DAY 4 : OUTLOOK
Advanced Distributed Training

2021-02-04 | Jenia Jitsev | Cross Sectional Team Deep Learning, Helmholtz AI @ JSC

JÜLICH
Forschungszentrum

DISTRIBUTED TRAINING ON VERY LARGE DATA

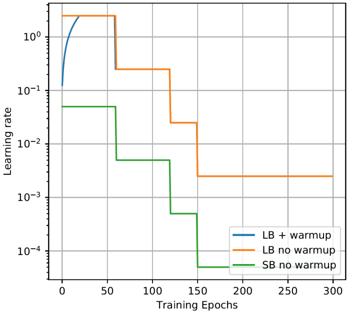
- ImageNet-1k : still test bench for data parallel distributed training
- Substantial speed up with data parallel mode without test accuracy loss
- Requires hyperparameter tuning to adapt training for larger batch sizes

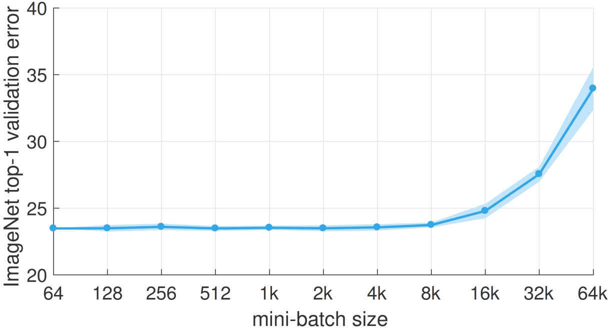
	Hardware	Software	Batch size	Optimizer	# Steps	Time/step	Time	Accuracy
Goyal *et al.* [6]	Tesla P100 $\times$ 256	Caffe2	8,192	SGD	14,076	0.255 s	1 hr	**76.3** %
You *et al.* [8]	KNL $\times$ 2048	Intel Caffe	32,768	SGD	3,519	0.341 s	20 min	75.4 %
Akiba *et al.* [7]	Tesla P100 $\times$ 1024	Chainer	32,768	RMSprop/SGD	3,519	0.255 s	15 min	74.9 %
You *et al.* [8]	KNL $\times$ 2048	Intel Caffe	32,768	SGD	2,503	0.335 s	14 min	74.9 %
Jia *et al.* [9]	Tesla P40 $\times$ 2048	TensorFlow	65,536	SGD	1,800	0.220 s	6.6 min	75.8 %
Ying *et al.* [13]	TPU v3 $\times$ 1024	TensorFlow	32,768	SGD	3,519	**0.037 s**	2.2 min	**76.3** %
Mikami *et al.* [10]	Tesla V100 $\times$ 3456	NNL	55,296	SGD	2,086	0.057 s	2.0 min	75.3 %
Yamazaki *et al.* [11]	Tesla V100 $\times$ 2048	MXNet	81,920	SGD	1,440	0.050 s	**1.2 min**	75.1 %

Osawa et al, 2020

DISTRIBUTED TRAINING ON VERY LARGE DATA

- Requires a package of measures to deal with large batch sizes
  - Learning rate scaling, schedules, warm-up, optimizers, ...
  - Often heuristics for specific scenarios
- ImageNet-1k is getting rusty: Larger, more diverse datasets upcoming
  - **ImageNet-21k**: **14x** larger; **JFT-300M**: **300x** larger, ...
  - able to further increase worker size to train efficiently in data parallel mode?
  - Reminder: data parallel training with $|\mathfrak{B}| = K \cdot |B_{\text{ref}}|$, K workers, large batch sizes

