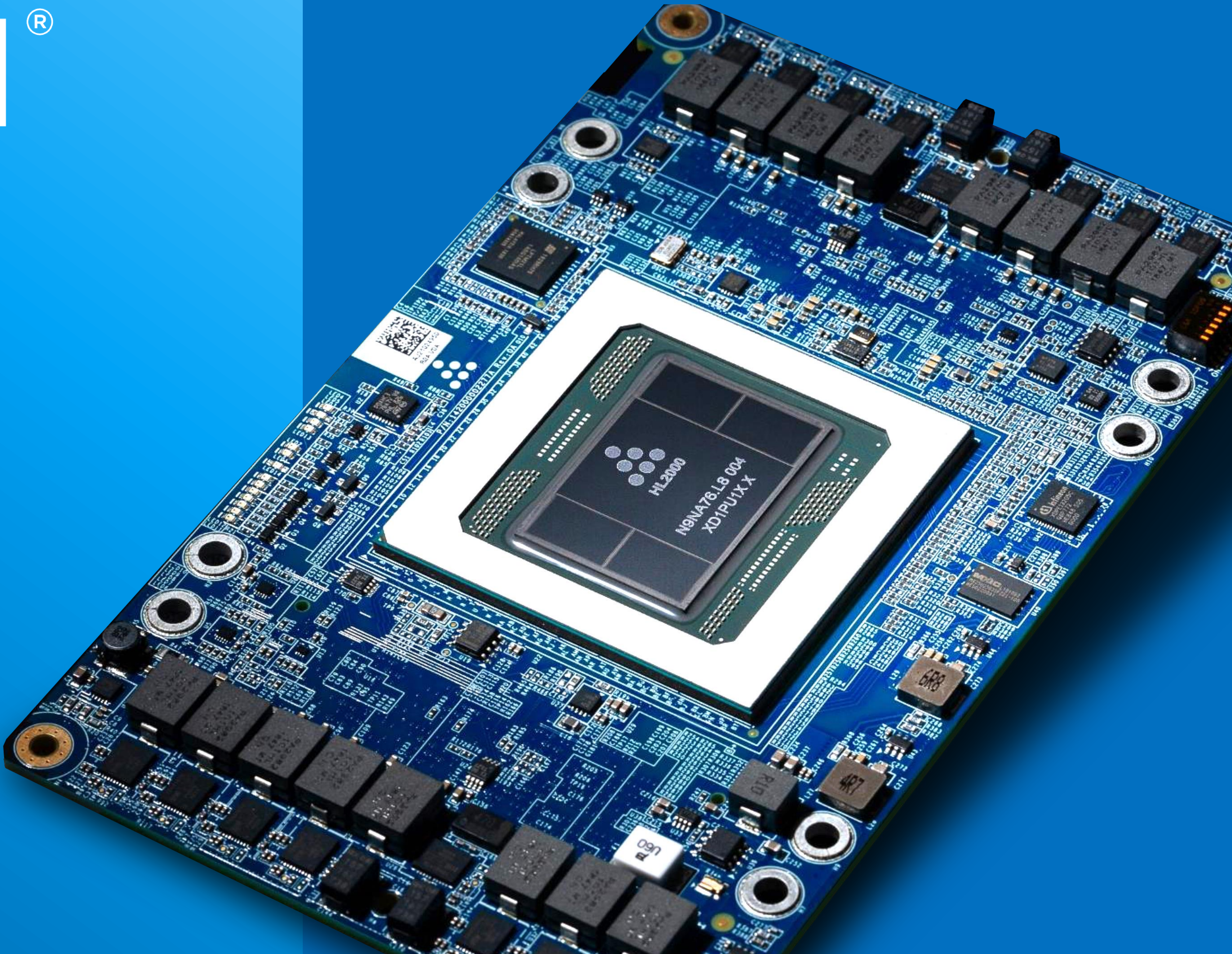
Industry Challenge:

How to give more customers access to more AI?

Source: IDC Semiannual Artificial Intelligence Tracker (2020H1, published Jan 2021)

GAUDI®

Purpose-built for AI training efficiency, usability and scale



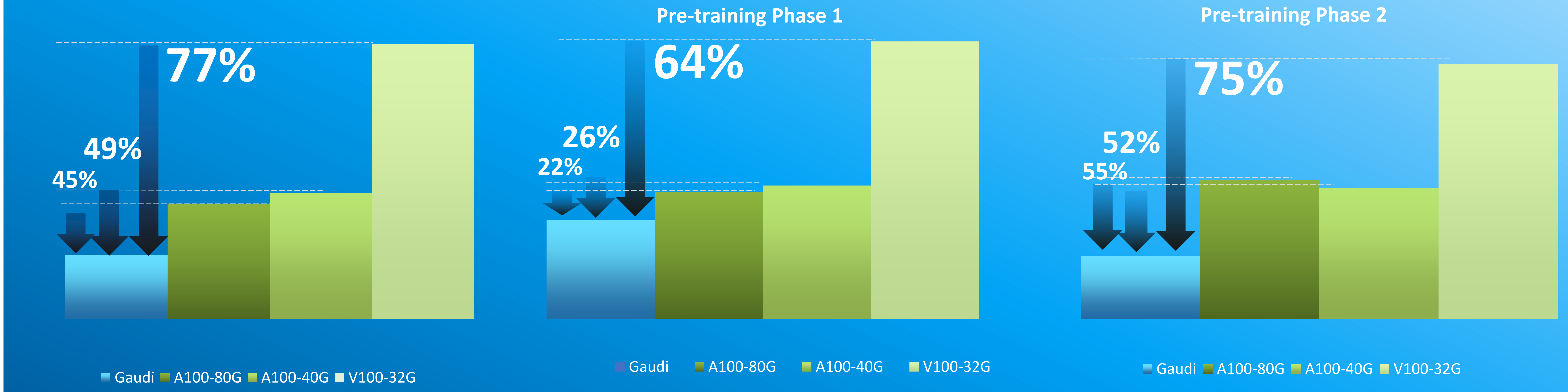
DL1 Model Training Cost Savings

ResNet50 $/image

(lower is better)

BERT-Large $/seq

(lower is better)



Cost savings based on Amazon EC2 On-Demand pricing for P3, P4d and DL1 instances respectively. Performance data was collected and measured using the following resources:

Habana BERT-Large Model: https://github.com/HabanaAI/Model-References/tree/master/TensorFlow/nlp/bert

Habana ResNet50 Model: https://github.com/HabanaAI/Model-References/tree/master/TensorFlow/computer_vision/Resnets/resnet_keras

Habana SynapseAI Container: https://vault.habana.ai/ui/repos/tree/General/gaudi-docker/1.4.0/ubuntu20.04/habanalabs/tensorflow-installer-tf-cpu-2.8.0

Habana Gaudi Performance: https://developer.habana.ai/resources/habana-training-models/

A100 / V100 Performance: https://ngc.nvidia.com/catalog/resources/nvidia:bert_for_tensorflow/performance, https://ngc.nvidia.com/catalog/resources/nvidia:resnet_50_v1_5_for_tensorflow/performance,

Results published for DGX A100-40G and DGX V100-32G. Results May Vary.

The Habana® Gaudi® AI Training Processor

GAUDI®

Designed to optimize AI performance, delivering higher AI efficiency than traditional CPUs and GPUs

Heterogeneous compute architecture enables high-efficiency on large AI workloads

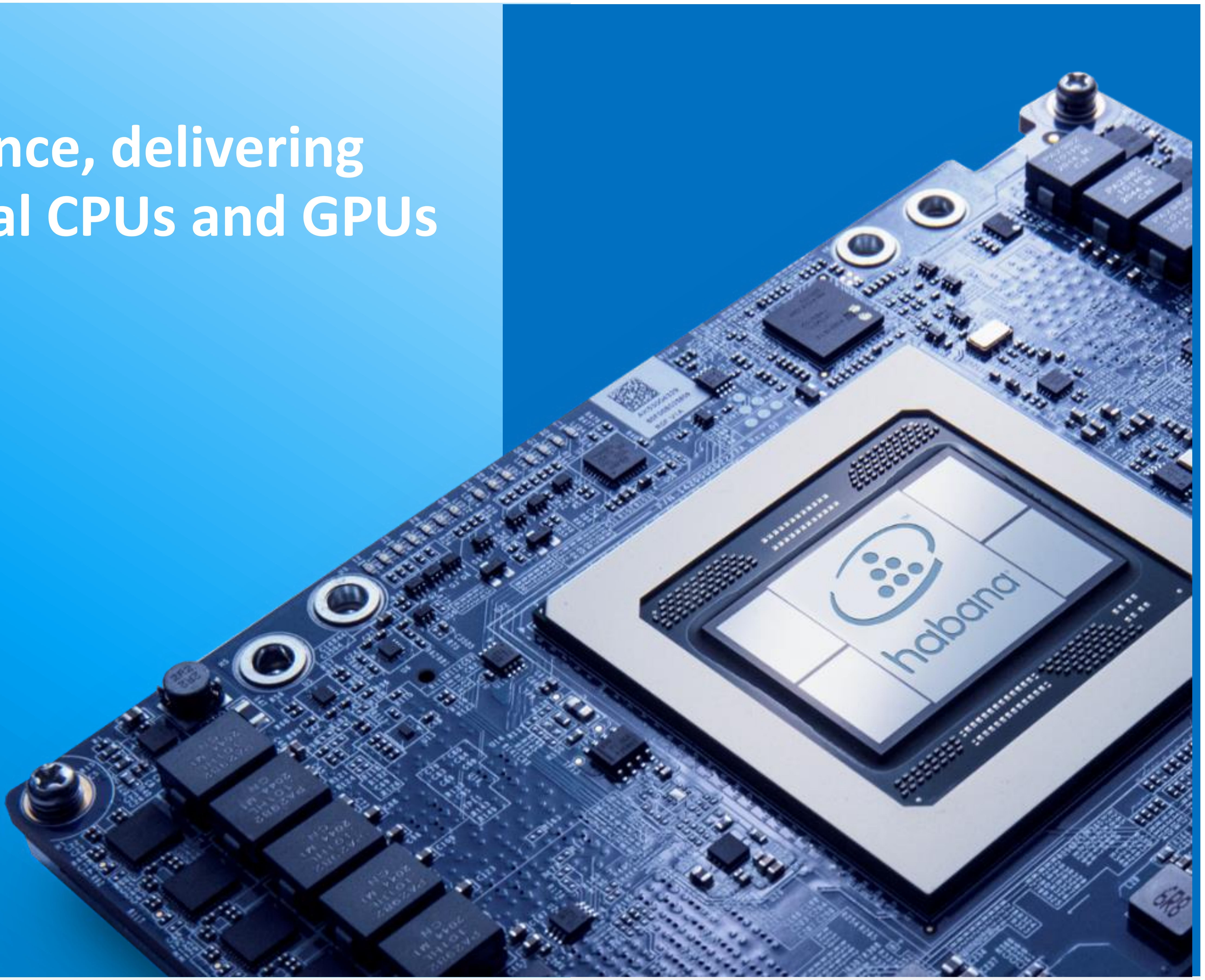
- GEMM engine (MME) excels at matrix multiplication
- While TPC runs non-linear and element wise ops

Software-managed memory architecture

- 32 GB of HBM2 memory

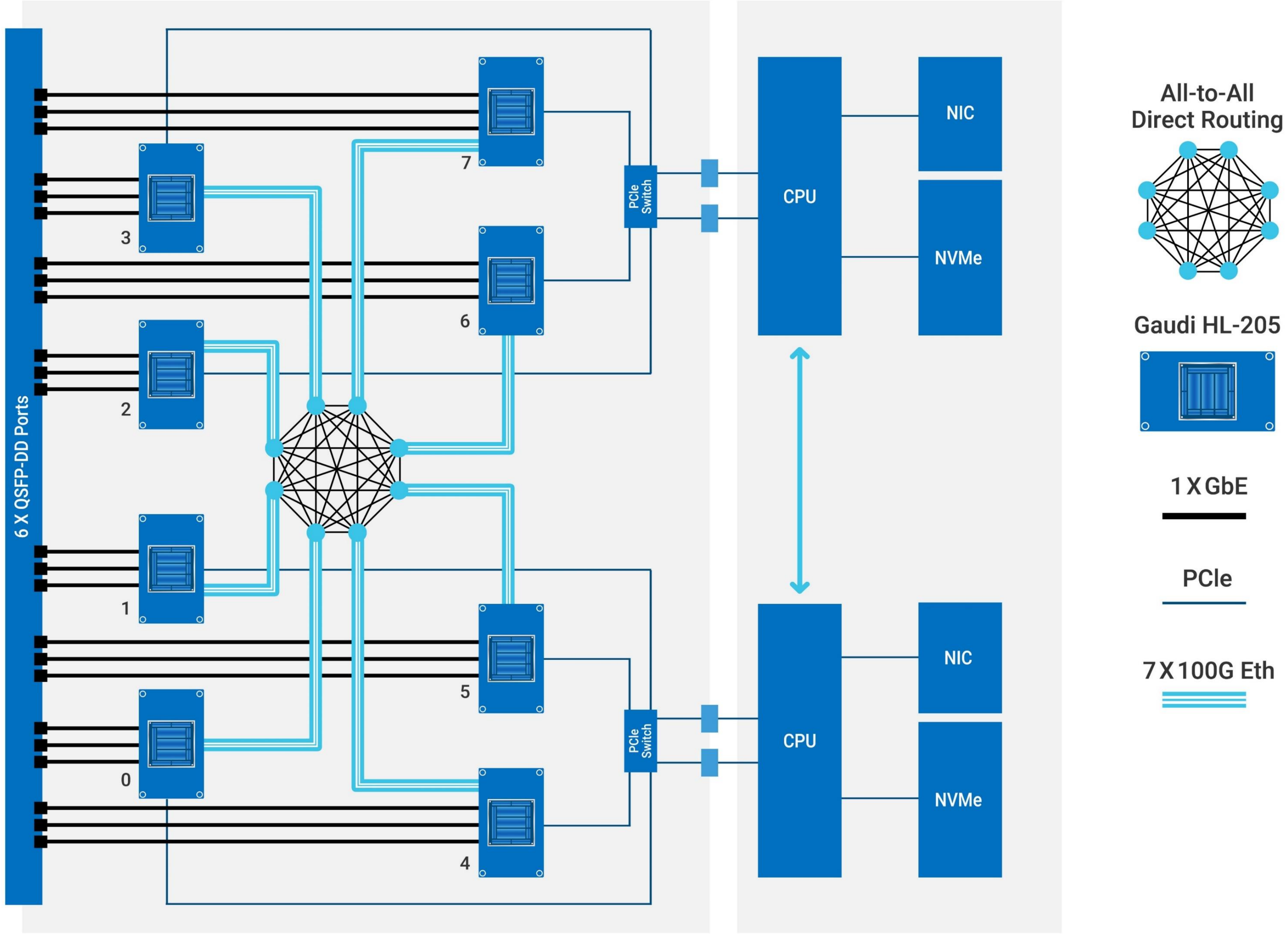
Integrates ten 100Gb Ethernet RoCE ports

- Scaling capacity
- Flexibility based on industry standard
- Cost-efficiency with integrated NIC



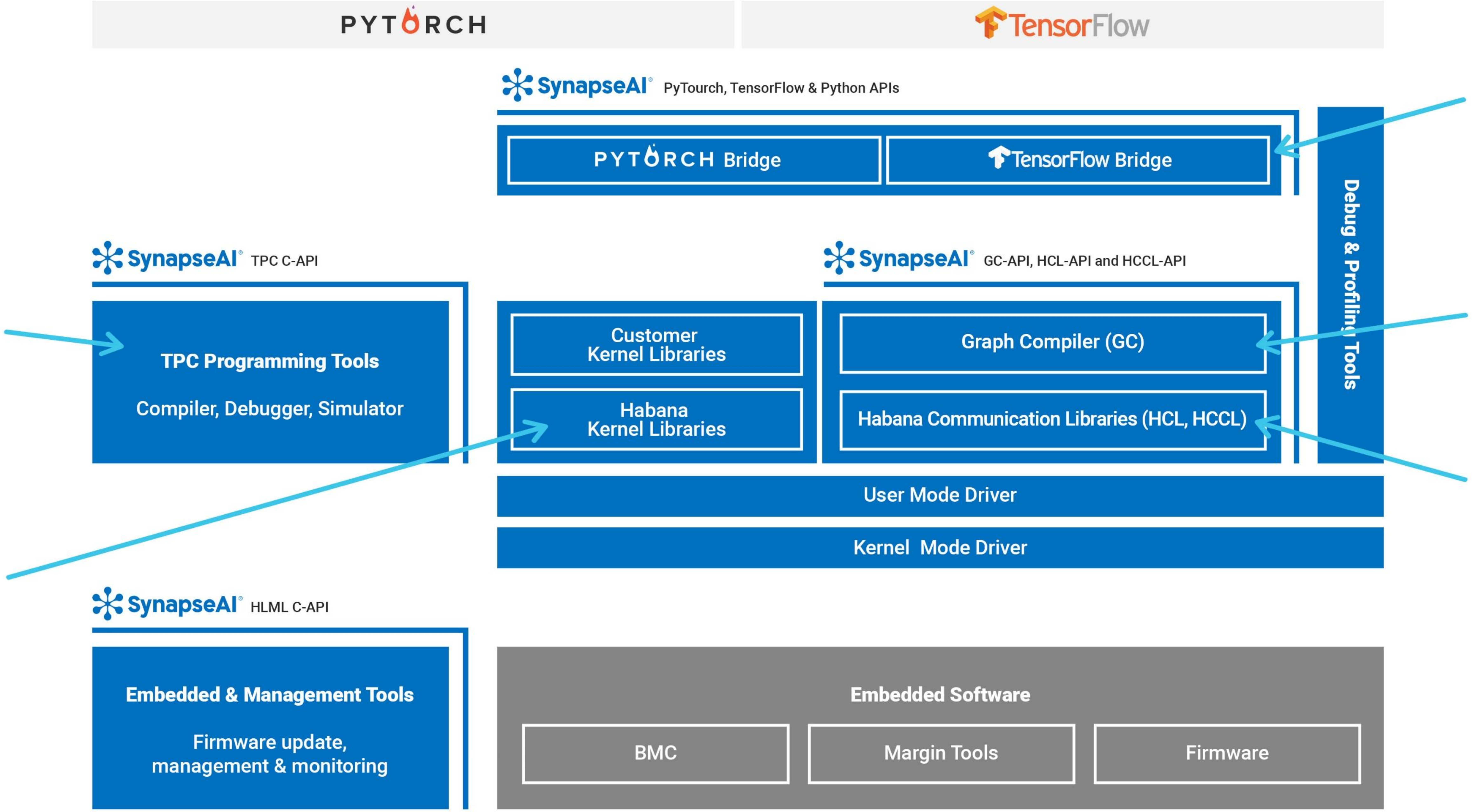
Scaling within a Gaudi Server

- 8 Gaudi OCP OAM cards
- 24 x 100GbE RDMA RoCE for scale-out
- Non-blocking, all-2-all internal interconnect across Gaudi AI processors
- Separate PCIe ports for external Host CPU traffic



Example of Integrated Server with eight Gaudi AI processors, two Xeon CPU and multiple Ethernet Interfaces

Software Suite Detail



LLVM-based TPC-C compiler, simulator and debugger facilitate development of custom TPC kernels. TensorFlow and PyTorch.

Rich TPC kernel library, with wide variety of operations (for example, elementwise, non-linear, non-GEMM operators).

TensorFlow and PyTorch integrated with SynapseAI and optimized for Gaudi. Developers enjoy the same abstraction that they are accustomed to today.

