DASYA

dasya.itu.dk

RAD

rad.itu.dk

IT UNIVERSITY OF COPENHAGEN

www.itu.dk

satisfying the data monster with fewer resources

Pınar Tözün

Associate Professor
IT University of Copenhagen

Innovationsfonden

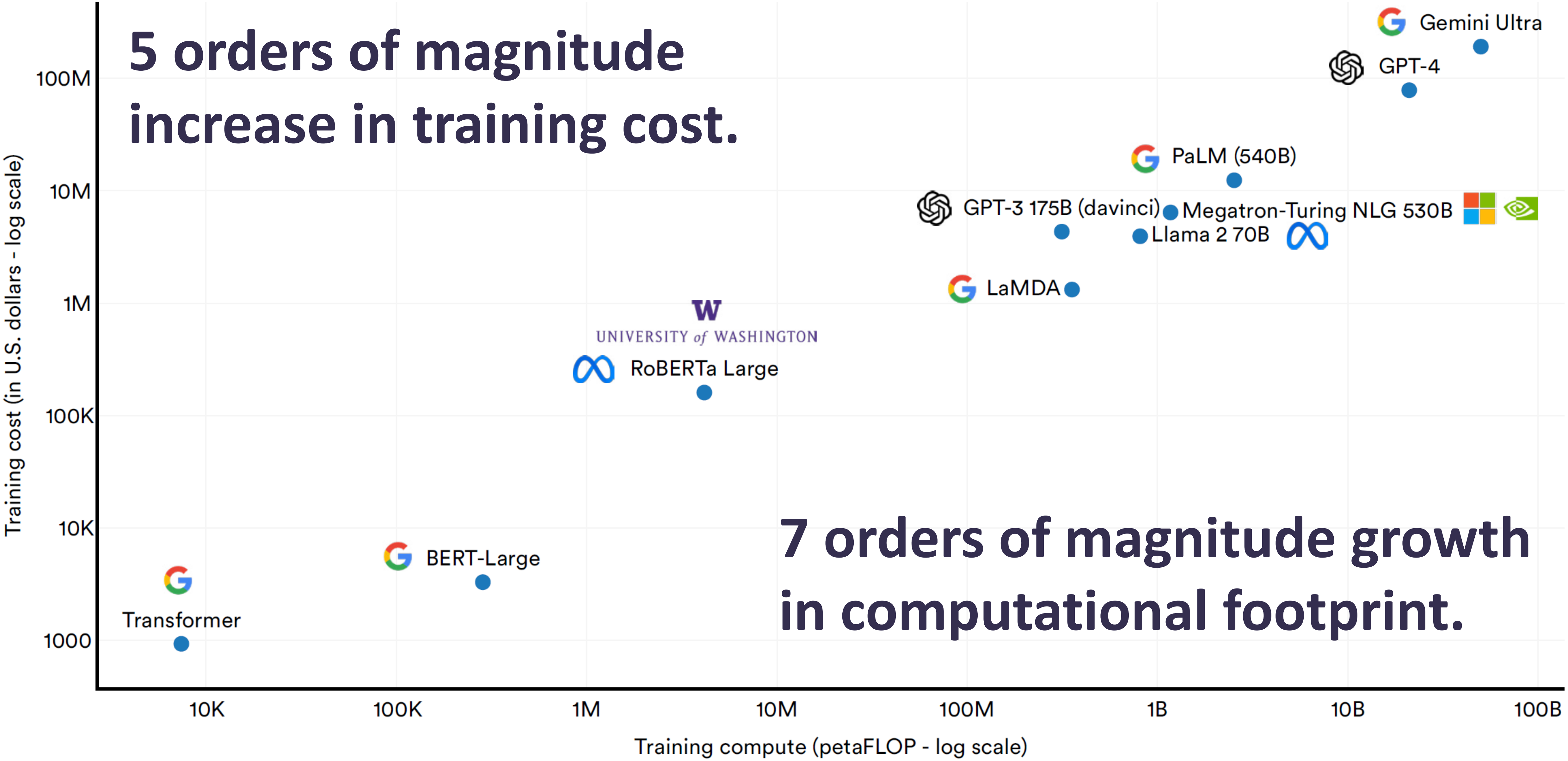
INDEPENDENT RESEARCH FUND DENMARK

novo nordisk foundation

BDA
23/10/2025

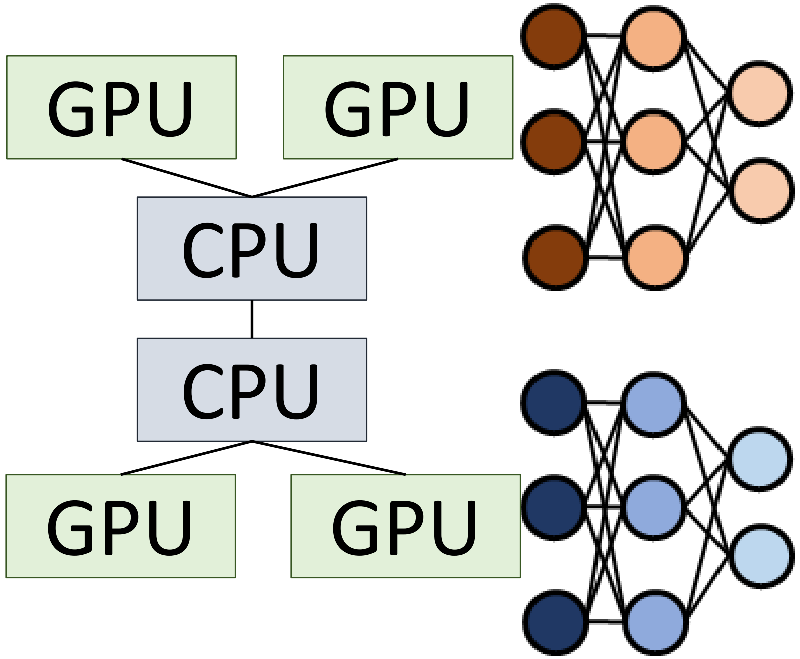
pito@itu.dk
pinartozun.com

language model training (2017 – today)

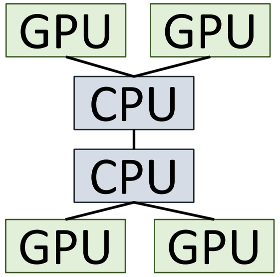


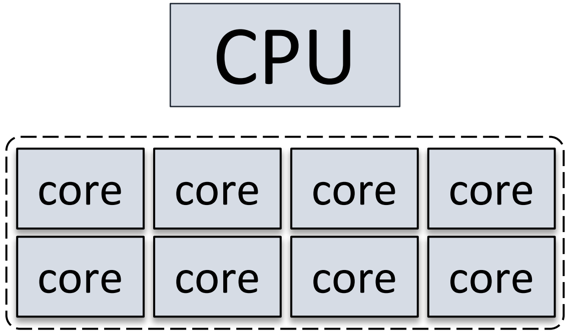
source: [Stanford AI Index Report 2024]

