Overview of Voyager - An Innovative AI Computing Resource for Deep Learning Applications in Science and Engineering

VOYAGER

Advancing AI for Science & Engineering

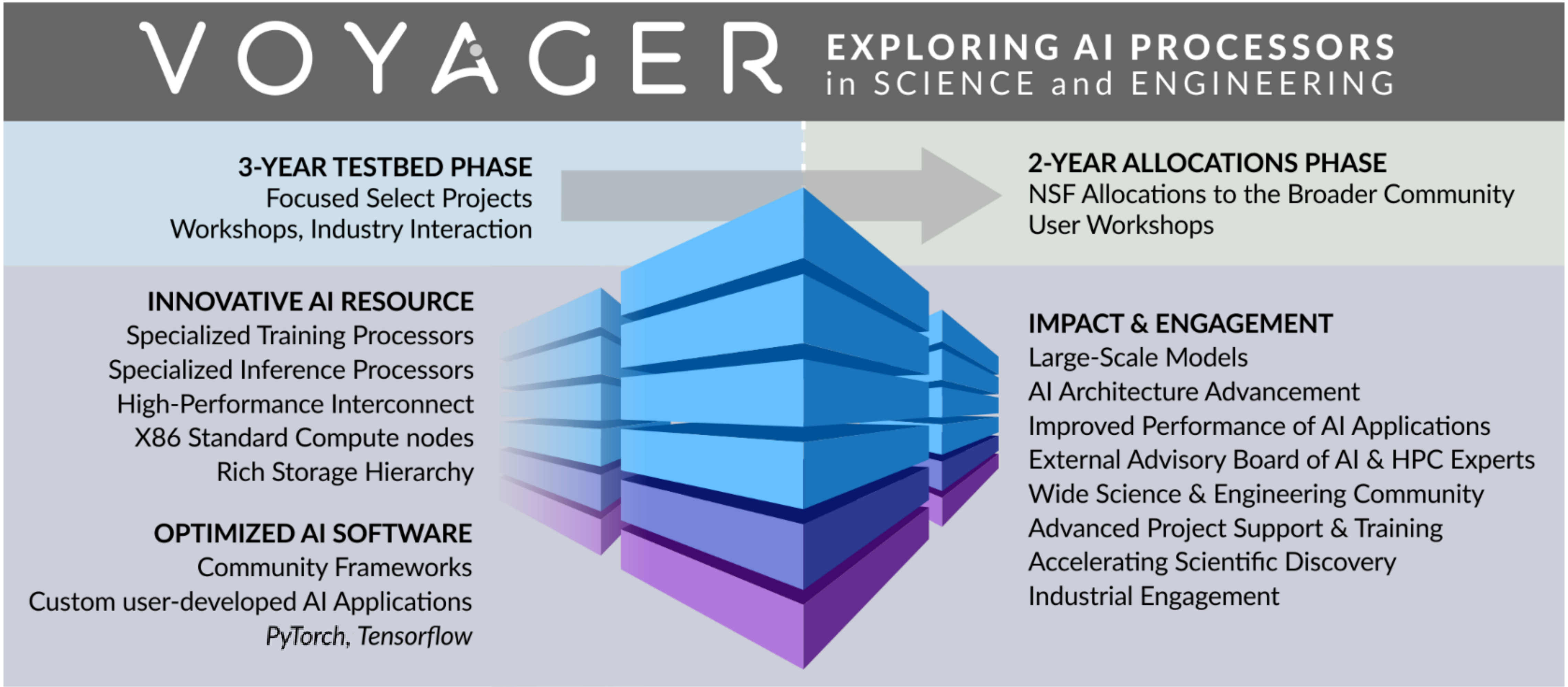
Amit Majumdar, San Diego Supercomputer Center (SDSC)
University of California San Diego (UCSD)
HPC and Data Science Summer Institute 2025

Contributions from many others at SDSC and UCSD
Contributions from Intel/Habana
Contributions from Supermicro and Arista

Voyager Motivation and Design

- The 2019 update to the National Artificial Intelligence Research and Development Strategic Plan recognized the need for research into new AI hardware, “While AI research is most commonly associated with advances in software, the performance of AI systems has been heavily dependent on the hardware upon which it runs. The current renaissance in deep machine learning is directly tied to progress in GPU-based hardware technology and its improved memory, input/output, clock speeds, parallelism, and energy efficiency. **Developing hardware optimized for AI algorithms will enable even higher levels of performance than GPUs**. …..”

- We deployed *Voyager* with vendor partners, Intel/Habana, Supermicro and Arista
- Focused on Deep Learning AI applications across a wide range of science and engineering domains
- Gaudi training nodes (Gaudi1 and Gaudi2 on Voyager; and Gaudi3 available soon on the cloud)
  - architected for performance and efficiency
- Gaudi servers support all-to-all connectivity
- First generation inference nodes and Gaudi training nodes both used for inference
- Three distinct networks support application performance (Arista 400 GbE switch for scale out training), data movement (bonded 50GbE (2 X 25GbE)) and system management (1 GbE out-of-band management network)
- Users implement and deploy AI applications via abstraction of PyTorch (in earlier years also TensorFlow)
- SynapseAI software stack optimizes performance on architecture transparent to users
- Various models optimized on Gaudi1 and Gaudi2 (and Gaudi3) by Habana
- Overall designed philosophy - easy and flexible model migration by users and transparently achieve performance



Category II System, NSF Award # 2005369

PI: Amit Majumdar (SDSC); Co PIs: Rommie Amaro (UCSD), Javier Duarte (UCSD), Mai Nguyen (SDSC), Robert Sinkovits (SDSC)

Voyager is a heterogeneous system designed to support complex AI workflows

- **42x Intel Habana Gaudi** training nodes, each with 8 training processors (**336 in total**); all-to-all network between processors on a node
- Gaudi processors feature specialized hardware units for AI, HBM2, and on-chip high-speed Ethernet
- **2x first generation inference nodes**, each with 8 inference processors (**16 in total**)
- **36x Intel x86 processors compute nodes** for general purpose computing and data processing
- **400 GbE interconnect** using RDMA over Converged Ethernet
- **3 PB Storage** system connected vis 25GbE. Deployed as Ceph; experiment with other filesystems
- 324 TB HFS; connectivity to compute via 25GbE
- Machine integrated by Supermicro; includes Arista switch

- **<u>2 Intel Habana Gaudi2 nodes – loaner nodes received from Intel/Supermicro in August 2024</u>**
- **<u>About to receive access to a Gaudi3 node also</u>**