Graph compiler generates optimized binary code that implements model topology on Gaudi

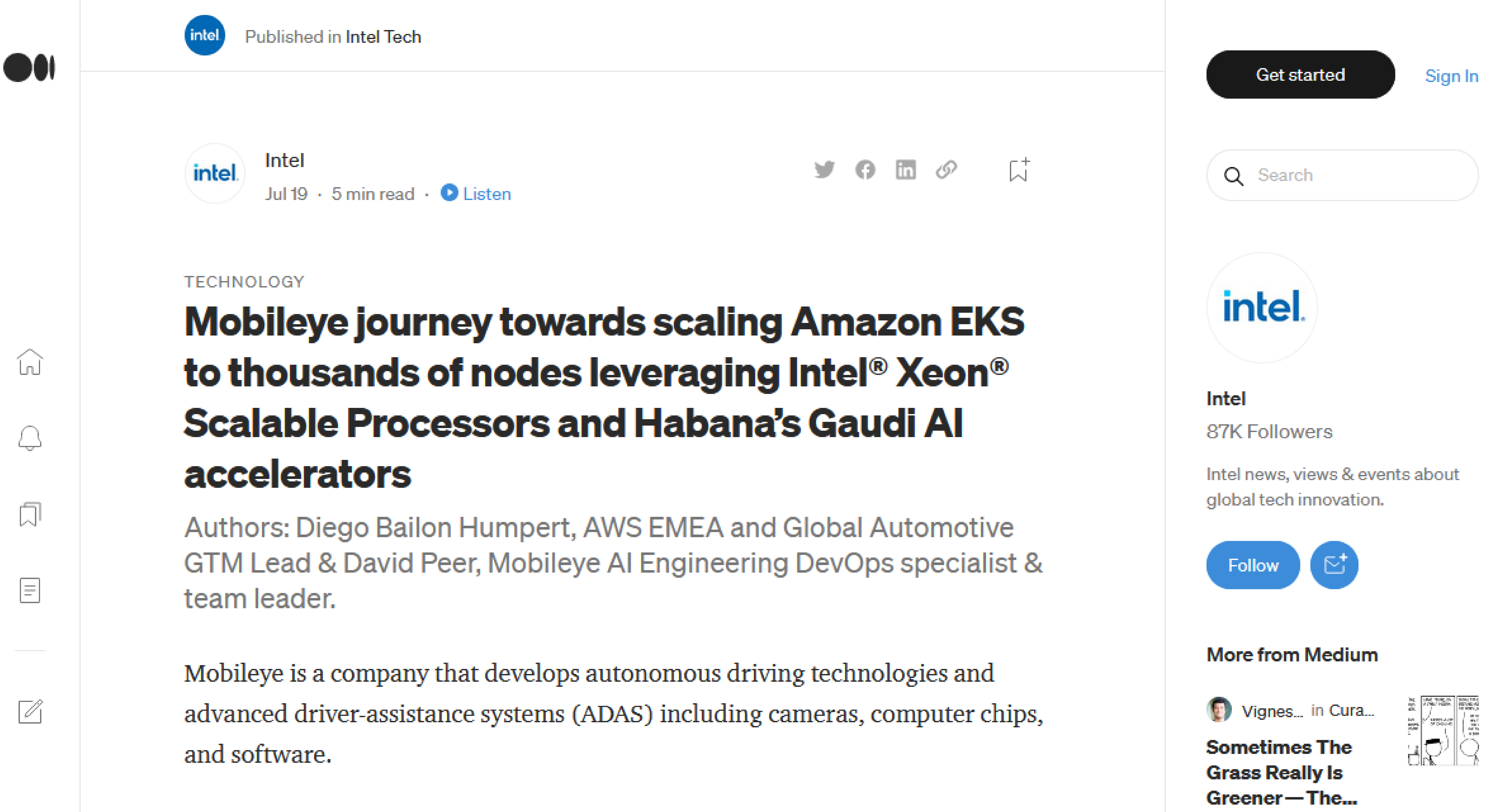
Implements efficient scale-up between Gaudi cards within a single node and scale-out across nodes for distributed training. HCL uses Gaudi integrated NICs for both scale-up and scale-out. HCCL allows users to enable Gaudi integrated NIC for scale-up and host NIC for scale-out, using NCCL-compatible API

Mobileye

Custom object detection
(2D and 3D) models trained on Gaudi

"Multiple teams across Mobileye have chosen to use Gaudi-accelerated training machines, either on Amazon EC2 DL1 instances or on-prem; Those teams consistently see significant cost-savings relative to existing GPU-based instances across model types, enabling them to achieve much better Time-To-Market for existing models or training much larger and complex models…We're excited to see Gaudi2's leap in performance"

Gaby Hayon, EVP R&D, Mobileye



Published in Intel Tech

Intel

Jul 19 · 5 min read

TECHNOLOGY

Mobileye journey towards scaling Amazon EKS to thousands of nodes leveraging Intel® Xeon® Scalable Processors and Habana's Gaudi AI accelerators

Authors: Diego Bailon Humpert, AWS EMEA and Global Automotive GTM Lead & David Peer, Mobileye AI Engineering DevOps specialist & team leader.

Mobileye is a company that develops autonomous driving technologies and advanced driver-assistance systems (ADAS) including cameras, computer chips, and software.

Intel

87K Followers

Intel news, views & events about global tech innovation.

More from Medium

Vignes... in Cura...

Sometimes The Grass Really Is Greener — The...

Accelerating Medical Imaging Applications

Objective

Demonstrate Gaudi DL1 AI processor cost-efficiency (price-performance ratio) for training deep learning models to detect novel coronavirus pneumonia in frontal chest X-ray images.

Models

- Pretraining: CheXNet, to detect and localize multiple kinds of diseases from chest X-ray images.
- Finetuning: COVID-CXNet, to detect novel coronavirus pneumonia in frontal chest X-ray images
  - Transfer learning of CheXNet with a focus on Grad-CAM visualizations.

Datasets

- 3200 normal images from NIH CXR dataset excluding age < 18 images based on paper
- 845 COVID-19 images from dataset used in the paper excluding age < 18 and early stage images.

Source: Leidos

DL1 Cost Savings

CheXNet-Keras

Dataset: ChestXray-NIHCC

Batch size: 32

Precision: FP32

Device count: 8

Instance	On-Demand hourly rate of EC2 instance [$/Hour]	Time per epoch [Seconds]	Cost per epoch [$]	DL1 Cost Savings to EC2 Customers [%]
8x V100-32 GB* (p3dn.24xlarge)	$31.21	4.6	$143.57	59%
8x Gaudi DL1.24xlarge**	$13.11	4.47	$58.56	

COVID-CXNet

Dataset: COVID-CXNet

Batch size: 16

Precision: BF16

Device Count: 1

Instance	On-Demand hourly rate of EC2 instance [$/Hour]	Time per epoch [Seconds]	Cost per epoch [$]	DL1 Cost Savings to EC2 Customers [%]
8x V100-32 GB* (p3dn.24xlarge)	$31.21	718	$6.22	67%
8x Gaudi DL1.24xlarge**	$13.11	565	$2.06	

Results may vary.
Pricing is published by Amazon: . https://aws.amazon.com/ec2/pricing/on-demand/
Pricing data accessed: May 2022

Source: Leidos

Summary

- Using Amazon EC2 DL1 instances for Chest X-Ray COVID Detection model pretraining and finetuning resulted in 60%+ savings in cost of training

- Successfully trained deep learning models on EC2 DL1 platform with minimal code changes

- Excellent support and documentation available on Habana Developer Site https://developer.habana.ai and GitHub with reference models

Source: Leidos

Accelerating Medical Benefit Application Processing

Leidos customer using NLP-based deep learning solution to facilitate medical benefit application processing

Objective: Demonstrate price/performance of Gaudi based EC2 DL1.24xlarge instance versus GPU based G4DN.12xlarge EC2 instance used by customer

- TensorFlow DistilBERT Model finetuned for a multi-labeling classification task
- Trained with 737k labeled examples and tested against 184k test examples

Attribute	G4DN.12xlarge (GPU based)	DL1.24xlarge (Gaudi Based)
Memory	192 GB	768 GB
Accelerator	4 x Tesla T4	8 x Gaudi
Accelerator Type	GPU	HPU
On Demand Cost (per hour)	$3.91	$13.11

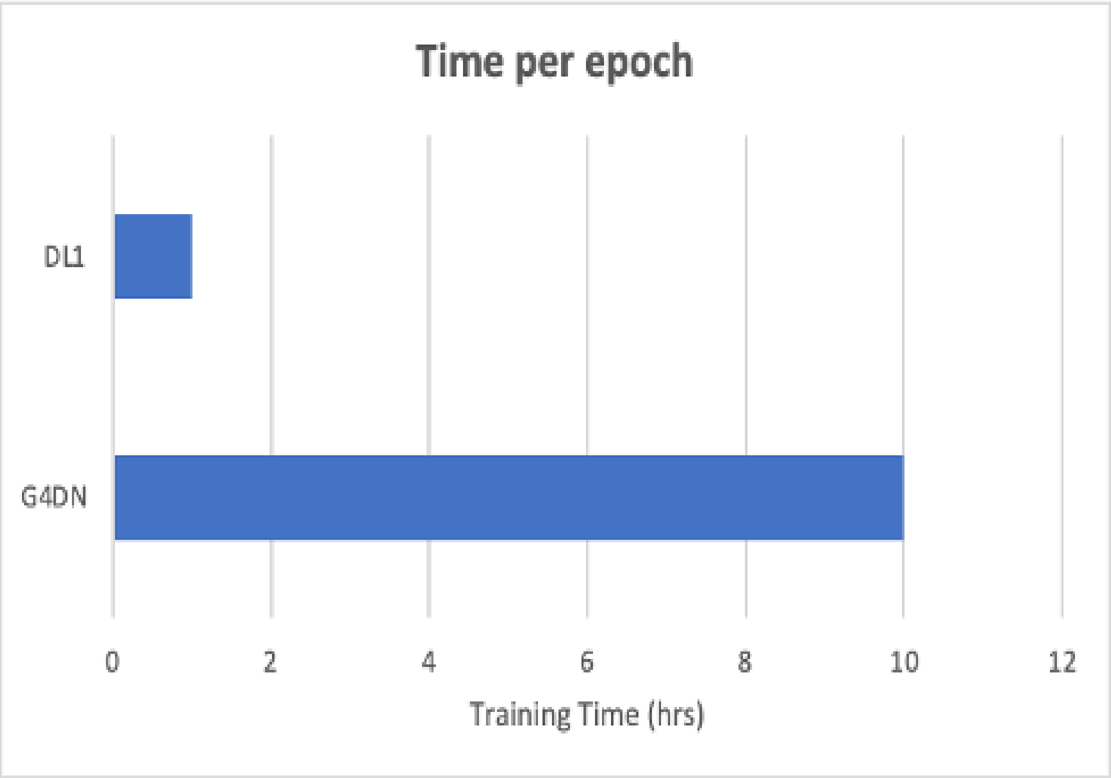
Source: Leidos

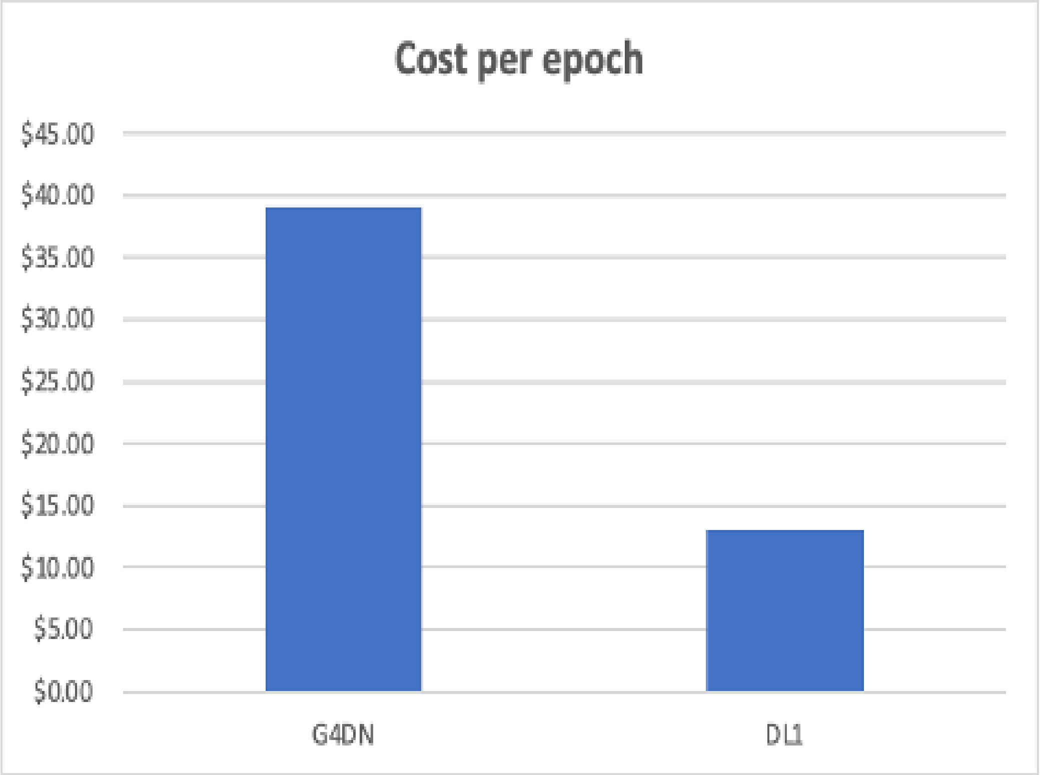
Results

Cost Performance

- Compared to GPU, training took only 45% of the time with 1x Gaudi processor and only 10% of the time when all 8 Gaudi processors were used
- Although the DL1 instance costs more per hour, due to shorter training time, the total cost for model training ended up being only 1/3 of the baseline, i.e., 66% cost savings.

Device	Training Time	Training Cost	DL1 cost savings
GPU (g4dn)	10 hrs	$39.12	
Gaudi x1 (dl1)	4.5 hrs	$59.00	
Gaudi x8 (dl1)	1 hr	$13.11	66%





User Experience

- Performing single card training is quite intuitive and simple
- Distributed training is simple when using Horovod and OpenMPI

Results may vary.
Pricing is published by Amazon: . https://aws.amazon.com/ec2/pricing/on-demand/
Pricing data accessed: May 2022

Source: Leidos

Summary

- Gaudi performs significantly better than NVIDIA Tesla T4 when looking at the time and cost metrics
- DL1 is worthy of strong consideration as part of cloud-forward strategy, especially when an organization anticipates using deep learning models.
- For Leidos' customers who are interested in continuing training, low cost is a very attractive feature

Areas for Future Work:

- With lower cost to train with Gaudi, one can potentially train/update more complex and accurate models
- Pre-training or continue to pre-train domain specific model, which is a more computing expensive task. This is critical for domain adaptation, which impacts a broad range of NLP tasks and is relevant to lot of our clients.

Source: Leidos

> 60% cost savings with DL1 vs. GPU instances

leidos

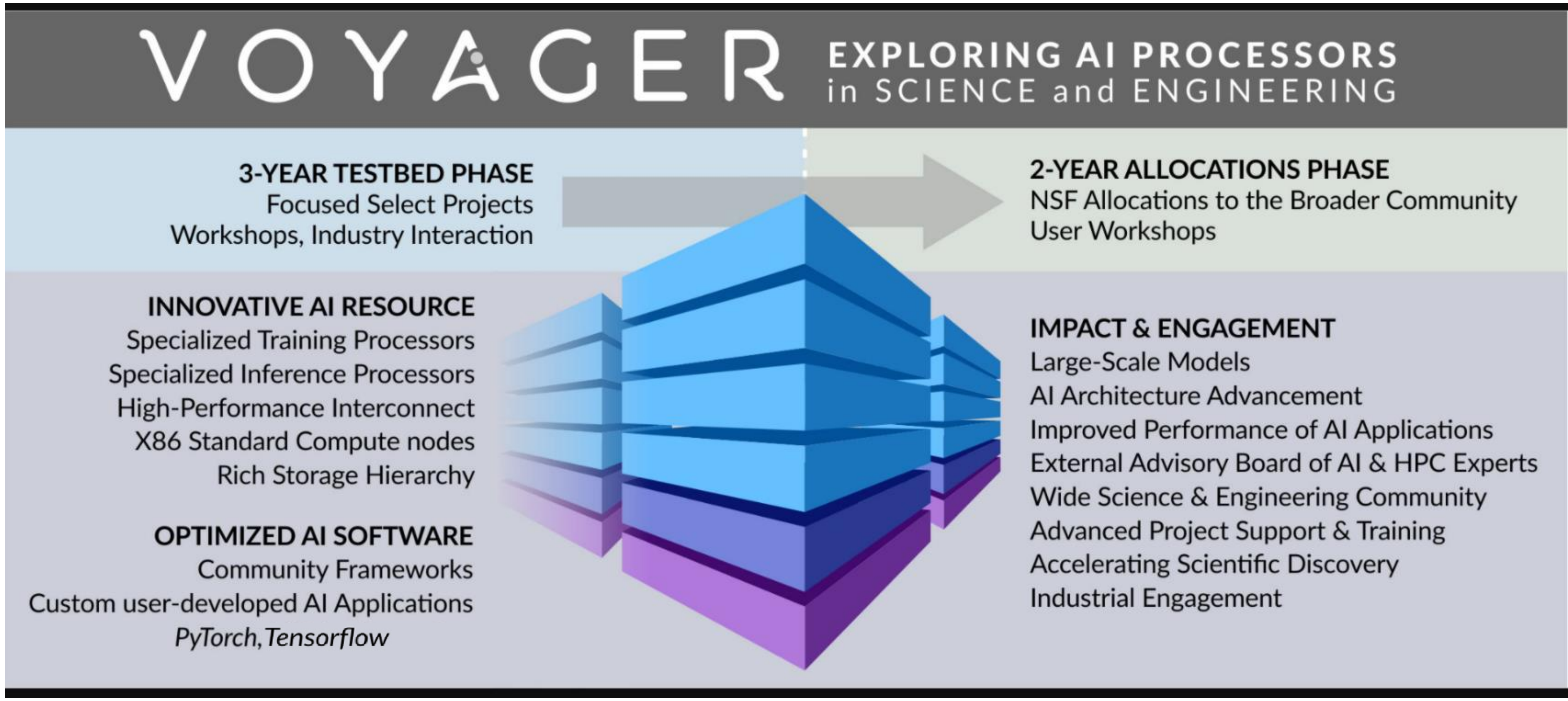
“Given Leidos and its customers’ need for quick, easy, and cost-effective training for deep learning models, we are excited to have begun this journey with Intel and AWS to use Amazon EC2 DL1 instances based on Habana Gaudi AI processors.”

Chetan Paul, CTO Health and Human Services at Leidos

Habana AI Powers SDSC's Voyager Research Program

336 Gaudi Training accelerators with native RoCE scaling and 16 Goya Inference processors

- In service since Fall of 2021
- Funded by $5M grant from National Science Foundation
  - Matching funds targeting community support and operation
- AI research conducted across range of science and engineering domains
  - Astronomy, climate sciences, chemistry, particle physics,
- Announced by SDSC in July 2020, more information here.