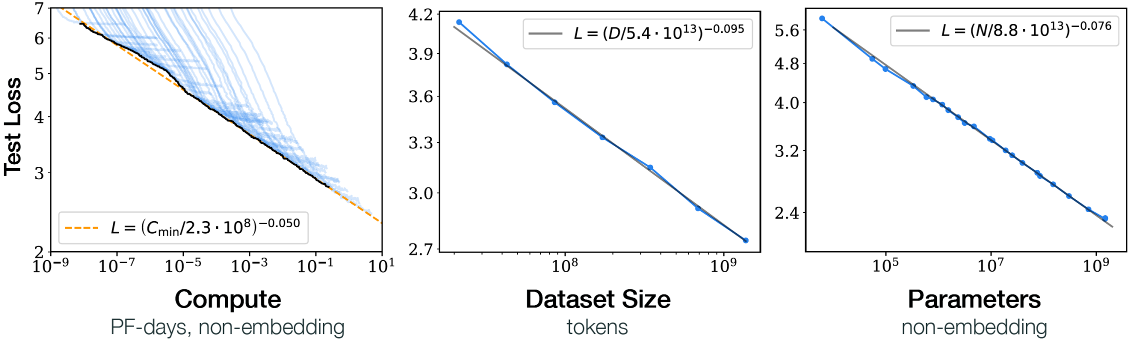




Goyal et al, 2017

DISTRIBUTED TRAINING ON VERY LARGE DATA

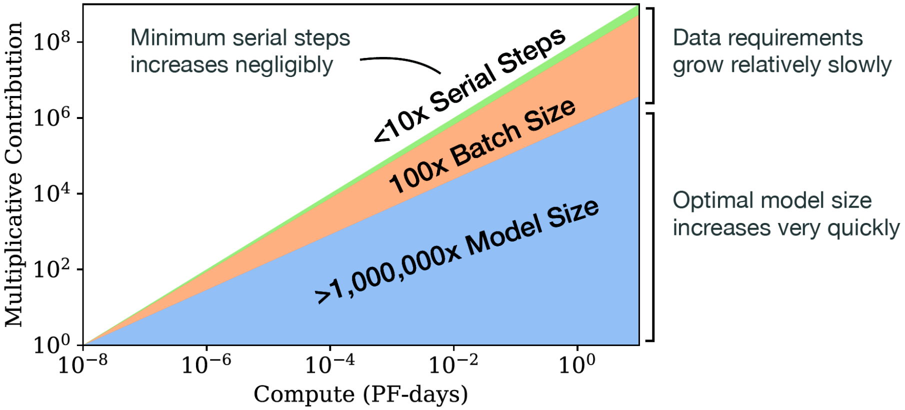
- Scaling Laws: larger models further improve generalization, especially when given enough data and compute
- This seems to be valid across different datasets and training scenarios
  - image, text; unsupervised learning, reinforcement learning



Kaplan et al, 2020

DISTRIBUTED TRAINING ON VERY LARGE DATA

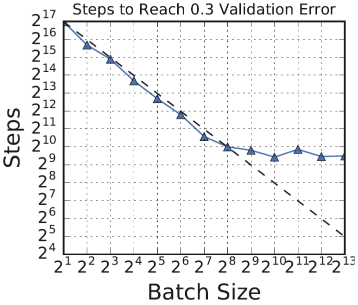
- Scaling Laws: increasing model size requires (modest) increase in data and batch size to achieve better test loss (generalization)
- Increasing batch size : is there a limit?



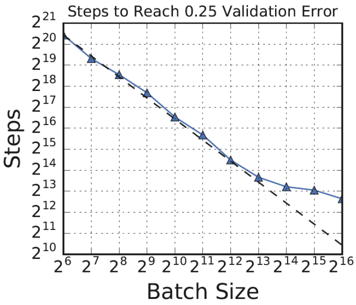
Kaplan et al, 2020

DISTRIBUTED TRAINING ON VERY LARGE DATA

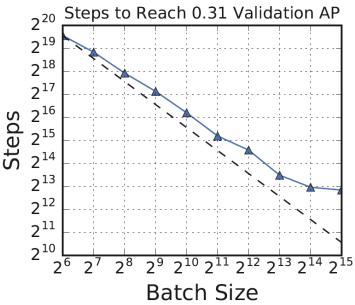
- Critical batch sizes $|\mathfrak{B}_{crit}|$: optimal batch size to train on, almost linear speed-up for **time to accuracy**
  - $|\mathfrak{B}| > |\mathfrak{B}_{crit}|$: diminishing speed up returns, wasting additional compute



