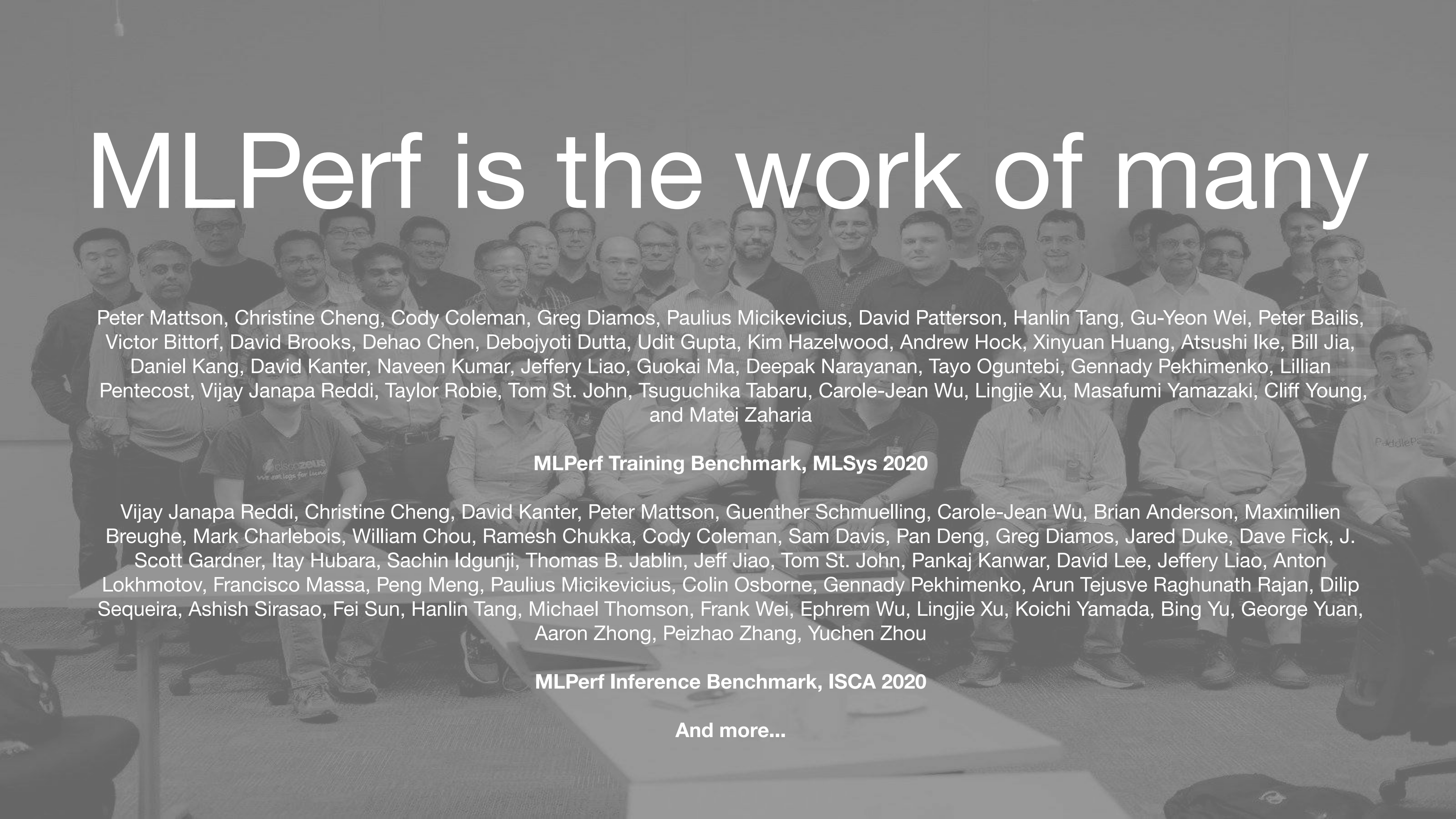
MLPerf™ Training & Inference Benchmarks

Peter Mattson, Google / MLCommons Association President



MLPerf is the work of many

Peter Mattson, Christine Cheng, Cody Coleman, Greg Diamos, Paulius Micikevicius, David Patterson, Hanlin Tang, Gu-Yeon Wei, Peter Bailis, Victor Bittorf, David Brooks, Dehao Chen, Debojyoti Dutta, Udit Gupta, Kim Hazelwood, Andrew Hock, Xinyuan Huang, Atsushi Ike, Bill Jia, Daniel Kang, David Kanter, Naveen Kumar, Jeffery Liao, Guokai Ma, Deepak Narayanan, Tayo Oguntebi, Gennady Pekhimenko, Lillian Pentecost, Vijay Janapa Reddi, Taylor Robie, Tom St. John, Tsuguchika Tabaru, Carole-Jean Wu, Lingjie Xu, Masafumi Yamazaki, Cliff Young, and Matei Zaharia