VOYAGER EXPLORING AI PROCESSORS in SCIENCE and ENGINEERING

3-YEAR TESTBED PHASE
Focused Select Projects
Workshops, Industry Interaction

2-YEAR ALLOCATIONS PHASE
NSF Allocations to the Broader Community
User Workshops

INNOVATIVE AI RESOURCE
Specialized Training Processors
Specialized Inference Processors
High-Performance Interconnect
X86 Standard Compute nodes
Rich Storage Hierarchy

OPTIMIZED AI SOFTWARE
Community Frameworks
Custom user-developed AI Applications
PyTorch,Tensorflow

IMPACT & ENGAGEMENT
Large-Scale Models
AI Architecture Advancement
Improved Performance of AI Applications
External Advisory Board of AI & HPC Experts
Wide Science & Engineering Community
Advanced Project Support & Training
Accelerating Scientific Discovery
Industrial Engagement

Combining Fire Science With AI For Wildfire Mitigation

DL algorithms of satellite images determine land covers across geographies in the context of wildfire management

SDSC SAN DIEGO SUPERCOMPUTER CENTER

"With innovative solutions optimized for deep learning operations and AI workloads, Habana accelerators are excellent choices to power Voyager's forthcoming AI research"

Amit Majumdar, Director of Data Enabled Scientific Computing Division, SDSC

First-generation Gaudi

IN THE CLOUD

- AWS EC2 DL1 Instances
- Leading AWS AI training efficiency

ON PREMISES

- Supermicro X12 Gaudi Server
- DDN AI 400X2 storage solution

MLOps SOFTWARE

cnvrg.io

Getting Started with TensorFlow on Gaudi®

```
import tensorflow as tf

from TensorFlow.common.library_loader import load_habana_module
load_habana_module()

(x_train, y_train), (x_test, y_test) =
tf.keras.datasets.mnist.load_data()
x_train, x_test = x_train / 255.0, x_test / 255.0

model = tf.keras.models.Sequential([
                          tf.keras.layers.Flatten(input_shape=(28, 28)),
                          tf.keras.layers.Dense(10),
])
loss = tf.keras.losses.SparseCategoricalCrossentropy(from_logits=True)
optimizer = tf.keras.optimizers.SGD(learning_rate=0.01)

model.compile(optimizer=optimizer, loss=loss, metrics=['accuracy'])

model.fit(x_train, y_train, epochs=5, batch_size=128)
model.evaluate(x_test, y_test)
```

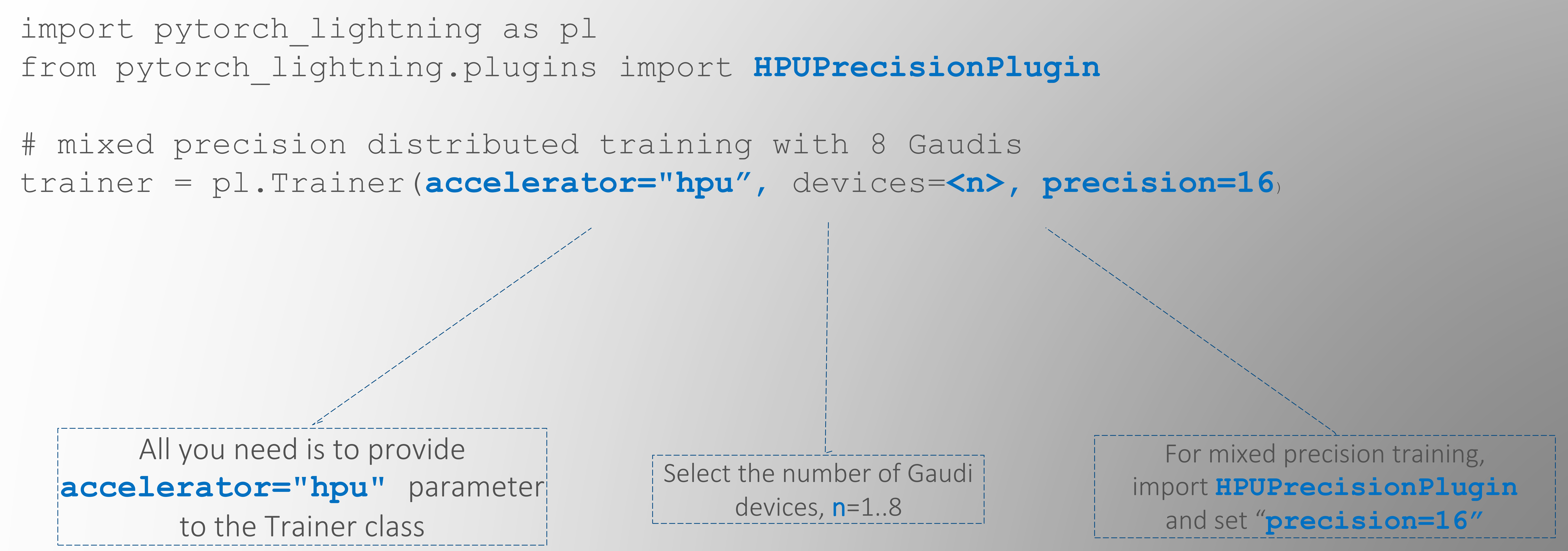
Load the Habana® libraries needed to use Gaudi aka **HPU** device

Once loaded, the **HPU** device is registered in TensorFlow

When an Op is available for both CPU and HPU, Op is assigned to the HPU

When an Op is not supported on HPU, it runs on the CPU

Getting Started With PyTorch Lightning On Gaudi



```
import pytorch_lightning as pl
from pytorch_lightning.plugins import HPUPrecisionPlugin

# mixed precision distributed training with 8 Gaudis
trainer = pl.Trainer(accelerator="hpu”, devices=<n>, precision=16)
```

All you need is to provide **`acelerator="hpu"`** parameter to the Trainer class

Select the number of Gaudi devices, n=1..8

For mixed precision training, import **`HPUPrecisionPlugin`** and set “**`precision=16`**”

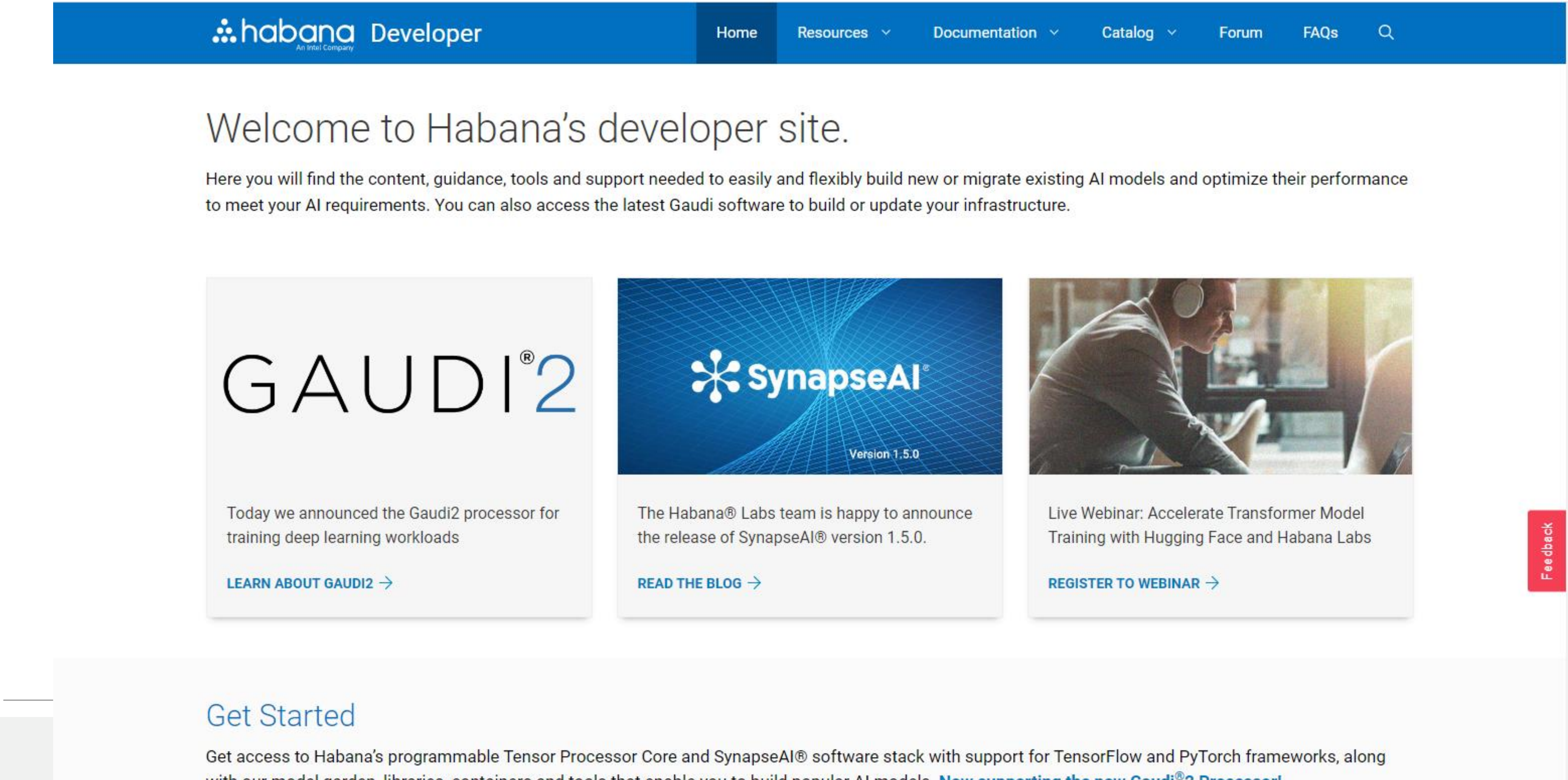
Lightning 1.6 now supports HPU with SynapseAI 1.4: https://pytorch-lightning.readthedocs.io/en/stable/accelerators/hpu.html

Getting Started With Huggingface On Gaudi

```
from optimum.habana import GaudiConfig, GaudiTrainer, GaudiTrainingArguments
from transformers import BertTokenizer, BertModel
…
tokenizer = BertTokenizer.from_pretrained("bert-base-uncased")
model = BertModel.from_pretrained("bert-base-uncased")
gaudi_config = GaudiConfig.from_pretrained("Habana/bert-base-uncased")
args = GaudiTrainingArguments(
    output_dir="/tmp/output_dir",
    use_habana=True,
    use_lazy_mode=True,
)
trainer = GaudiTrainer(
    model=model,
    gaudi_config=gaudi_config,
    args=args,
    tokenizer=tokenizer,
)
trainer.train()
```

- Uses Optimum Habana library
- Model instantiated the same way as in the Transformers library
- Only difference is to load Gaudi configuration and provide to the Gaudi trainer

Habana Developer Platform---developer.habana.ai



Welcome to Habana's developer site.

Here you will find the content, guidance, tools and support needed to easily and flexibly build new or migrate existing AI models and optimize their performance to meet your AI requirements. You can also access the latest Gaudi software to build or update your infrastructure.

Today we announced the Gaudi2 processor for training deep learning workloads

LEARN ABOUT GAUDI2 →

The Habana® Labs team is happy to announce the release of SynapseAI® version 1.5.0.

READ THE BLOG →

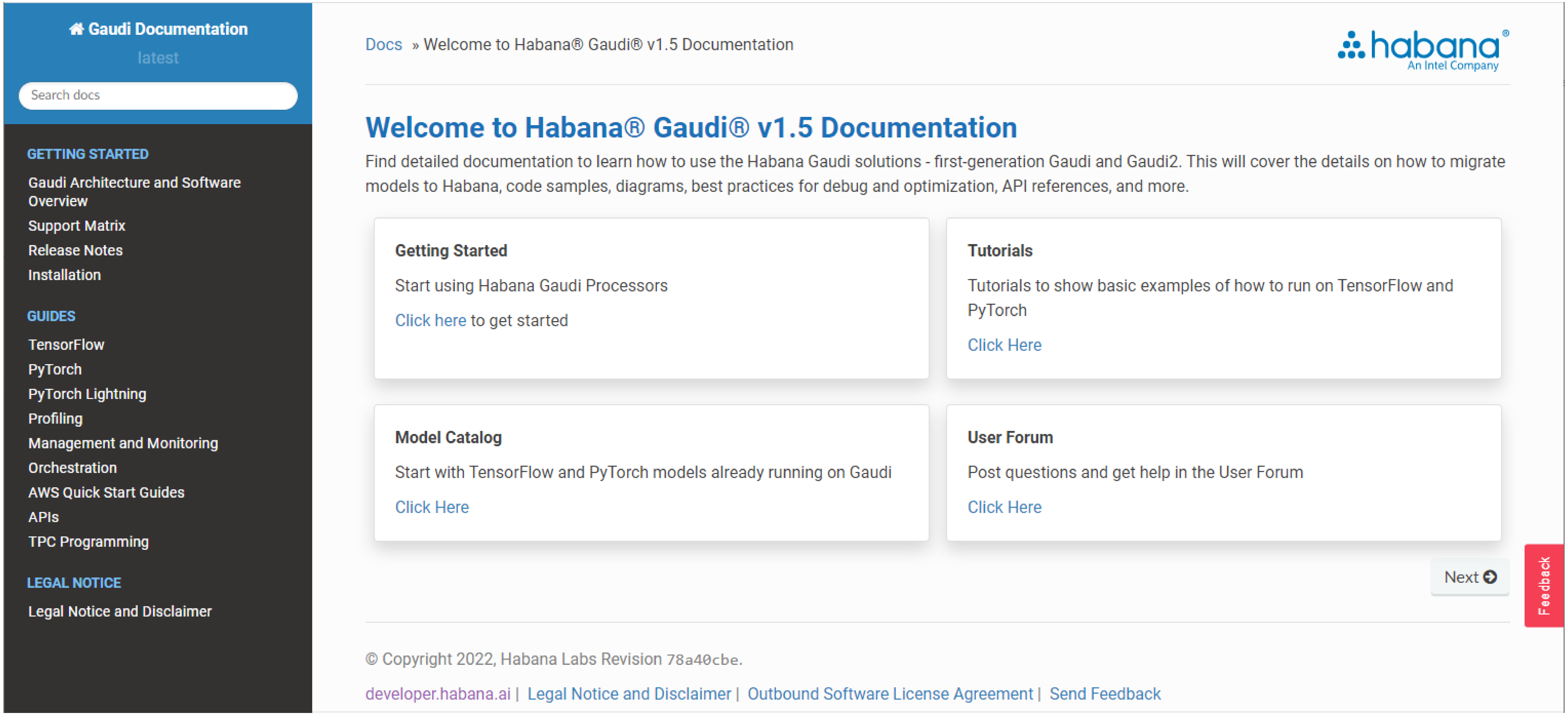
Live Webinar: Accelerate Transformer Model Training with Hugging Face and Habana Labs

REGISTER TO WEBINAR →

Get Started

Get access to Habana's programmable Tensor Processor Core and SynapseAI® software stack with support for TensorFlow and PyTorch frameworks, along with our model garden, libraries, containers and tools that enable you to build popular AI models. Now supporting the new Gaudi®2 Processor!

Habana Developer Documentation---docs.habana.ai



GETTING STARTED

- Gaudi Architecture and Software Overview
- Support Matrix
- Release Notes
- Installation

GUIDES

- TensorFlow
- PyTorch
- PyTorch Lightning
- Profiling
- Management and Monitoring
- Orchestration
- AWS Quick Start Guides
- APIs
- TPC Programming

LEGAL NOTICE

- Legal Notice and Disclaimer

Docs » Welcome to Habana® Gaudi® v1.5 Documentation

Welcome to Habana® Gaudi® v1.5 Documentation

Find detailed documentation to learn how to use the Habana Gaudi solutions - first-generation Gaudi and Gaudi2. This will cover the details on how to migrate models to Habana, code samples, diagrams, best practices for debug and optimization, API references, and more.

Getting Started

Start using Habana Gaudi Processors

Click here to get started

Tutorials

Tutorials to show basic examples of how to run on TensorFlow and PyTorch

Click Here

Model Catalog

Start with TensorFlow and PyTorch models already running on Gaudi

Click Here

User Forum

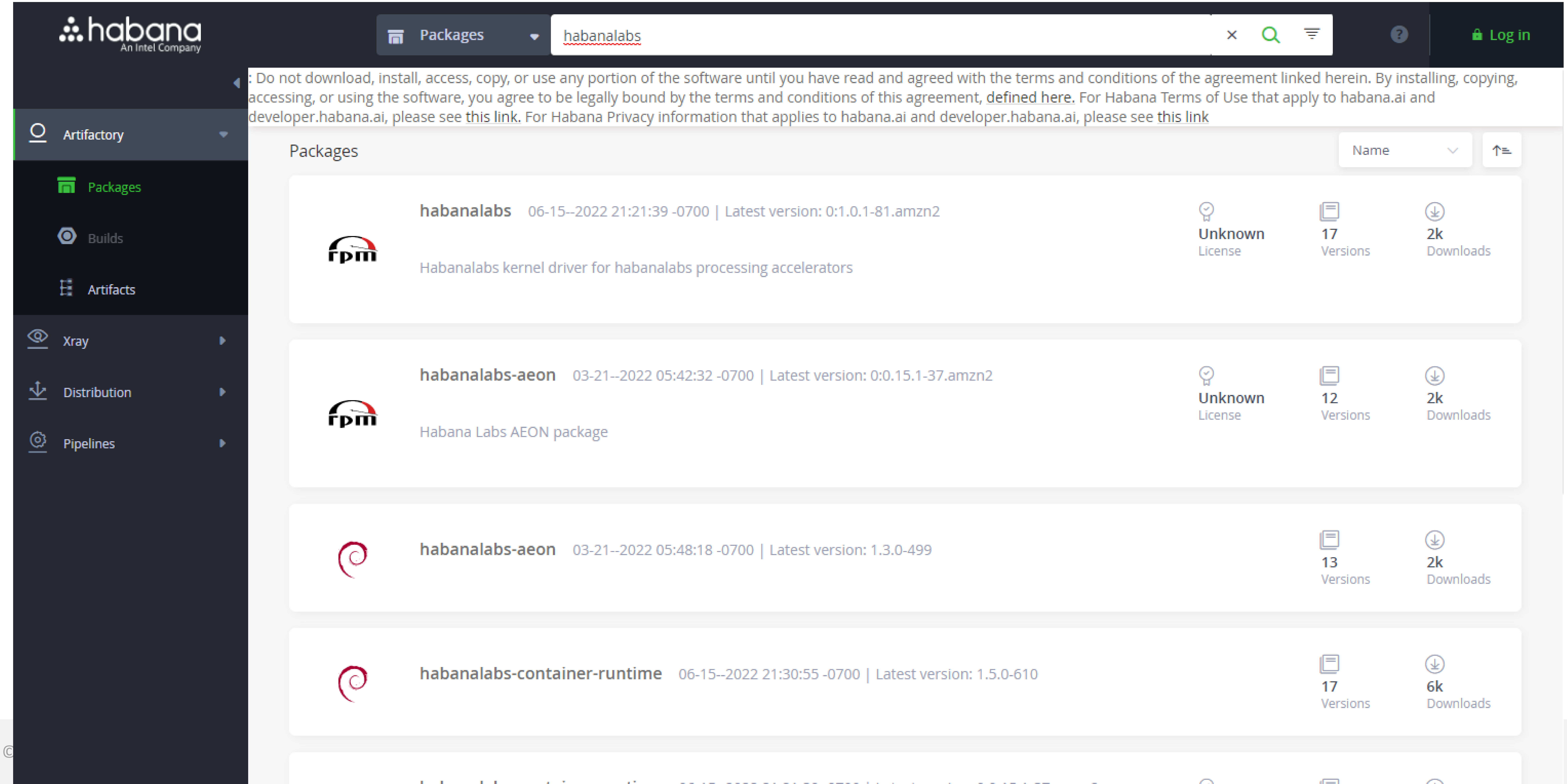
Post questions and get help in the User Forum

Click Here

© Copyright 2022, Habana Labs Revision `78a40cbe`.

developer.habana.ai | Legal Notice and Disclaimer | Outbound Software License Agreement | Send Feedback

Habana Developer Software---vault.habana.ai



Packages: habanalabs

: Do not download, install, access, copy, or use any portion of the software until you have read and agreed with the terms and conditions of the agreement linked herein. By installing, copying, accessing, or using the software, you agree to be legally bound by the terms and conditions of this agreement, defined here. For Habana Terms of Use that apply to habana.ai and developer.habana.ai, please see this link. For Habana Privacy information that applies to habana.ai and developer.habana.ai, please see this link