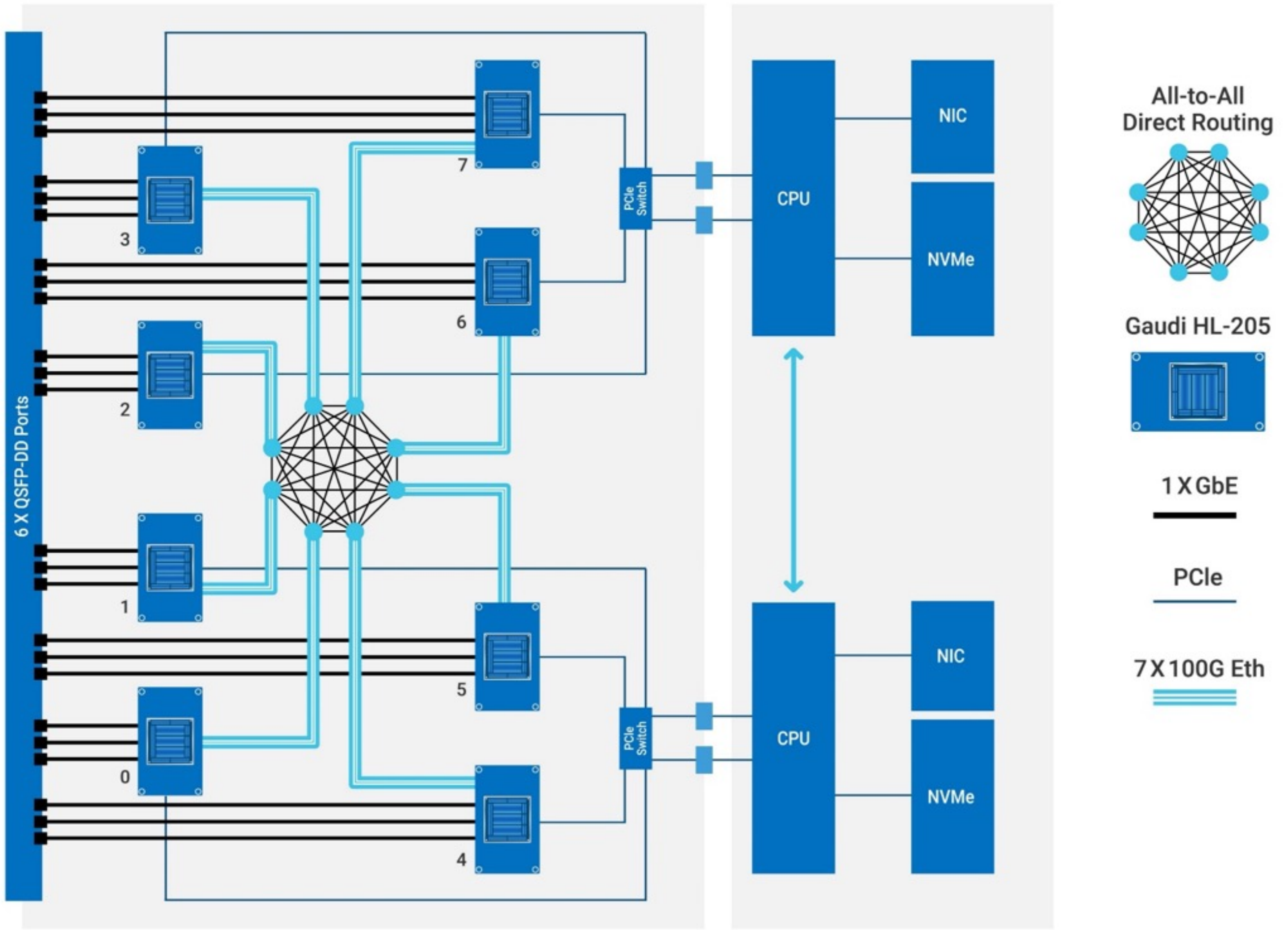
System Component	*Configuration*
INTEL GAUDI TRAINING NODES	
Node count	42
Training processors/node	8
Host x86 processors/node	2
Memory/node	512 GB DDR4
Memory/training processor	32 GB HBM2
Local NVMe	6.4 TB
INTEL GOYA INFERENCE NODES	
Node count	2
Inference processors/node	8
Host x86 processors/node	2
Memory/node	512 GB DDR4
Memory/inference processor	16 GB DDR4
Local NVMe	3.2 TB
STANDARD COMPUTE NODES	
Node count	36
x86 processors/node	2
Memory capacity	384 GB
Local NVMe	3.2 TB
STORAGE SYSTEM	
High performance storage: HDD:NVMe	3 PB:140 TB
High performance filesystems	Ceph, Lustre
Home filesystem storage: HDD:NVMe	324 TB: 12.4 TB
File system	NFS

Designed for flexible and easy model migration

Ease of use	*Customization*	*Balanced compute & memory*
Integrated with PyTorch; minimal code changes to get started *➔ SynapseAI maps model topology onto Gaudi devices*	*SynapseAI TPC SDK facilitates development of custom kernels*	*32GB HBM2 memories similar to GPUs, so existing DL models will fit into Gaudi memory (96 GB HBM2 in Gaudi2)*
*Developers can enjoy the **same abstraction** they are accustomed to today*	*Developers can **customize** models to extract best performance*	*Developers can spend **less effort** to port their models to Gaudi*

Gaudi servers supports all-to-all connectivity

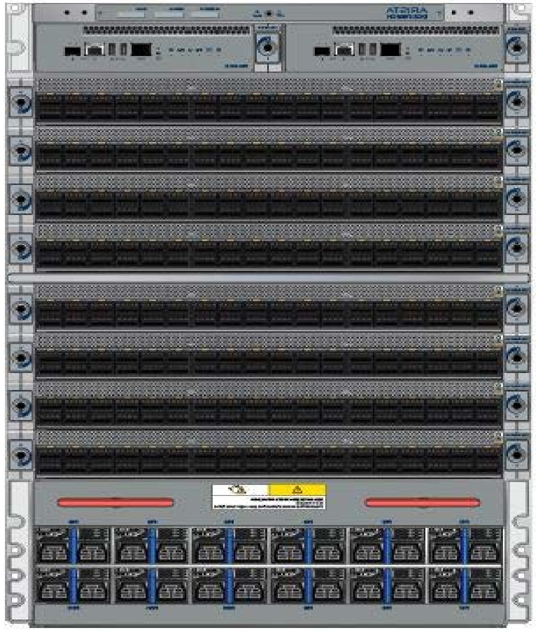
- 8 Gaudi OCP OAM cards
- 24 x 100GbE RDMA RoCE for scale-out
- Non-blocking, all-2-all internal interconnect across Gaudi AI processors
- Separate PCIe ports for external Host CPU traffic



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

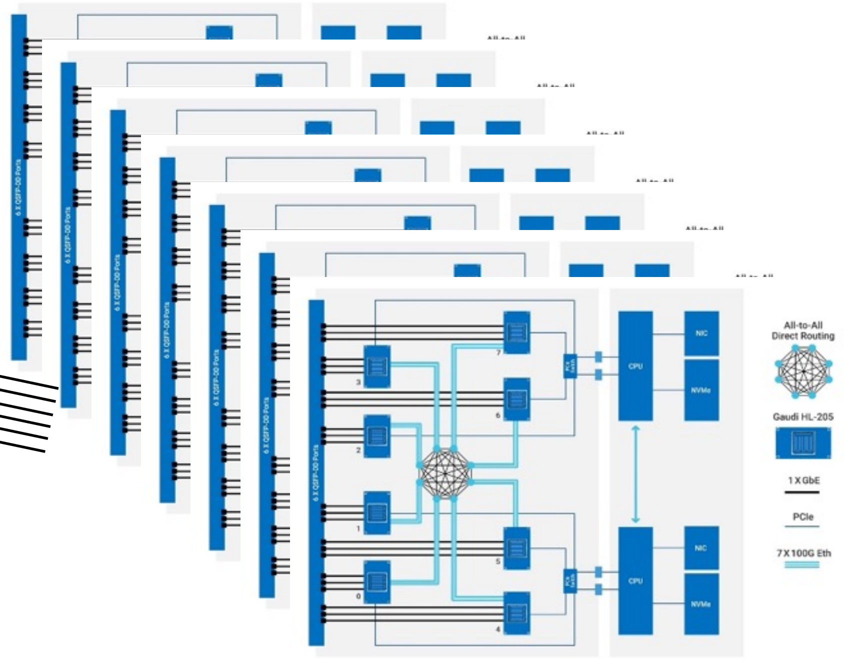
Gaudi design enables highly efficient scaling