ResNet-8, CIFAR-10



ResNet-50, ImageNet-1k

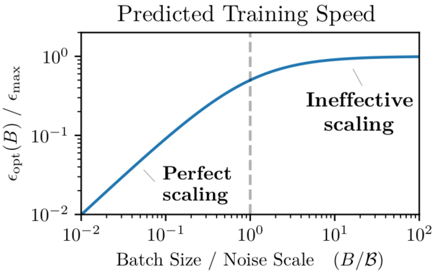


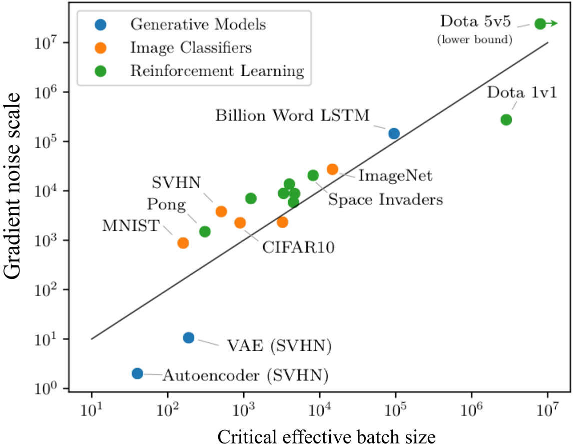
ResNet-50, OpenImages

Shallue et al, JMLR, 2019

DISTRIBUTED TRAINING ON VERY LARGE DATA

- Critical batch sizes $|\mathfrak{B}_{crit}|$: optimal batch size to train on
- $|\mathfrak{B}_{crit}|$ existence across different datasets and training scenarios
  - image, text; unsupervised learning, reinforcement learning
  - measures like gradient noise scale (gradient variance estimate) may provide estimate for $|\mathfrak{B}_{crit}|$

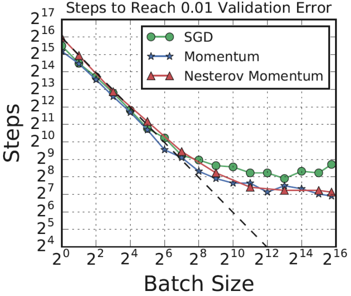




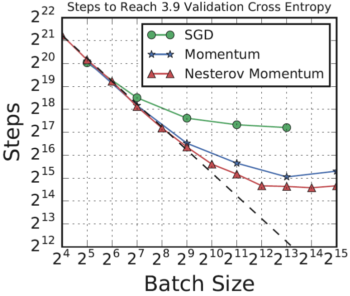
McCandlish et al, 2018

DISTRIBUTED TRAINING ON VERY LARGE DATA

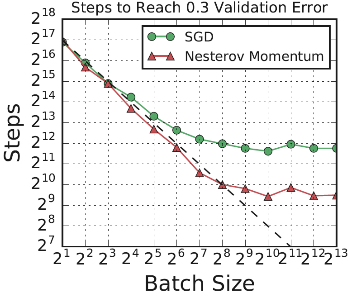
- Critical batch sizes $|\mathfrak{B}_{crit}|$: optimal batch size to train on
- Still debated whether $|\mathfrak{B}_{crit}|$ in turn depends on training hyperparameters



(a) Simple CNN on MNIST



(b) Transformer on LM1B

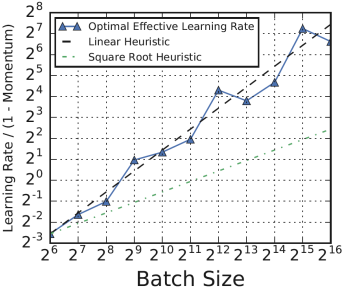


(c) ResNet-8 on CIFAR-10

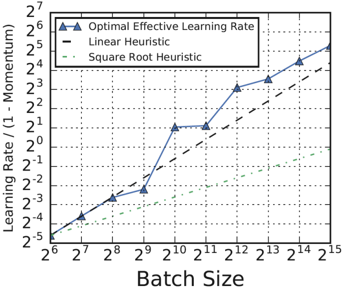
Shallue et al, JMLR, 2019

DISTRIBUTED TRAINING ON VERY LARGE DATA

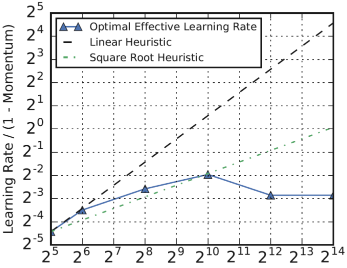
- Large batch sizes $|\mathfrak{B}|$ for efficient data parallel training
- Hyperparameter tuning for each $|\mathfrak{B}|$: no simple scheme for derivation from a reference $|B_{ref}|$ (e.g rescaling)