Packages

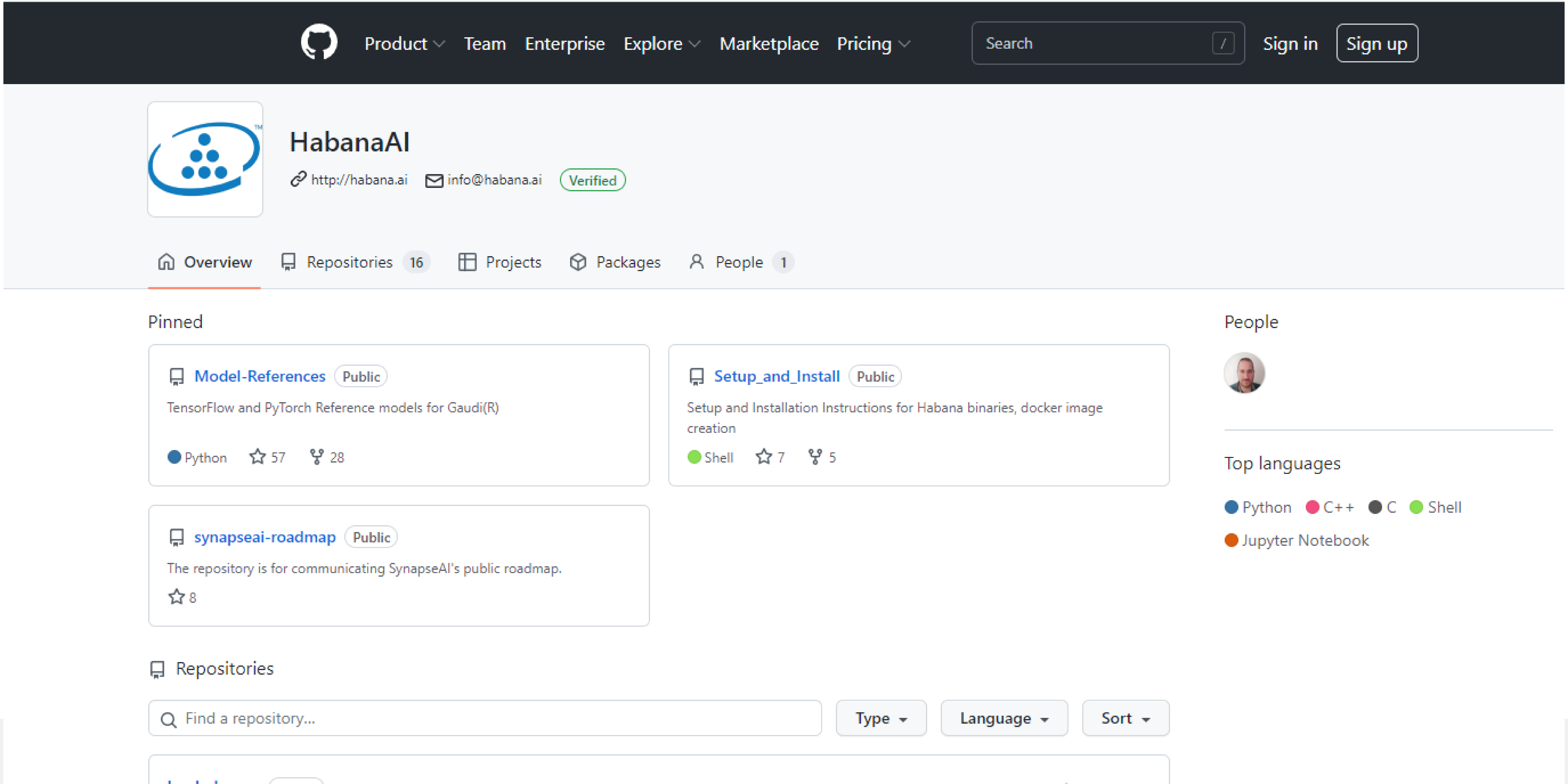
habanalabs 06-15--2022 21:21:39 -0700 | Latest version: 0:1.0.1-81.amzn2
Habanalabs kernel driver for habanalabs processing accelerators
Unknown License | 17 Versions | 2k Downloads

habanalabs-aeon 03-21--2022 05:42:32 -0700 | Latest version: 0:0.15.1-37.amzn2
Habana Labs AEON package
Unknown License | 12 Versions | 2k Downloads

habanalabs-aeon 03-21--2022 05:48:18 -0700 | Latest version: 1.3.0-499
13 Versions | 2k Downloads

habanalabs-container-runtime 06-15--2022 21:30:55 -0700 | Latest version: 1.5.0-610
17 Versions | 6k Downloads

Habana GitHub Repositories---github.com/HabanaAI



HabanaAI

http://habana.ai info@habana.ai Verified

Pinned

Model-References Public
TensorFlow and PyTorch Reference models for Gaudi(R)
● Python ☆ 57 28

Setup_and_Install Public
Setup and Installation Instructions for Habana binaries, docker image creation
● Shell ☆ 7 5

synapseai-roadmap Public
The repository is for communicating SynapseAI's public roadmap.
☆ 8

People

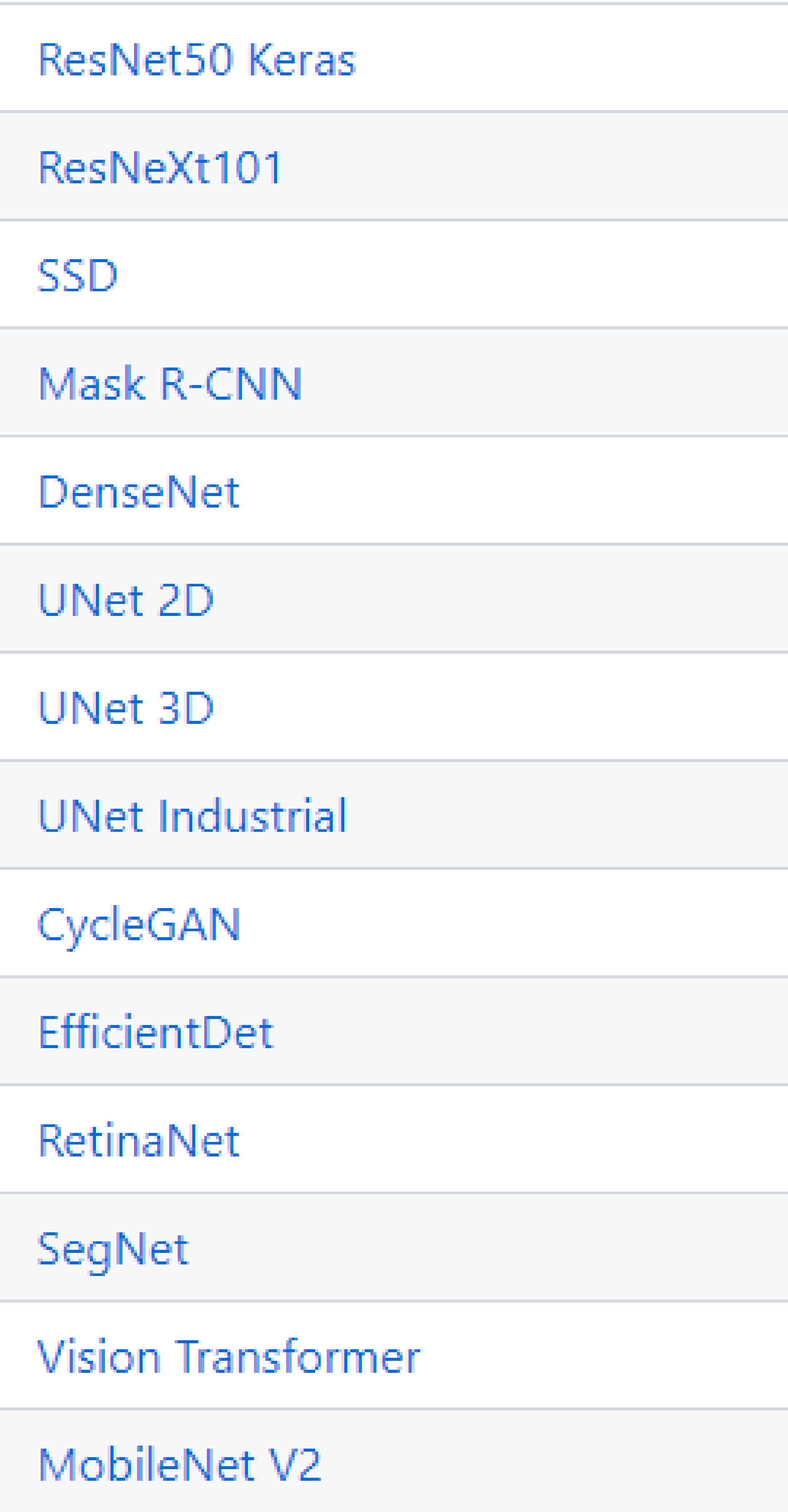
Top languages

● Python ● C++ ● C ● Shell ● Jupyter Notebook

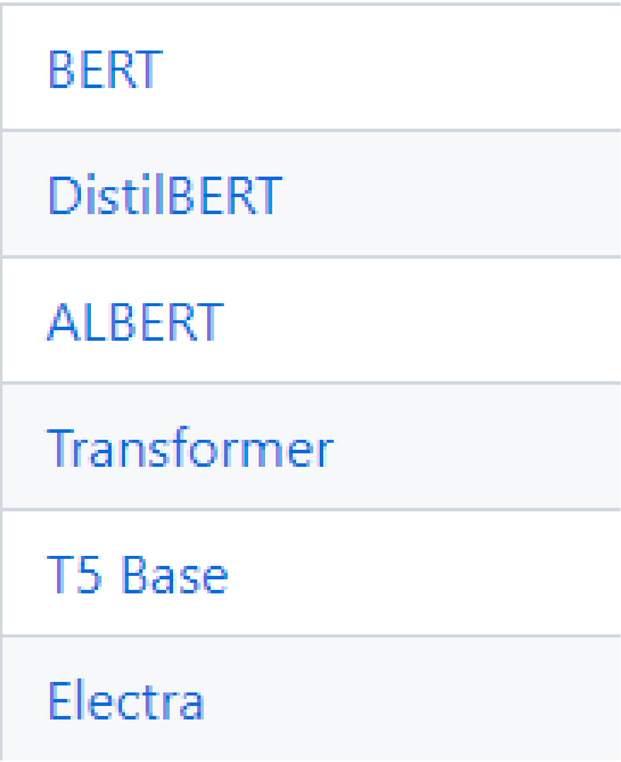
Repositories

Gaudi Reference Models---August 2022

TensorFlow

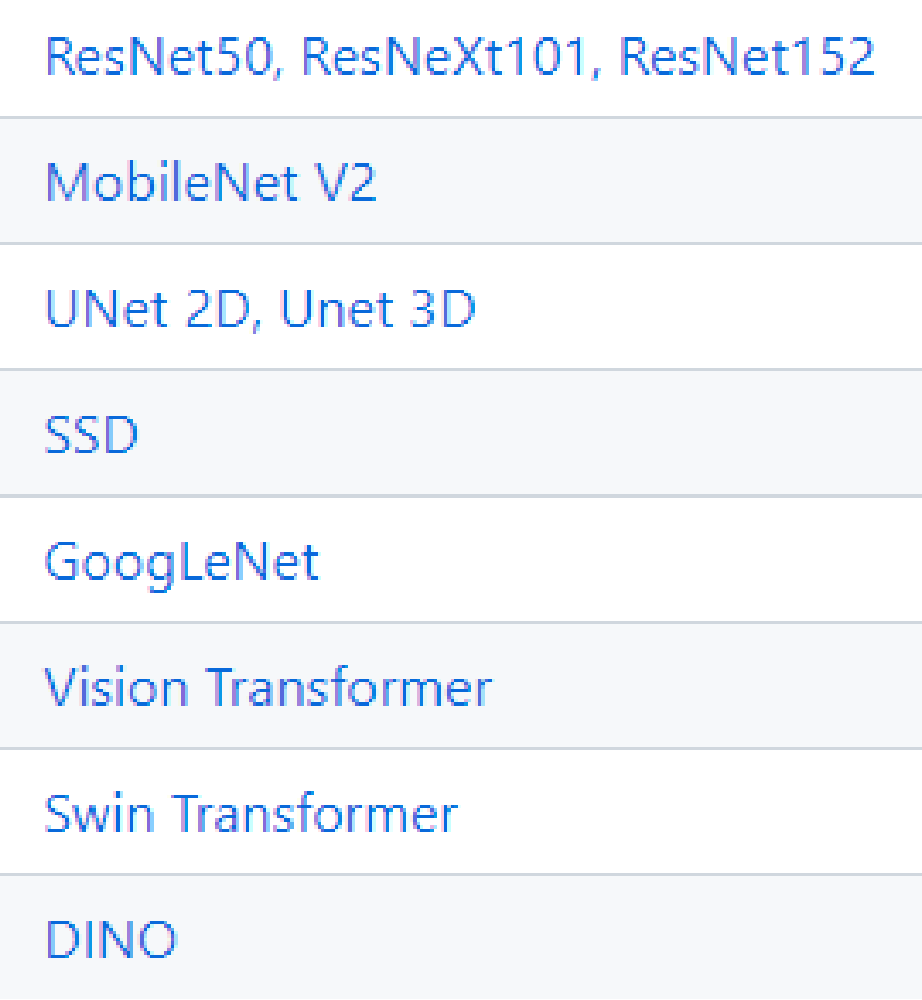


- ResNet50 Keras
- ResNeXt101
- SSD
- Mask R-CNN
- DenseNet
- UNet 2D
- UNet 3D
- UNet Industrial
- CycleGAN
- EfficientDet
- RetinaNet
- SegNet
- Vision Transformer
- MobileNet V2

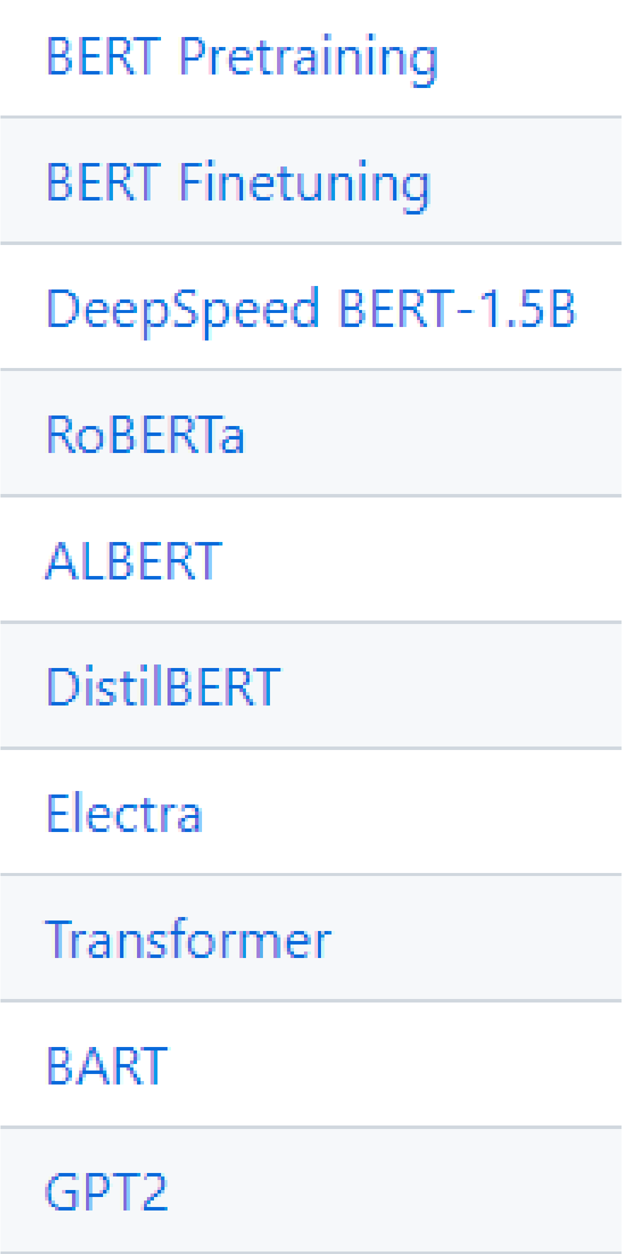


- BERT
- DistilBERT
- ALBERT
- Transformer
- T5 Base
- Electra

PyTorch

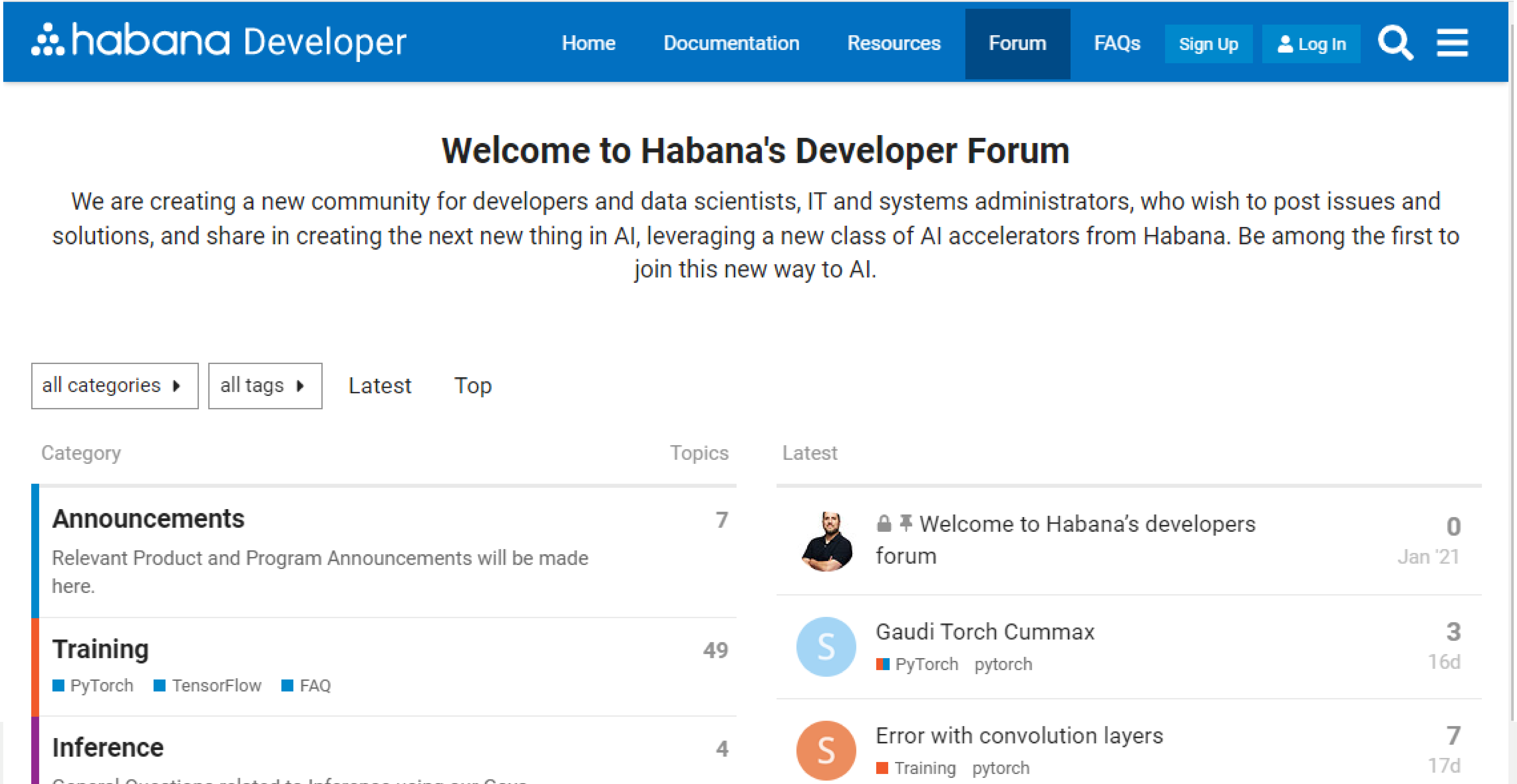


- ResNet50, ResNeXt101, ResNet152
- MobileNet V2
- UNet 2D, Unet 3D
- SSD
- GoogLeNet
- Vision Transformer
- Swin Transformer
- DINO



- BERT Pretraining
- BERT Finetuning
- DeepSpeed BERT-1.5B
- RoBERTa
- ALBERT
- DistilBERT
- Electra
- Transformer
- BART
- GPT2

Habana Developer Forum---forum.habana.ai



Welcome to Habana's Developer Forum

We are creating a new community for developers and data scientists, IT and systems administrators, who wish to post issues and solutions, and share in creating the next new thing in AI, leveraging a new class of AI accelerators from Habana. Be among the first to join this new way to AI.