- *Natively integrated RoCE on Gaudi processor*
- *6x Quad-100 GbE per node (8x Gaudi)*
- *7808 Arista 400 GbE switch*



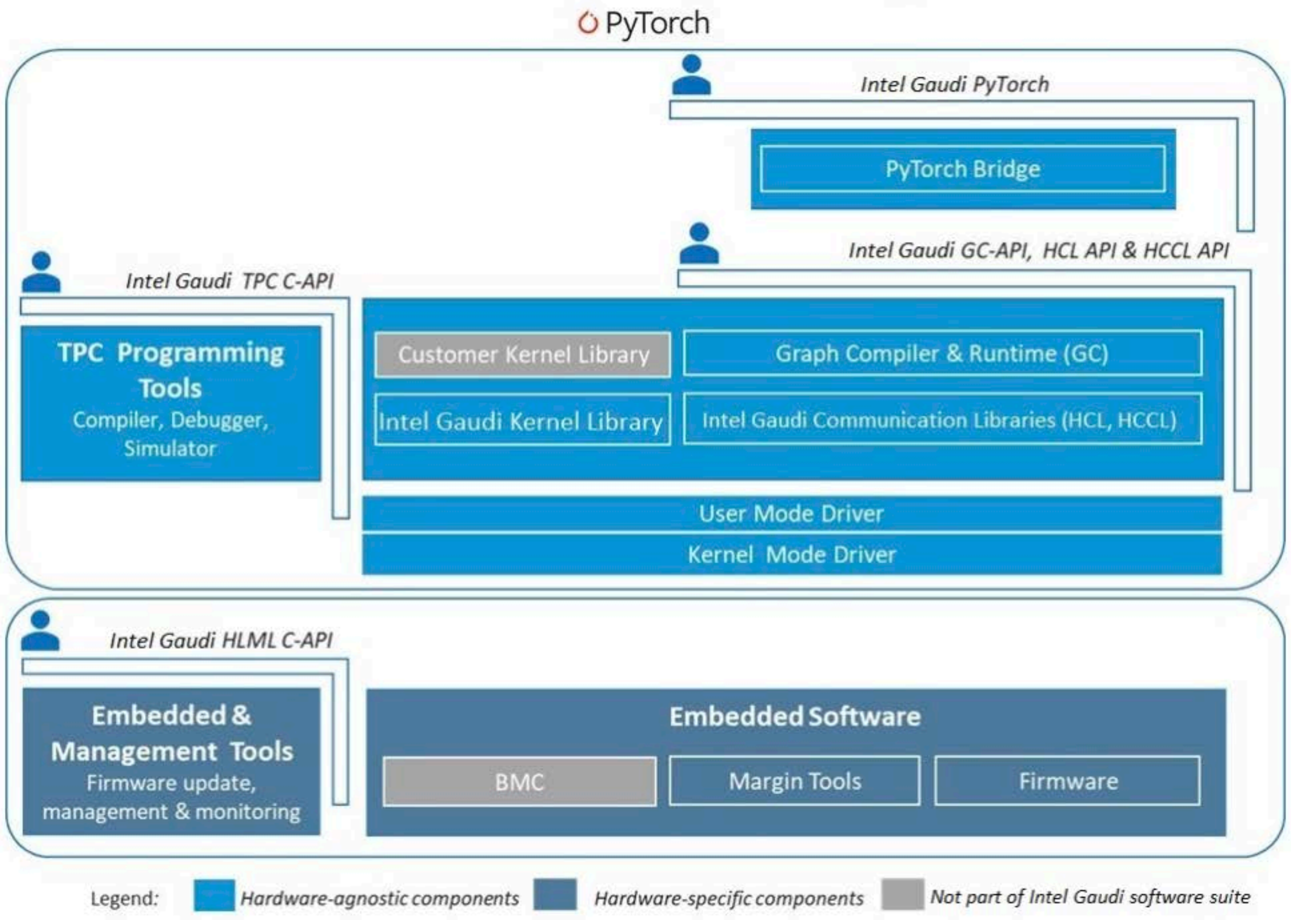
7808 Arista 400GbE

6x 400GbE per Gaudi Node



6x Gaudi nodes per rack

Intel Gaudi Software Suite for High Performance DL Training/Inference



Intel Gaudi Software Suite

- Shared software suite for training and inference
- Start running on Habana accelerators with minimal code changes
- Integrated with PyTorch. TensorFlow supported on older Synapse versions.
- Rich library of performance-optimized kernels
- Advanced users can write their custom kernels
- Components include graph compiler and runtime, TPC kernel library, firmware and drivers, and developer tools

Reference: https://docs.habana.ai/en/latest/Gaudi_Overview/Intel_Gaudi_Software_Suite.html

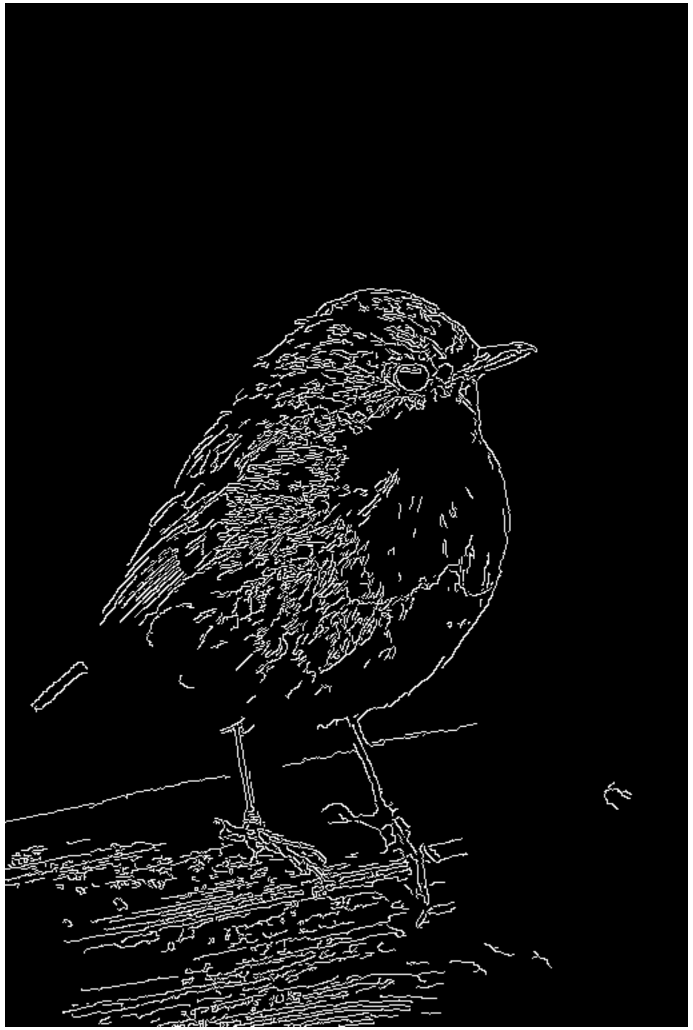
Enabling the community to utilize the Habana AI focused architecture path

