ML Benchmark Design Challenges

Peter Mattson, General Chair MLPerf
petermattson@google.com

(Work by many people in MLPerf community)

Hot Chips 2019

Agenda

Introduction

Design

Training

Inference

Presentation

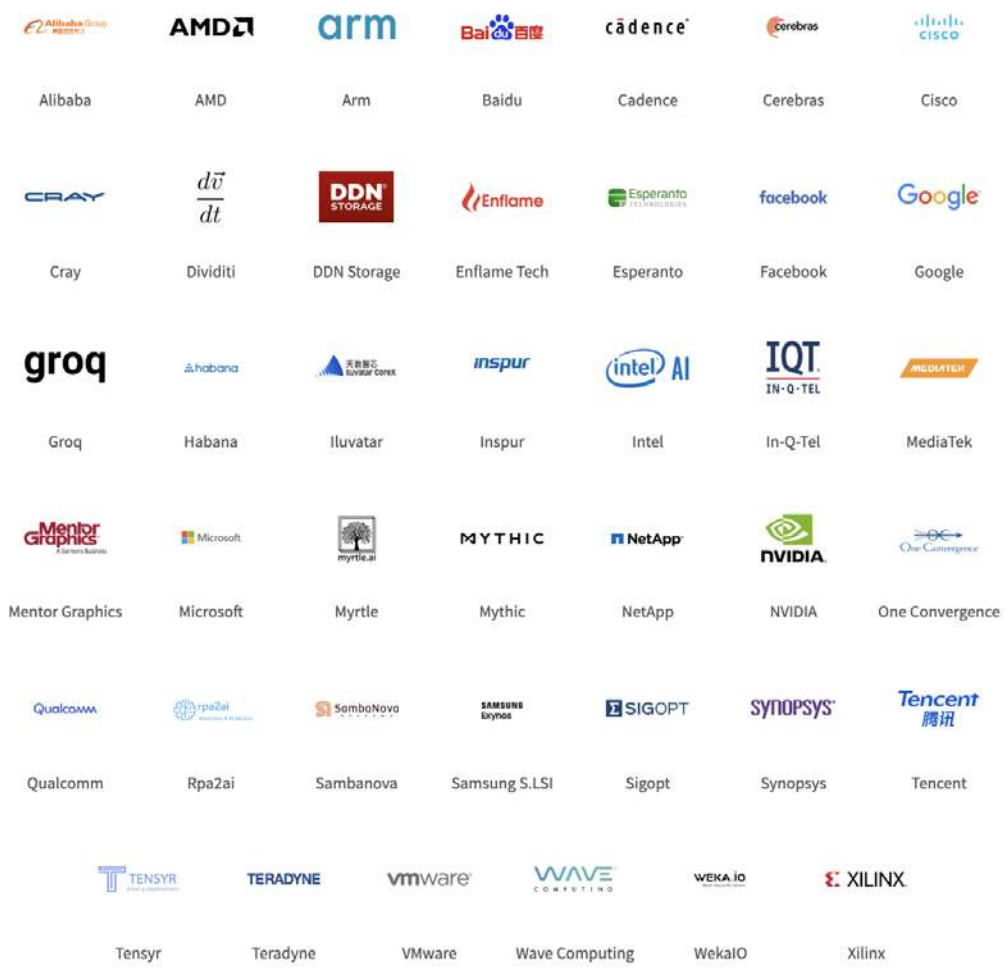
Results

Future plans

MLPerf

What is MLPerf?

A machine learning performance benchmark suite with broad industry and academic support.