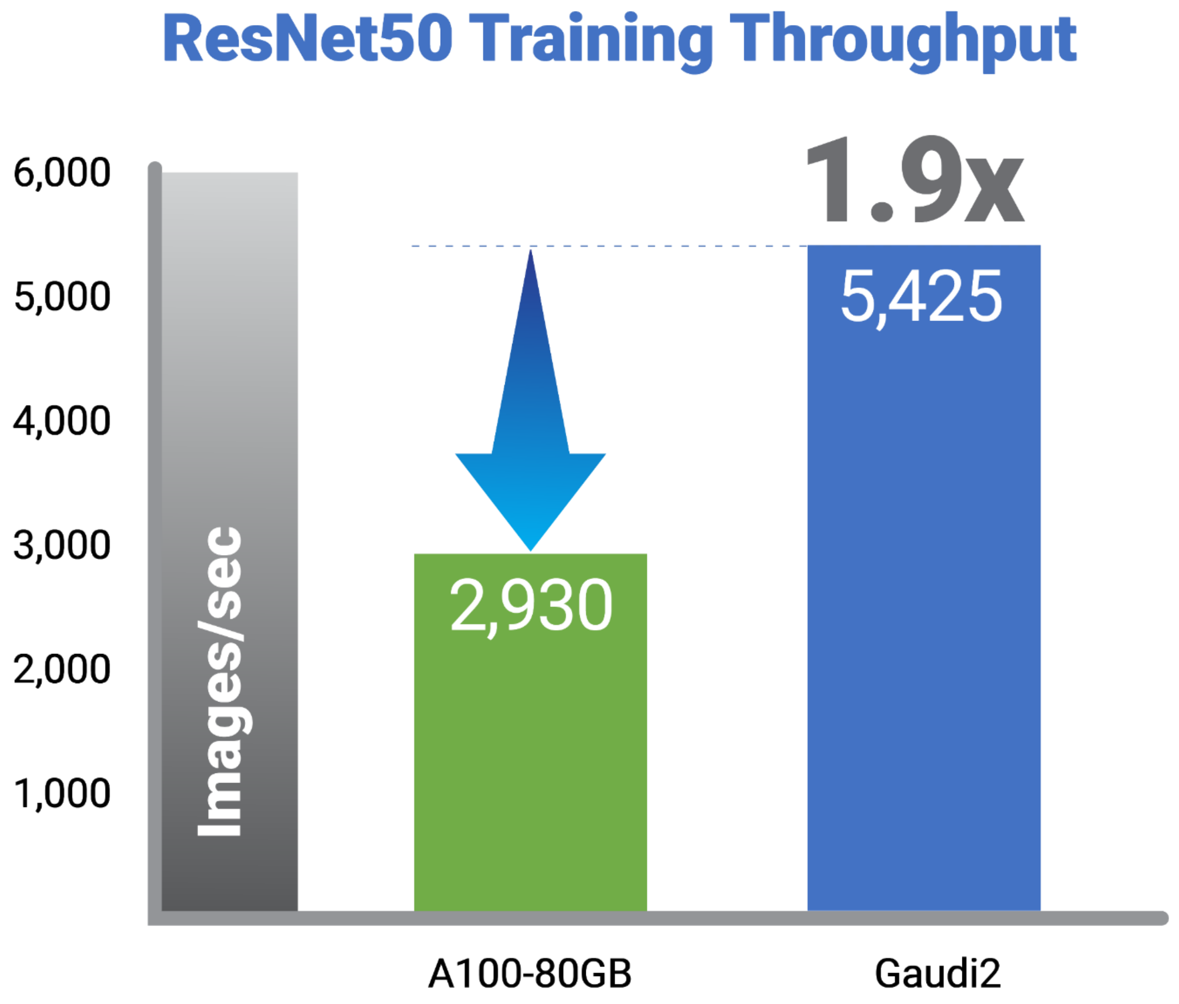
Category	Topics
Announcements Relevant Product and Program Announcements will be made here.	7
Training PyTorch TensorFlow FAQ	49
Inference	4

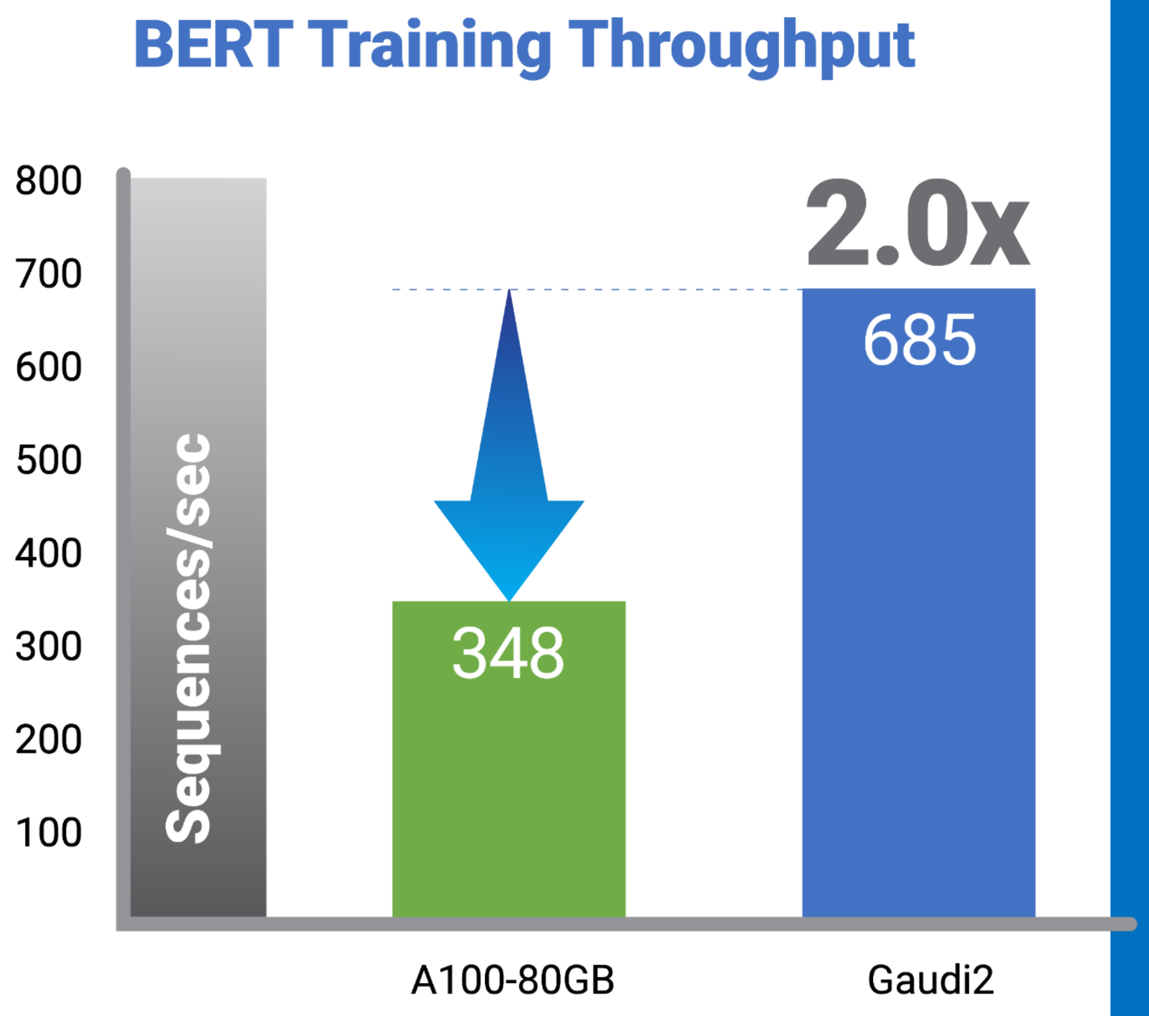
Latest		
Welcome to Habana's developers forum	0	Jan '21
Gaudi Torch Cummax PyTorch pytorch	3	16d
Error with convolution layers Training pytorch	7	17d

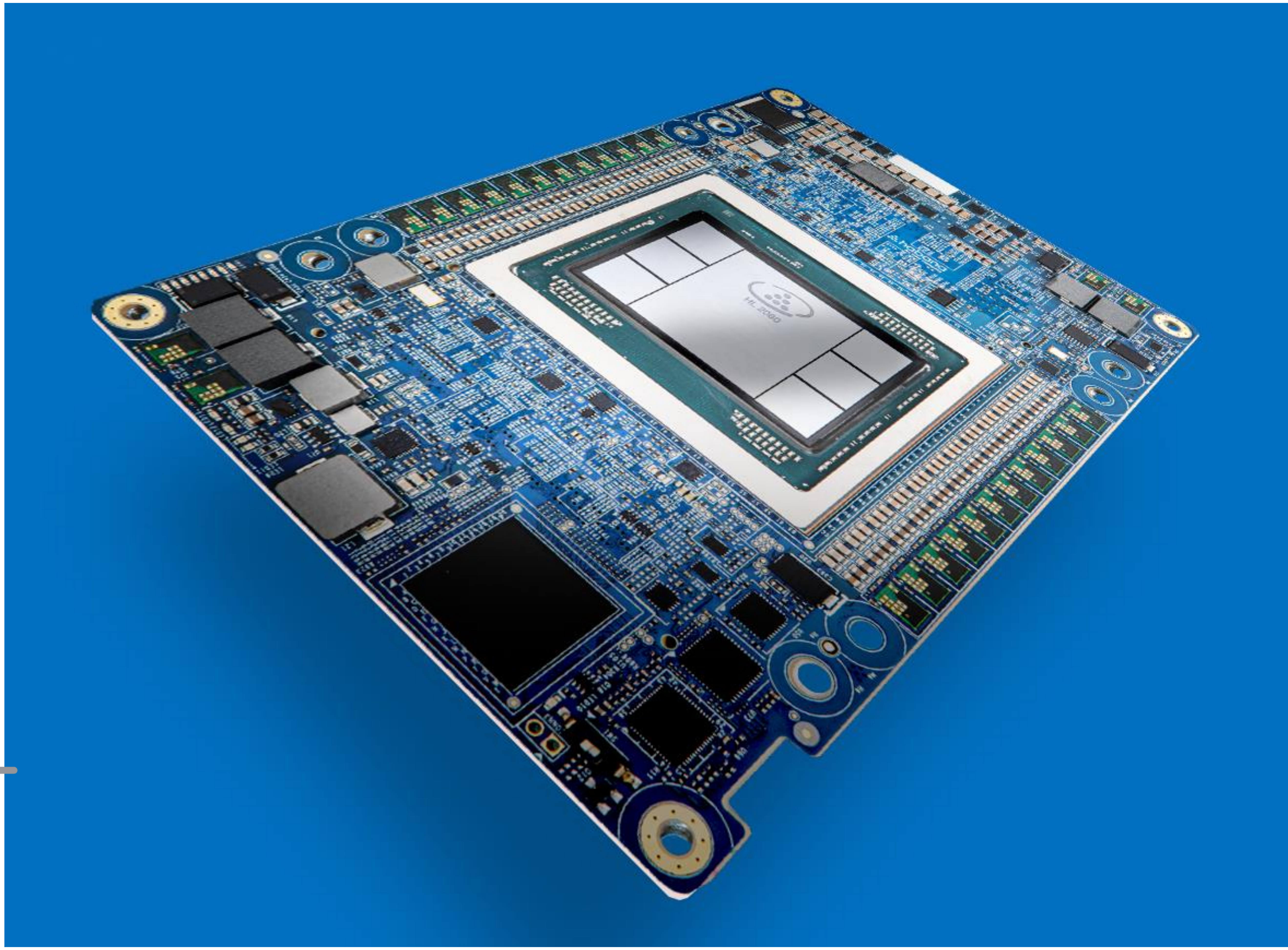
GAUDI® 2

Leadership Performance

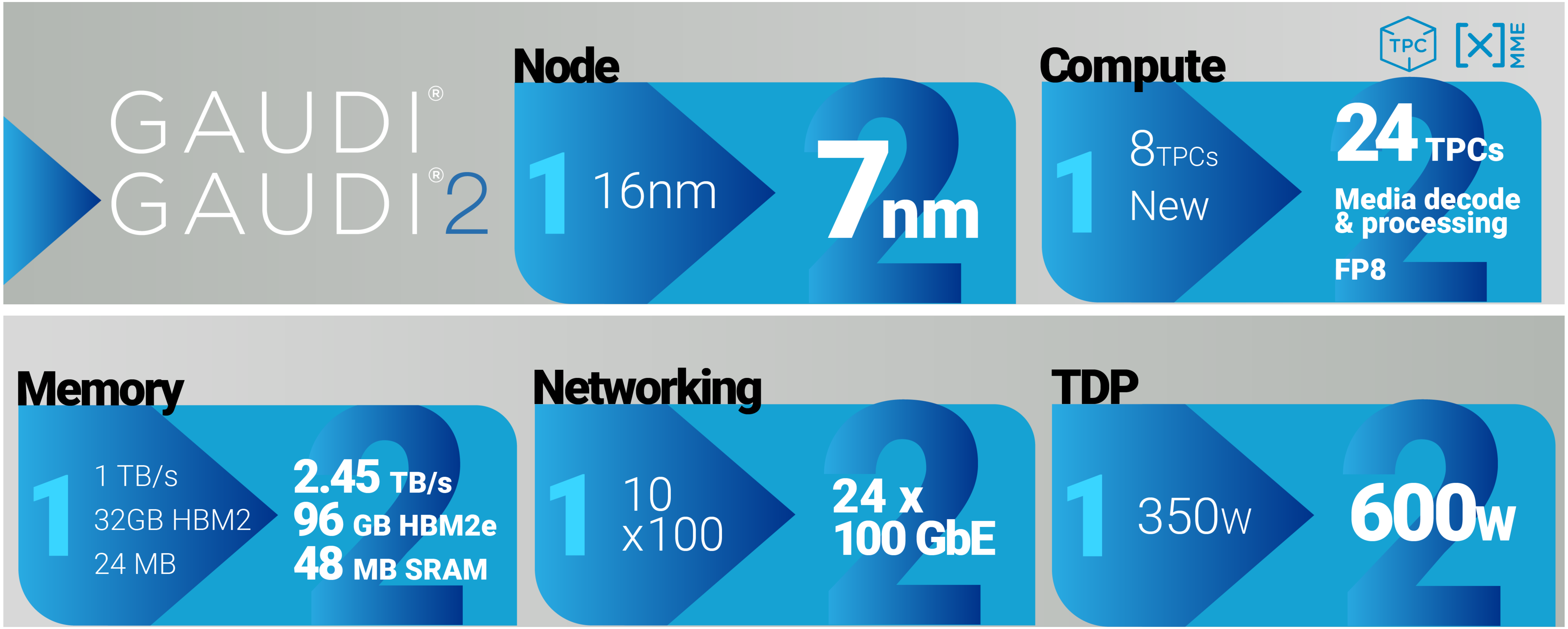
~2x better throughput vs A100
for popular vision and language models







