

Distributed Training

02457 Machine Learning Operations

Nicki Skafte Detlefsen,

Postdoc

DTU Compute

What is distributed computations?



We focus on training as it is the most computationally expensive part of MLOps

Devices



- Three types of devices

- CPU

- General compute unit
 - 2-128 threads

- GPU

- Rendering unit
 - 1000-10000 threads

- TPU

- Specialized unit
 - 8-2048 threads



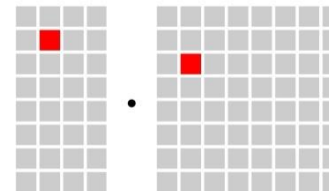
CPU



GPU

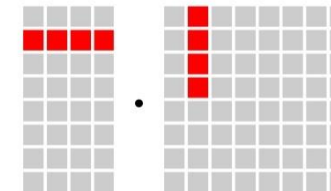


TPU

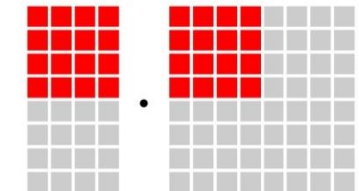


scalar

Compute Primitive



vector



tensor

Note that we are comparing apples to bananas!

Memory



Equally important to what device you are using, is the amount of memory that you have available

With more memory

- Faster data transfer
- Higher data modality
- Larger models

	CPU	GPU	TPU
Standard	32-64 GiB	12 GiB	64 GiB
Maximum	256 GiB	24 GiB	32 TiB

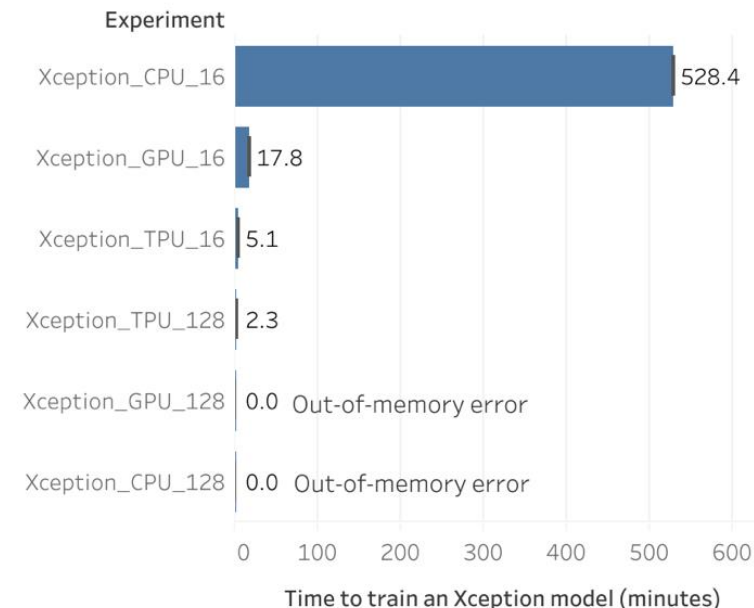
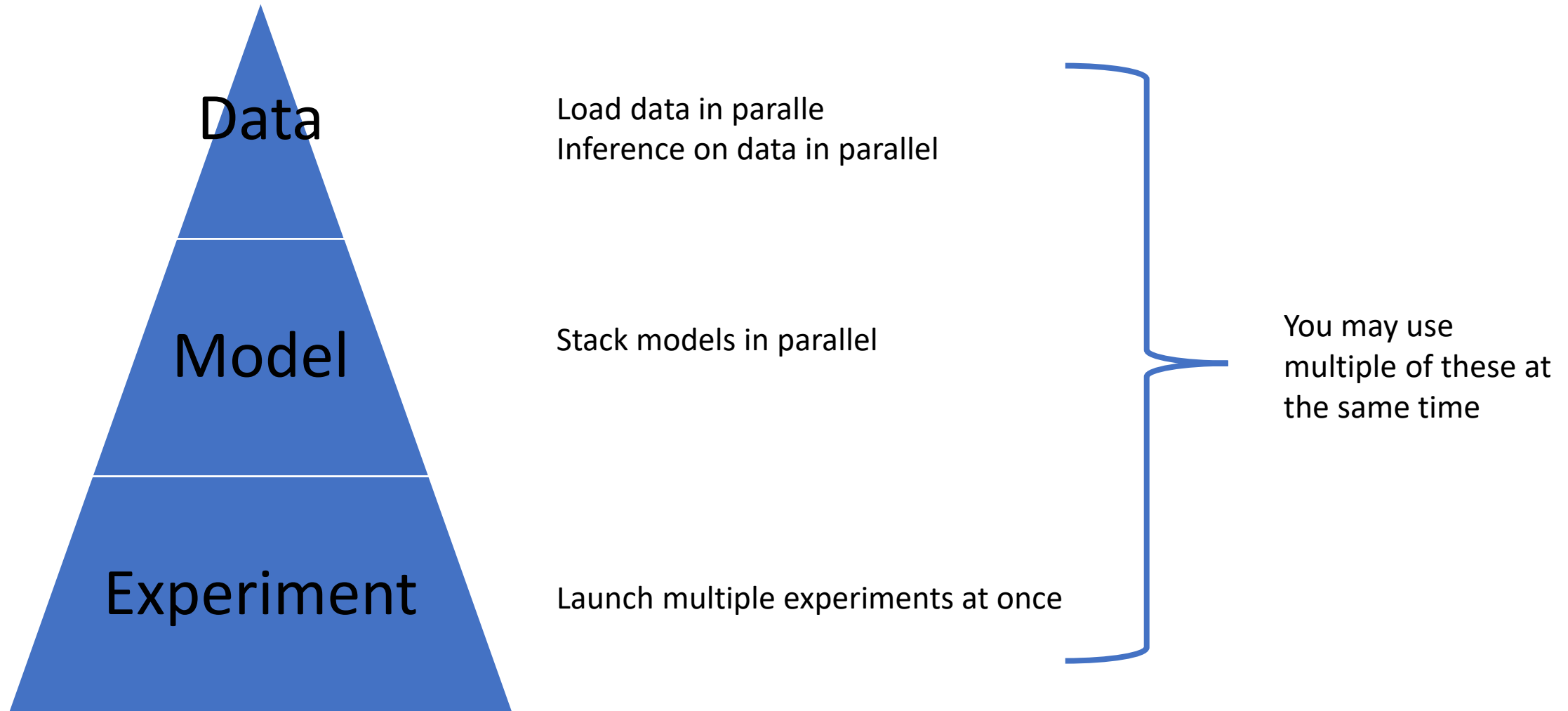