ResNet-50, ImageNet-1k



ResNet-50, OpenImages

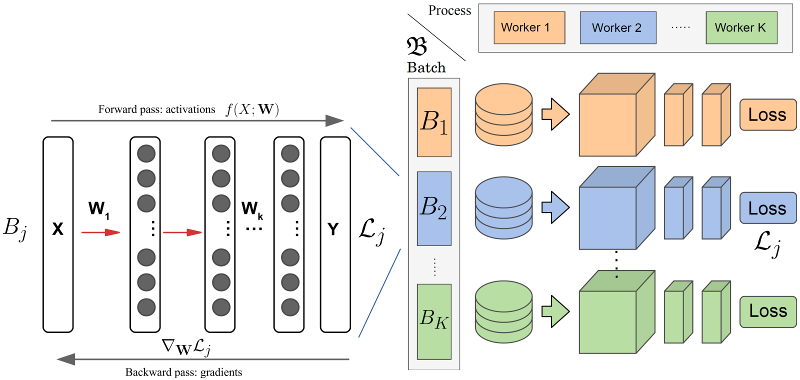


Transformer, Common Crawl

Shallue et al, JMLR, 2019

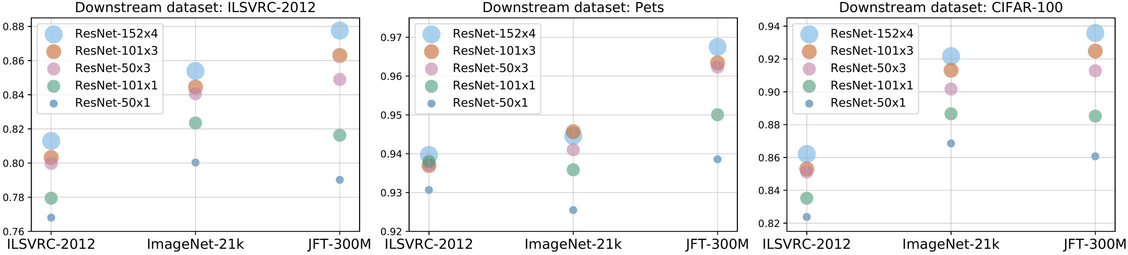
DISTRIBUTED TRAINING ON VERY LARGE DATA

- Alternative data parallel schemes that do not rely on increasing $|\mathfrak{B}|$ with number of workers K
- Local SGD: giving up consistency between model parameters across different workers after each update
  - run local mini-batch SGD without increasing effective global batch size
- Post Local SGD: combining coupled global SGD and decoupled local SGD
  - usual global batch SGD in early training phase, decoupled local SGD with occasional syncing in later phase (Li et al, ICLR, 2020)



DISTRIBUTED TRAINING ON VERY LARGE DATA

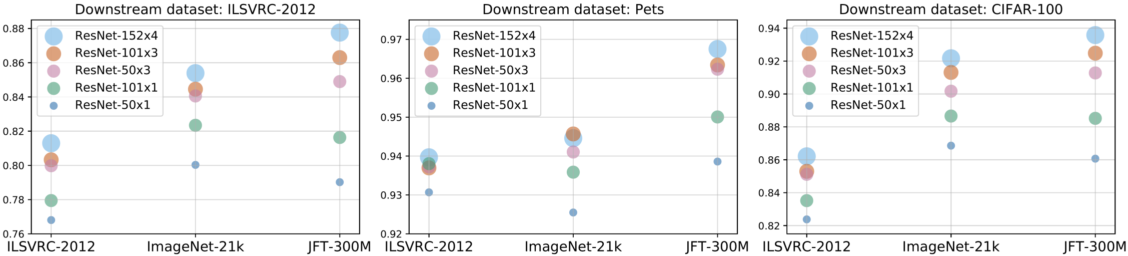
- Growing data: labeled data?
- ImageNet-21k : 21k classes with labels, 14x larger than ImageNet-1k
- JFT-300M : ≈ 18K classes, noisy labels, 300x larger than ImageNet-1k
- Still **supervised** training
  - Evidence for strong transfer learning performance when using large networks



Kolesnikov et al, ECCV, 2020

DISTRIBUTED TRAINING ON VERY LARGE DATA

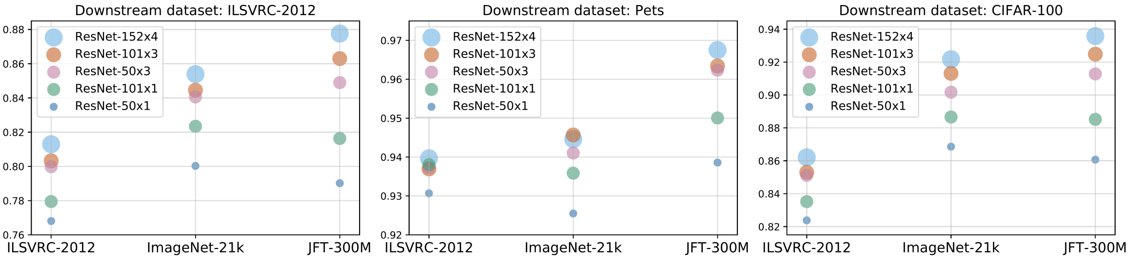
- Growing data: labeled data
- **Supervised** training on very large datasets
  - Evidence for strong transfer learning performance when using large networks
  - Performance increase only evident after **many** epochs - **8 GPU-months** until seeing progress reported! (after 8 GPU weeks - learning seemingly stalled)