MLPerf is the work of many

Founding leads: Peter Bailis (Stanford), Greg Diamos (Baidu), Peter Mattson (Google), David Patterson (UC Berkeley / Google), Gu-Yeon Wei (Harvard), Matei Zaharia (Stanford)

Training chairs: Victor Bittorf (Google), Paulius Micikevicius (NVIDIA), Andy Hock (Cerebras)

Inference chairs: Christine Cheng (Intel), David Kanter (RWI), Vijay Reddi (Harvard), Carole-Jean Wu (Facebook), Guenther Schmuelling (Microsoft), Hanlin Tang (Intel), Bing Yu (MediaTek)

Many others see **mlperf.org/about**

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

ML benchmark design overview

Big Questions	Training	Inference

Agenda

Introduction

Design

Training

Inference

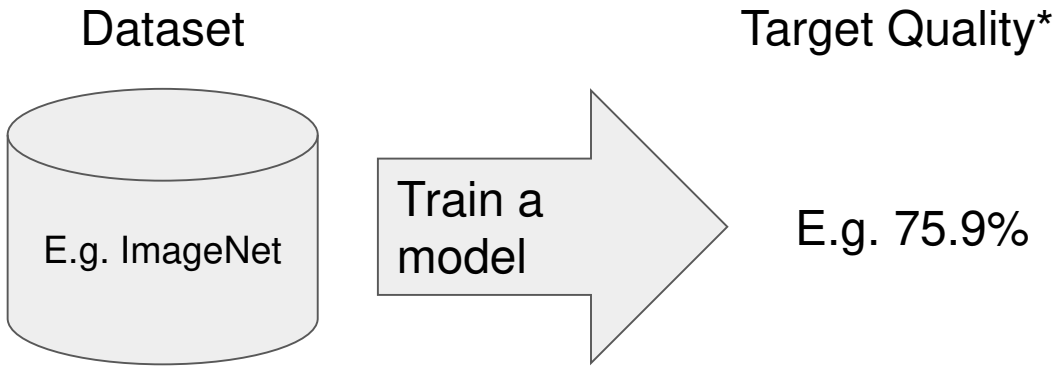
Presentation

Results

Future plans

MLPerf

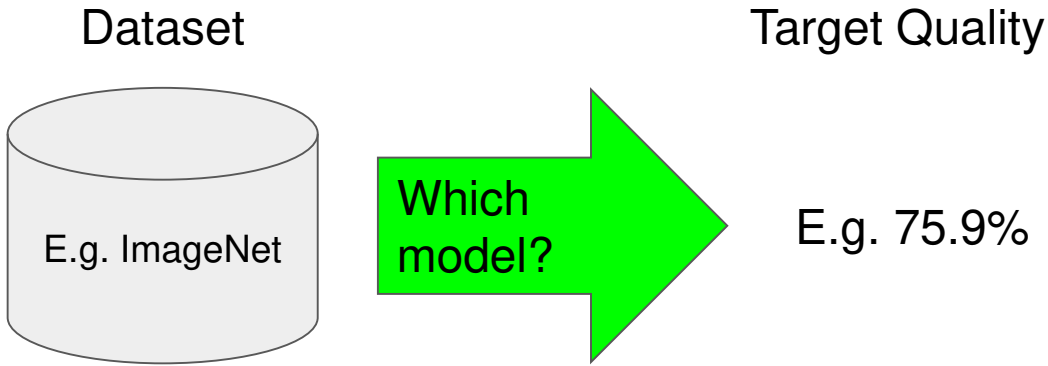
Training benchmark definition



* Target quality set by experts in area, raised as SOTA improves

MLPerf

Do we specify the model?



Choice: two divisions
Closed division: model is specified
Open division: model is not specified

Training benchmark selection

Training closed division model selection

Model Range	Example	Principle

Training v0.5, v0.6 benchmark selection

Area	Problem	Dataset	Model
Vision	Image recognition	ImageNet	ResNet
	Object detection	COCO	SSD
	Object segmentation	COCO	Mask R CNN
Language	Translation	WMT Eng.-German	NMT
	Translation	WMT Eng.-German	Transformer
Commerce	Recommendation	Movielens-20M	NCF
Other	Go	n/a	Mini go

Also driven by availability of data and readiness of code.
Need to broaden, evolve.