MLPerf Training Benchmark, MLSys 2020

Vijay Janapa Reddi, Christine Cheng, David Kanter, Peter Mattson, Guenther Schmuelling, Carole-Jean Wu, Brian Anderson, Maximilien Breughe, Mark Charlebois, William Chou, Ramesh Chukka, Cody Coleman, Sam Davis, Pan Deng, Greg Diamos, Jared Duke, Dave Fick, J. Scott Gardner, Itay Hubara, Sachin Idgunji, Thomas B. Jablin, Jeff Jiao, Tom St. John, Pankaj Kanwar, David Lee, Jeffery Liao, Anton Lokhmotov, Francisco Massa, Peng Meng, Paulius Micikevicius, Colin Osborne, Gennady Pekhimenko, Arun Tejusve Raghunath Rajan, Dilip Sequeira, Ashish Sirasao, Fei Sun, Hanlin Tang, Michael Thomson, Frank Wei, Ephrem Wu, Lingjie Xu, Koichi Yamada, Bing Yu, George Yuan, Aaron Zhong, Peizhao Zhang, Yuchen Zhou

MLPerf Inference Benchmark, ISCA 2020

And more...

Why benchmark ML?
(as presented at Hotchips 2019)

Why benchmark machine learning?

ML hardware is projected to be a ~$60B industry in 2025.
(Tractica.com $66.3B, Marketsandmarkets.com: $59.2B)

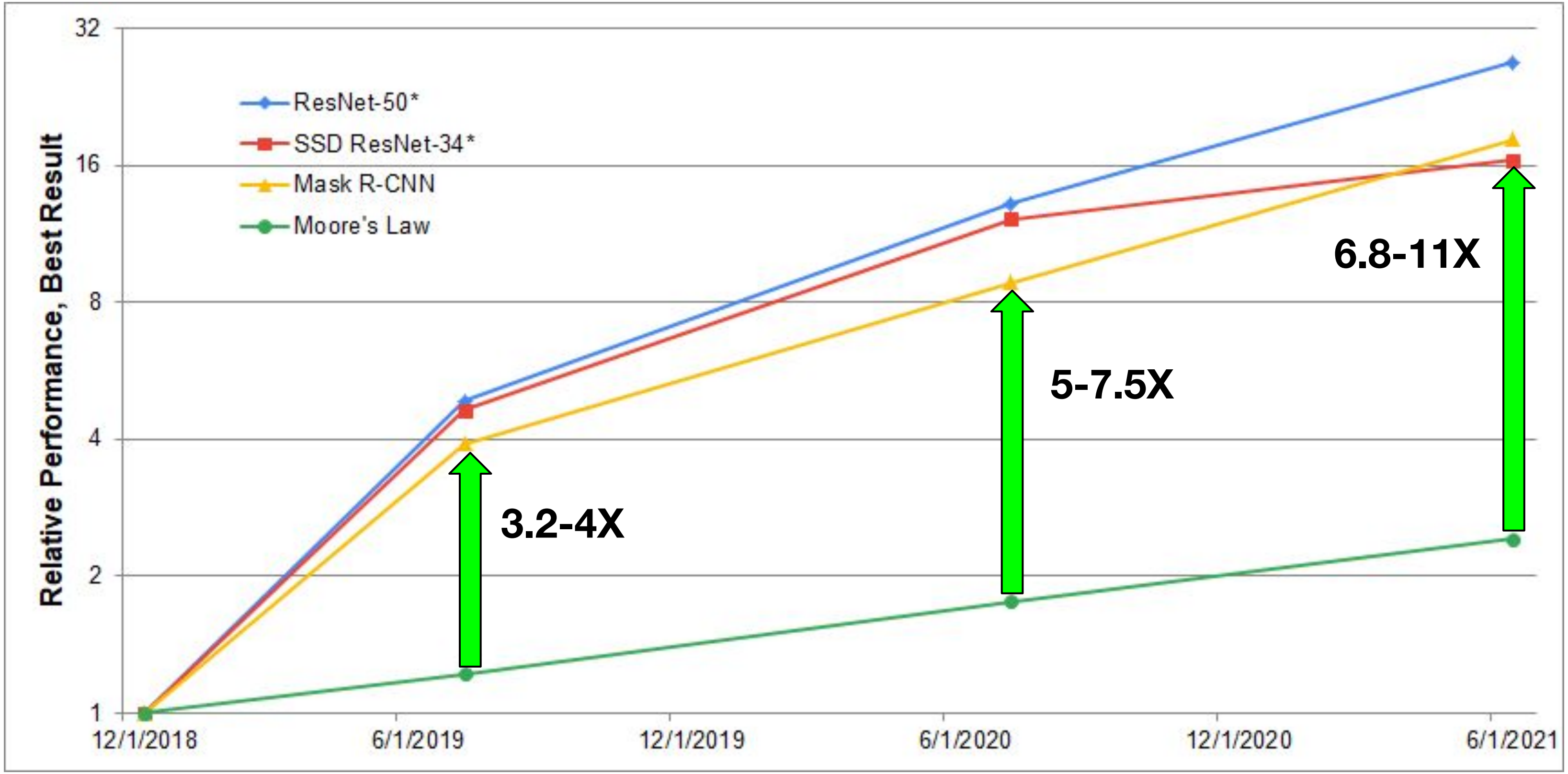
"What get measured, gets improved." — Peter Drucker

Benchmarking aligns research with development, engineering with marketing, and competitors across the industry in pursuit of a clear objective.