Kolesnikov et al, ECCV, 2020

DISTRIBUTED TRAINING ON VERY LARGE DATA

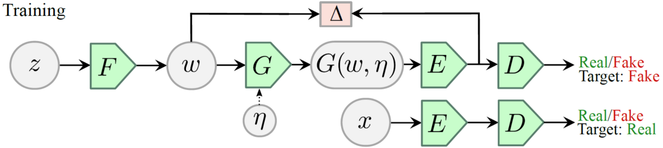
- Growing data: labeled data
- **Supervised** training on very large datasets
  - Performance increase only evident after **many** epochs - **8 GPU-months** until seeing progress
  - Data parallel training: $\approx$ **5.625 hours** on **1024 GPUs** (if scaling goes very well)

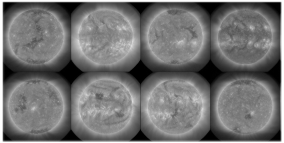


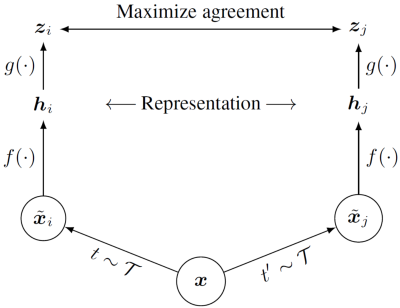
Kolesnikov et al, ECCV, 2020

DISTRIBUTED TRAINING ON VERY LARGE DATA

- Growing data: unlabeled data
- **Unsupervised** learning in different flavors
  - human-made labels not required
- Often, using auxiliary tasks - self-supervised learning
  - contrastive losses (SimCLR), reconstruction based losses (eg VAEs), . . .
  - adversarial losses (eg. GANs -> see Day 5 Special!)







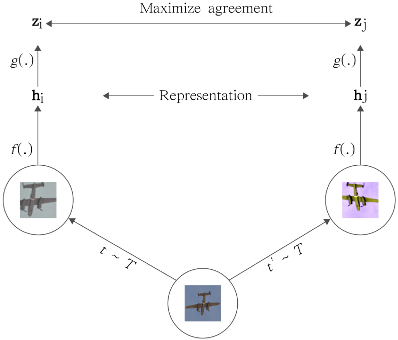
Pidhorskyi et al, 2020; Effenberger et al, 2020; Chen et al, 2020

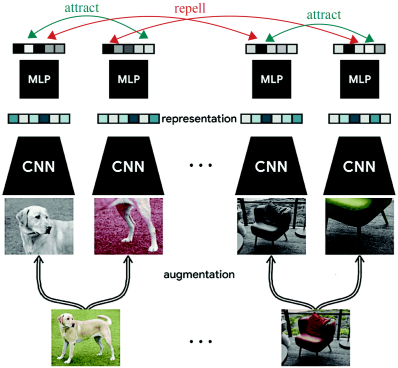
DISTRIBUTED TRAINING ON VERY LARGE DATA

- Growing data: unlabeled data
- Contrastive losses: construct losses from transformed pairs of inputs

$$\mathbf{z}_i = g(\mathbf{h}_i), \quad \mathbf{z}_j = g(\mathbf{h}_j), \quad \mathrm{sim}(\mathbf{z}_i, \mathbf{z}_j) = \frac{\mathbf{z}_i^\top \mathbf{z}_j}{\|\mathbf{z}_i\| \|\mathbf{z}_j\|}$$

$$\mathcal{L}_{i,j} = -\log \frac{\exp(\mathrm{sim}(\mathbf{z}_i, \mathbf{z}_j)/\tau)}{\sum_{k=1}^{2n} \mathbf{1}_{[k \neq i]} \exp(\mathrm{sim}(\mathbf{z}_i, \mathbf{z}_k)/\tau)}$$

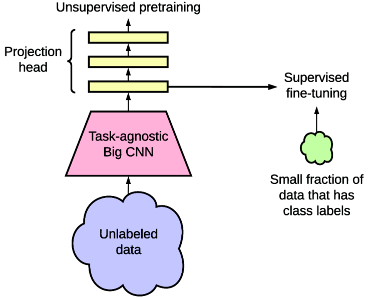


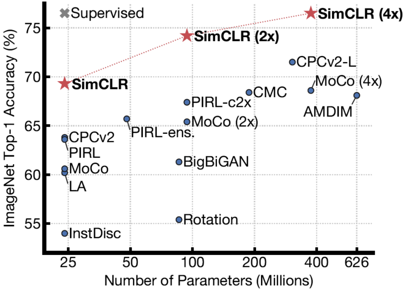


Chen et al, 2020

DISTRIBUTED TRAINING ON VERY LARGE DATA

- Growing data: unlabeled data
- Contrastive losses: larger models do better unsupervised learning!
- Evidence for better representations in larger networks after unsupervised training