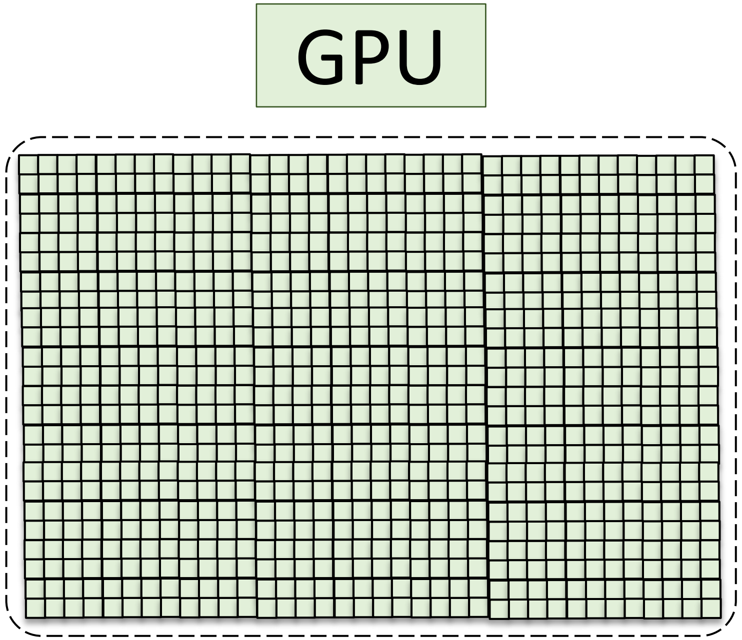
deep learning hardware



deep learning commodity hardware







central processing unit

➔ several (complex) cores

➔ good for latency-oriented tasks & single-core performance

- throughput- vs latency-oriented designs exist among CPUs as well

graphics processing unit

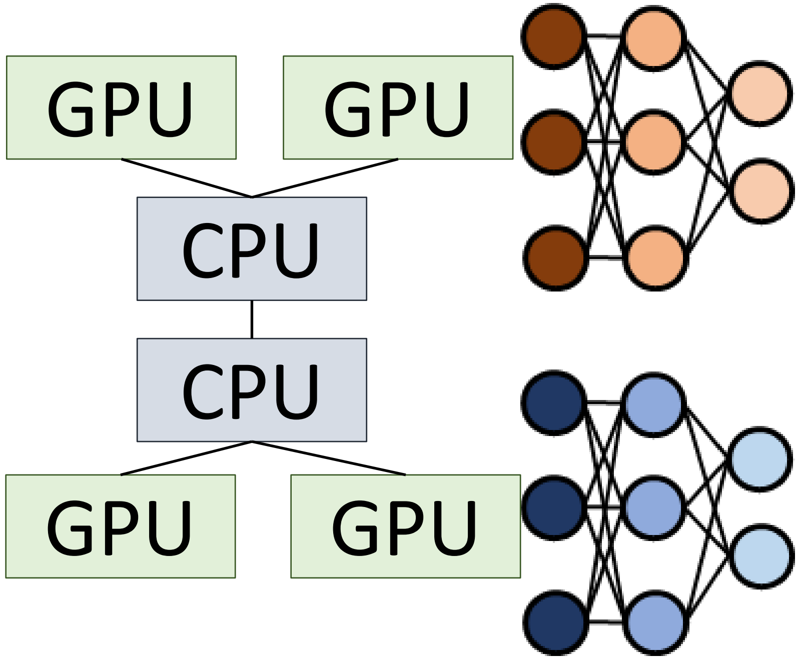
➔ many (simple) cores

➔ good for throughput-oriented & embarrassingly parallel tasks

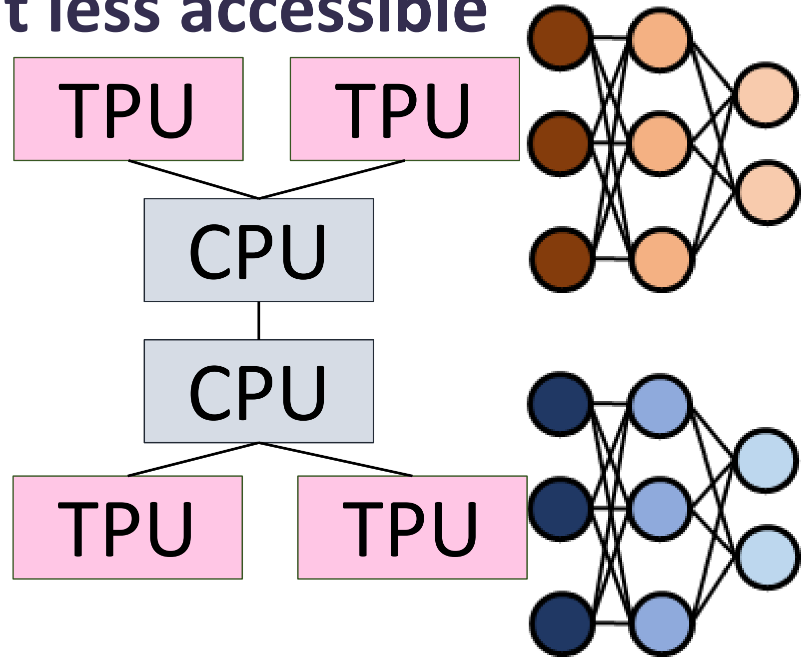
➔good for deep learning

- e.g., large matrix operations

deep learning hardware

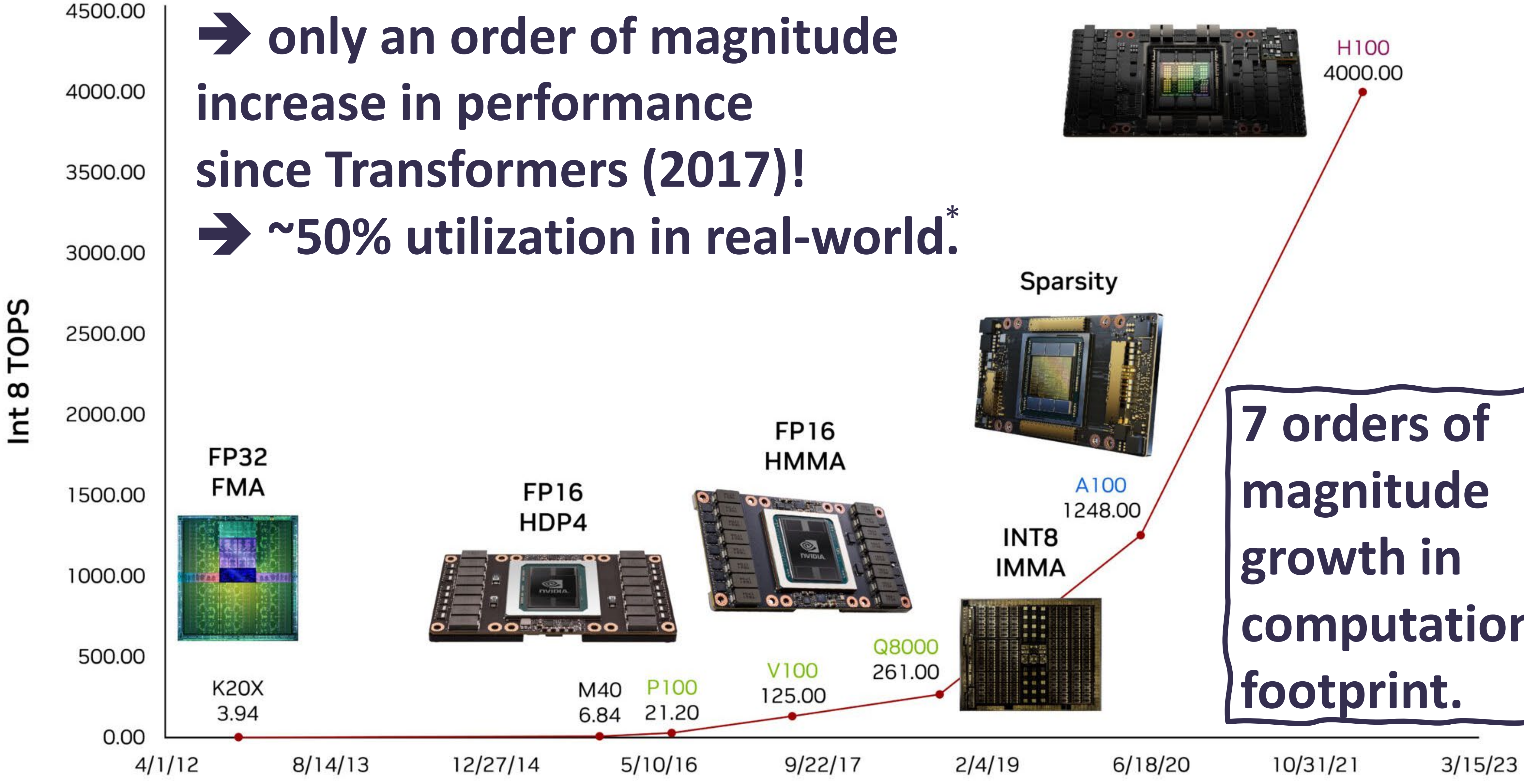


- **tend to be more efficient**
- **but less accessible**



costs & progress depend on the performance & utilization of the available hardware.

NVIDIA GPUs (2012 – 2023)



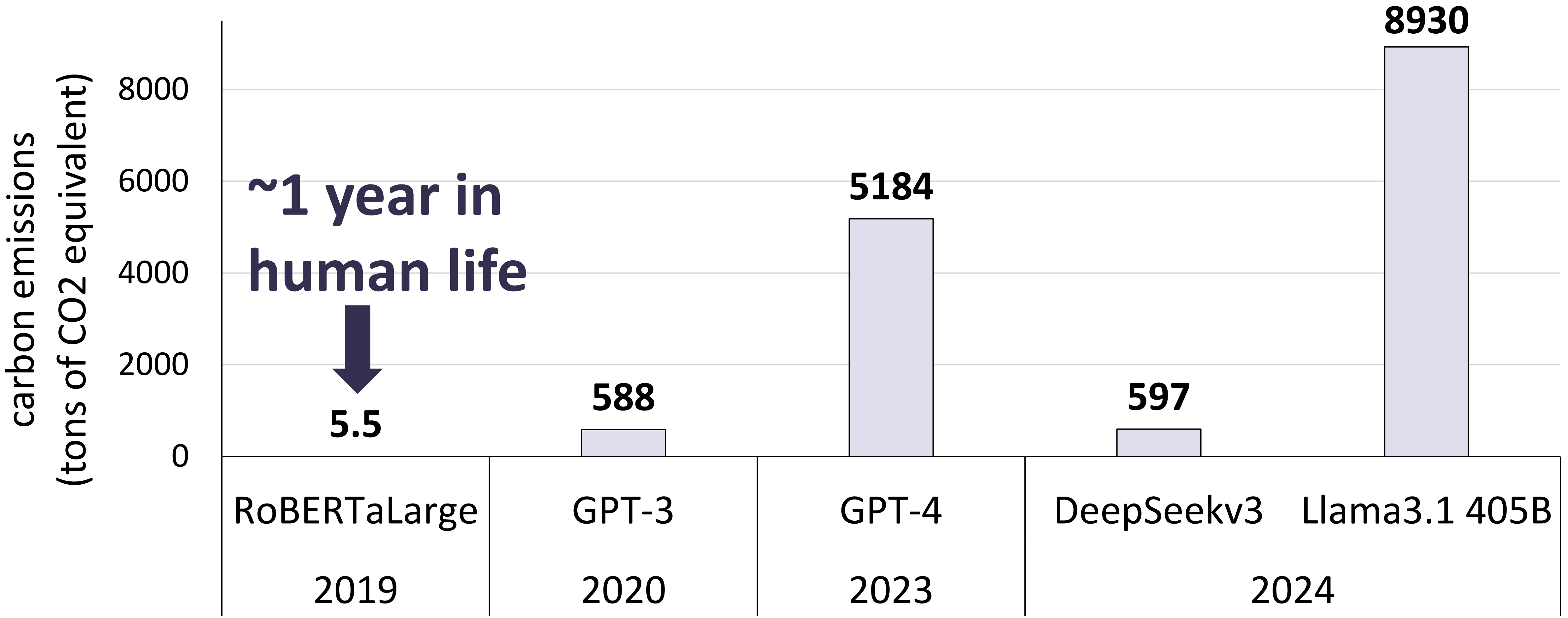
➔ only an order of magnitude increase in performance since Transformers (2017)!

➔ ~50% utilization in real-world.*

7 orders of magnitude growth in computational footprint.

sources: figure by [Bill Dally, NVIDIA] & *Gao et al. "[An Empirical Study on Low GPU Utilization of Deep Learning Jobs]." ICSE 24

carbon footprint of language model training



can we do better while using fewer resources?

model accuracy cannot be the only metric to aim for!

source: [Stanford AI Index Report 2025]

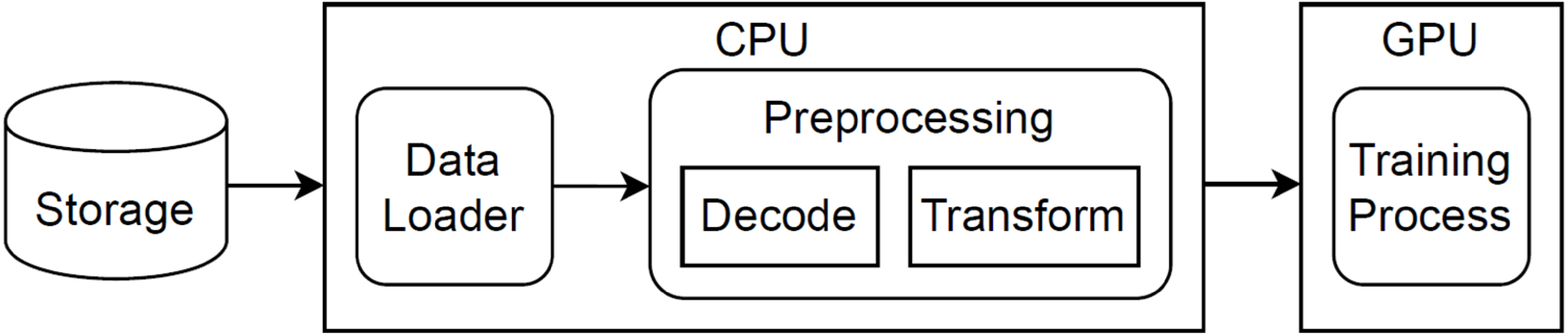
deep learning with fewer resources

IT UNIVERSITY OF COPENHAGEN

- GPU-centric data path
- data & work sharing
- impact of data selection

for model training

journey of data in deep learning training



CPU feeds the GPU

- 16-64 CPU cores per GPU (recommended)
- 96 CPU cores per TPU*

➔ otherwise, GPU/TPU may be underutilized

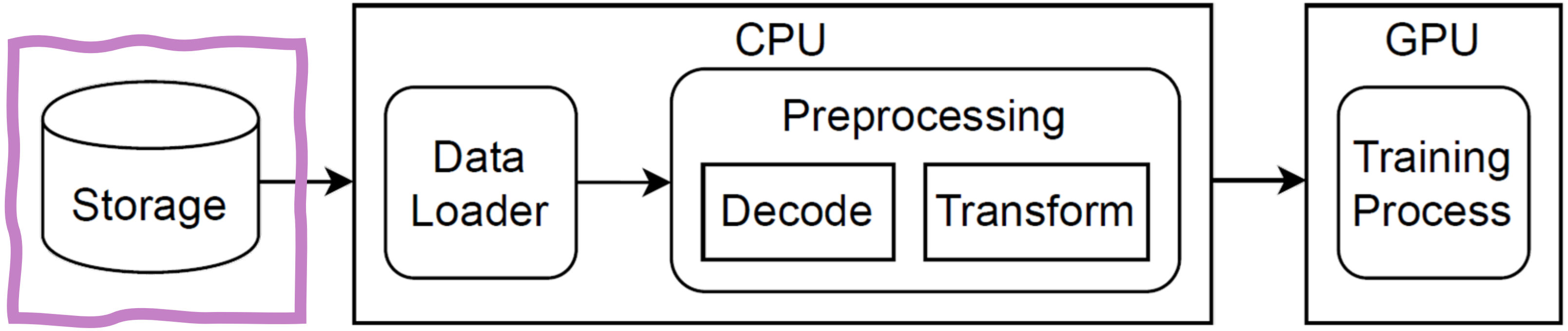
➔ can we do more with fewer CPUs & less of the CPU?