MLPerf

Does it work?

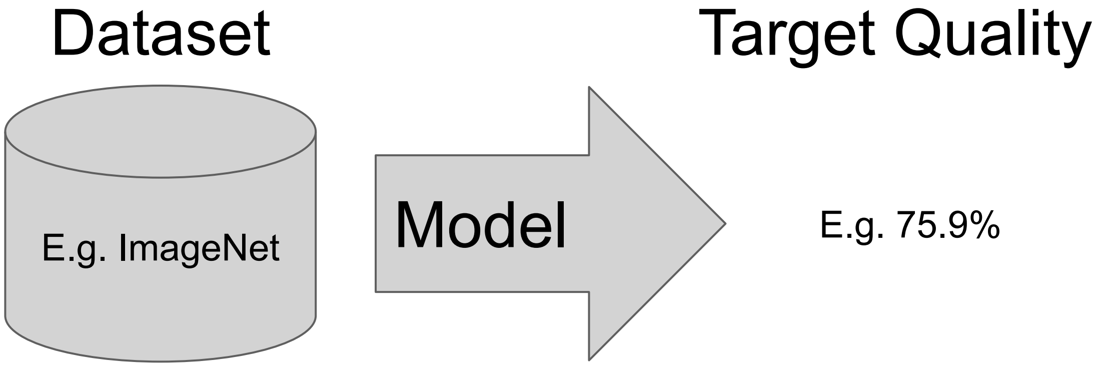
MLPerf™ Training Results Outstrip Moore's Law



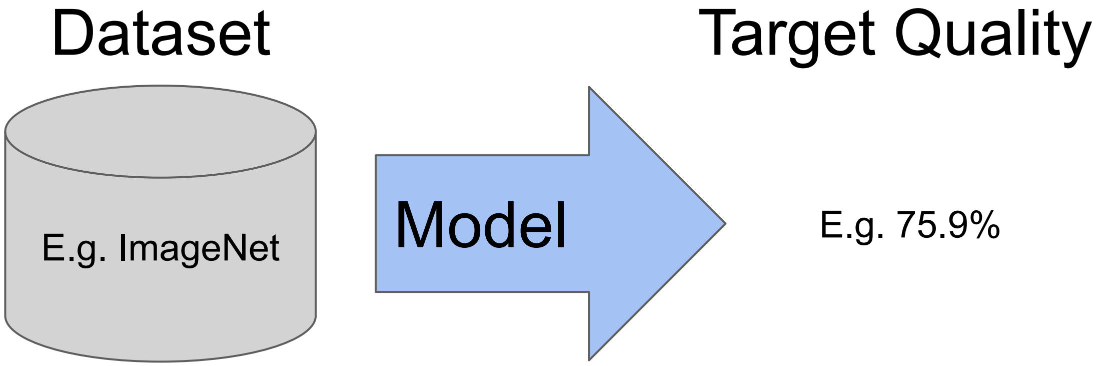
How does it work?

MLPerf Training

MLPerf Training benchmark definition



Two divisions with different model restrictions



Closed division: specific model e.g. ResNet v1.5 → direct comparisons

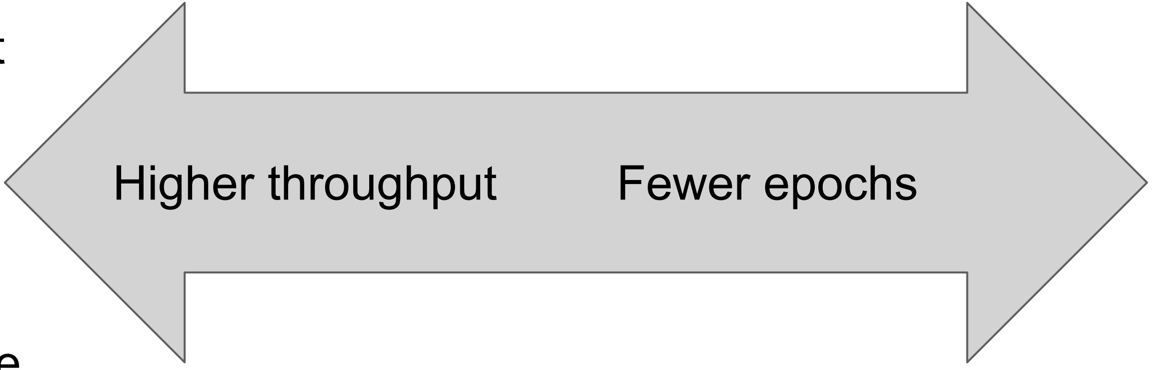
Open division: any model → innovation

Metric: time-to-train

Alternative is throughput
- Easy / cheap to measure

But can increase throughput at cost of total time to train!