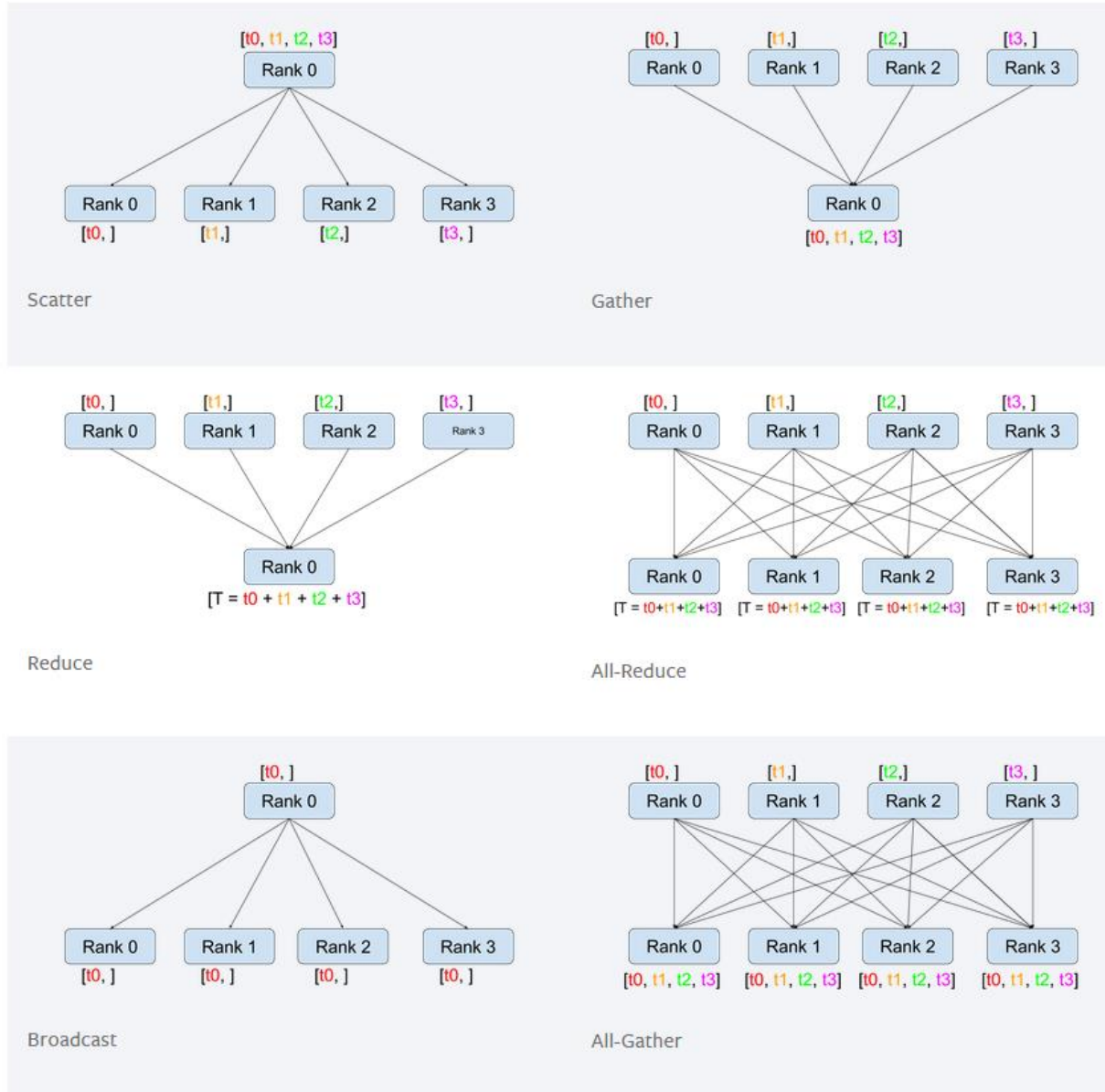
Figure 3: CPUs vs GPUs vs TPUs for training an Xception model for 12 epochs. Y-Axis labels indicate the choice of model, hardware, and batch size for each experiment. Increasing the batch size to 128 for TPUs resulted in an additional ~2x speedup.