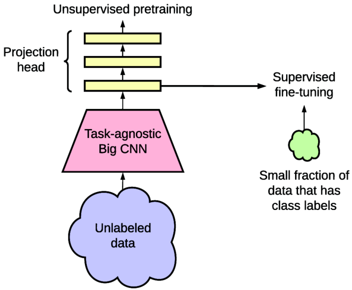


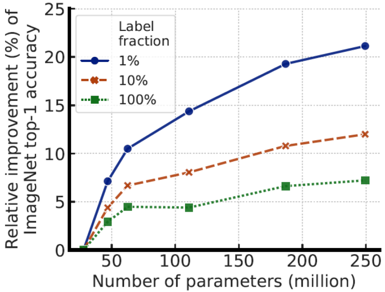


Chen et al, 2020

DISTRIBUTED TRAINING ON VERY LARGE DATA

- Growing data: unlabeled data
- Contrastive losses: larger models do better unsupervised learning!
- Evidence for better transfer learning when using only very few labels
- Single training run: 128 TPU v3, 1.5 hours for a (small) ResNet-50 (25M weights)
  - batch size 4096, 100 Epochs;
  - learning rate rescaling, schedule & LARS optimizer

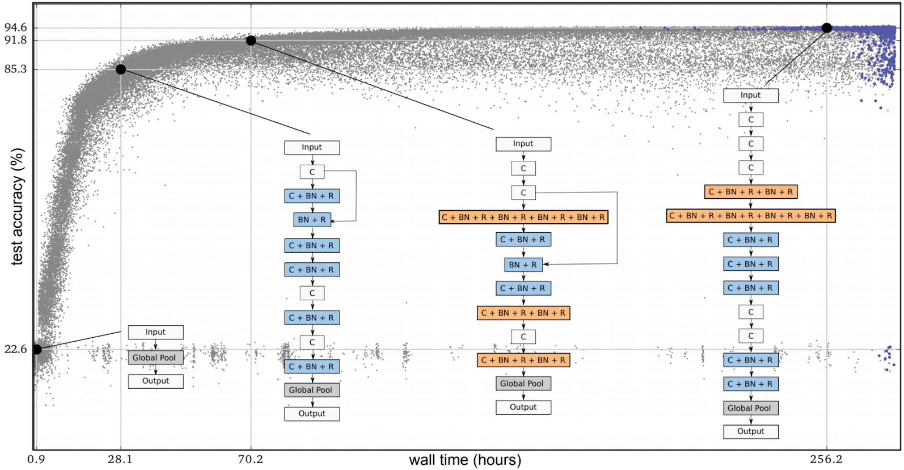




Chen et al, 2020

DISTRIBUTED TRAINING ON VERY LARGE DATA

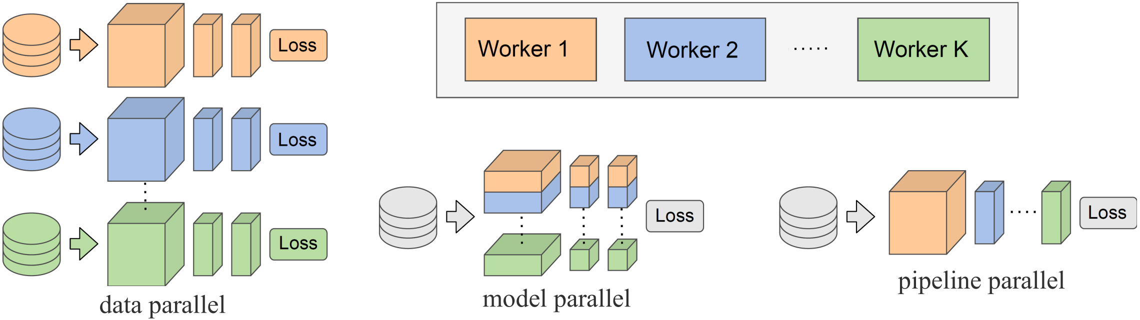
- Neural Architecture Search: training thousands of different networks to find a strong architecture for a (set of) tasks
- May use either supervised or unsupervised training for each candidate network