*Audibert et al., "tf.data service: A Case for Disaggregating ML Input Data Processing." ACM SoCC 2023

deep learning with fewer resources

[Path to GPU-Initiated I/O for Data-Intensive Systems]
Karl B. Torp, Simon Lund, Pınar Tözün.
DaMoN 2025

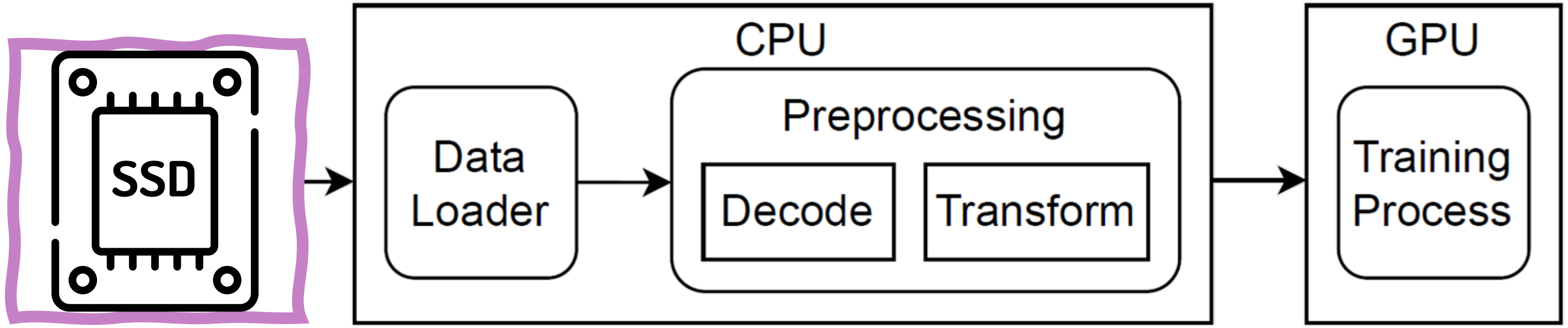
- GPU-centric data path
- data & work sharing
- impact of data selection



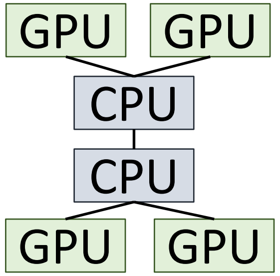
deep learning with fewer resources

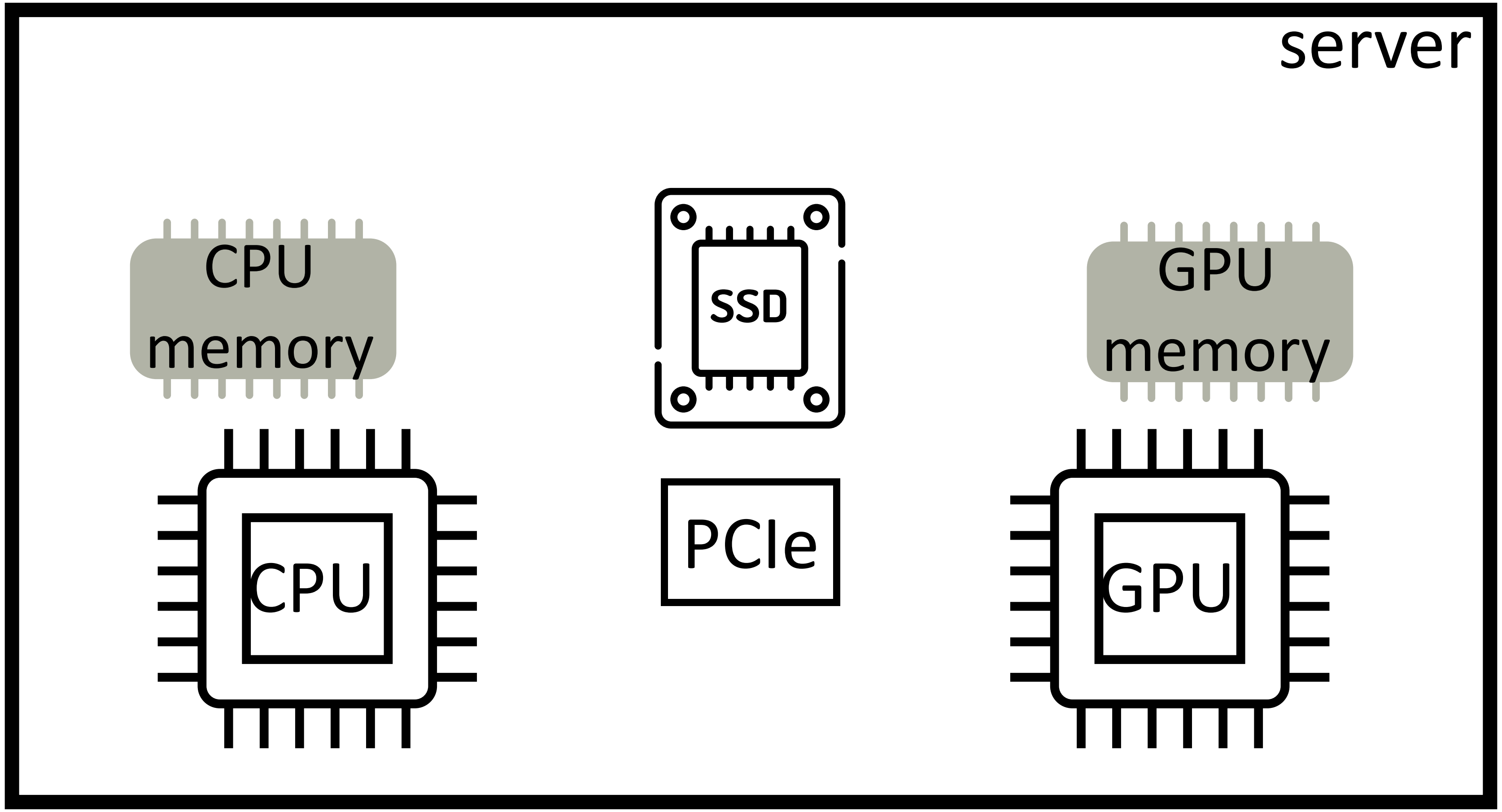
[Path to GPU-Initiated I/O for Data-Intensive Systems]
Karl B. Torp, Simon Lund, Pınar Tözün.
DaMoN 2025

- GPU-centric data path
- data & work sharing
- impact of data selection



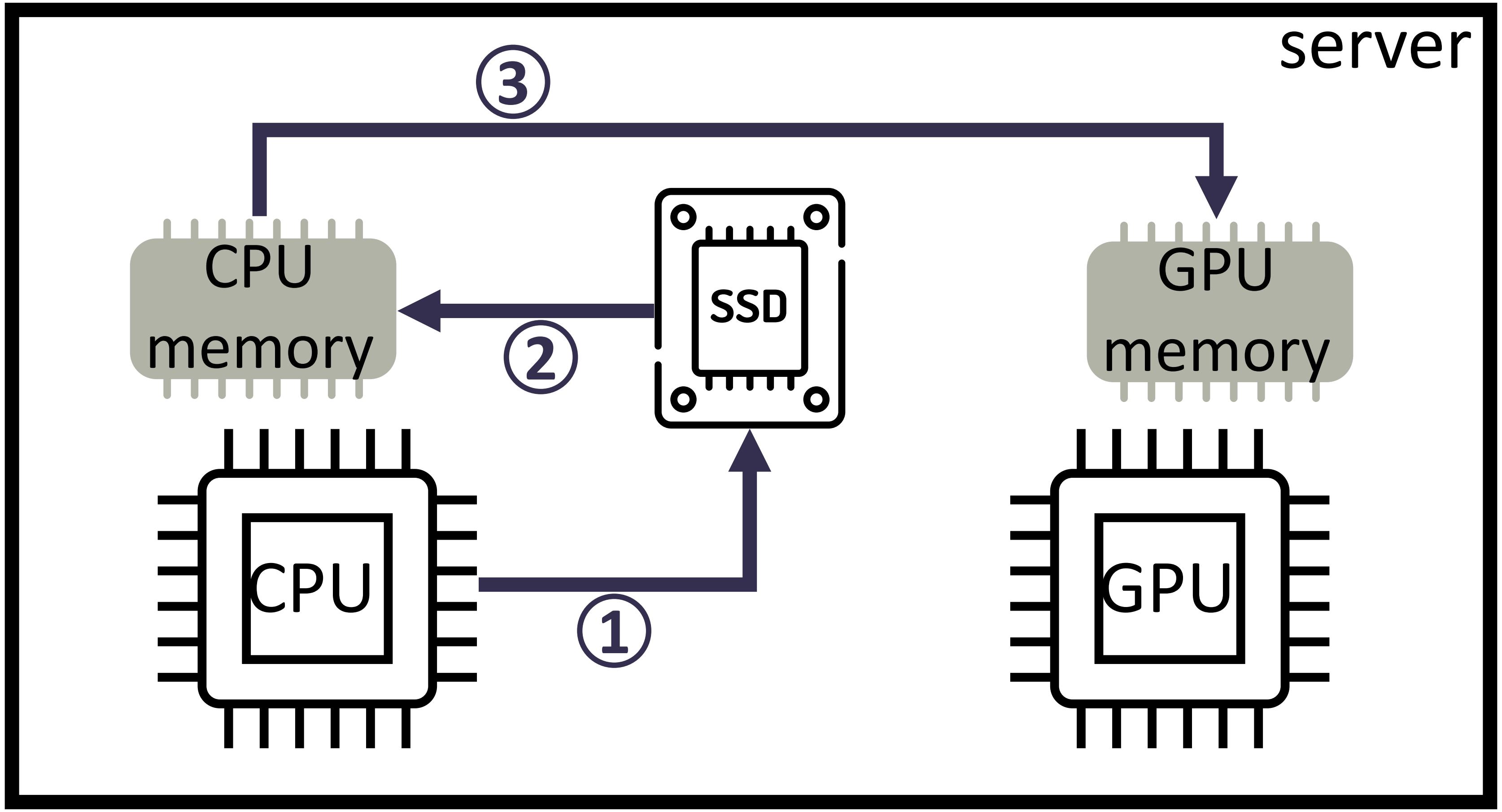
target hardware setup





* PCIe is dropped in the remaining figures for the sake of simplicity in illustrations.

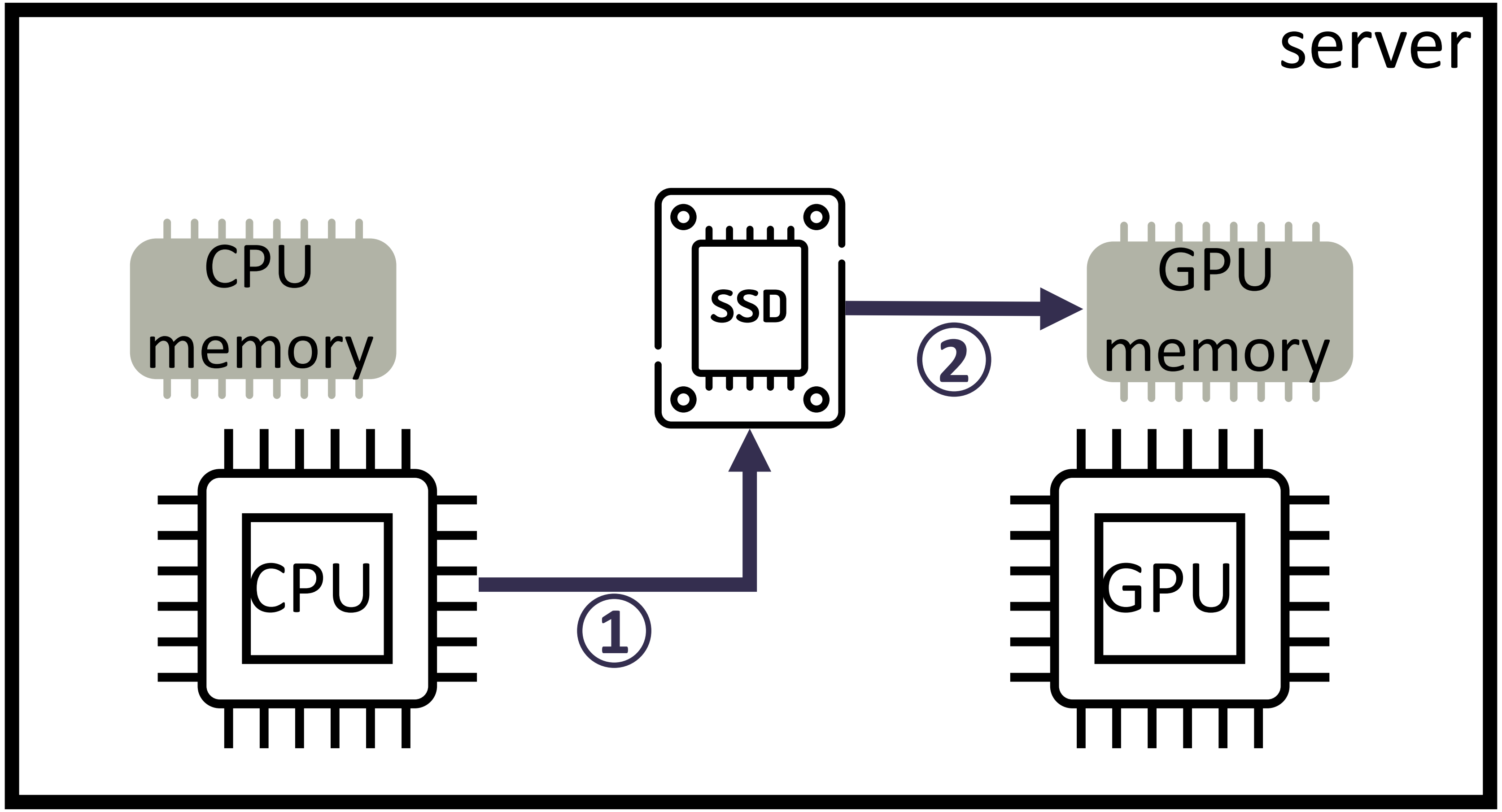
conventional: CPU-centric



- ✓ **ecosystem support**
- × **CPU-bound & overhead from memory copy**

GDS: GPU-centric & CPU-initiated

GPU Direct Storage [NVIDIA'19]