Time-to-train (end-to-end)
- Time to solution!
- Computationally expensive
- High variance
- **Least bad choice**



Lower precision
Higher batch size

Higher precision
Lower batch size

Time-to-train excludes

System initialization

Depends on cluster configuration and state

Model initialization

Disproportionate for big systems with small benchmarking datasets

Data reformatting

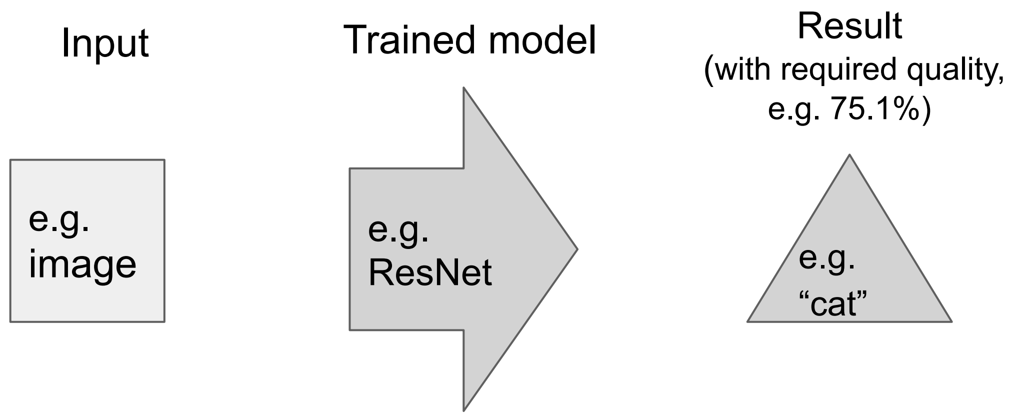
Mandating format would give advantage to some systems

MLPerf v1.0 Training Workloads

Use Case	Neural Network
Vision	ResNet-50 v1.5
	SSD ResNet-34
	Mask R-CNN
	3D UNET
Speech	RNN-T
Language	BERT Large
Commerce	DLRM
Research	Mini-Go

MLPerf Inference

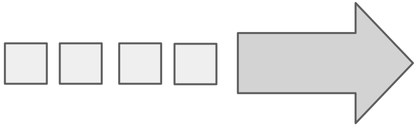
MLPerf inference definition



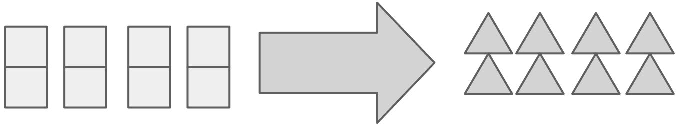
Closed division: specific model e.g. ResNet v1.5 → direct comparisons

Open division: any model → innovation

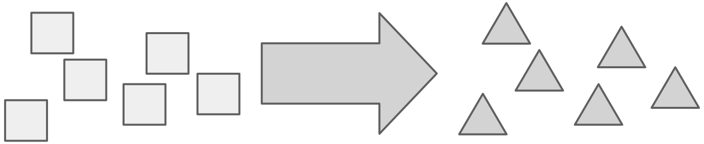
Four **scenarios** to handle different use cases



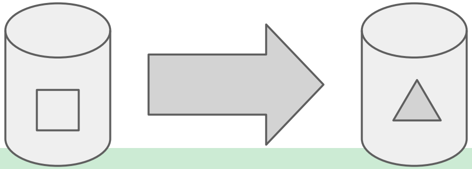
Single stream
(e.g. cell phone augmented vision)



Multiple stream
(e.g. multiple camera driving assistance)

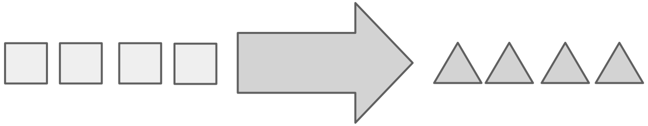


Server
(e.g. translation app)



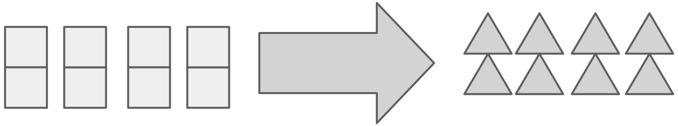
Offline
(e.g. photo sorting app)

Different metric for each scenario



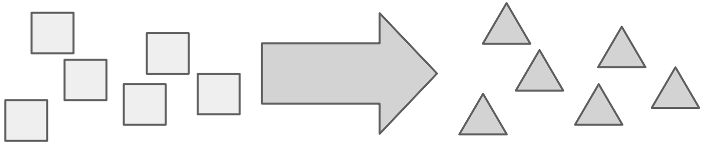
Single stream
e.g. cell phone
augmented vision

Latency



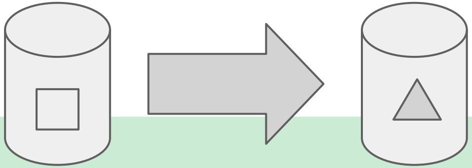
Multiple stream
e.g. multiple camera
driving assistance