MLPerf

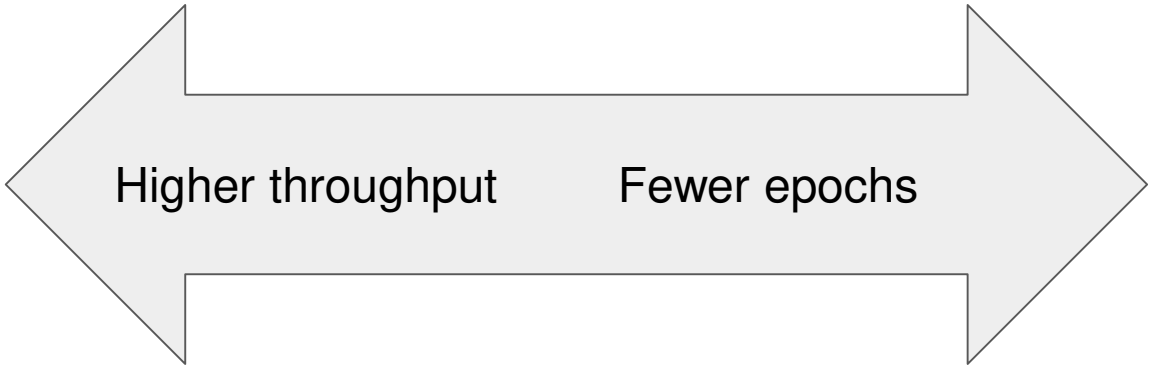
Training metric: throughput vs. time-to-train

Throughput (samples / sec)
- Easy / cheap to measure

Can increase throughput at cost of total time to train!