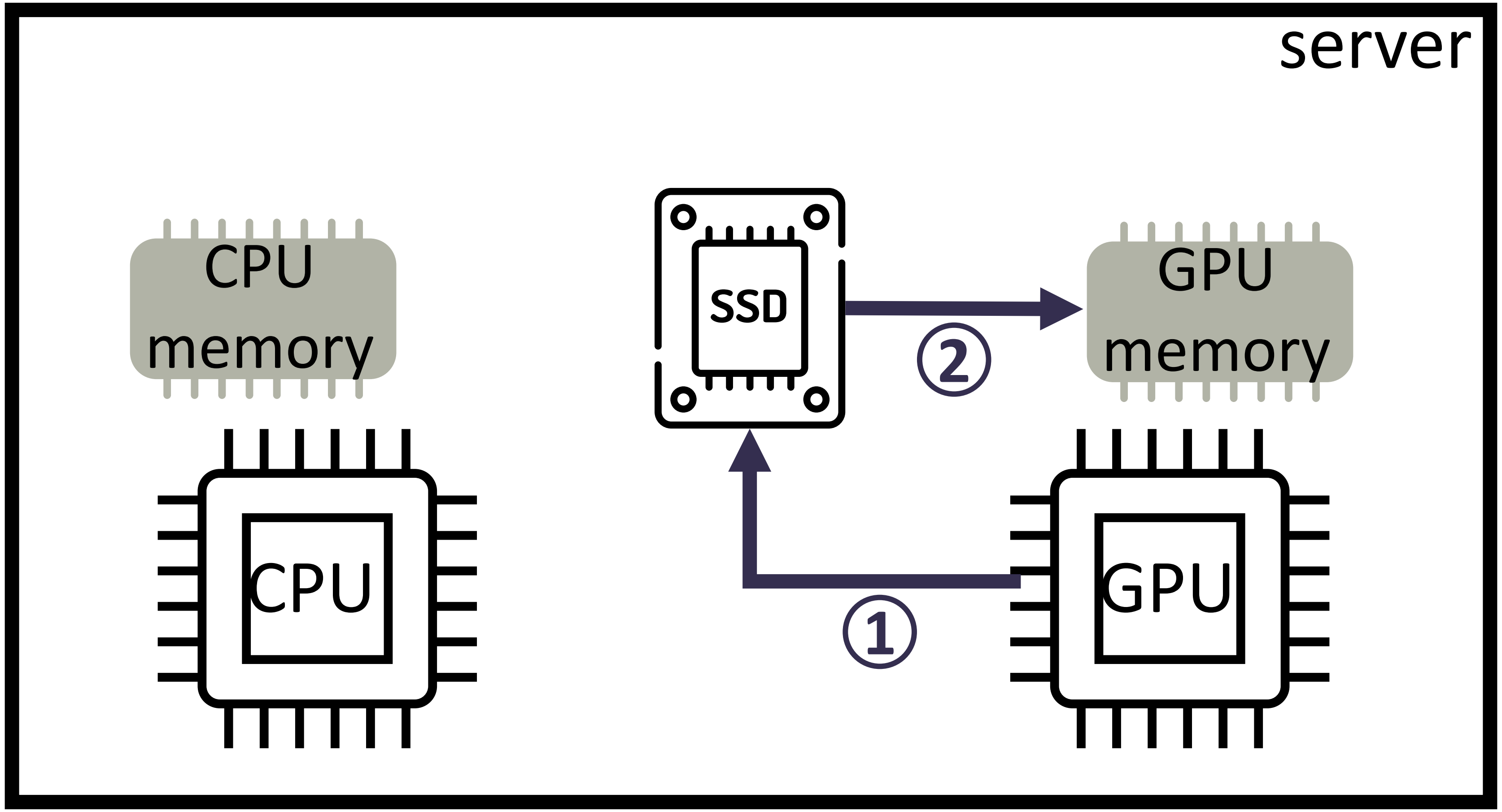


✓ **eliminates the extra memory copy**

× **still CPU-bound**

BaM: GPU-centric & GPU-initiated

Big Accelerator Memory [ASPLOS'23]



✓ **eliminates the CPU on the path**

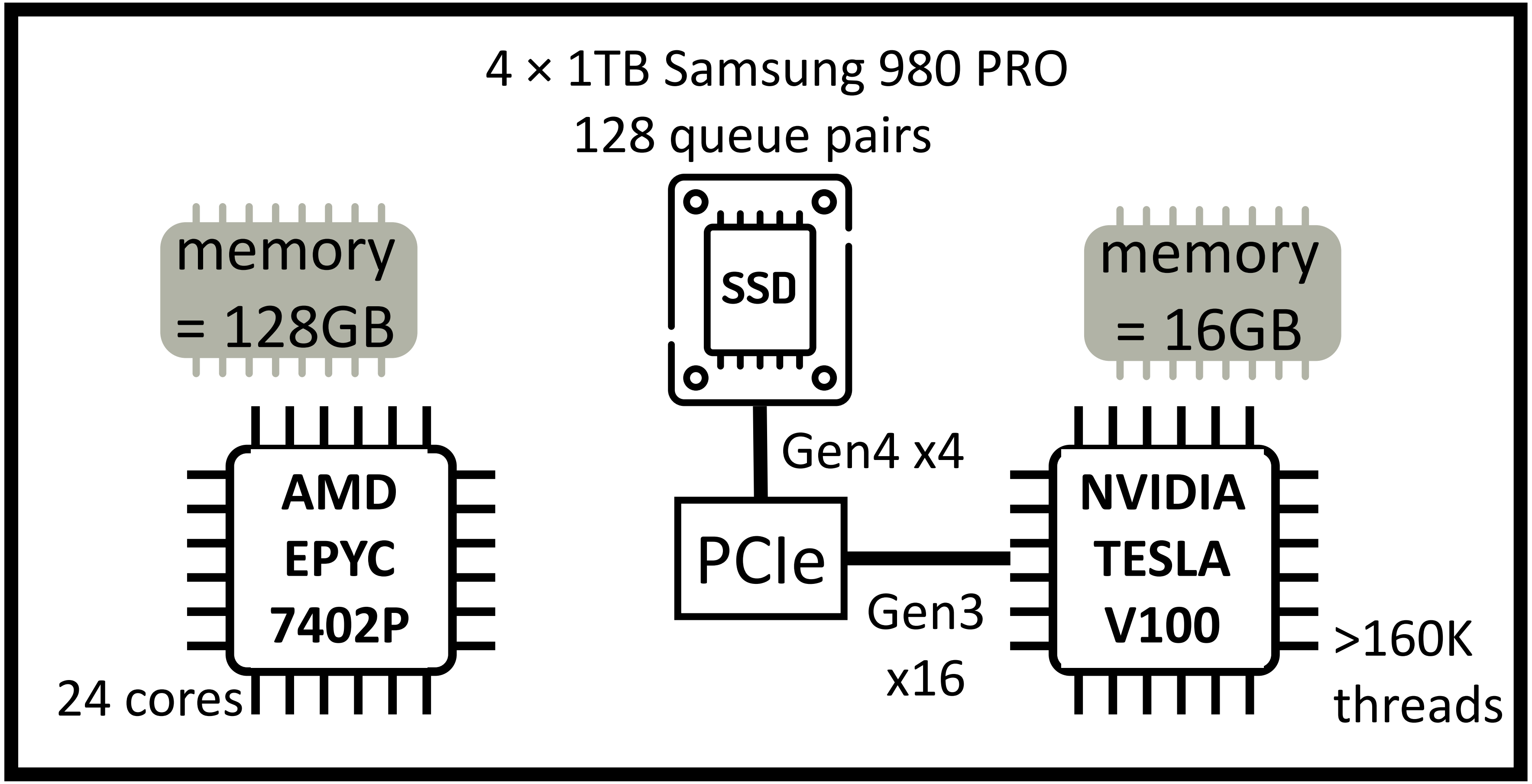
× **ecosystem missing & saturates GPU**

CPU- vs GPU-centric storage access

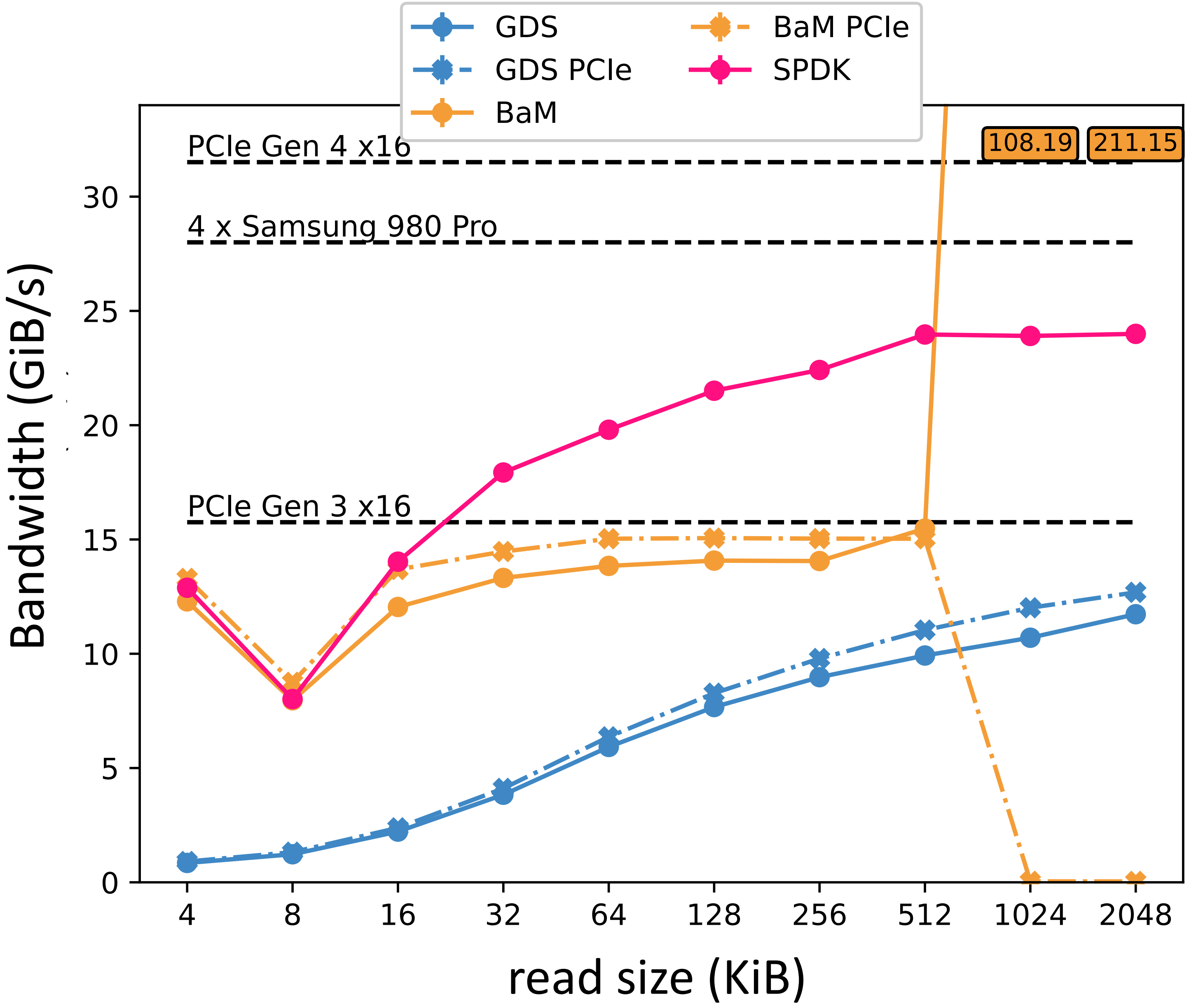
mechanisms: CPU-centric: SPDK & GPU-centric: GDS, BaM

workload: random reads
→ each mechanism has their own tool for benchmarking

hardware



bandwidth utilization – 4 SSDs & PCIe



GDS is CPU-compute heavy.

➔ 16 logical cores utilized

BaM is limited by the PCIe Gen3 link & heavy on the GPU resources.

➔ whole GPU utilized

CPU-centric SPDK is resource-efficient but has a longer path to the GPU.