Number streams
subject to latency
bound



Server
e.g. translation site

QPS
subject to latency
bound

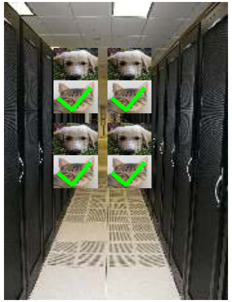


Offline
e.g. photo sorting

Throughput

MLPerf v1.0 Inference Workloads

Datacenter / Edge Inference



Use Case	Neural Network
Vision	ResNet-50 v1.5
	SSD ResNet-34
	SSD MobileNet v1 (edge only)
	3D UNET
Speech	RNN-T
Language	BERT Large
Commerce	DLRM (datacenter only)

Data Center: Offline, Server scenarios
Edge: Single Stream, Offline, Multi stream scenarios

Mobile Inference

Use Case	Neural Network
Vision	MobileNetEdge
	MobileDet
	DeepLabv3
Language	Mobile-BERT

Single Stream, Offline scenarios

Challenges and Contributions

MLPerf Training

ML Training benchmarking challenges

Diverse software stacks and hardware systems	• Can’t use the same executable • Can’t use the same *code*

ML Training benchmarking challenges

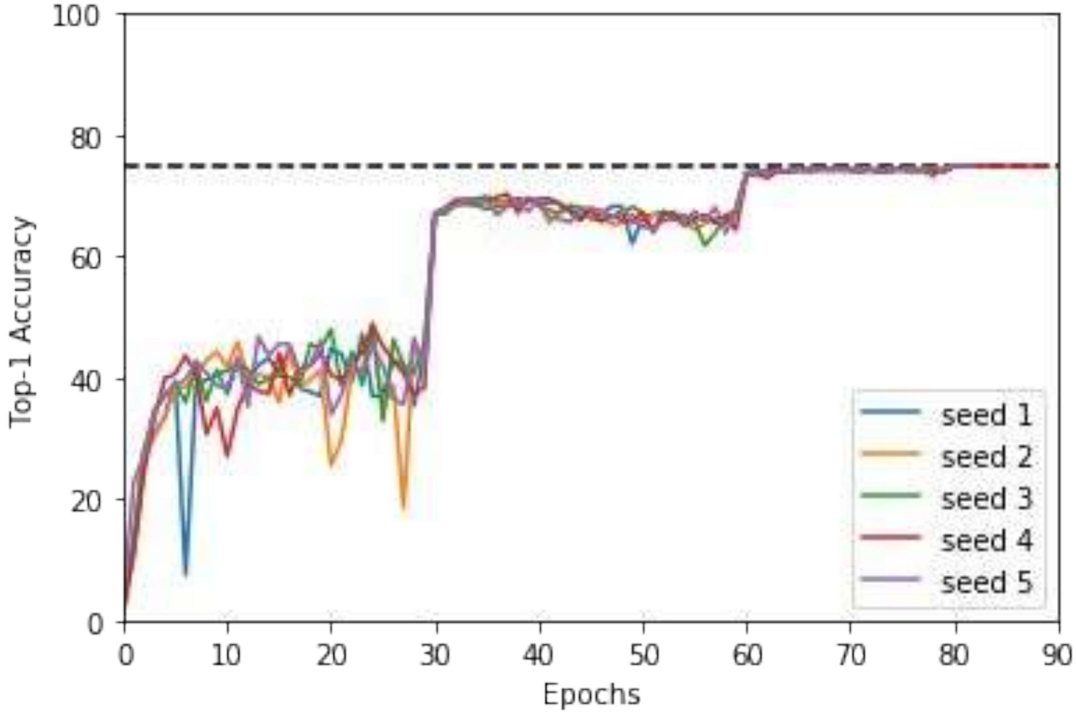
Diverse software stacks and hardware systems	• E.g.: larger systems → larger SGD mini batches → different optimizer hyperparams • Hyperparameter tuning is computationally expensive, can be unfair
Different scales and/or numerics require tuning	