Time-to-train (end-to-end)
- Time to solution!
- Expensive
- High variance

Least bad choice



MLPerf

Training reimplementation equivalence

There are multiple competing ML frameworks

Not all architectures support all frameworks

Implementations still require some degree of tuning, especially at scale

Temporary solution: allow submitters to **reimplement** the benchmarks

Require models be mathematically equivalent

Exceptions: floating point, whitelist of minor differences

Training specific: hyperparameter tuning

Different system sizes ⇒ different batch sizes ⇒ different hyperparameters

But, some working hyperparameters are better than others

Finding good hyperparameters is expensive and not the point of the benchmark

Solution v0.5, v0.6: hyperparameter “borrowing” during review process

Training specific: variance

ML convergence has relatively high variance

Solution (kind of): run each benchmark multiple times

To reduce variance by x, need to run x^2 times = $$$

Settled for high margins of error

For vision: 5 runs, 90% of runs on same system within 5%

Agenda

Introduction

Design

Training

Inference

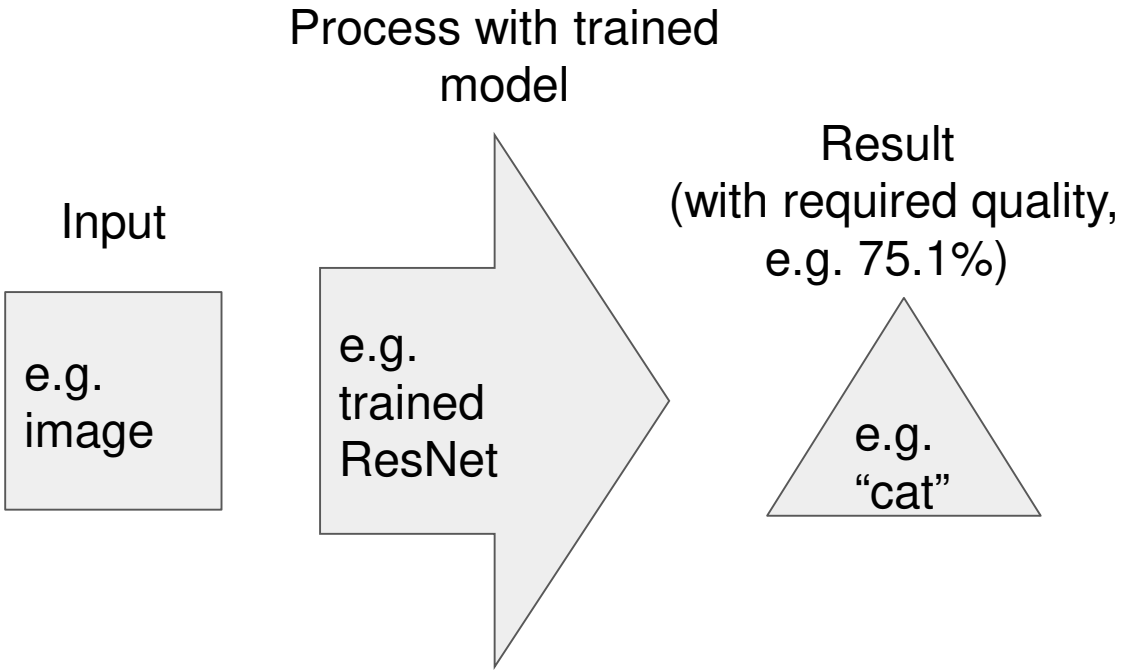
Presentation

Results

Future plans

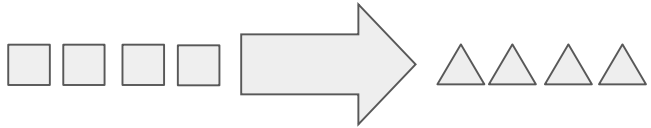
MLPerf

Inference benchmark definition

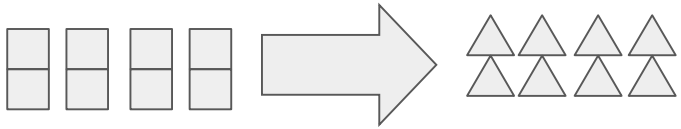


Do you specify the model? Again, Closed division does, Open division does not.

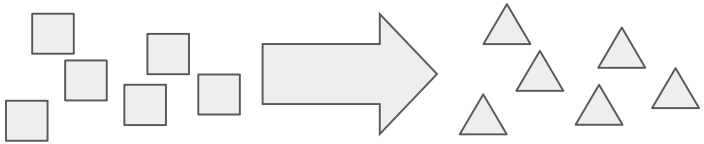
But how is inference really used? Four **scenarios**.



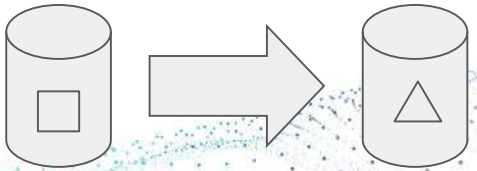
Single stream
(e.g. cell phone augmented vision)



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



Server
(e.g. translation app)



Offline
(e.g. photo sorting app)

MLPerf

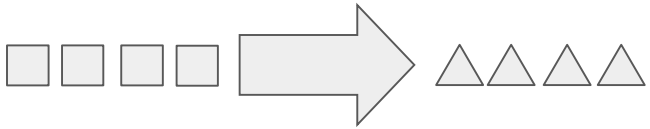
Inference benchmark selection v0.5

Minimum-viable-benchmark, maximize submitters, reflect real use cases

Area	Task	Model	Dataset
Vision	Image classification	Resnet50-v1.5	ImageNet (224x224)
Vision	Image classification	MobileNets-v1 224	ImageNet (224x224)
Vision	Object detection	SSD-ResNet34	COCO (1200x1200)
Vision	Object detection	SSD-MobileNets-v1	COCO (300x300)
Language	Machine translation	GNMT	WMT16