- Gaudi1
  - 16nm process
  - 8 TPC, single MME
  - 32 GB HBM memory
  - 10 x 100 Gigabit Ethernet for multi-chip scale-out training
- Gaudi2
  - 7nm process
  - 24 TPC, 2 MME
  - 96 GB HBM2E
  - 24 X 100 Gigabit Ethernet for multi-chip scale-out training
- Gaudi3
  - 5nm process
  - 64 TPC, 8 MME
  - 128 GB HBM2E memory
  - 24 X 200 Gigabit Ethernet for multi-chip scale-out training

- **Such architecture path has tremendous impact on AI capabilities**
- **User community can see the impact of generation of Habana AI focused hardware on their AI applications**

Stable diffusion: depth diffuser model controlnet_sd21

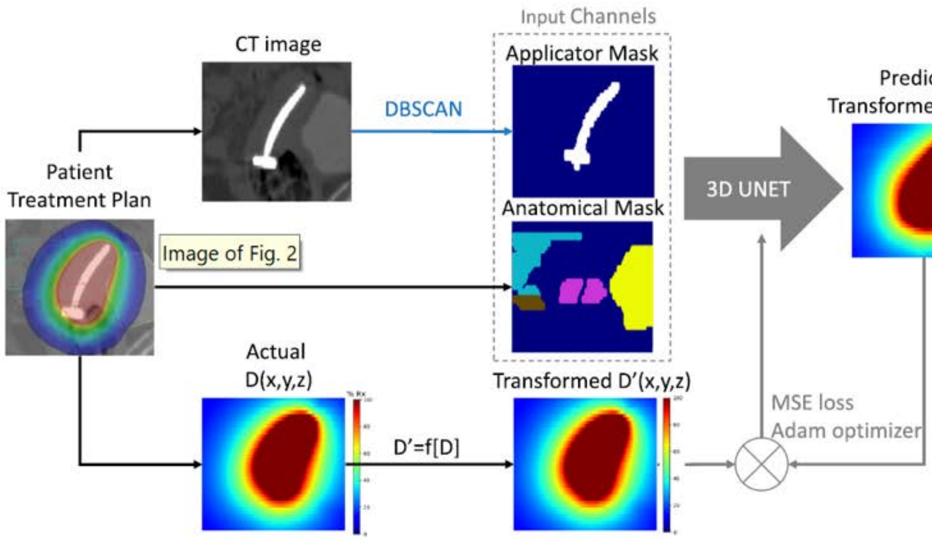
(generated on *Voyager* Gaudi)



3D-UNET Model for Dose Prediction in Cervical Brachytherapy

Sandra Meyers PI, and Lance Moore,
Department of Radiation Medicine & Applied Sciences, UC San Diego Health

- Cervical cancer is treated with radiation, but it is very difficult to predict in 3D how dosage from applicators will impact other tissue.
- PI's team developed a workflow with inputs of possible dosage plan, segmented tissue and applicator mask, and applied a series of DL vision models

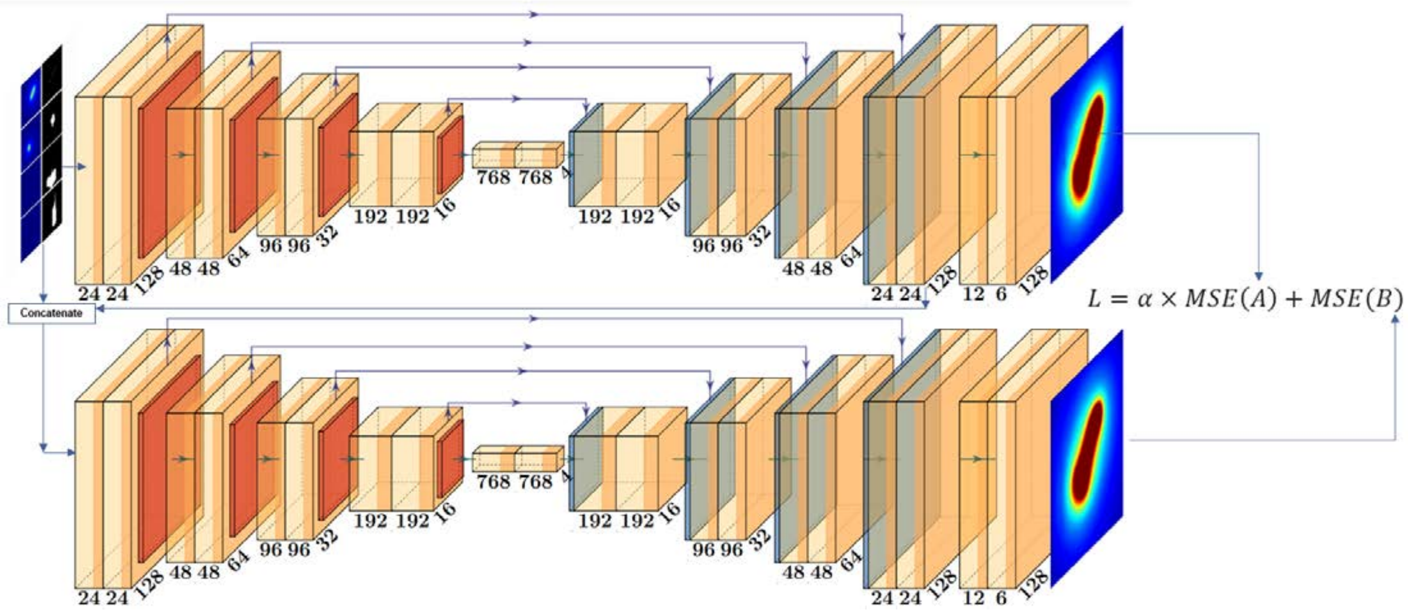


Reference: Cortes KG, Kallis K, Simon A, Mayadev J, Meyers SM, Moore KL. Knowledge-based three-dimensional dose prediction for tandem-and-ovoid brachytherapy. Brachytherapy. 2022 Jul-Aug;21(4):532-542.

The workflow of data transformations and input preparation for the UNET model

3D-UNET Model for Dose Prediction

- PIs developed UNET model for a single GPU device execution in PyTorch , we helped port and parallelize the model
- PIs assembled largest dataset for this task, where each input is multiple 3D images (128x128x128) images



Publications

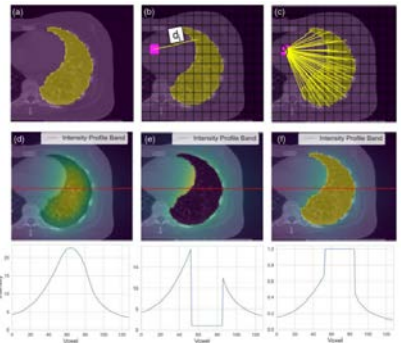
- Several papers have been published:

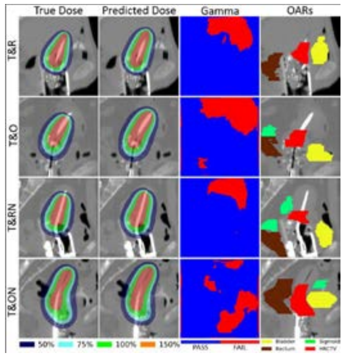
Comparing models [1]

Investigating transfer learning to external clinic data[2]

Novel loss and masking algorithm,

and assessing data scarcity [3]





1. *Moore, L.C., et al. Rapid Auto-Planning for Cervical Brachytherapy: International Conference on the use of Computers in Radiation Therapy,*
2. *Moore, L.C., et al. Improving Dose Prediction Model Performance on an External Clinic's Data. 66th Annual Meeting of the American Association of Physicists in Medicine,*
3. *Moore L.C., et al. Neural network dose prediction for cervical brachytherapy: Overcoming Data Scarcity for Applicator-Specific Models. Medical Physics.2024*