ML Training benchmarking challenges

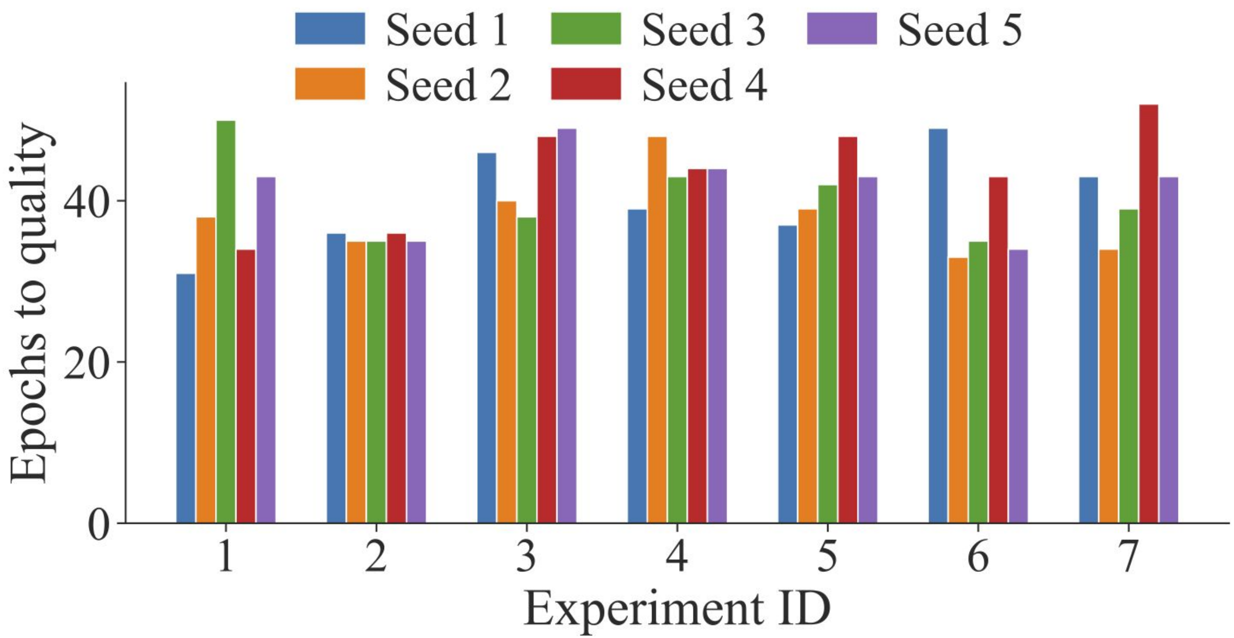
Diverse software stacks and hardware systems	• Random weight initialization • Non-deterministic floating point effects
Different scales and/or numerics require tuning	
Convergence is stochastic	

MLC

Convergence variance: ResNet



Convergence variance: MiniGo



MLPerf contributions

Diverse software stacks and hardware systems	**Reference implementations** **Rules for reimplementation**
Different scales and/or numerics require tuning	
Convergence is stochastic	

ML
C

MLPerf contributions

Diverse software stacks and hardware systems	Reference implementations Rules for reimplementation
Different scales and/or numerics require tuning	**Limited tunable hyperparameters; limited values**
Convergence is stochastic	

MLPerf contributions

Diverse software stacks and hardware systems	Reference implementations Rules for reimplementation
Different scales and/or numerics require tuning	Limited tunable hyperparameters; limited values
Convergence is stochastic	**Require multiple runs** **Drop low and high, average**

MLPerf Inference

MLPerf Inference challenges

Even more diverse software stacks / hardware systems	● Can't use the same executable ● Can't use the same *code*

ML
•C

MLPerf Inference challenges

Even more diverse software stacks / hardware systems

Different approaches to quantization

- Quantization is used in practice
- Don’t want a quantization algorithm contest

MLPerf Inference challenges

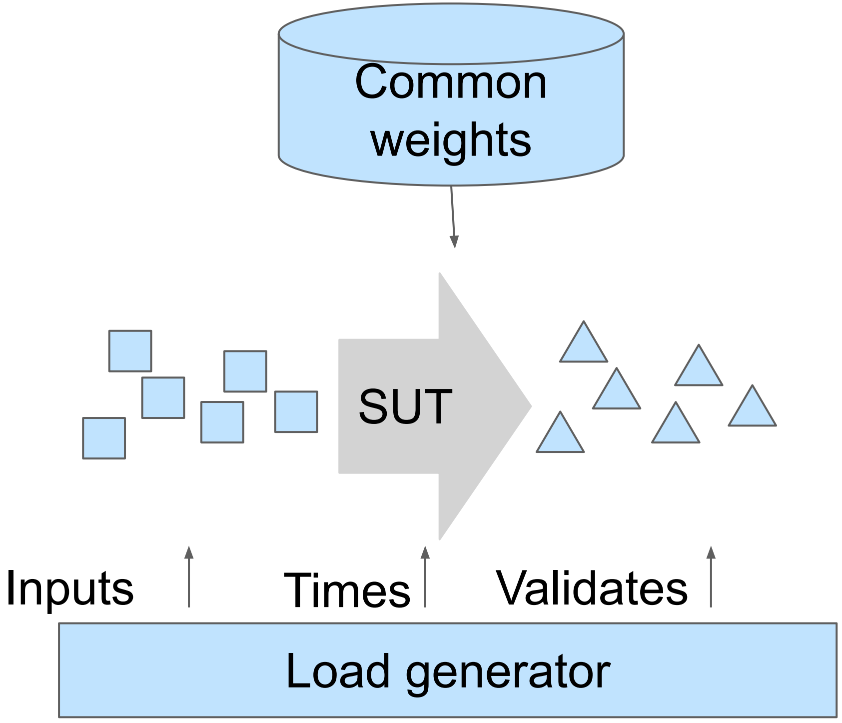
Even more diverse software stacks / hardware systems	• Infinitely scalable • Need to normalize for meaningful comparison • Chips, list price, TCO, TDP, power?
Different approaches to quantization	
Infinitely parallel	