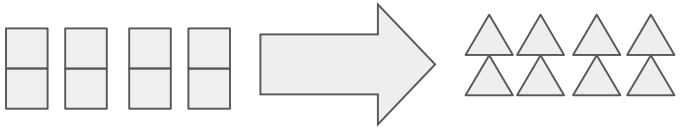
MLPerf

Inference metric: one metric for each scenario



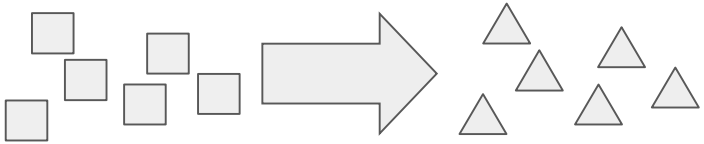
Single stream
e.g. cell phone
augmented vision

Latency



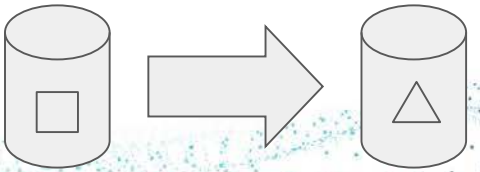
Multiple stream
e.g. multiple camera
driving assistance

Number streams
subject to latency
bound



Server
e.g. translation site

QPS
subject to latency
bound



Offline
e.g. photo sorting

Throughput

MLPerf

Inference implementation equivalence

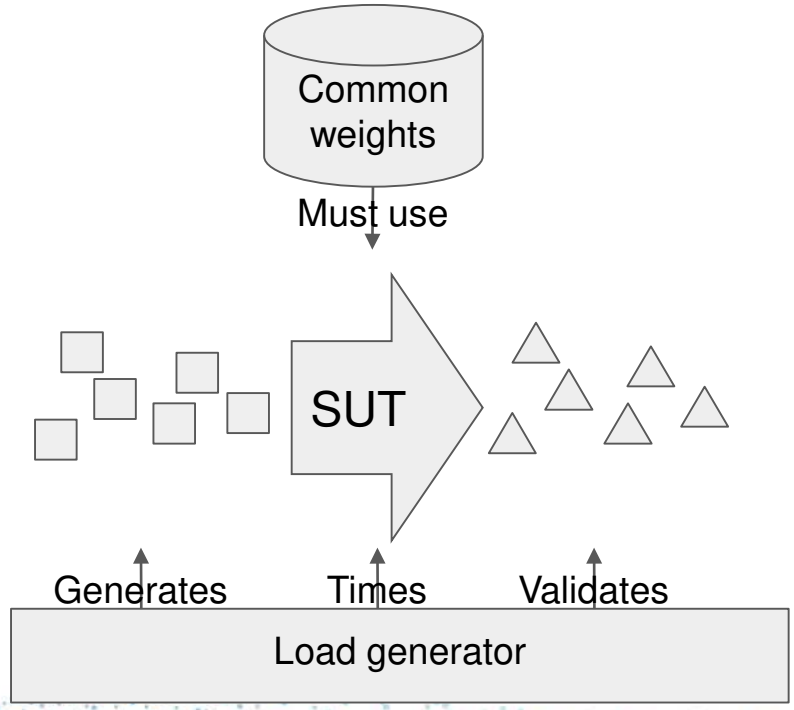
Even greater range of software and hardware solutions

So, allow submitters to reimplement subject to mathematical equivalence

But require:

Use standard set of **pre-trained weights for Closed Division**

Use **standard C++ "load generator"** that handles scenarios and metrics



MLPerf

Inference specific: quantization and retraining

Quantization is key to efficient inference, but do not want a quantization contest

Can the Closed division **quantize**?