Gaudi2: Second-generation Training & Inference



GAUDI®
GAUDI®2

Node

1 16nm

2 7nm

Compute

1 8TPCs
New

2 24 TPCs
Media decode & processing
FP8

Memory

1 1 TB/s
32GB HBM2
24 MB

2 2.45 TB/s
96 GB HBM2e
48 MB SRAM

Networking

1 10 x100

2 24 x 100 GbE

TDP

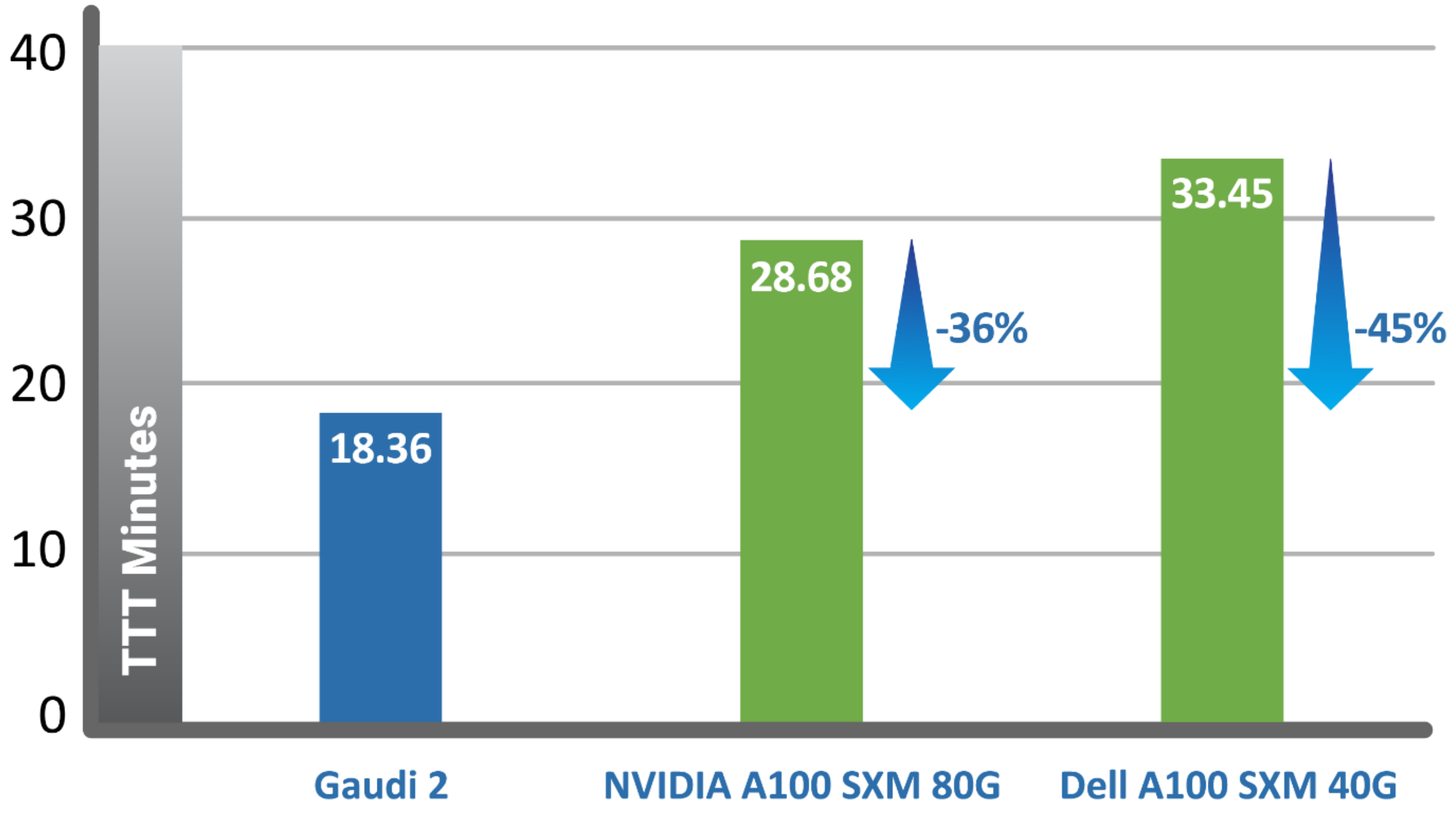
1 350W

2 600W

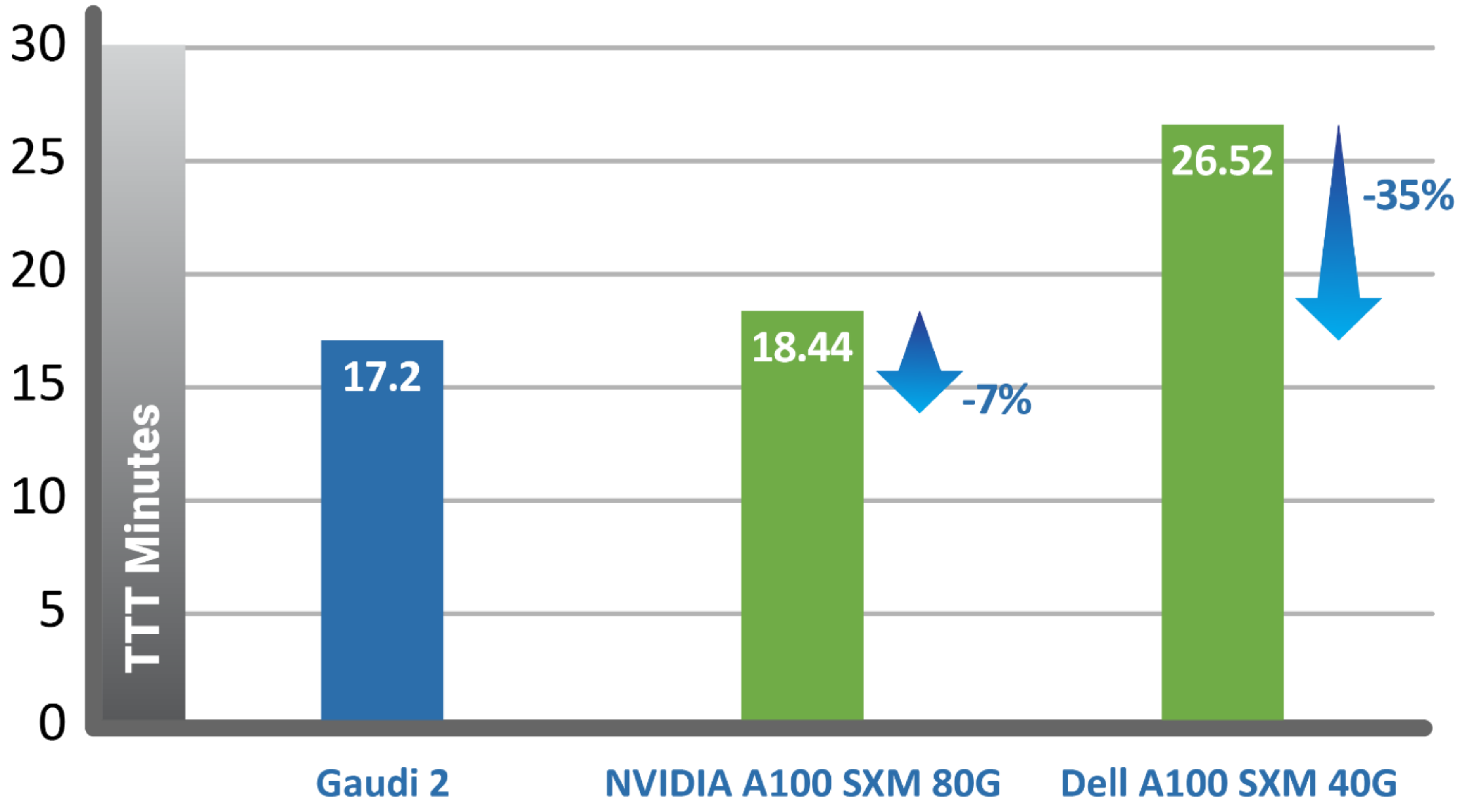
Gaudi2 MLPerf June '22 Training Benchmark Results

Gaudi2 outperformed Nvidia A100 MLPerf submissions on both ResNet and BERT

MLPERF ResNet-50 Training Time [lower is better]
(8 accelerator server)



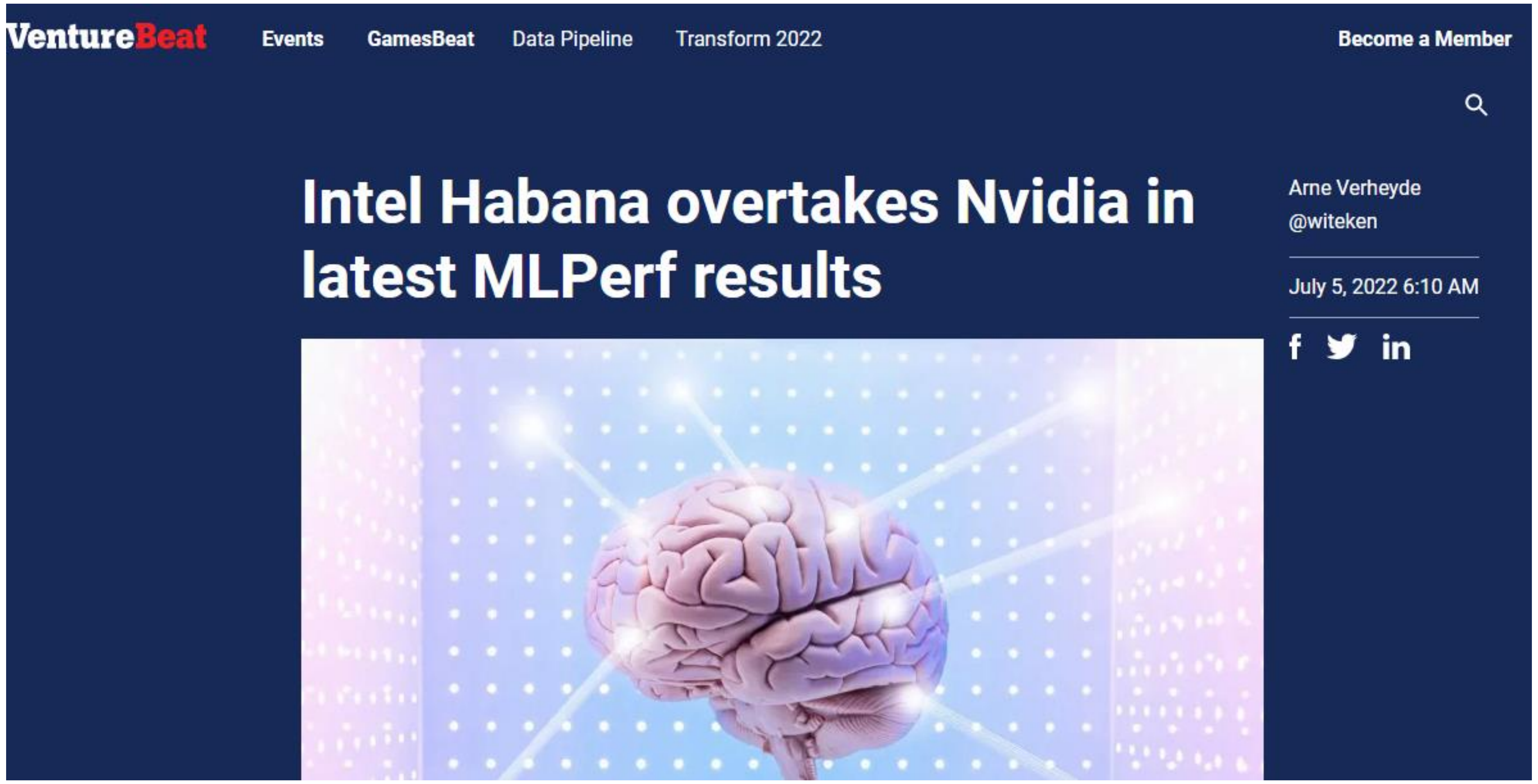
MLPERF BERT Training Time [lower is better]
(8 accelerator server)



…and First-gen Gaudi achieved near-ideal linear scale on 128- and 256-accelerators

Gaudi2 Time-to-Train (TTT) improved by 3 to 4.7x compared to First-gen Gaudi

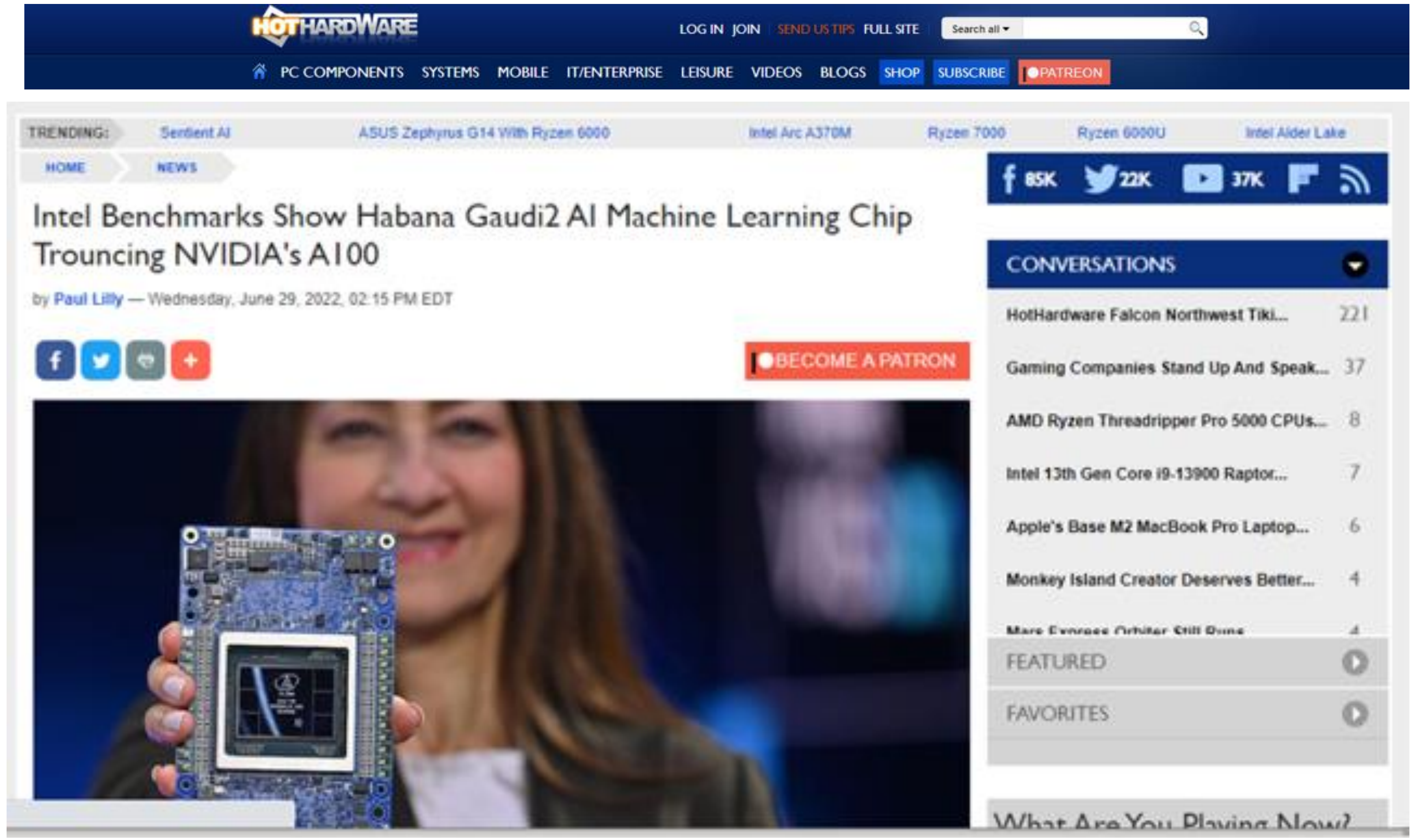
MLPerf Press Coverage



Intel Habana overtakes Nvidia in latest MLPerf results

Arne Verheyde
@witeken

July 5, 2022 6:10 AM



Intel Benchmarks Show Habana Gaudi2 AI Machine Learning Chip Trouncing NVIDIA's A100

by Paul Lilly — Wednesday, June 29, 2022, 02:15 PM EDT

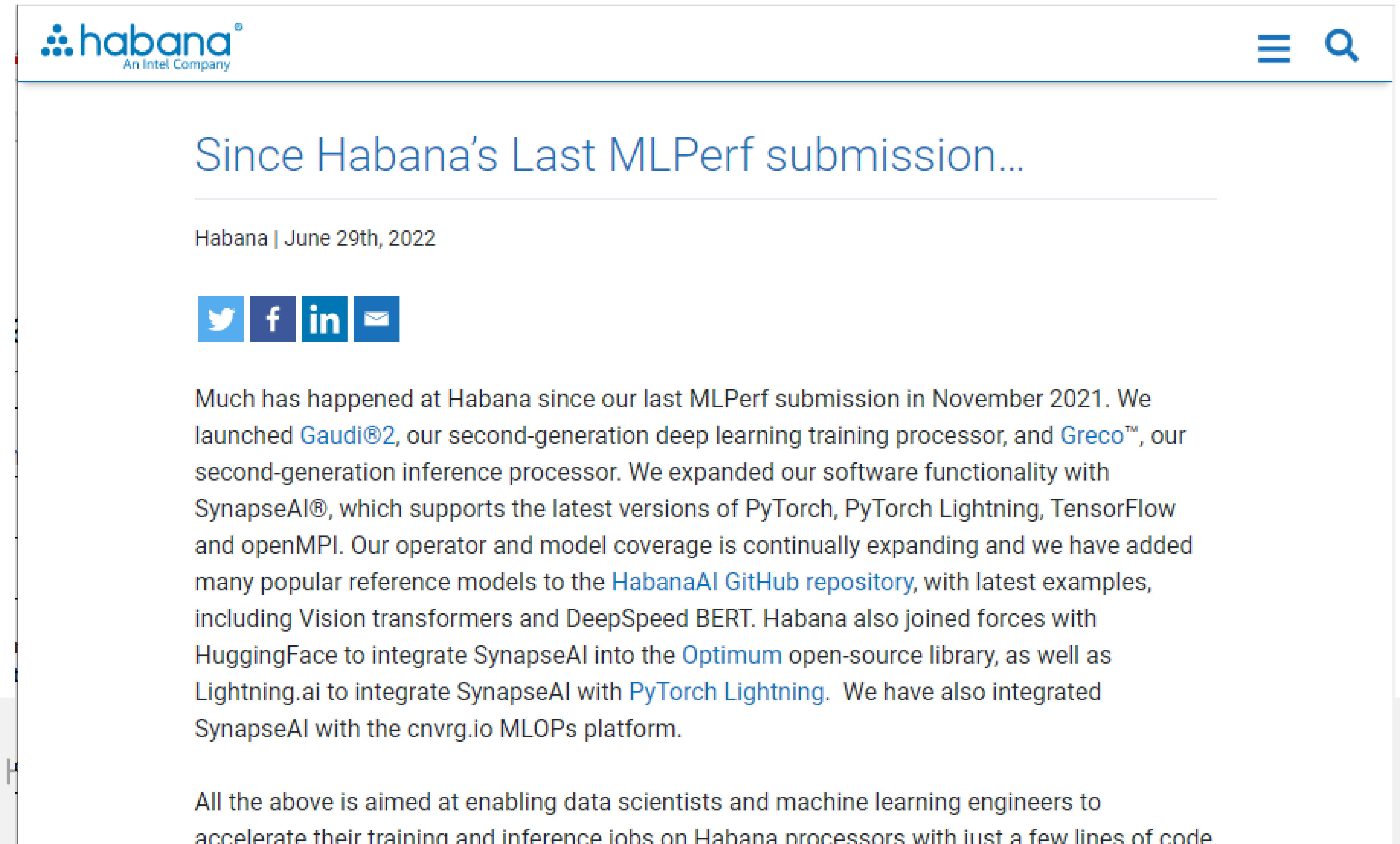
CONVERSATIONS

HotHardware Falcon Northwest Tiki...	221
Gaming Companies Stand Up And Speak...	37
AMD Ryzen Threadripper Pro 5000 CPUs...	8
Intel 13th Gen Core i9-13900 Raptor...	7
Apple's Base M2 MacBook Pro Laptop...	6
Monkey Island Creator Deserves Better...	4
Mars Express Orbiter Still Runs	4

FEATURED

FAVORITES

What Are You Playing Now?

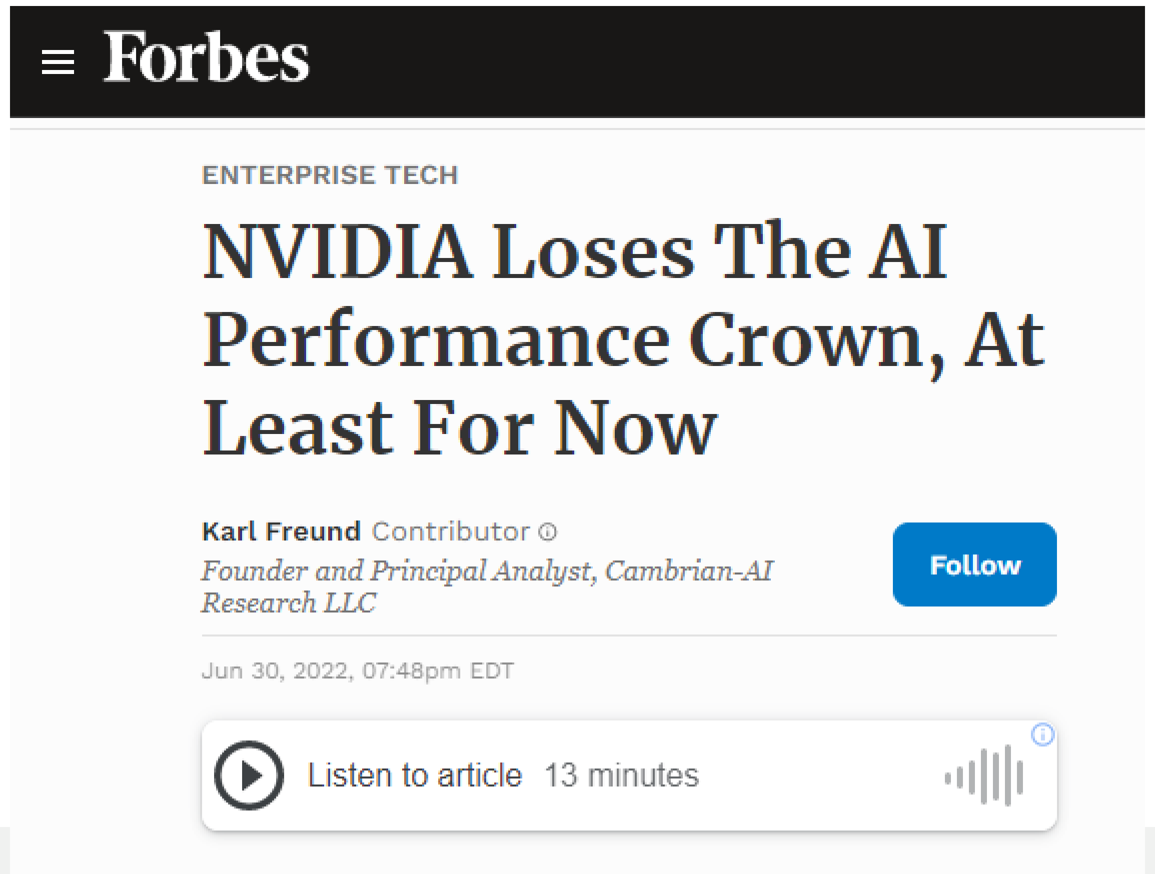


Since Habana's Last MLPerf submission…

Habana | June 29th, 2022

Much has happened at Habana since our last MLPerf submission in November 2021. We launched Gaudi®2, our second-generation deep learning training processor, and Greco™, our second-generation inference processor. We expanded our software functionality with SynapseAI®, which supports the latest versions of PyTorch, PyTorch Lightning, TensorFlow and openMPI. Our operator and model coverage is continually expanding and we have added many popular reference models to the HabanaAI GitHub repository, with latest examples, including Vision transformers and DeepSpeed BERT. Habana also joined forces with HuggingFace to integrate SynapseAI into the Optimum open-source library, as well as Lightning.ai to integrate SynapseAI with PyTorch Lightning. We have also integrated SynapseAI with the cnvrg.io MLOPs platform.