➔ 2 logical cores utilized

path to GPU-centric storage access

- need to reduce the dependency on CPUs
  for more efficient deep learning pipelines
- GPU-centric data path is a way to do that
  & we have the mechanisms today (e.g., GDS, BaM)
  - GDS has dependency on CPUs still
  - BaM requires a lot of GPU resources

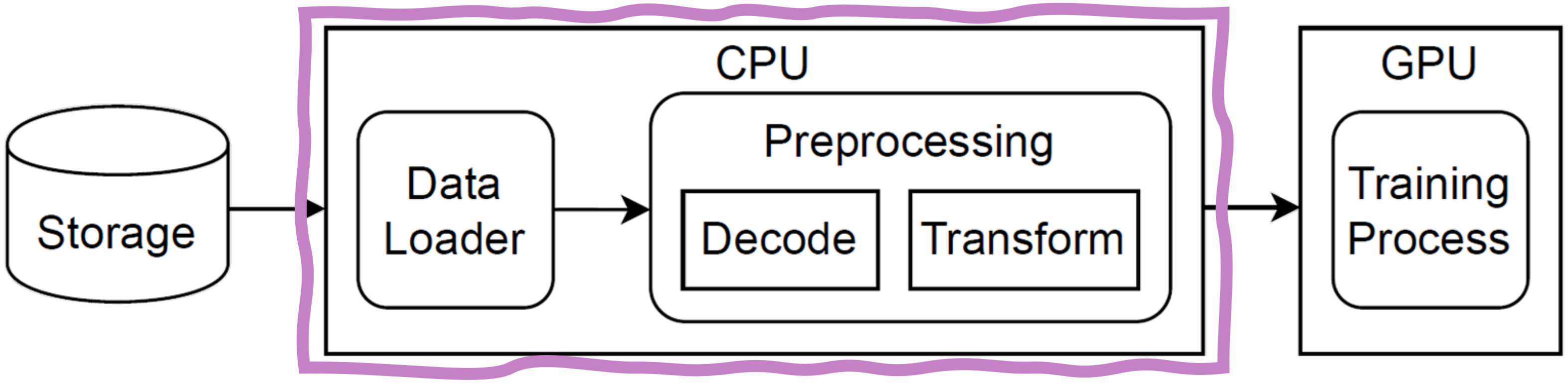
➔ when to use which mechanism while being resource-aware?

➔ how to best integrate them into popular deep learning frameworks (or GPU databases) for wider-scale use?

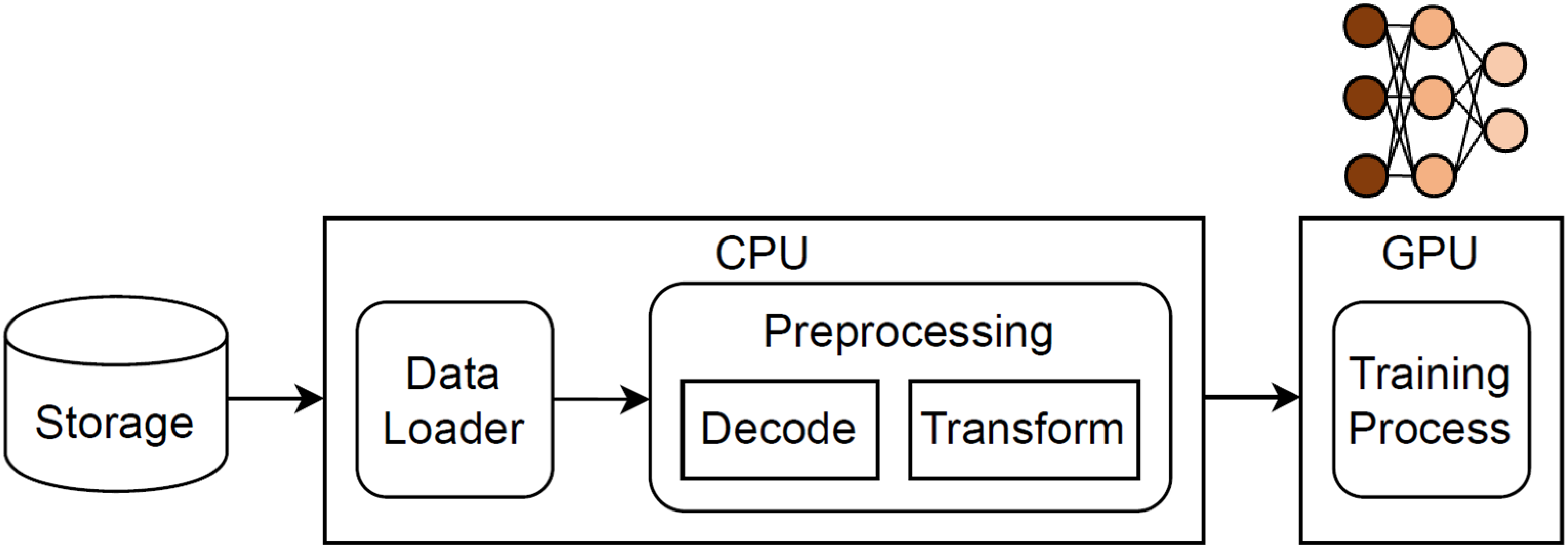
deep learning with fewer resources

- GPU-centric data path
- data & work sharing
- impact of data selection

[TensorSocket: Shared Data Loading for Deep Learning Training]
Ties Robroek, Neil Kim Nielsen, Pınar Tözün.
SIGMOD 2026



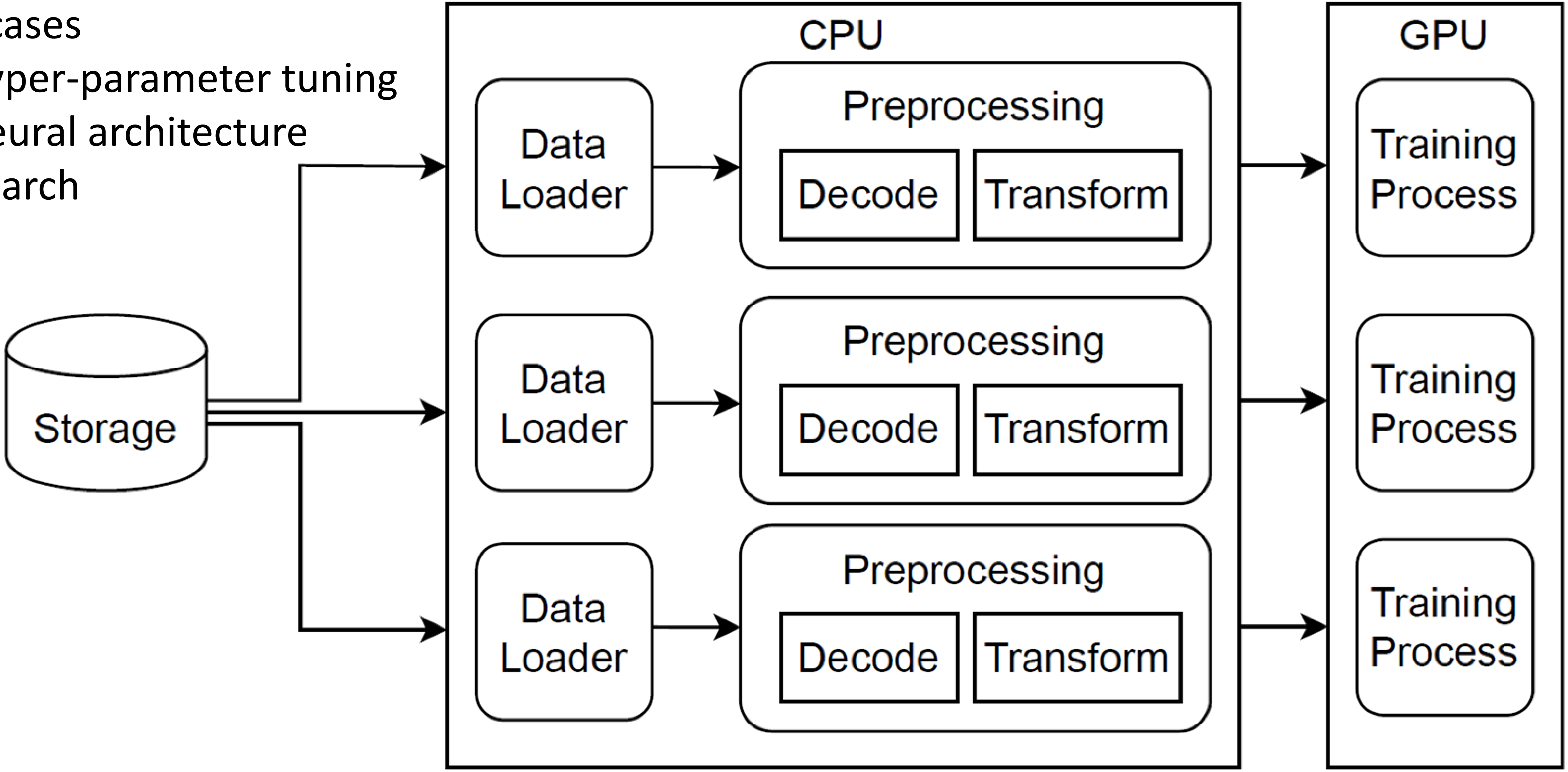
conventional journey of data while training



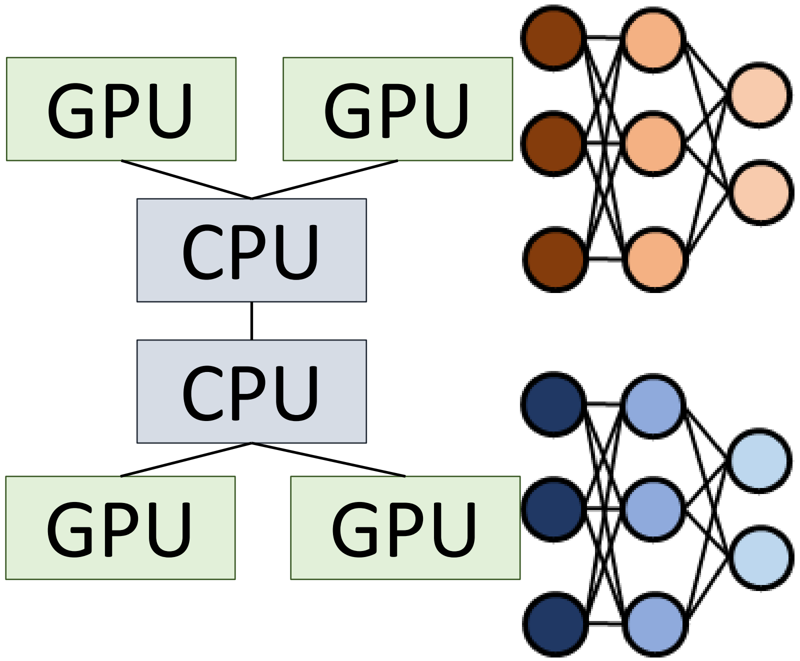
multiple model training on the same data