Real et al, ICML, 2017

DISTRIBUTED TRAINING WITH VERY LARGE MODELS

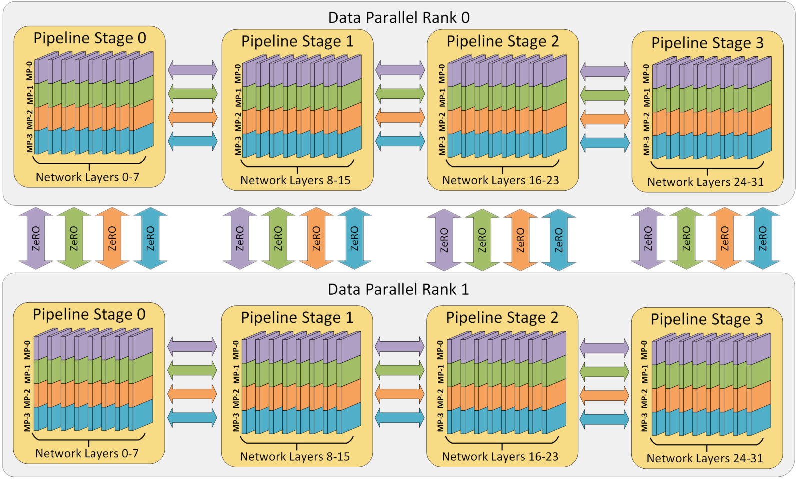
- Growing models: only data parallel scheme not sufficient
  - Language Modelling: GPT 3 - 175 Billion parameters; Switch Transformers (Google) - over 1 Trillion parameters . . .
- Model parallelism, Pipeline Parallelism: can split a very large model across accelerators
- Different libraries: DeepSpeed (Microsoft), HyPar-Flow, Mesh TensorFlow, Tarantella (Fraunhofer), HeAT (Helmholtz - KIT/**JSC**), . . .



Laskin et al, 2020

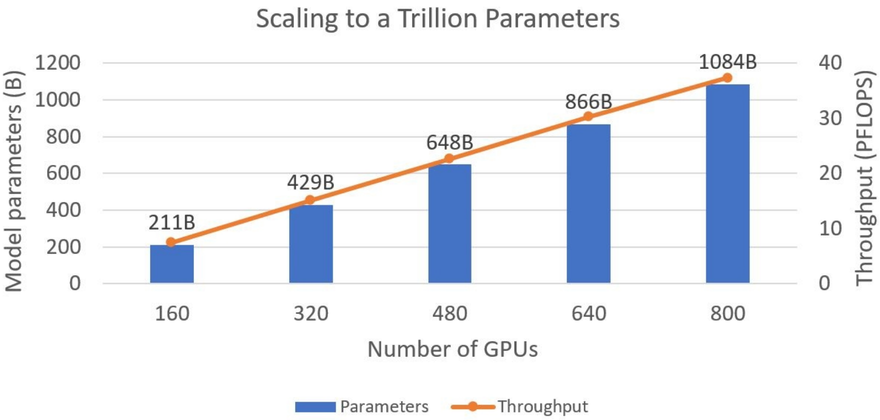
DISTRIBUTED TRAINING WITH VERY LARGE MODELS

- Upcoming: hybrid parallel schemes
  - using data, model and pipeline parallelism simultaneously
- Distributed training that combines memory and compute efficiency
- DeepSpeed: supports hybrid parallelism



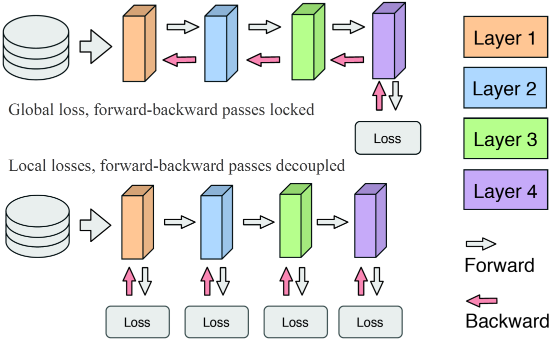
DISTRIBUTED TRAINING WITH VERY LARGE MODELS

- Upcoming: hybrid parallel schemes
  - using data, model and pipeline parallelism simultaneously
- DeepSpeed: “3D Parallelism”
  - executing and speeding up a Trillion size model on 800 A100 GPUs



DISTRIBUTED TRAINING WITH VERY LARGE MODELS

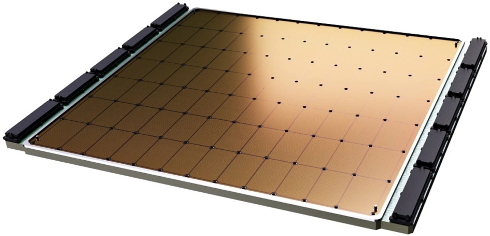
- Upcoming: local updates, decoupled gradients
- Getting rid of global forward-backward pass dependency alltogether
- Asynchronous local updates, highly beneficial for parallelization
- Towards “truly” neuromorphic design, in-memory computing
- New generic losses for unsupervised learning