All the above is aimed at enabling data scientists and machine learning engineers to accelerate their training and inference jobs on Habana processors with just a few lines of code



ENTERPRISE TECH

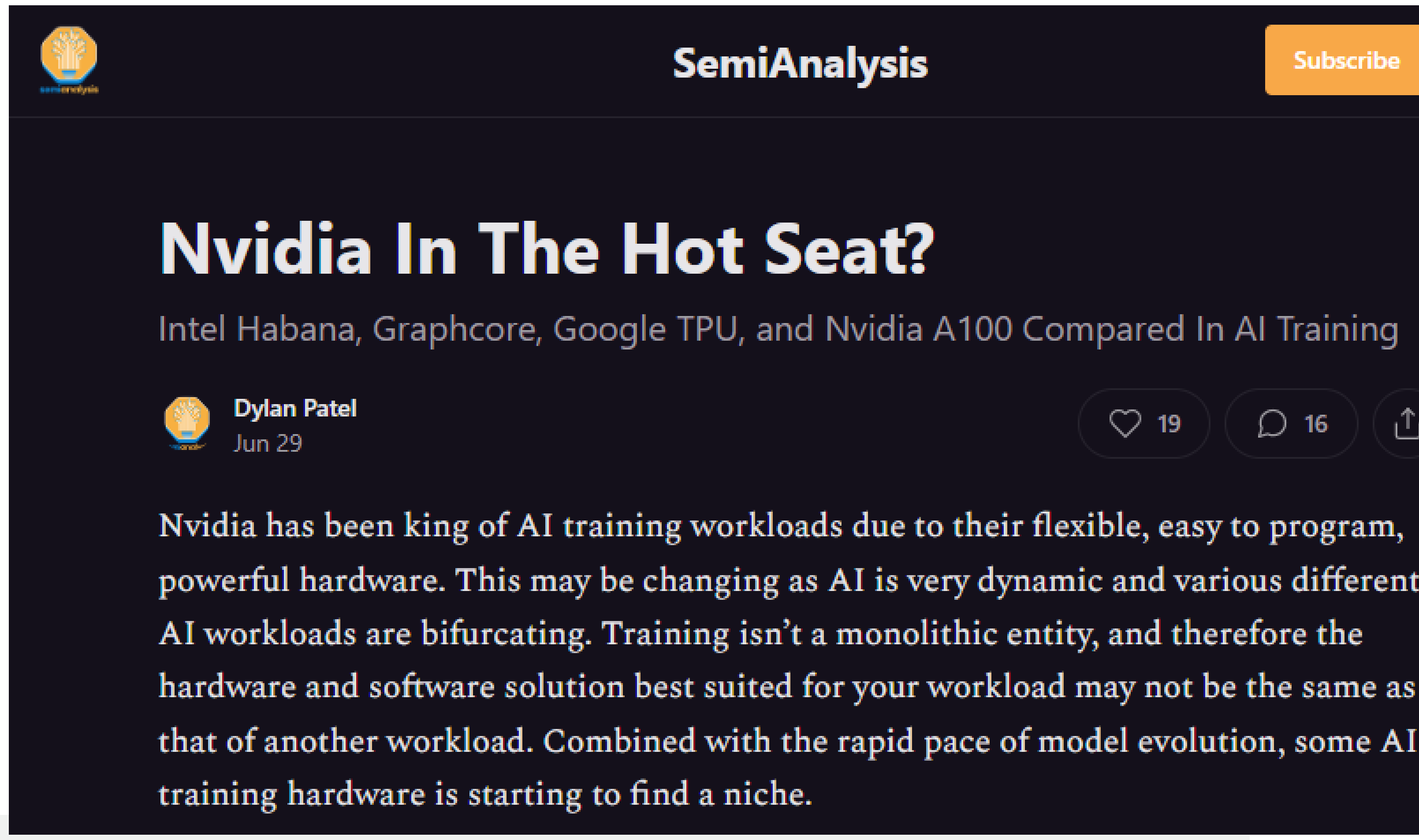
NVIDIA Loses The AI Performance Crown, At Least For Now

Karl Freund Contributor

Founder and Principal Analyst, Cambrian-AI Research LLC

Jun 30, 2022, 07:48pm EDT

Listen to article 13 minutes



Nvidia In The Hot Seat?

Intel Habana, Graphcore, Google TPU, and Nvidia A100 Compared In AI Training

Dylan Patel
Jun 29

Nvidia has been king of AI training workloads due to their flexible, easy to program, powerful hardware. This may be changing as AI is very dynamic and various different AI workloads are bifurcating. Training isn't a monolithic entity, and therefore the hardware and software solution best suited for your workload may not be the same as that of another workload. Combined with the rapid pace of model evolution, some AI training hardware is starting to find a niche.

GRECO™

Deep Learning
Inference Efficiency

Greco: Second-Generation Inference for Deep Learning

GOYA™

GRECO™

Node

16nm

7nm

Memory

40GB/s DDR4

16GB

50MB

204GB/s LPDDR5

16GB

128MB on chip SRAM

Compute

BF16, FP16, INT4

Media decode and processing

Form Factor

Dual-slot PCIe >

Single-slot HHHL

TDP

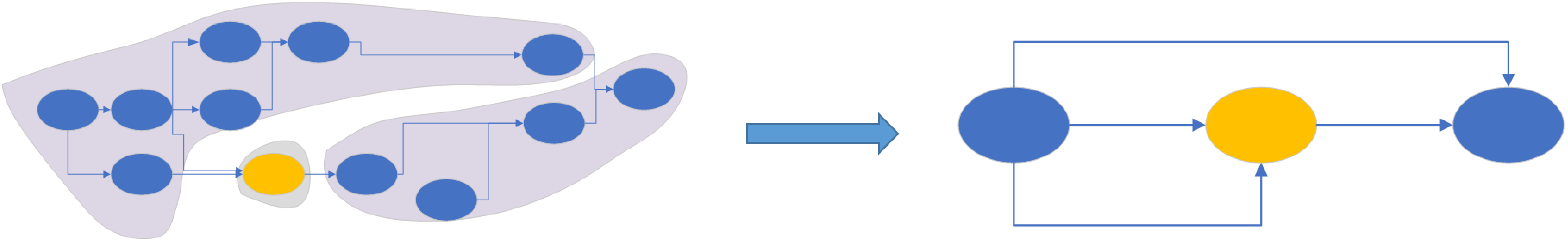
200w

75w

habana®

An Intel Company

TensorFlow integration with SynapseAI



SynapseAI receives a computational graph of the model from the framework

It identifies subgraphs (blue nodes) that can be accelerated by Gaudi

The rest of the graph runs on CPU (yellow node)

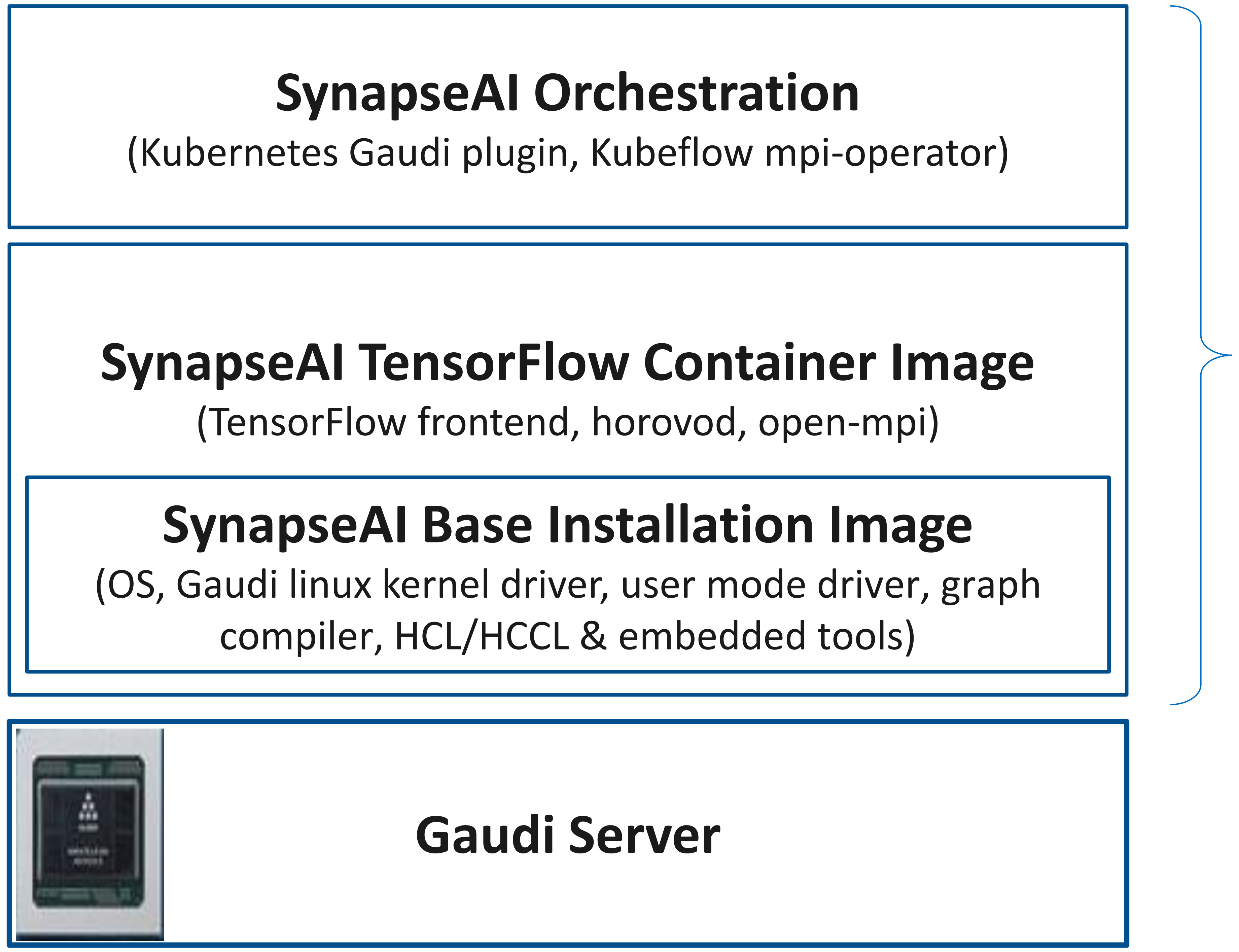
The original graph is modified to replace the Gaudi subgraphs with encapsulated nodes (blue)

The framework runtime executes the modified graph

For each encapsulated node, SynapseAI generates optimized binary code that runs on Gaudi

Software Installation and Deployment

Setup and Install repository on Habana GitHub provides instructions on how to setup your environment with the SynapseAI software stack



SynapseAI Orchestration
(Kubernetes Gaudi plugin, Kubeflow mpi-operator)

SynapseAI TensorFlow Container Image
(TensorFlow frontend, horovod, open-mpi)

SynapseAI Base Installation Image
(OS, Gaudi linux kernel driver, user mode driver, graph compiler, HCL/HCCL & embedded tools)

Gaudi Server

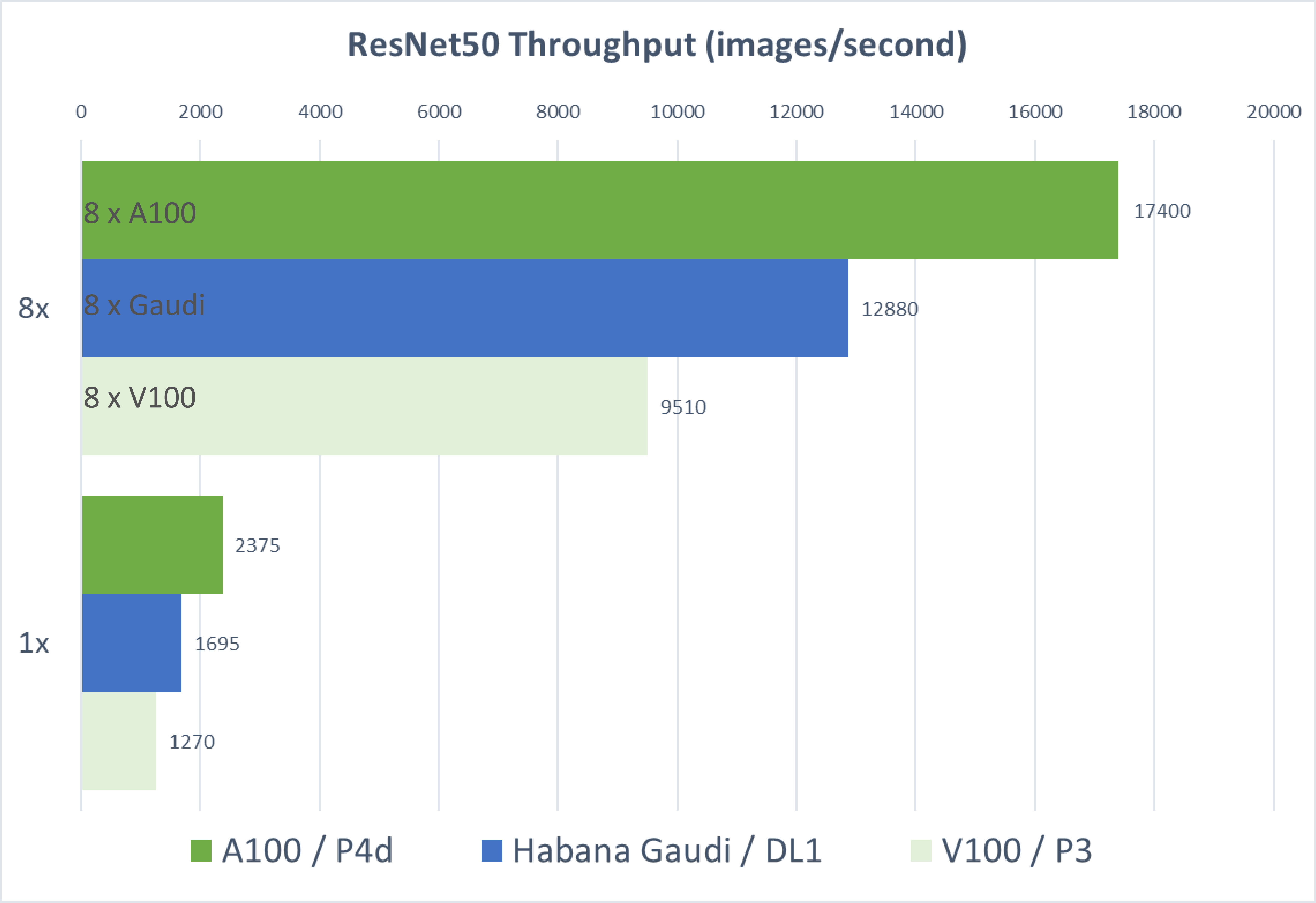
Gaudi-optimized Docker container images with all necessary dependences*

Official releases publicly available on Habana Vault

Orchestration	Kubernetes (1.19)
Frameworks	TensorFlow2 and PyTorch
Operating Systems	Ubuntu 18.04 and 20.04
Container Runtimes	Docker (Docker CE version 18.09)
Distributed Training Schemes	TensorFlow with Horovod and tf.distribute PyTorch distributed (native)

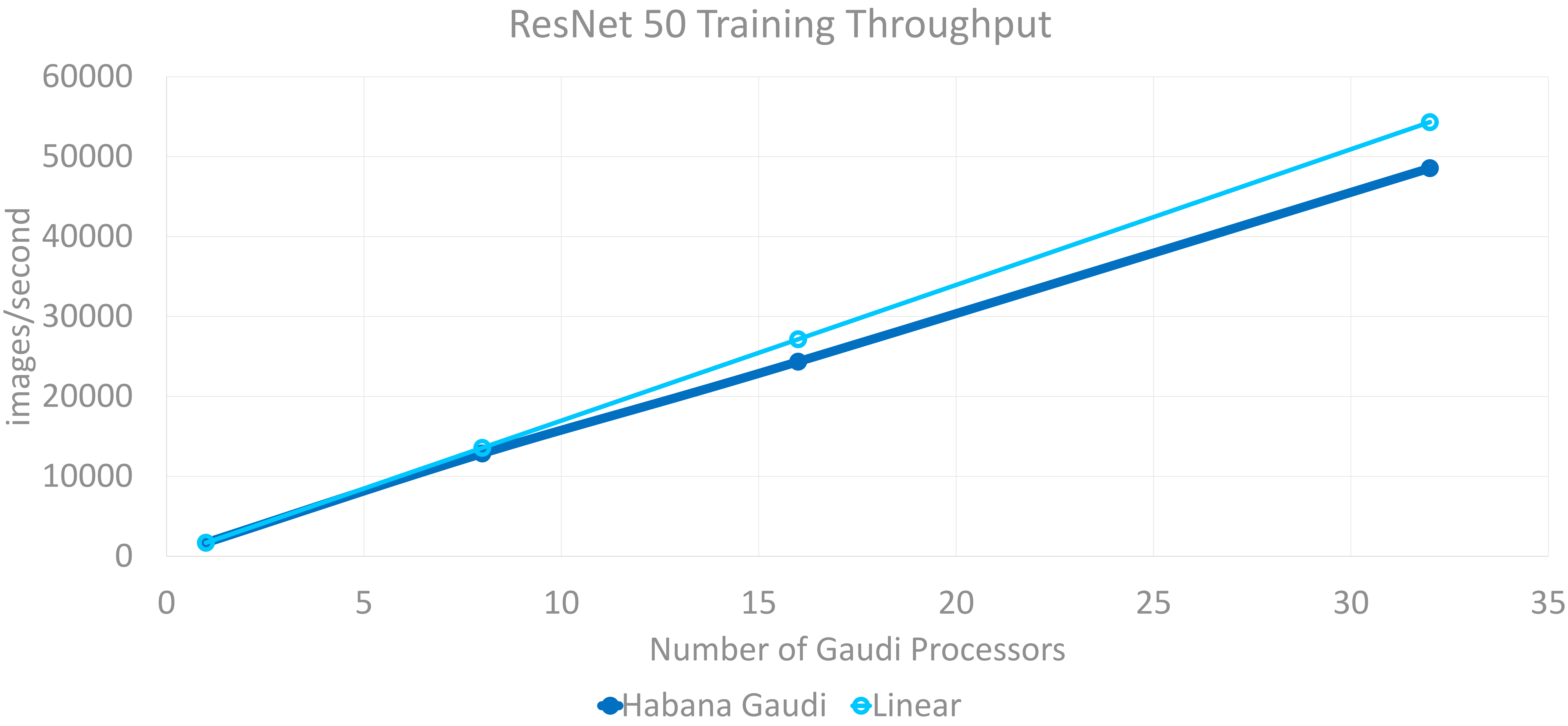
** Habana GitHub will have repository with Dockerfiles to "build your own" Docker images*