Cardiac Image Analysis

Mai H. Nguyen, SDSC, Voyager Co-PI

Motivation: Cardiac imaging allows visualization of heart structure and function non-invasively

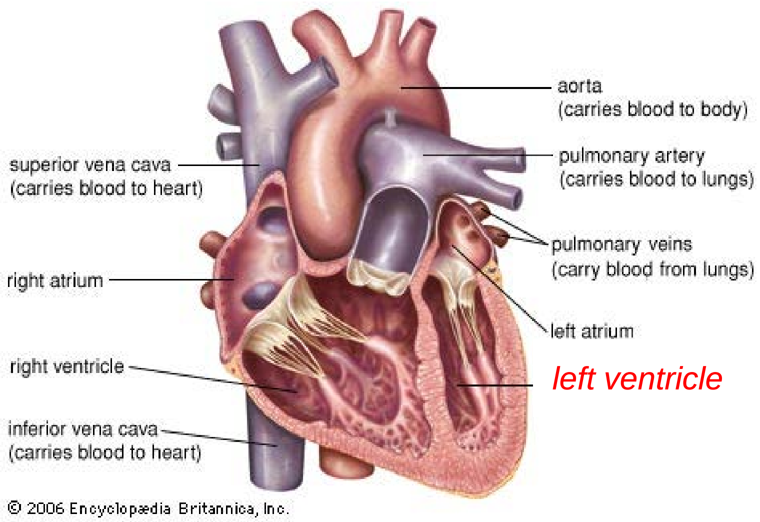
Goal: Develop deep learning approach to provide automated, efficient, and consistent segmentation of cardiac structures from MRI

Approach: U-Net model used to segment left ventricle from cardiac MRI

Collaborators:
- *M. H. Nguyen, G. W. Cottrell, I. Altintas (UC San Diego)*

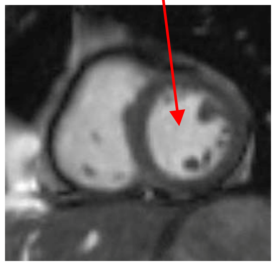
Students:
- *1 undergraduate*

References:
- *IEEE International Conference on Big Data*
- *arxiv:1909.08028*
- *IEEE 14th International Conference on e-Science*



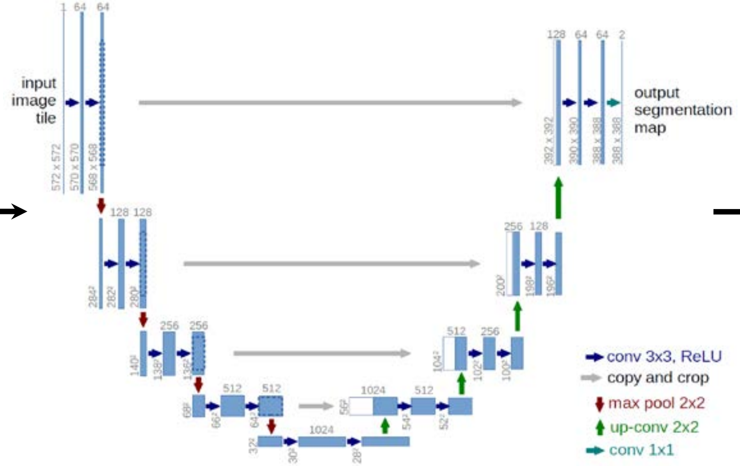
cardiac structures

left ventricle



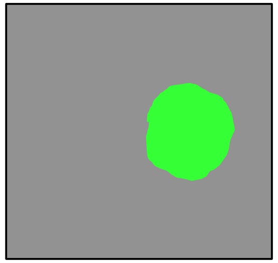
Cardiac MRI Frame

→



https://lmb.informatik.uni-freiburg.de/people/ronneber/u-net/

→



Segmented Left Ventricle Endocardium

Model identifies and segments left ventricle from cardiac MRI

Cardiac Image Analysis - Porting

- Code implemented in TensorFlow/Keras
- Porting to Gaudi for training
  - Straightforward process
  - Followed Habana guidelines
- Porting to Goya for inference
  - More extensive setup and code additions were needed compared to porting to Gaudi
  - Required more assistance from Habana team

Requirements for fine tuning an LLM

Madhu Gujral, SDSC

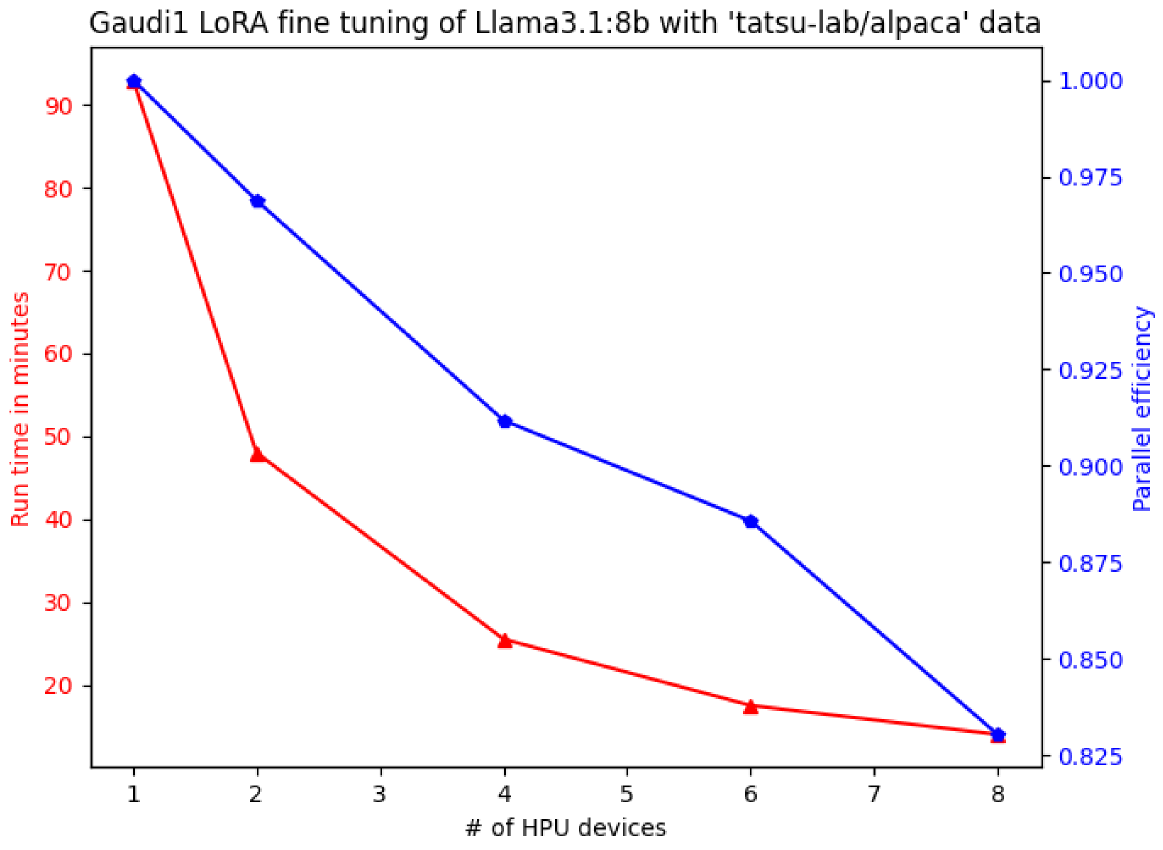
- *Fine tuning requires Optimum[Habana] libraries.*
- *Which are developed by Habana and Hugging face teams jointly.*
- *These libraries act as interface between Gaudi hardware and transformer libraires.*
- *Software stack (Synapse AI images) for Voyager are frequently updated. Hence, there is a need to bring in suitable release of Optimum[Habana] library.*
- *One needs to create a Hugging face portal account.*
- *Most LLMs are are gated. Hence, one needs a Hugging face API key for an LLM.*