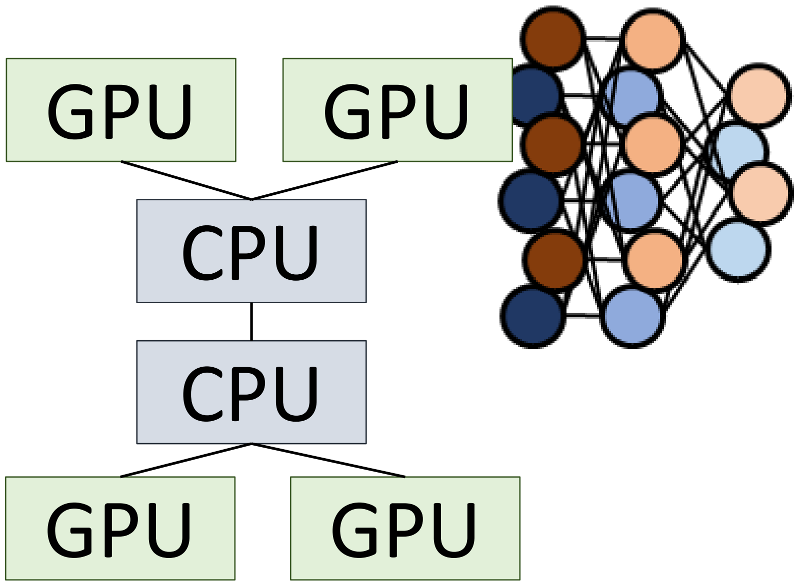
use cases

- hyper-parameter tuning
- neural architecture search
- …



collocated training

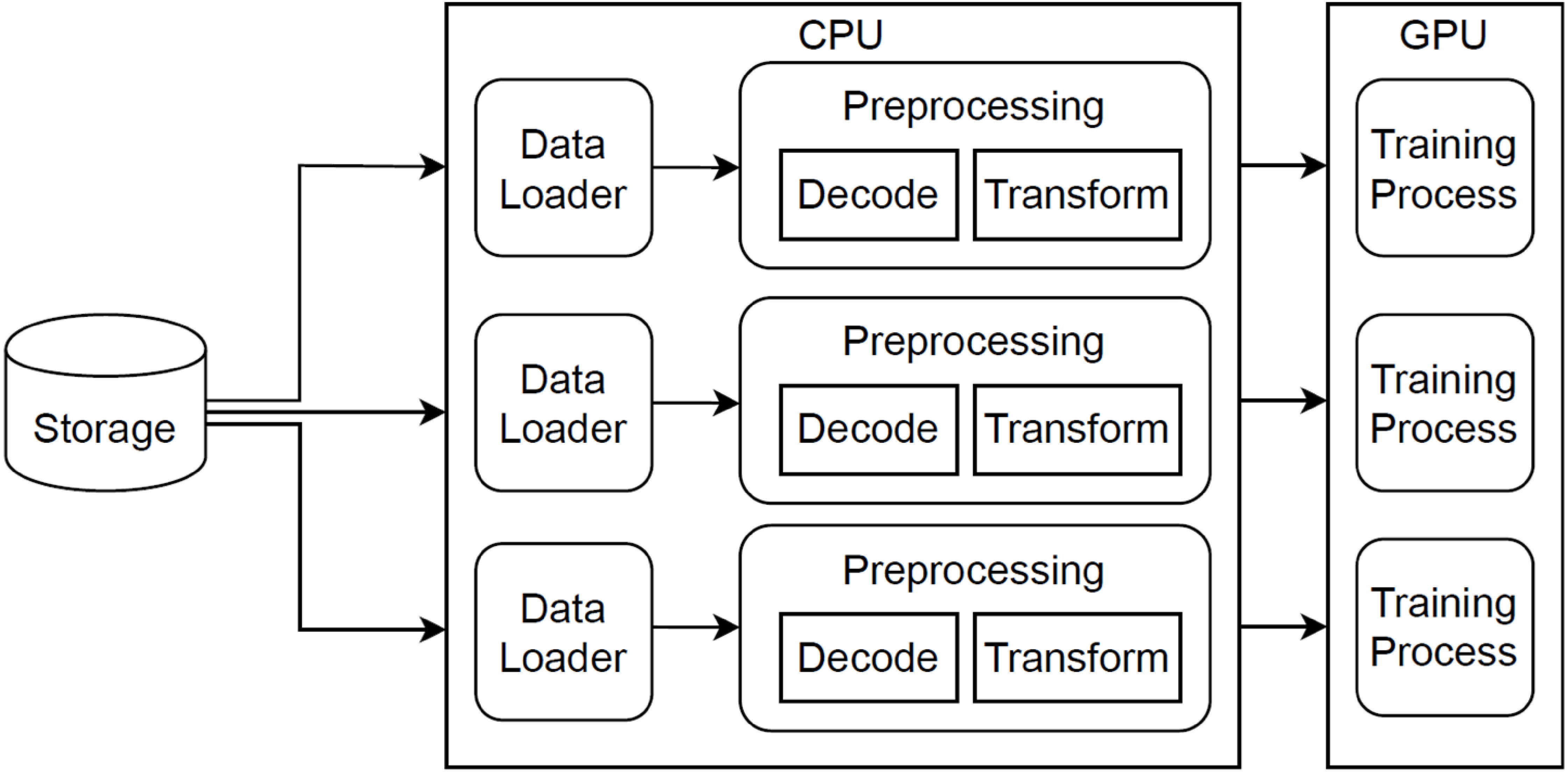




→ training more models with fewer hardware resources

→ leads to better hardware utilization & reduces costs

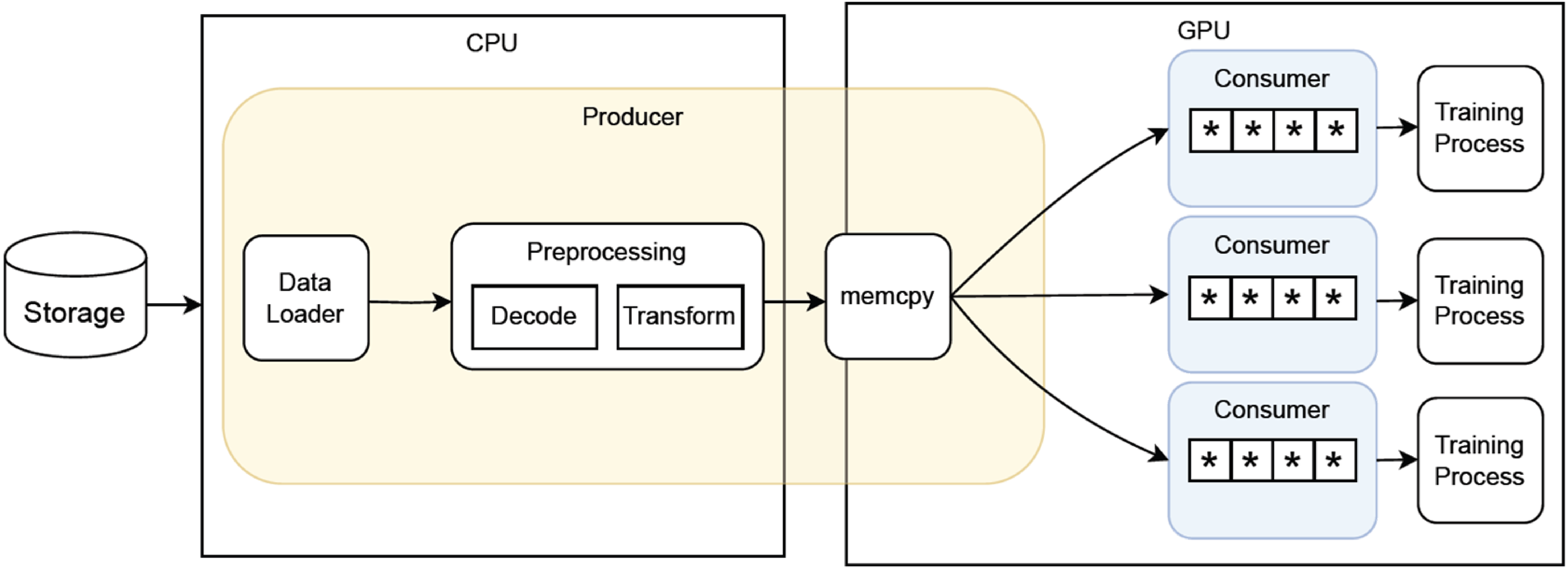
multiple model training on the same data



redundant work & CPU use!

data sharing for collocated training

TensorSocket



minimize the redundancy!

TensorSocket requirements & limitations

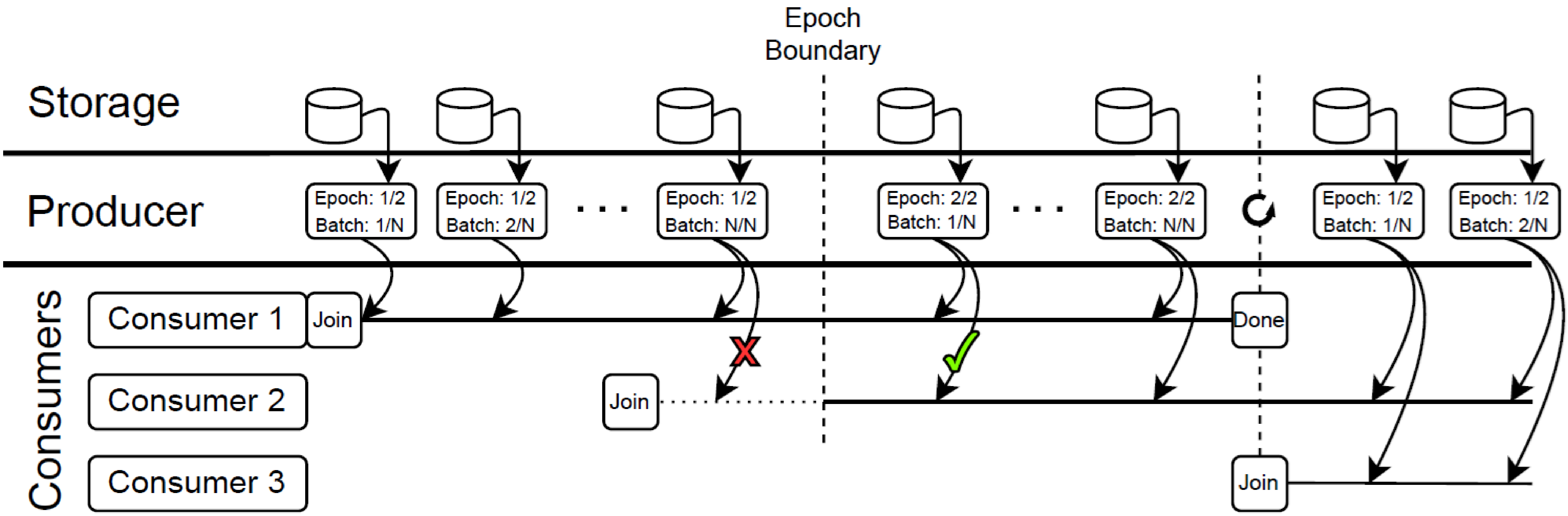
IT UNIVERSITY OF COPENHAGEN

➔ **consumers go through the *same dataset* at the *same rate***

but consumers can …

- join at different epochs of training
- have different batch sizes
- be different models

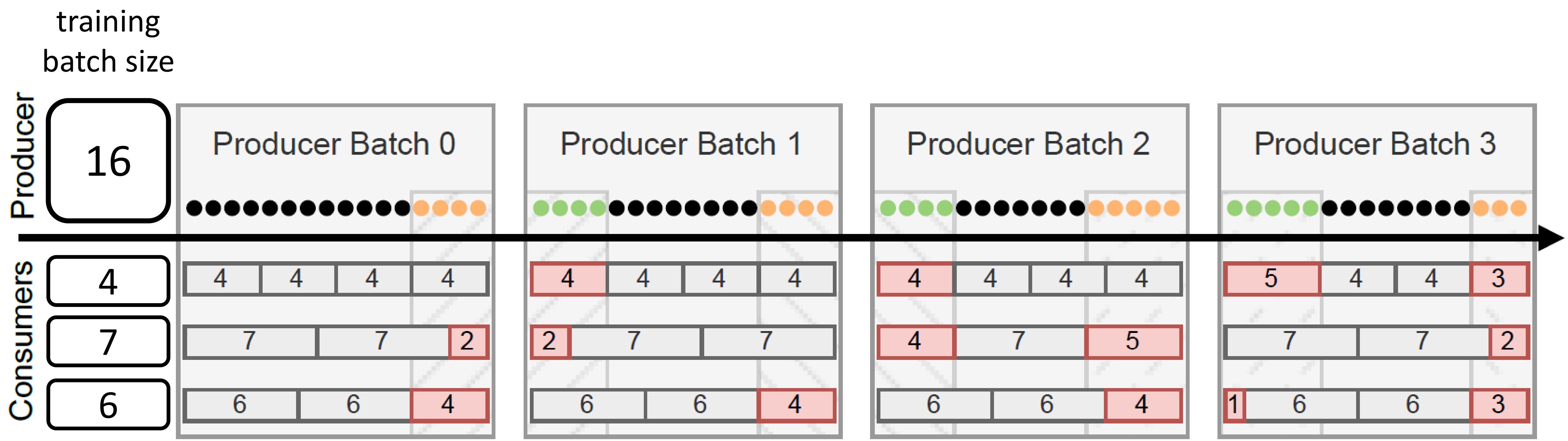
consumers joining at different epochs



trade-off of training latency for throughput & resources.

training is trail-&-error → latency is less critical.

flexible batch sizing



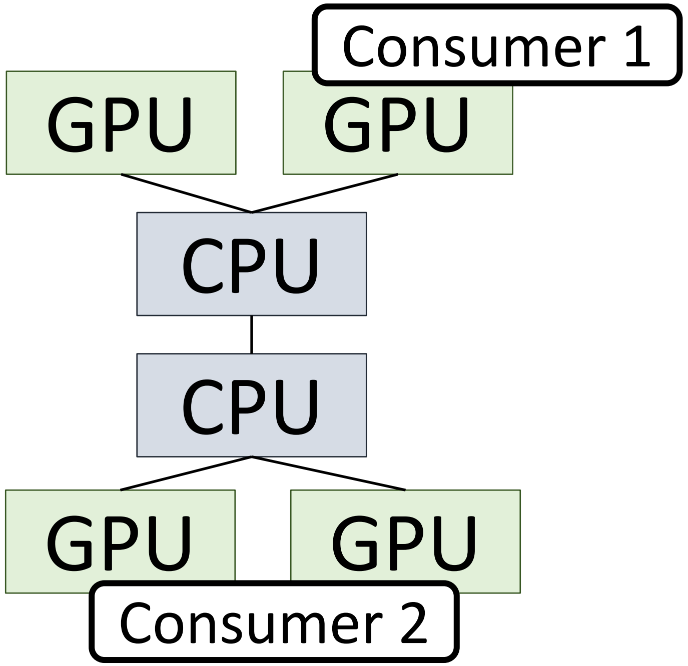
trade-off of repeated data to get flexibility.