DL1 Vision Model Training Performance

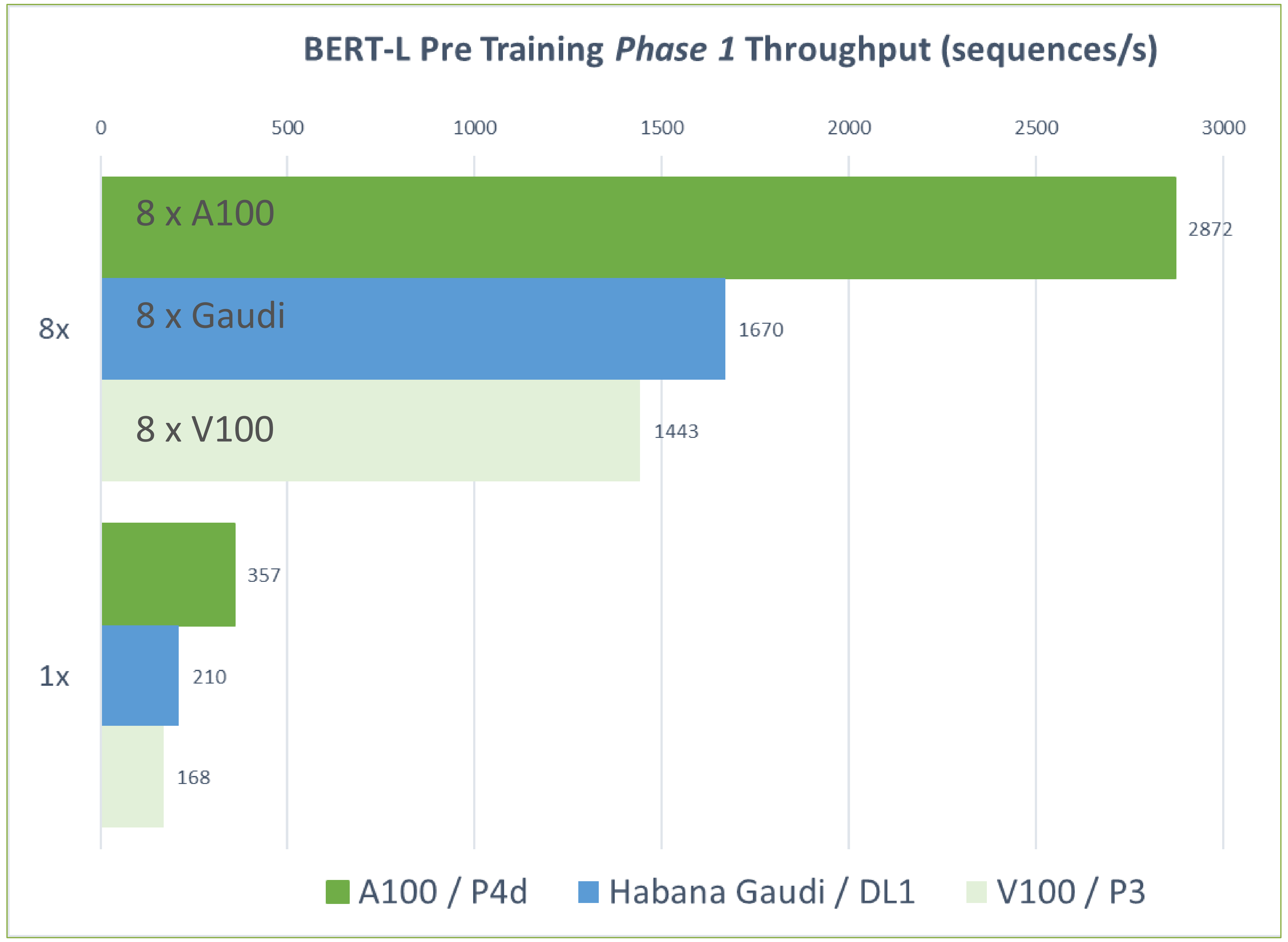


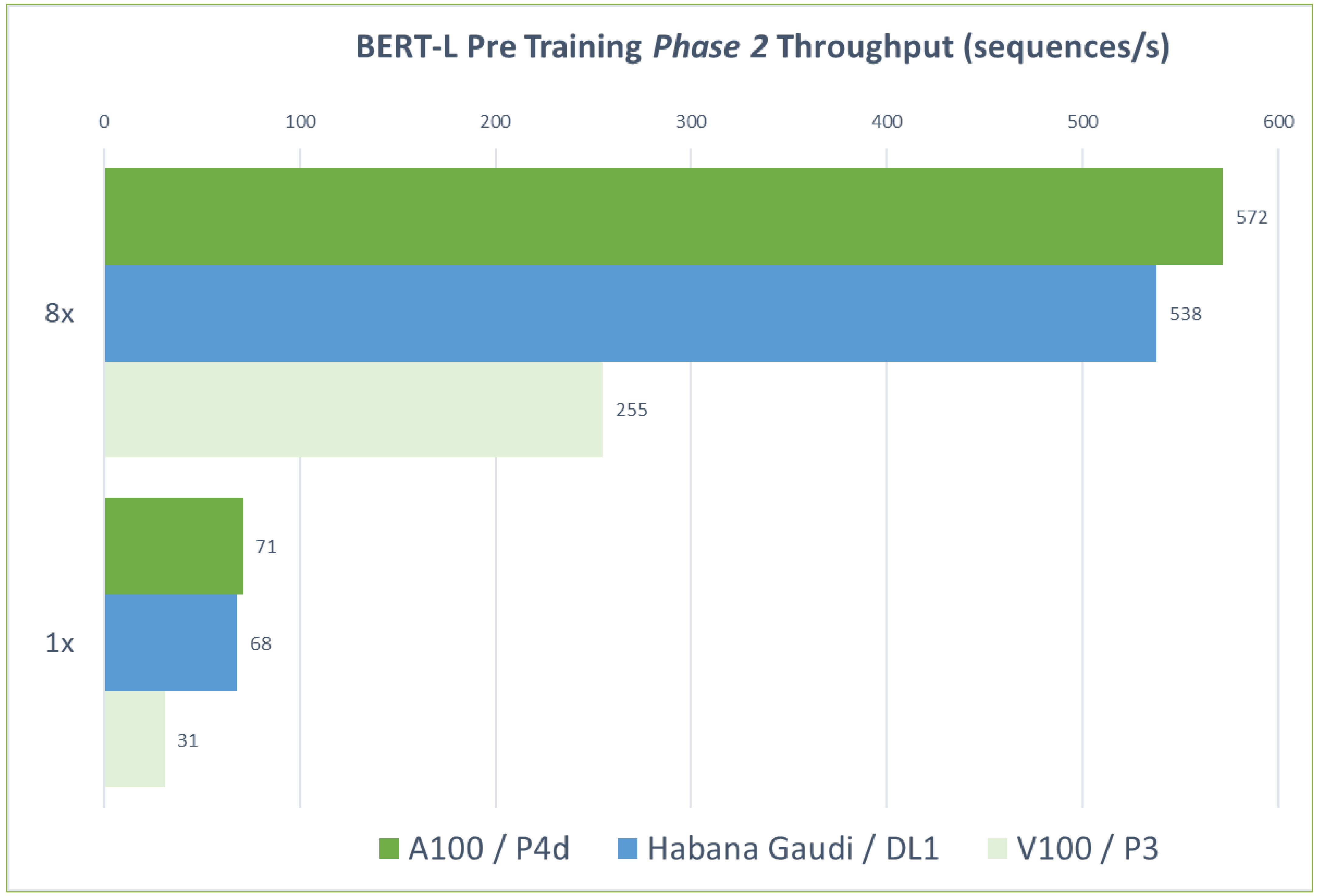
Habana ResNet50 Model: https://github.com/HabanaAI/Model-References/tree/master/TensorFlow/computer_vision/Resnets/resnet_keras
Habana SynapseAI Container: https://vault.habana.ai/ui/repos/tree/General/gaudi-docker/1.2.0/ubuntu20.04/habanalabs/tensorflow-installer-tf-cpu-2.7.0
Habana Gaudi Performance: https://developer.habana.ai/resources/habana-training-models/
A100 / V100 Performance Source: https://ngc.nvidia.com/catalog/resources/nvidia:resnet_50_v1_5_for_tensorflow/performance, results published for DGX A100-40G and DGX V100-32G

Vision Model Training Scalability



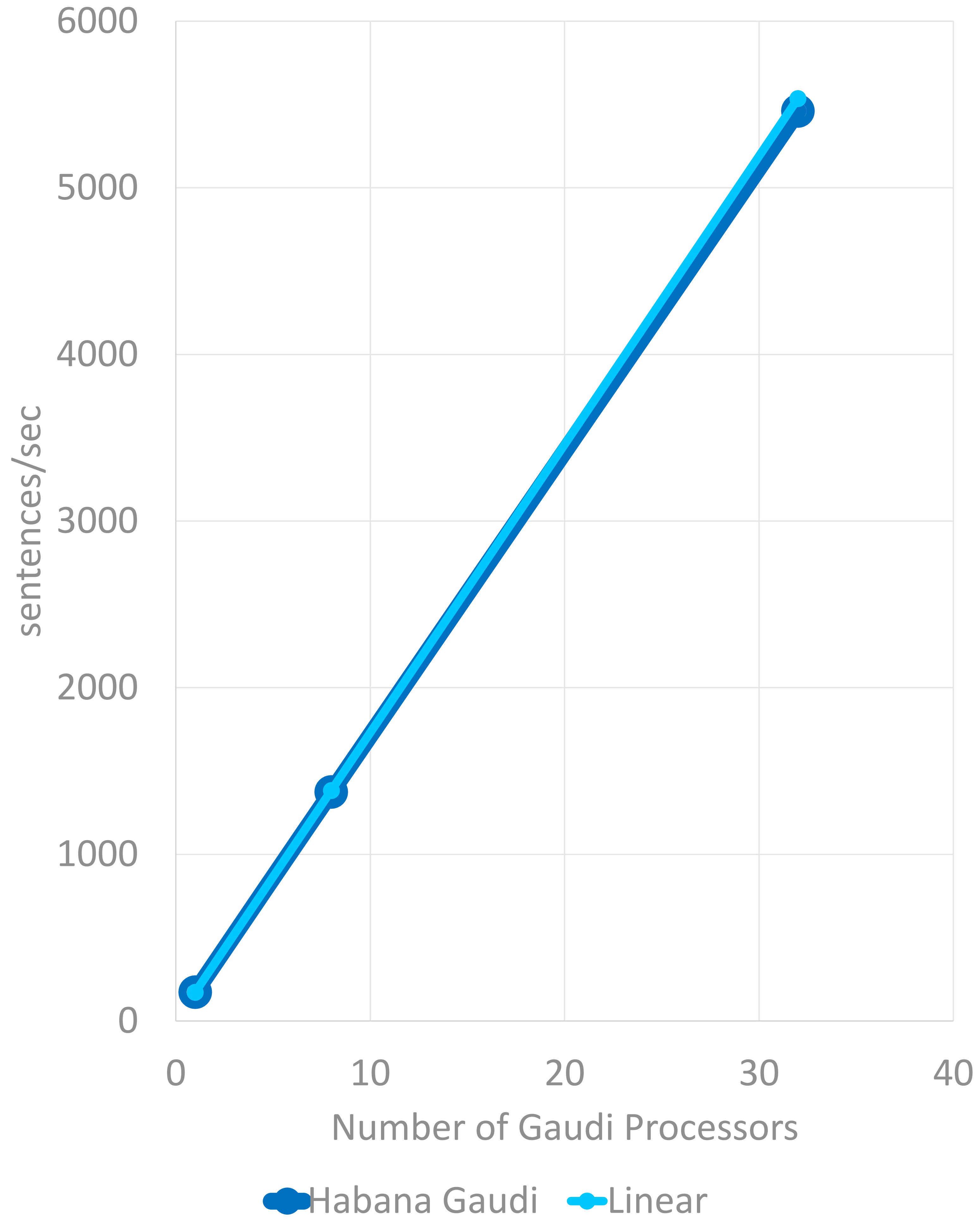
DL1 NLP Model Training Performance



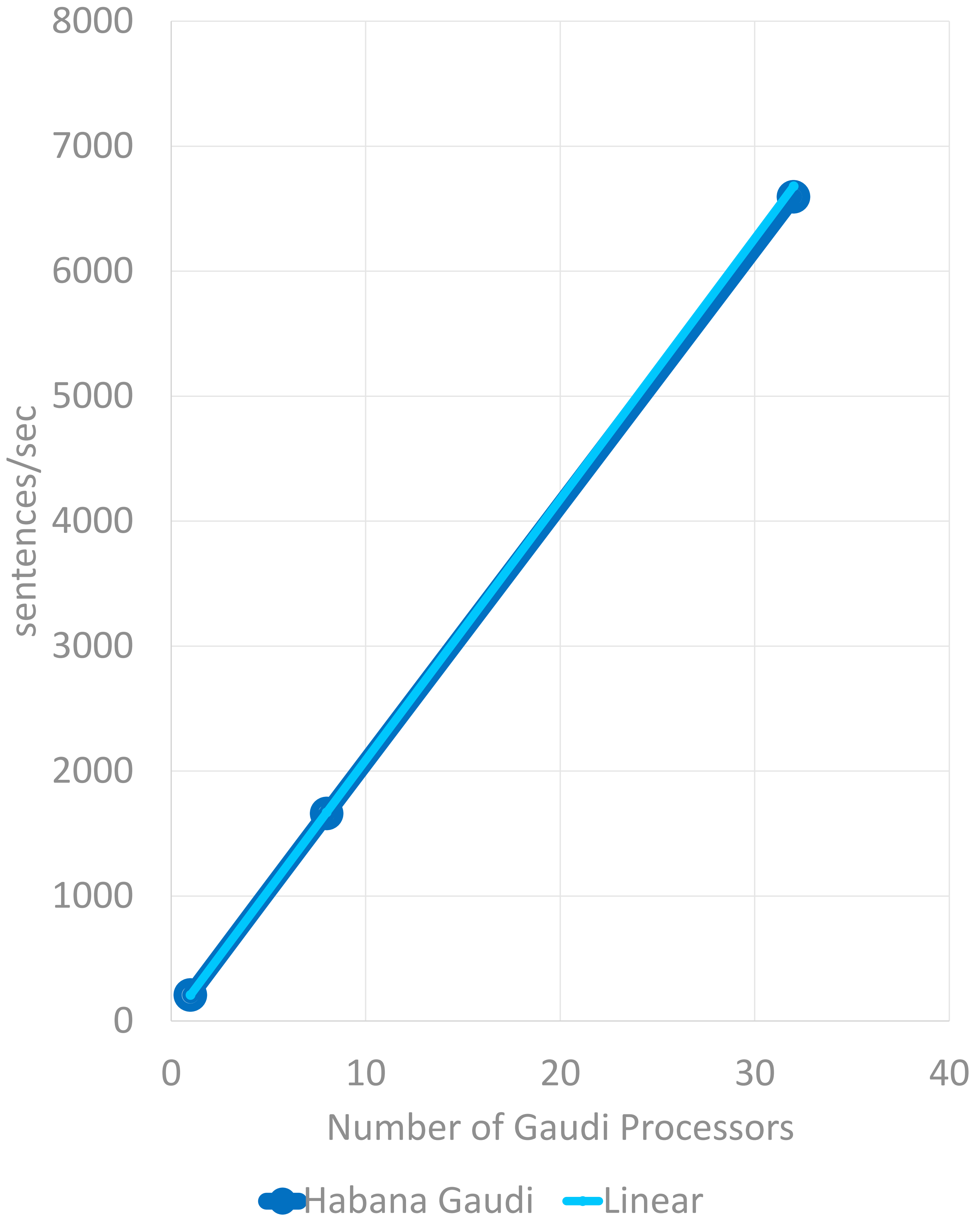


Habana BERT-Large Model: https://github.com/HabanaAI/Model-References/tree/master/TensorFlow/nlp/bert
Habana SynapseAI Container: https://vault.habana.ai/ui/repos/tree/General/gaudi-docker/1.2.0/ubuntu20.04/habanalabs/tensorflow-installer-tf-cpu-2.7.0
Habana Gaudi Performance: https://developer.habana.ai/resources/habana-training-models/
A100 / V100 Performance Sources: https://ngc.nvidia.com/catalog/resources/nvidia:bert_for_tensorflow/performance, results published for DGX A100-40G and DGX V100-32G

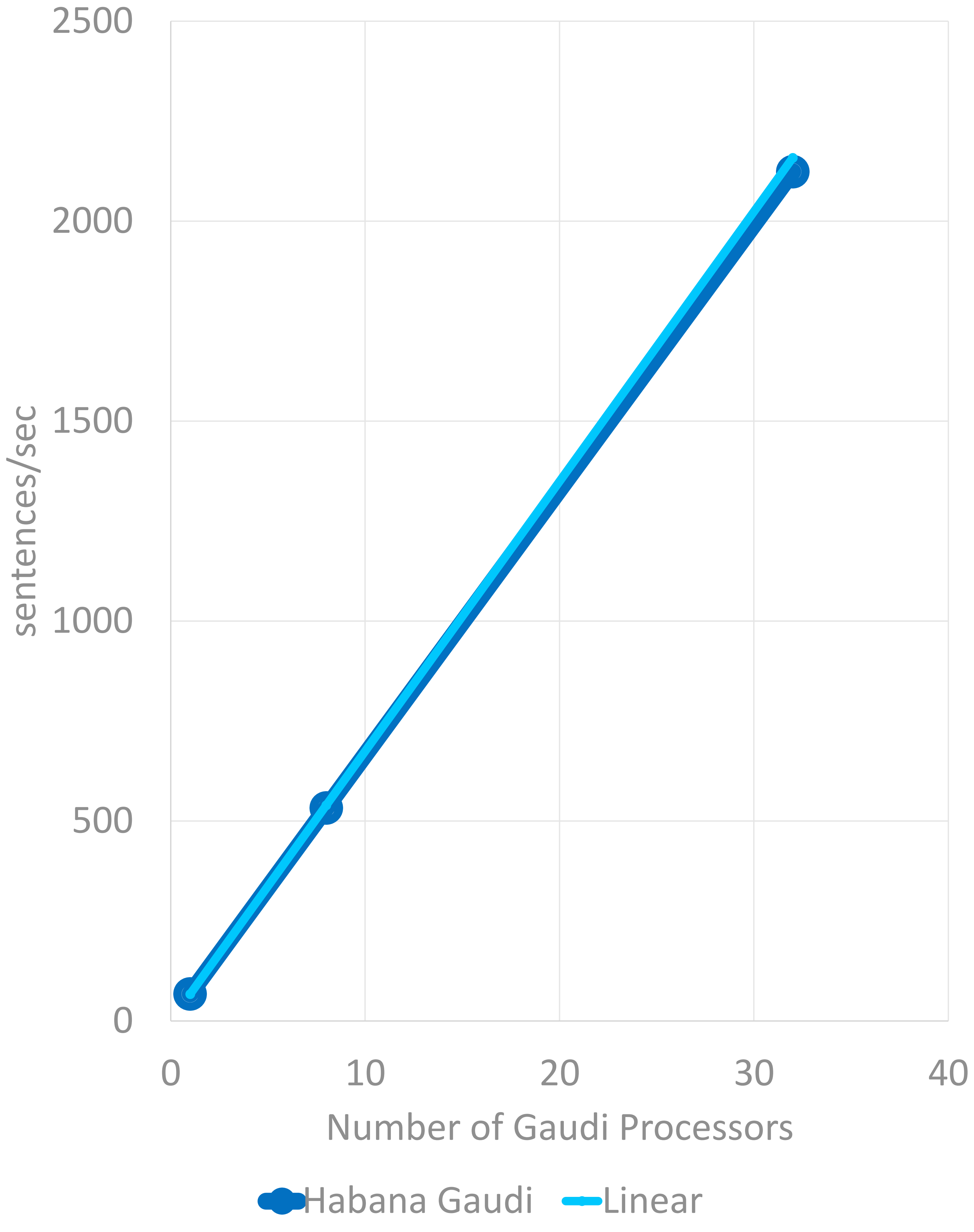
NLP Model Training Scalability



BERT Large Pretraining Throughput (Phase 1 & Phase 2)



BERT Large Finetuning Throughput (Phase 1)



BERT Large Pretraining Throughput (Phase 2)