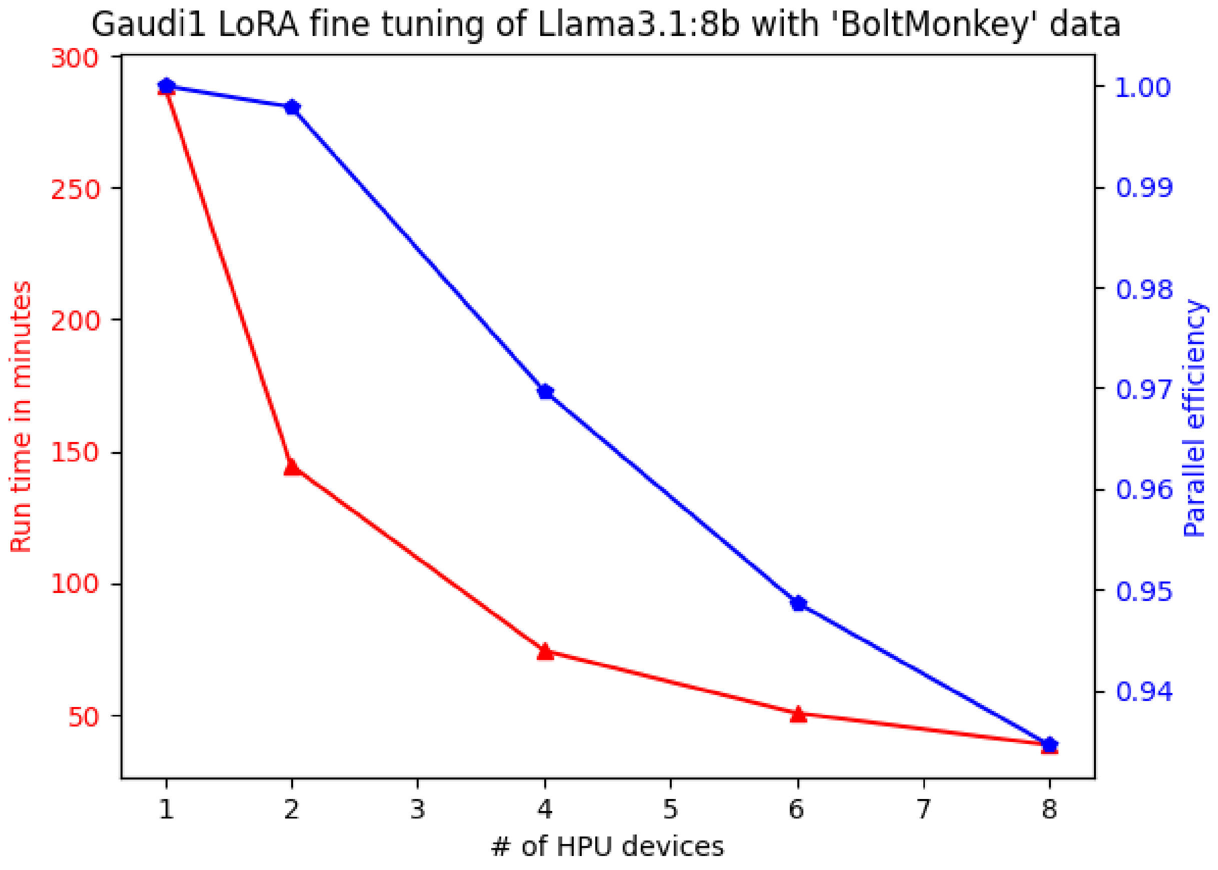
Fine tuning of Llama3.1:8B with different instruct data sets

SynapseAI image 1.18

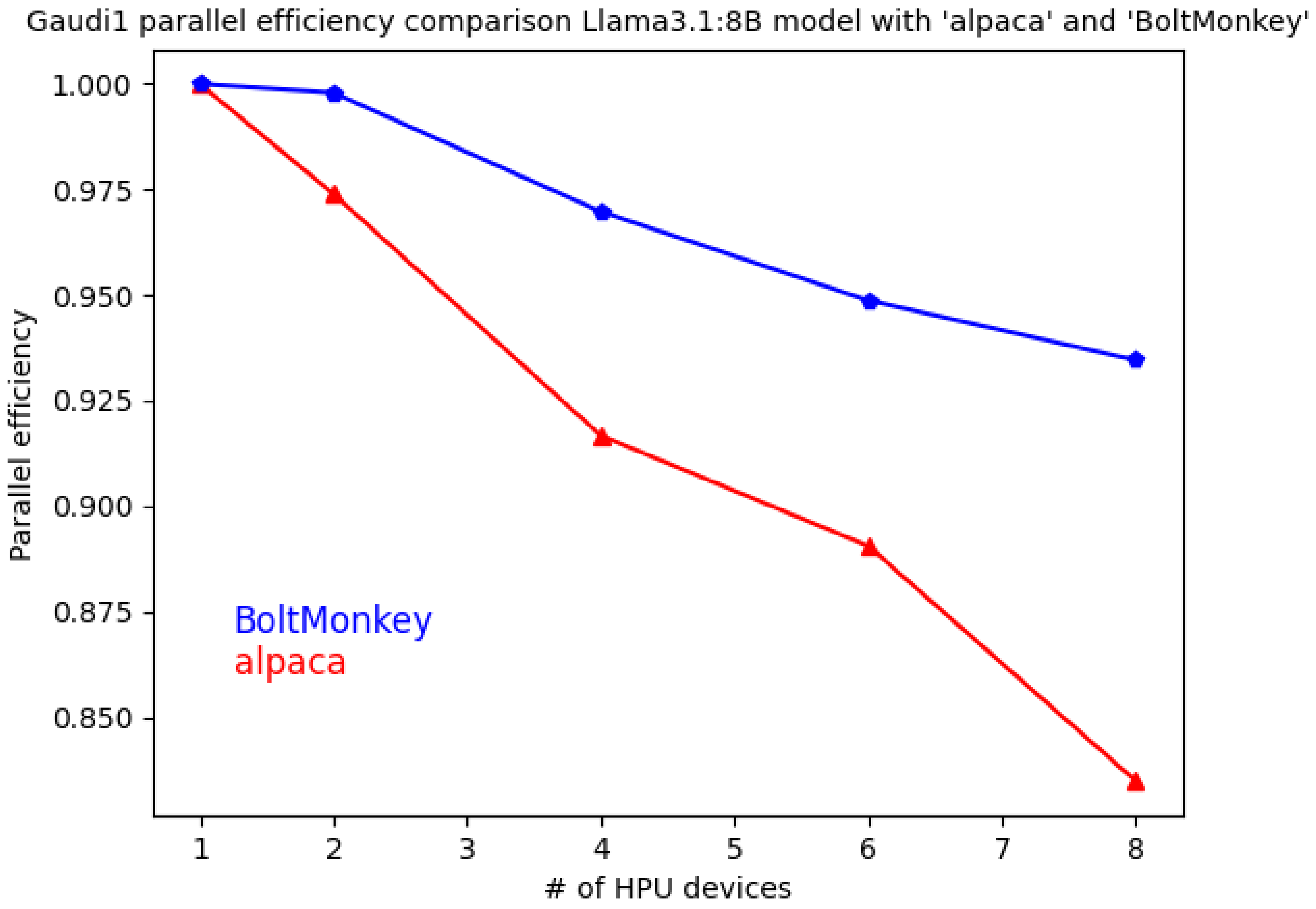
tatsu-lab/alpaca: ~52k records (Generic data)

BoltMonkey: ~197k records (Psychology data)
BoltMonkey data is 3.8 times of alpaca data.