MLPerf contributions

Even more diverse software stacks / hardware systems	**Reference implementations** **Rules for reimplementation**
Different approaches to quantization	
Infinitely parallel	

Additional constraints to ensure equivalence

- Must use standard set of **pre-trained weights for Closed Division**
- Must use **standard C++ "load generator"** that handles scenarios and metrics



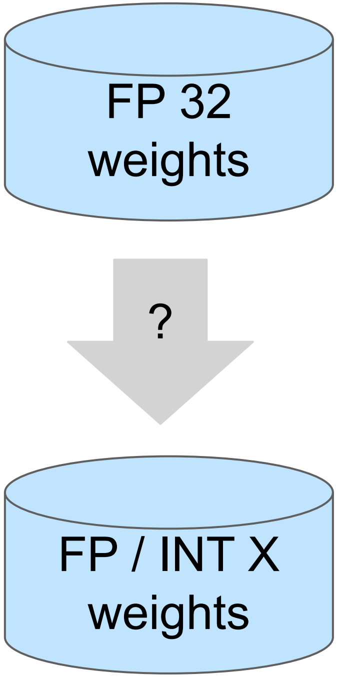
MLPerf contributions

Even more diverse software stacks / hardware systems	Reference implementations Rules for reimplementation
Different approaches to quantization	**Rules for limited quantization**
Infinitely parallel	

ML
C

Quantization allowed with constraints

- Quantization is key to efficient inference, but do not want a quantization contest
- Can the Closed division **quantize**?
  - **Yes**, but must be principled: describe reproducible method
- Can the Closed division **calibrate**?
  - **Yes**, but must use a fixed set of calibration data
- Can the Closed division **retrain**?
  - **No**, not a retraining contest. But, provide retrained 8 bit models..



MLPerf contributions

Diverse software stacks and hardware systems	Reference implementations Rules for reimplementation
Different approaches to quantization	Limited tunable hyperparameters; limited values
Infinitely parallel	**Usage dependent; left to result user!**

Submission Process

Pre-submit

Download **reference implementation**, read rules, join submitters working group

↓

Reimplement benchmark for system under test (SUT)

↓

For Training: tune hyperparameters (allowed by list, to allowed values)