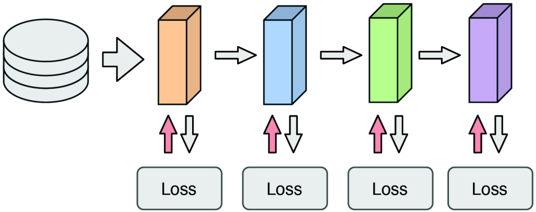
Laskin et al, 2020

DISTRIBUTED TRAINING WITH VERY LARGE MODELS

- Upcoming: local updates, decoupled gradients
- Asynchronous local updates, highly beneficial for parallelization
- Energy efficient distributed training on specialized hardware, in-memory computing
  - Graphcore IPU: Colossus Mk2
  - Cerebras : Wafer Scale Engine 2 (WSE - 850k Cores!)

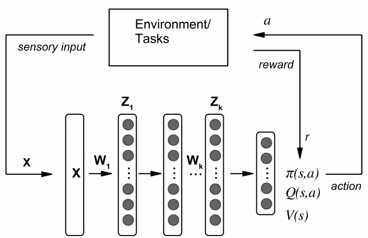


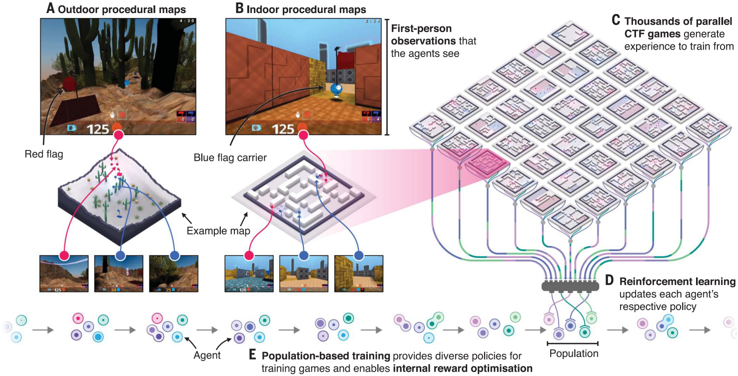
Local losses, forward-backward passes decoupled



DISTRIBUTED TRAINING: BEGINNING OF A JOURNEY

- Large Scale Learning in Simulated Environments
  - **Distributed Reinforcement Learning**: Data Selection and Generation in the Loop
  - **Differentiable simulators** integrated into learning loop - physics-based regularization and learning
- Modular Supercomputing containing different accelerator types
  - Modular Supercomputers are designed at JSC





Jadeberg et al, Science, 2019

DISTRIBUTED TRAINING: BEGINNING OF A JOURNEY

Outlook

- Large-scale distributed training for transfer on smaller datasets
- Large-scale self-supervised learning with auxiliary tasks
- Training of very large models with hybrid parallelism
- Energy efficient large scale learning with neuromorphic hardware
- Distributed reinforcement learning, simulators in the learning loop
- Modular Supercomputers

DISTRIBUTED TRAINING: ACTIVITIES AT JSC

- COVIDNetX: Large-Scale Distributed Training for Transfer Learning
  - Cross-Sectional Team Deep Learning (CST-DL) & Helmholtz AI Consultants Team (HLST)
  - https://tinyurl.com/CovidNetXHelmholtz
- SunGAN: Distributed GAN Training for Generating High Resolution Solar Observations
  - GFZ Potsdam & JSC, CST-DL & Helmholtz AI HLST
- HeAT (Helmholtz Analytics Toolkit): numPy for MPI, large-scale generic tensor computing
  - https://github.com/helmholtz-analytics/heat/
- Modular Supercomputers : JUWELS & JUWELS Booster, more to come

DISTRIBUTED TRAINING: ACTIVITIES AT JSC

- Distributed Training for Hyperspectral Remote Sensing
- Helmholtz AI Research Group at INM-1: distributed deep learning for neuroimaging
- Juelich Data Sets Challenge : Platform for collaborative datasets and model training
  - CST-DL & Helmholtz AI HLST: https://data-challenges.fz-juelich.de/
- TOAR: Earth System Data Exploration (ESDE) Lab
- JULAIN: Juelich Artificial Intelligence Network, join in!
  - mailing list: https://lists.fz-juelich.de/mailman/listinfo/ml