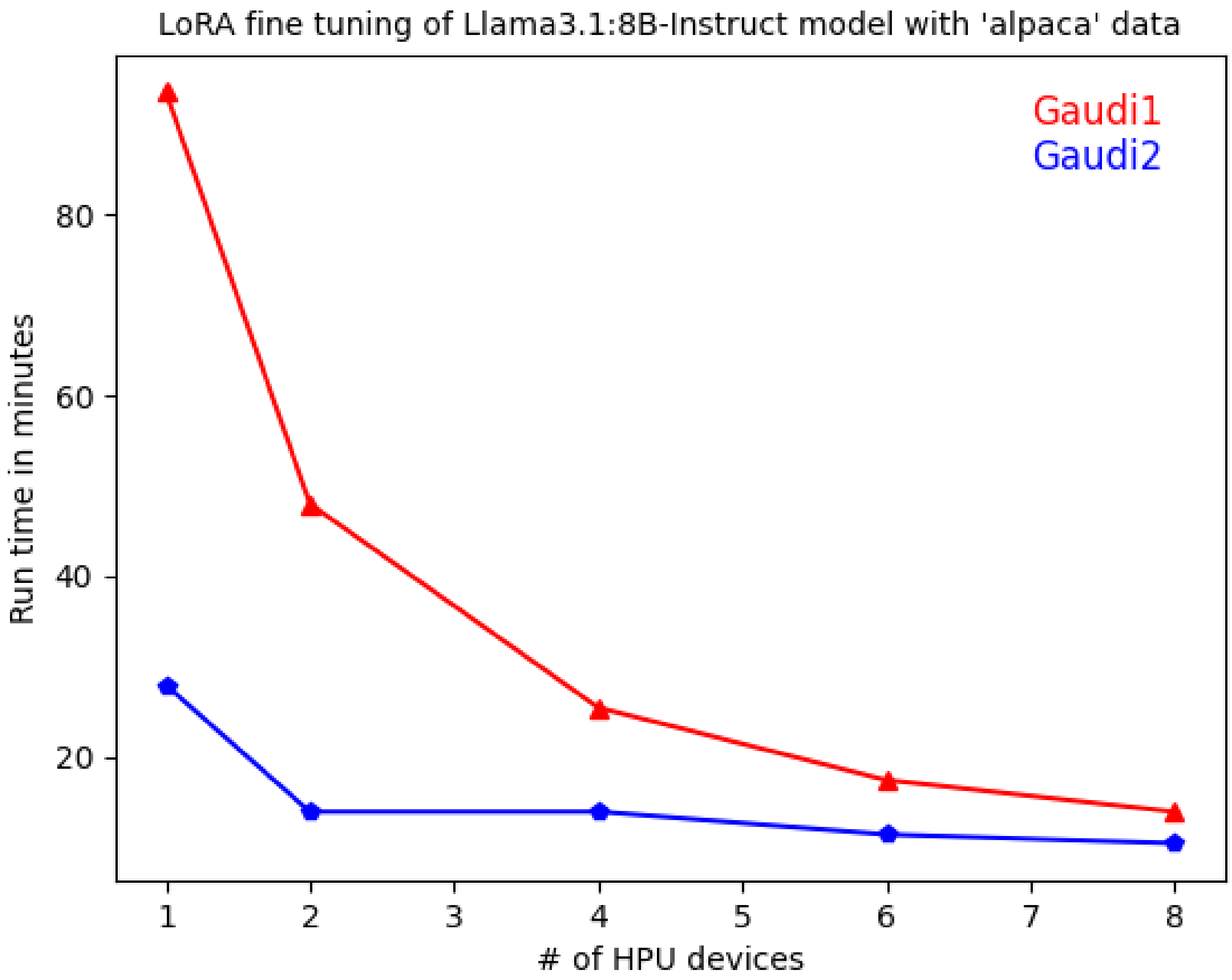




Improved parallel efficiency with the larger data set



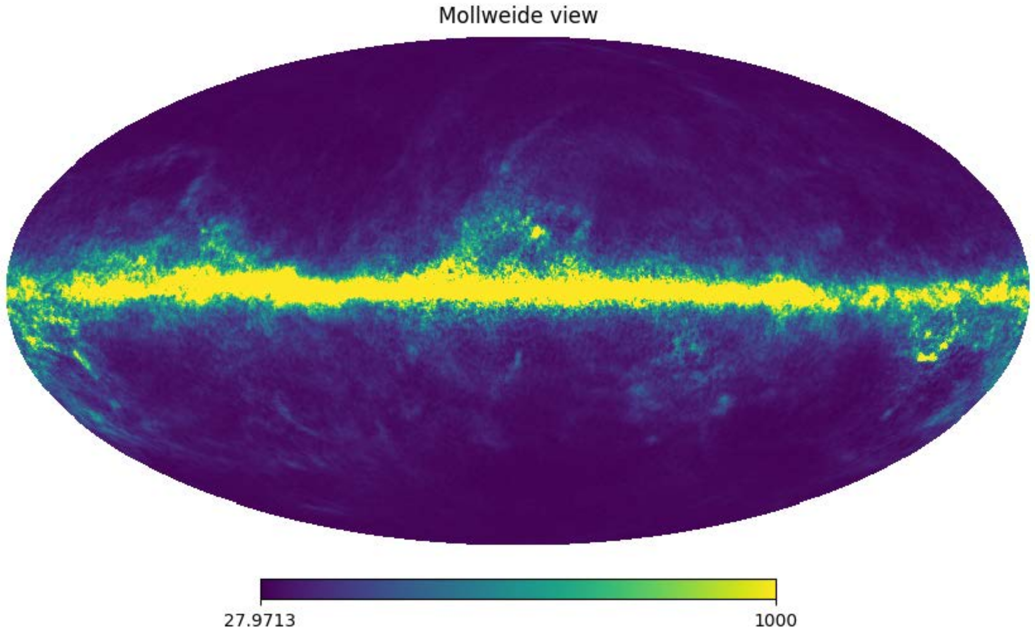
Comparison of fine-tuning performance on Gaudi1 and Gaudi2 nodes



Super-Resolution modeling for cosmological simulations of the cosmic microwave background (CMB)

Javier Hernandez-Nicolau, Andrea Zonca, SDSC-UCSD

- Advancements in cosmological instrumentation have led to higher resolution and lower noise levels.
- However, predicting outcomes from new instruments remains challenging due to the lack of adequate resolution in existing data.
- Consequently, simulations are often used to augment observational data.
- AI offers a solution by enabling the generation of high-resolution maps from low-resolution inputs.