↓

Run benchmark required number of times

↓

Submit logs from all runs, code, metadata in Github by deadline

↓

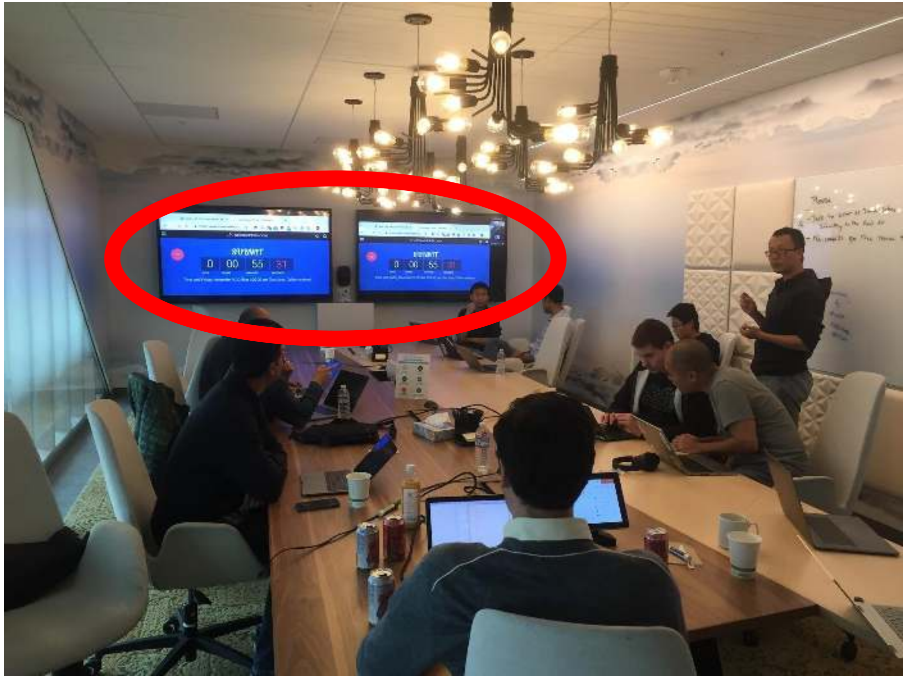
Post-submit

All submitters **peer review** all submissions, raise issues

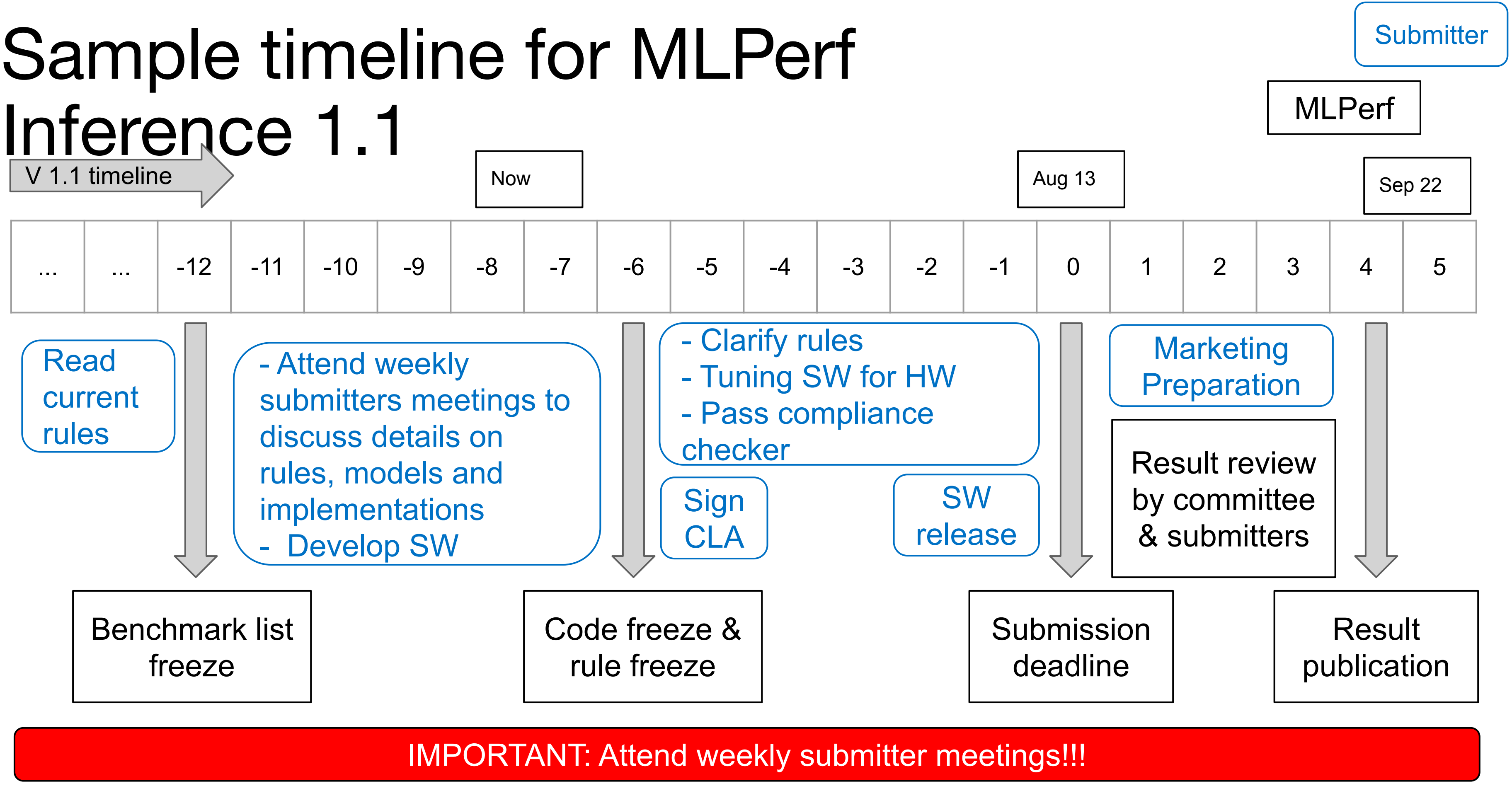
For training: borrow hyperparameters from other submissions and resubmit if desired

MLPerf posts all results and
makes logs, metadata, and code public under Apache-2

Celebrate!!!



Sample timeline for MLPerf Inference 1.1



Recent Developments / Future Work

MLPerf continues to grow and evolve

- Growing number of submitters each round
  - Training 1.0 had 13 submitting orgs
  - Inference 1.0 had 17 submitting orgs

- Improving benchmarking technology
  - E.g. MLPerf Training developed “reference convergence points (RCP)” methodology to verify equivalent convergence behavior

- New benchmarks
  - 3D medical imaging (3D-UNET)
  - Speech-to-text (RNN-T)

Advisory boards

- Why form advisory boards?
  - Enable practitioner ML users to define ML benchmarks
  - Ensure MLPerf benchmarks reflect real use cases
  - Avoid submitter bias - advisors not affiliated with submitters

Advisory boards

2019 **2020**

Recommendation
DLRM with the Terabyte CTR dataset
Inference Benchmark
- Submission results on v0.7 (Oct. 2020) → v1.0 (April 2021)
- **24 → 31 system submissions over *CPUs*, *GPUs*, and *FPGAs***

Medical Imaging
3DUNet with the BraTS dataset
Inference Benchmark
- Submission results on v0.7 (Oct. 2020)
- 20 → 33 system submissions (datacenter) & 11 → 18 system submissions (edge)

2021

Automotive
Data sets/model architectures → 3D Object Detection; Inputs from cameras and LIDAR; Waymo Open Dataset

Vision
Vision tasks on the horizon
Data sets/model architectures
- Recommendations to be adopted for v2.0

NLP/Speech
Board formation
- 18 nominations
- 7 candidates with expertise in DL+NLP/Speech

How to get involved?

https://mlcommons.org/en/get-involved/

Q&A