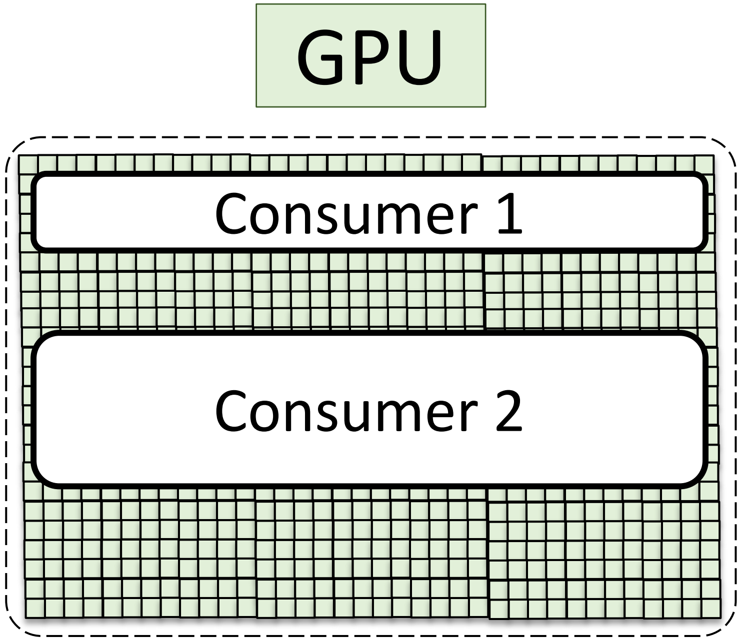
in practice, batch sizes tend to be multiples of 2.

different consumers / models

Consumer 1
MobileNet Small

Consumer 2
MobileNet Large





can adjust the hardware resources per consumer to ensure each goes over the data at the same rate

TensorSocket requirements & limitations

➔ **consumers go through the *same dataset* at the *same rate***

but consumers can …

- join at different epochs of training
- have different batch sizes
- be different models

➔ **target is smaller scale**

- collocation of model training on a single server
- models fit into the memory of a single GPU

- Varoquaux et al. Hype, Sustainability, and the Price of the Bigger-is-Better Paradigm in AI
- Margot Seltzer, SIGMOD'25 keynote

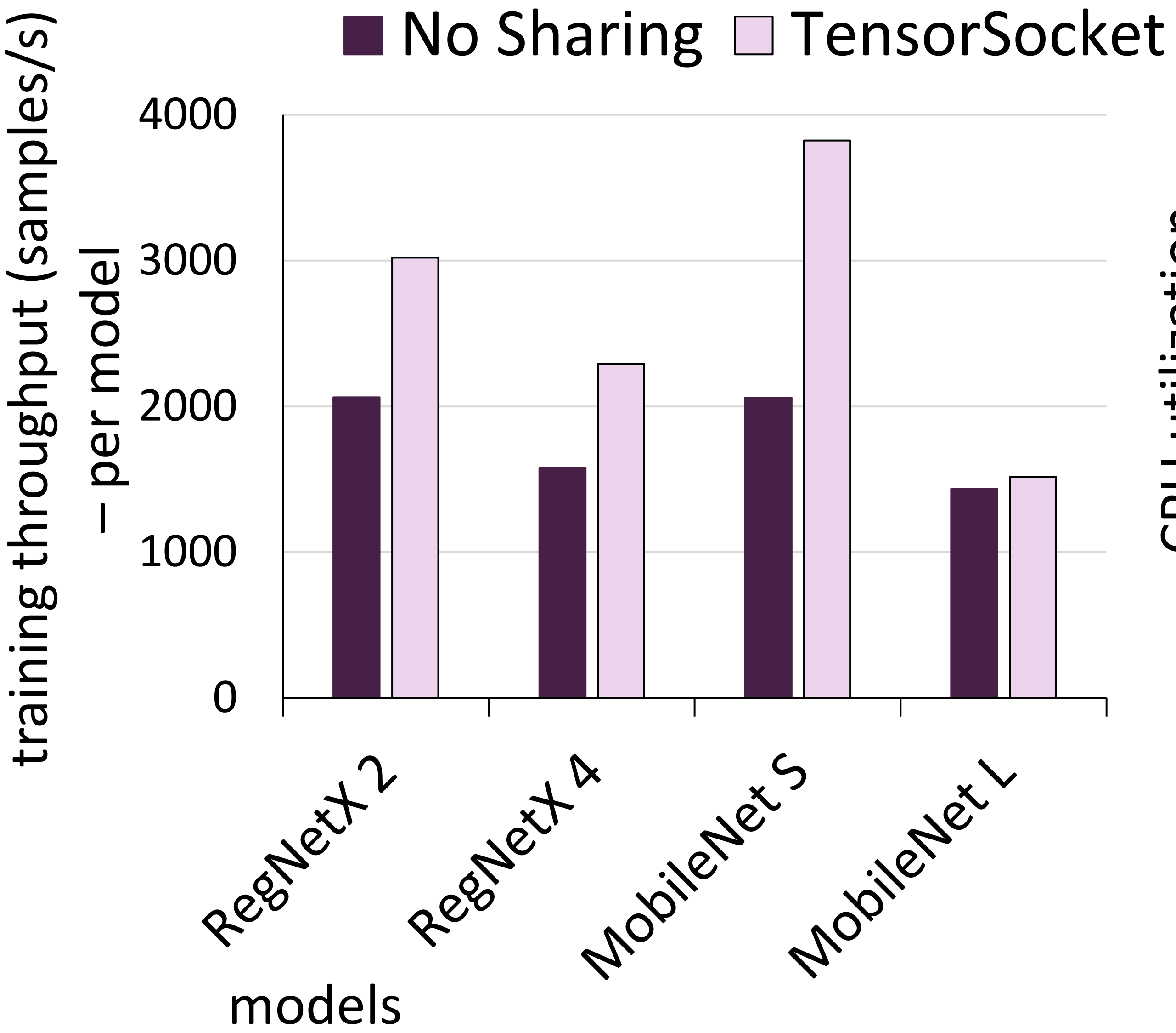
not everyone needs "big" models & scale!

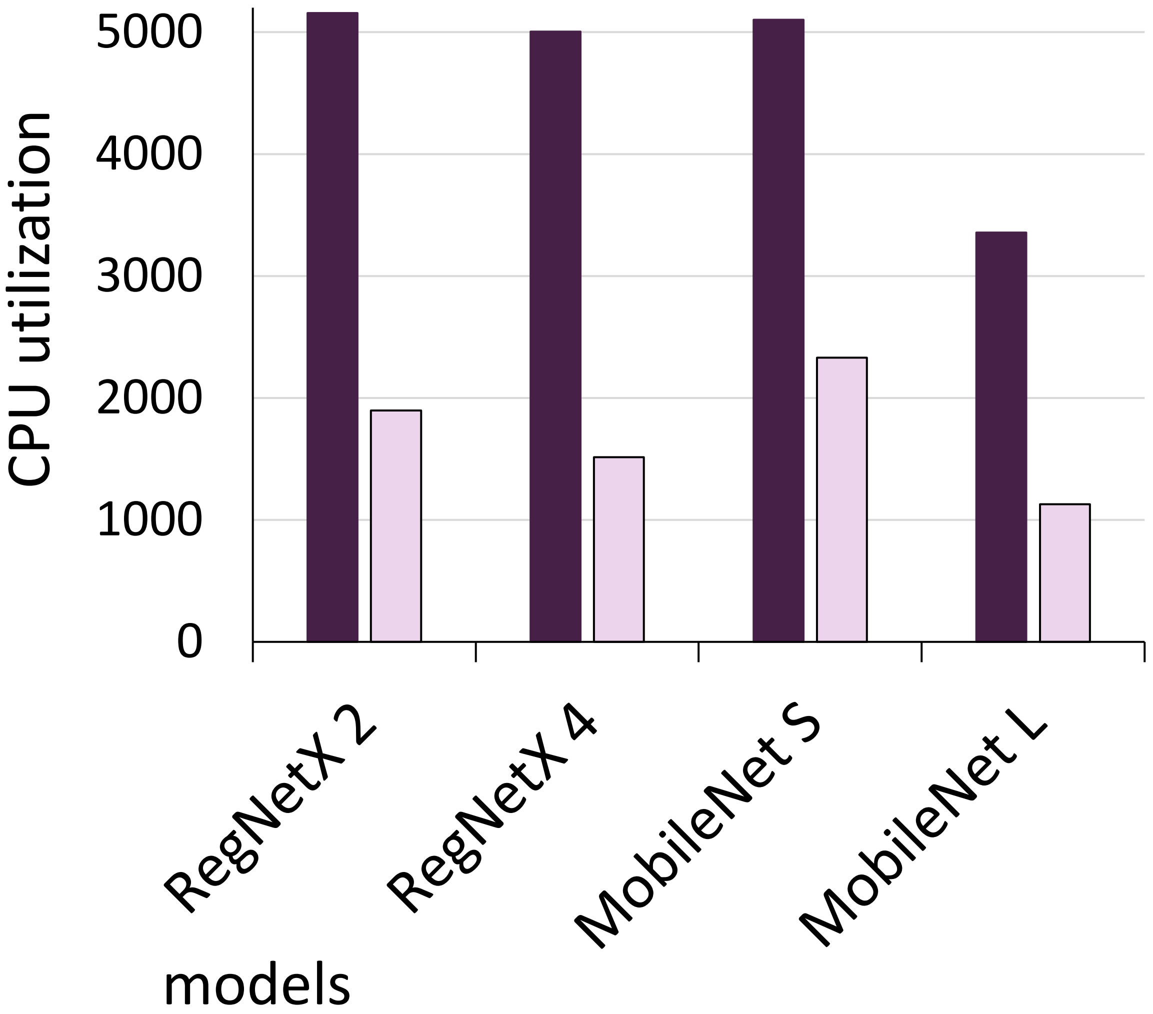
for larger scales, check out *tf.data service*, *CoorDL* …

[SoCC'23] [PVLDB'21]

impact of data sharing

- on PyTorch
- a server with 4 A100 (40GB) GPUs
- one model training on each

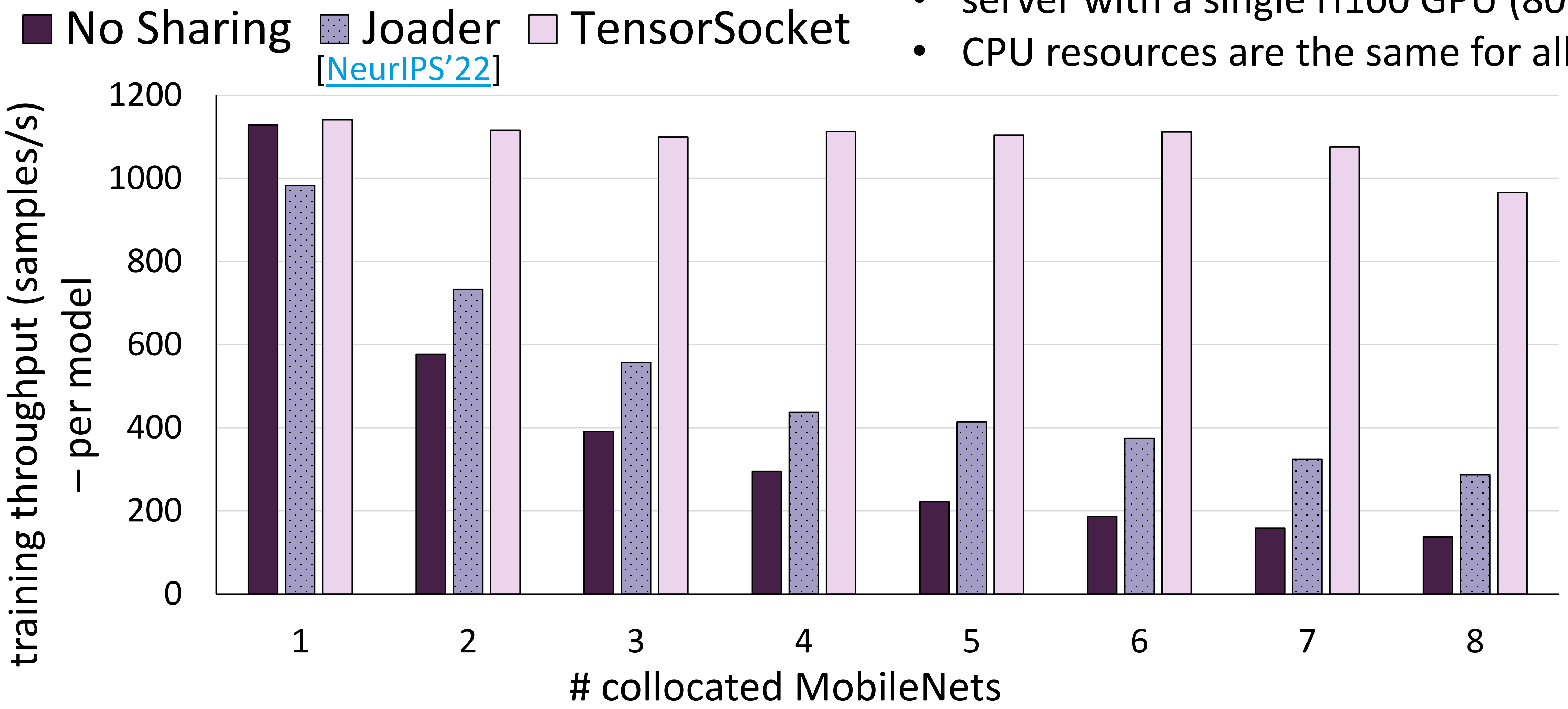




higher overall throughput & reduced CPU need!