Galactic dust emission generated by PySM

Super-Resolution modeling for cosmological simulations of the cosmic microwave background (CMB)

- A Super-Resolution model, based on Diffusion, has been used to enhance the resolution of CMB images generated by PySM.
- The original AI model [1], which implements Sharia *et al* [2] work, has been ported to Voyager and its Intel Gaudi cards.
- The model has been also parallelized to run in multiple cards (and nodes) using Pytorch DDP.

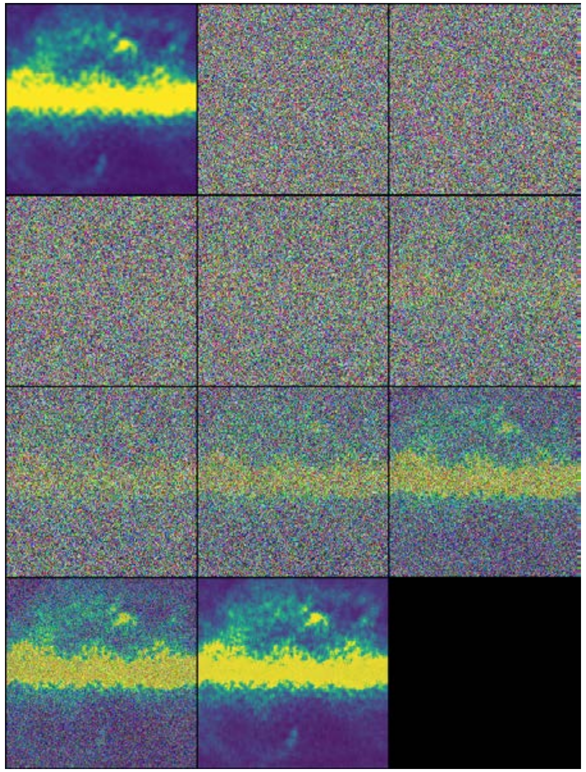
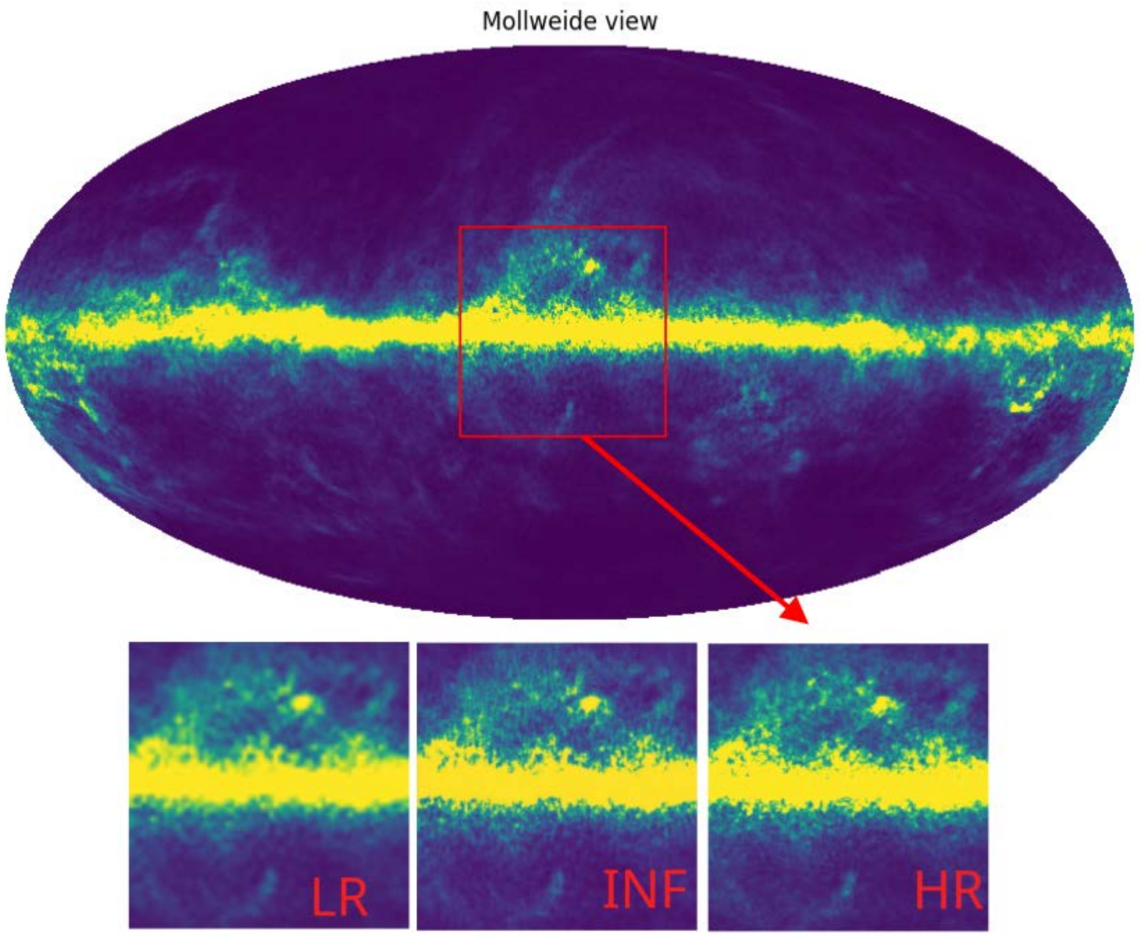


Image reconstruction during inference

[1] *https://github.com/Janspiry/Image-Super-Resolution-via-Iterative-Refinement*
[2] *https://arxiv.org/pdf/2104.07636*

Super-Resolution modeling for cosmological simulations of the cosmic microwave background (CMB)



- Model has been trained on ~1000 256x256 PySM images (to enhance resolution from 64x64 to 256x256).
- SR model has 500M+ parameters and training takes ~40 Gaudi-hours for this task.
- Very promising initial results. SR images exhibit same power spectrum.
- A Discover ACCESS allocation (1.5M credits) has been awarded.
- Future work: Use 1MPix images.

Porting of user applications has been relatively straightforward

- Several applications are now running on *Voyager*
  - *Training*
  - *Finetuning*
  - *Inference, LLMs*
- So far, our experience is that most codes do not need major changes
- Users can run with the familiar PyTorch framework (TensorFlow was available before)
- Typically, the changes are minor and involved using the Habana module and the Habana integrated versions of PyTorch
- Rich suite of models supported and documented on Model Reference page (https://github.com/HabanaAI/Model-References) and documented on developer site (https://developer.habana.ai/)
- SDSC *Voyager* experts assisted users in porting with support from Habana staff as needed
- Typically do an onboarding call for new projects followed by additional meetings/tickets as needed

Voyager would not be possible without a dedicated team of professionals and experts

Rommie Amaro
Haisong Cai*
Trevor Cooper
Chris Cox*
Javier Duarte
Javier Hernandez Nicolau
Tom Hutton*
Christopher Irving*
Andy Goetz
Madhu Gujral
Marty Kandes
Amit Majumdar
Tim McNew*
Dmitry Mishin
Sonia Nayak
Mai Nguyen

Susan Rathbun
Paul Rodriguez
Scott Sakai
Manu Shantharam
Robert Sinkovits
Fernando Silva*
Shawn Strande
Tom Tate*
Mahidhar Tatineni
Mary Thomas
Cindy Wong
Nicole Wolter

Supermicro Team
Habana/Intel Team
Arista

*A special thanks to the HPC Systems Group and the Data Center staff who have been performing onsite work under COVID restrictions