<https://towardsdatascience.com/when-to-use-cpus-vs-gpus-vs-tpus-in-a-kaggle-competition-9af708a8c3eb>

Many layers of distributed computations



The six important communication types

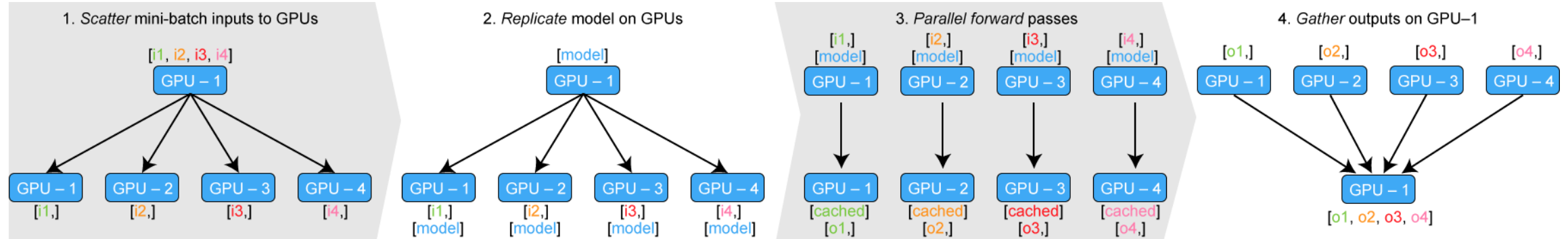


Data Parallel

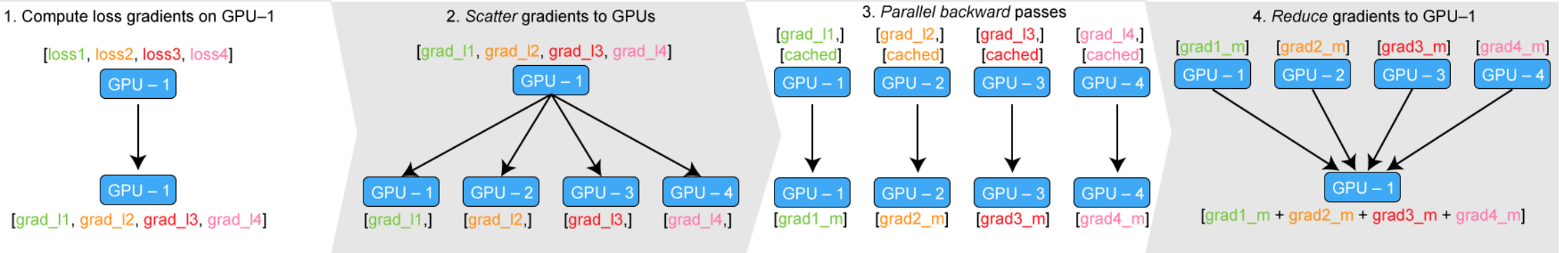


Simple as `parallel_model = torch.nn.DataParallel(model)`

Forward



Backward



Why using a training framework



- Reduce boilerplate = increase turn-around time
- Focus on what is important
- Reproduceability
- Shareability
- Consistency
- Scalability

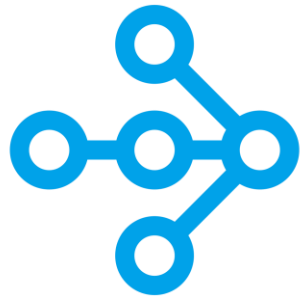
Training Frameworks



Many frameworks exist for reducing boilerplate



Many frameworks for accelerating training



RAY



Seperating engineering and research code