comparison to other techniques

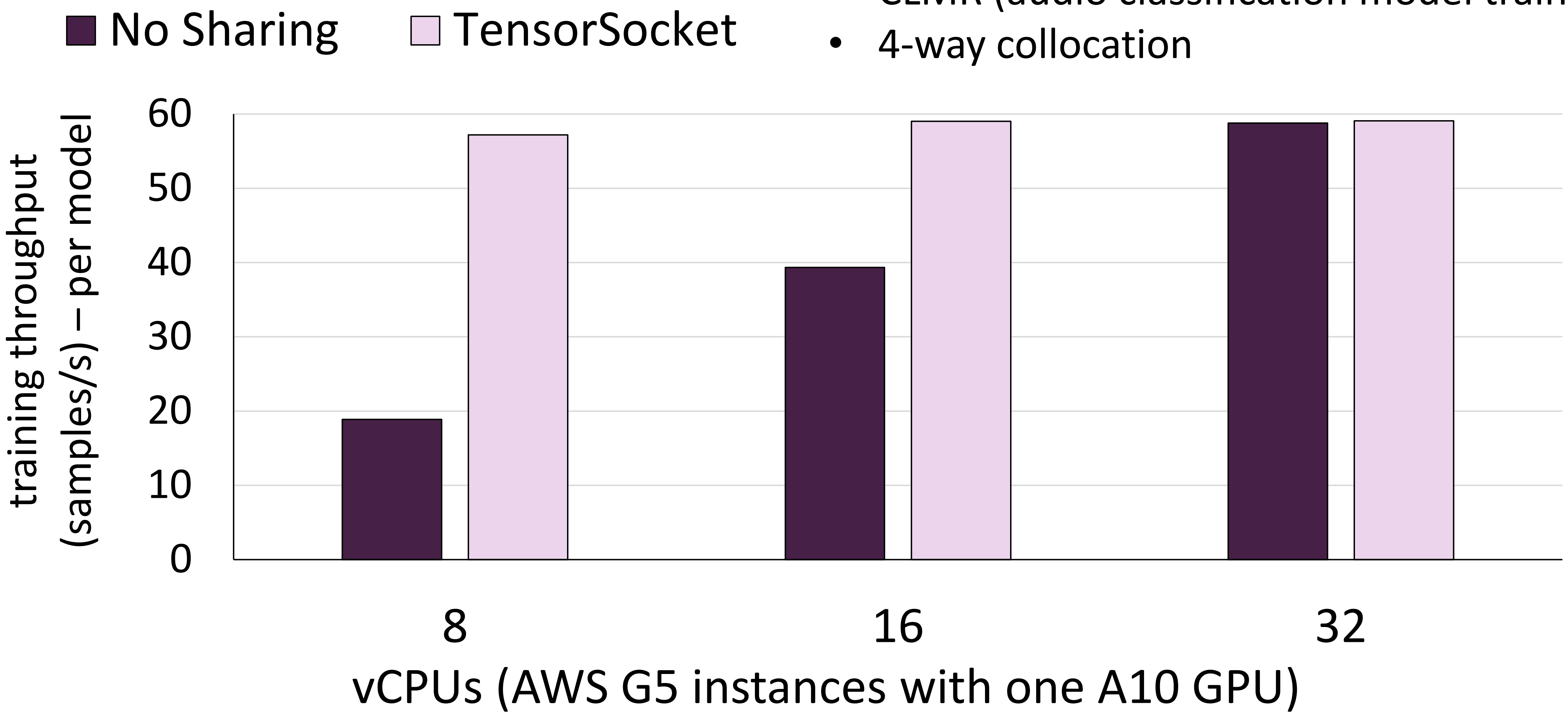
- server with a single H100 GPU (80GB)
- CPU resources are the same for all



TensorSocket sustains throughput even with GPU collocation & reduces both CPU and GPU needs for the whole workload.

cloud cost savings

- CLMR (audio classification model training)
- 4-way collocation



75% less vCPU need for the same training throughput

➔ 50% cost savings on AWS

sharing for deep learning training

- workload collocation allows data & work sharing

- ***TensorSocket*** enables data & work sharing for collocated training jobs on the same dataset.

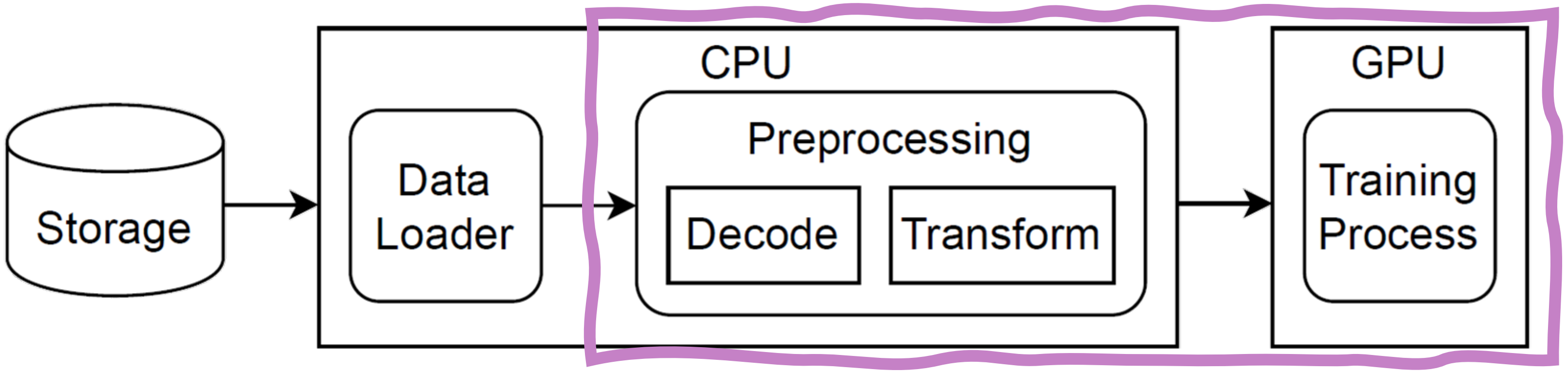
reduces both the CPU & GPU needs (& costs) of training while increasing training throughput!

deep learning with fewer resources

IT UNIVERSITY OF COPENHAGEN

- GPU-centric data path
- data & work sharing
- impact of data selection

Collaboration started at 2024 Dagstuhl seminar: Resource-Efficient Machine Learning.
T. Robroek, M. Böther, ***N. Christiansen, D. Sepehri***, D. Kainmüller, T. Rekatsinas, S. Scherzinger, A. Klimovic, P. Tözün.



why data selection?

- reduce the dataset size
- increase model accuracy
- fine-tuning

➔ **unclear impact on the end-to-end training time!**

➔ **trade-off: computational complexity vs “better” data selection**

preliminary results

base model = Llama-3.2-1B-Instruct
dataset = TruthfulQA

server with a single H100 GPU (80GB)

	duration (mins)	GPU energy use (Wh)	accuracy gain over base model
LESS (25%) [ICML'24]	318	1099	80%
Full	84	417	84%
Random (50%)	43	215	63%
Random (25%)	23	118	39%

if the data selection can be used across different fine-tuning processes, the costs may amortize.

deep learning with fewer resources

- GPU-centric data path

  Path to GPU-Initiated I/O for Data-Intensive Systems
  Karl B. Torp, Simon Lund, Pınar Tözün.
  DaMoN 2025

- data & work sharing

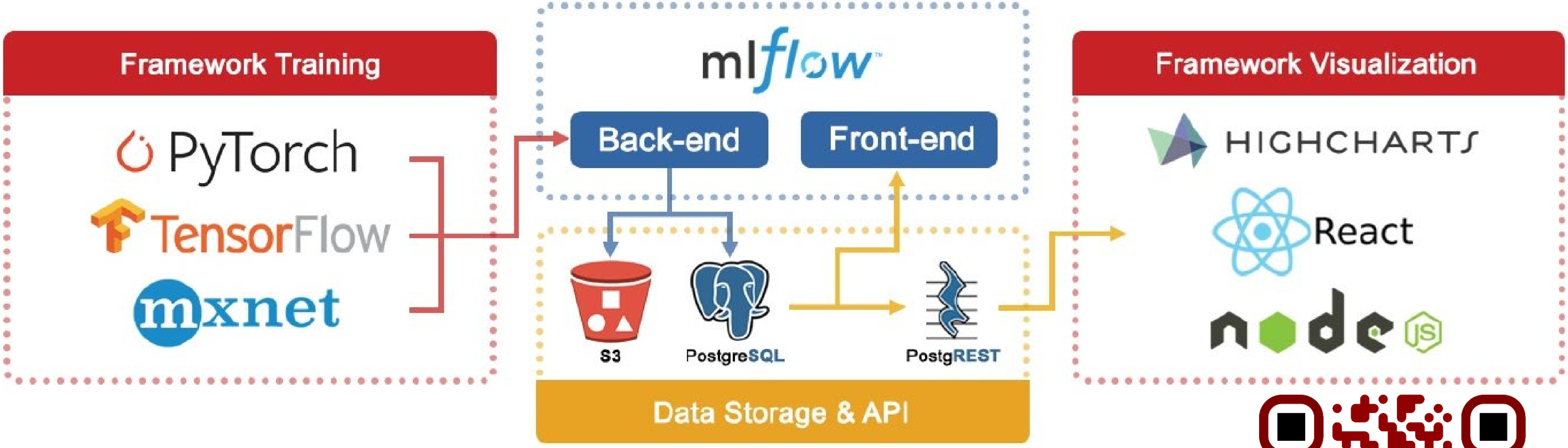
  TensorSocket: Shared Data Loading for Deep Learning Training
  Ties Robroek, Neil Kim Nielsen, Pınar Tözün.
  SIGMOD 2026

- impact of data selection

  Collaboration started at 2024 Dagstuhl seminar:
  Resource-Efficient Machine Learning.
  T. Robroek, M. Böther, N. Christiansen, D. Sepehri, D. Kainmüller,
  T. Rekatsinas, S. Scherzinger, A. Klimovic, P. Tözün.

how to monitor hardware? – radT

[DEEM'23]



- ➔ easy, extensible, and scalable tracking of hardware metrics (GPU utilization, storage access, carbon footprint …)
- ➔ frontend for data exploration

used by our group & data scientists @ITU for systematic benchmarking of deep learning training

RAD - resource-aware data systems

IT UNIVERSITY OF COPENHAGEN

postdocs

Ties
Robroek

Ehsan
Yousefzadeh-
Asl-Miandoab

phd students

Robert
Bayer

Jens Birk
Andersen

collaborators

Pamela Delgado
HES-SO

Tilmann Rabl
HPI

Ana Klimovic
ETH

Julian Priest
ITU

SAMSUNG

deep learning with fewer resources

- GPU-centric data path

Path to GPU-Initiated I/O for Data-Intensive Systems
Karl B. Torp, Simon Lund, Pınar Tözün.
DaMoN 2025

- data & work sharing

TensorSocket: Shared Data Loading for Deep Learning Training
Ties Robroek, Neil Kim Nielsen, Pınar Tözün.
SIGMOD 2026

- impact of data selection

Collaboration started at 2024 Dagstuhl seminar:
Resource-Efficient Machine Learning.
T. Robroek, M. Böther, N. Christiansen, D. Sepehri, D. Kainmüller,
T. Rekatsinas, S. Scherzinger, A. Klimovic, P. Tözün.

thank you!