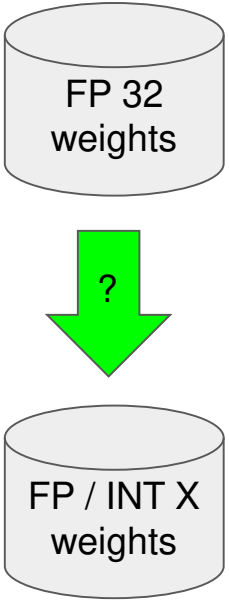
Yes, but must be principled: describe reproducible method

Can the Closed division **calibrate**?

Yes, but must use a fixed set of calibration data

Can the Closed division **retrain**?

No



MLPerf

Agenda

Introduction

Design

Training

Inference

Presentation

Results

Future plans

MLPerf

Presentation: normalization and/or scale

Do you present only the results?

System
Foo
Bar

Results lack scale information.

If so, an inefficient larger system can look better than an efficient smaller system.

Need supplemental normalization and/or scaling information

MLPerf provides some scale information

Current: number of chips
Planned: power

Presentation: results or summarize

Should we have a single MLPerf score that summarizes all results?

System	ResNet	GNMT
Foo	3m	4m
Bar	1m	6m

Pro:

- Easy to communicate
- Do it consistently

Con:

- Oversimplifies
- Some vendors submit subsets
- Users care about different subsets

MLPerf doesn't summarize.
We recommend weighted geometric mean.

Agenda

Introduction

Design

Training

Inference

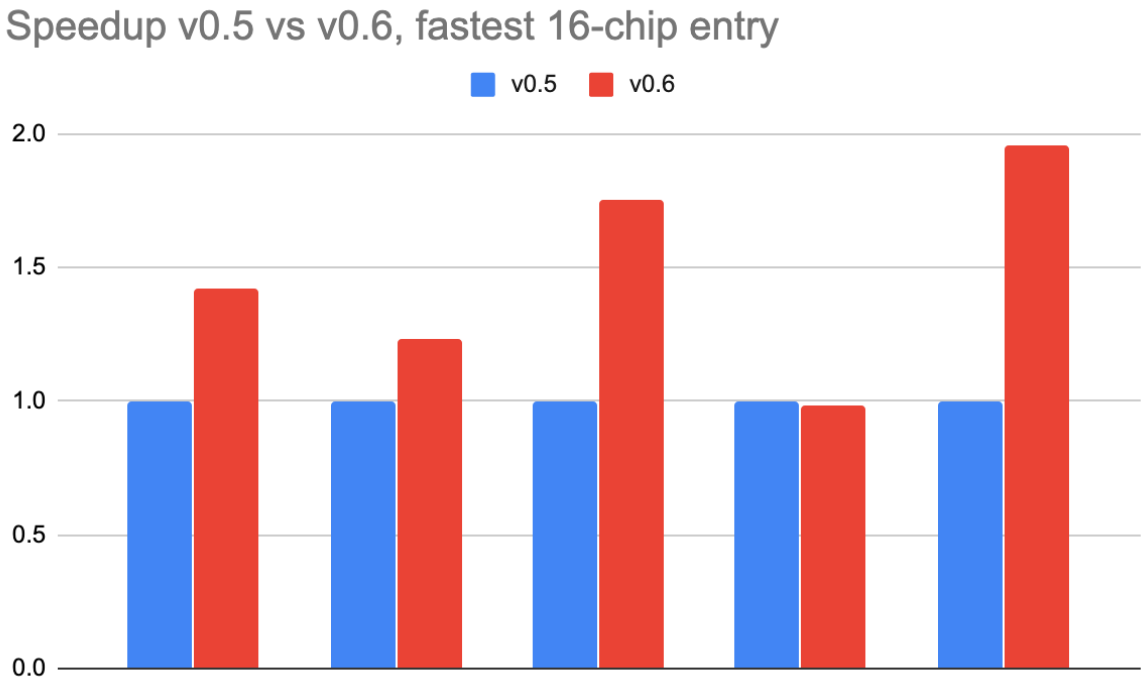
Presentation

Results

Future plans

MLPerf

MLPerf drives performance improvements



Over 6 months

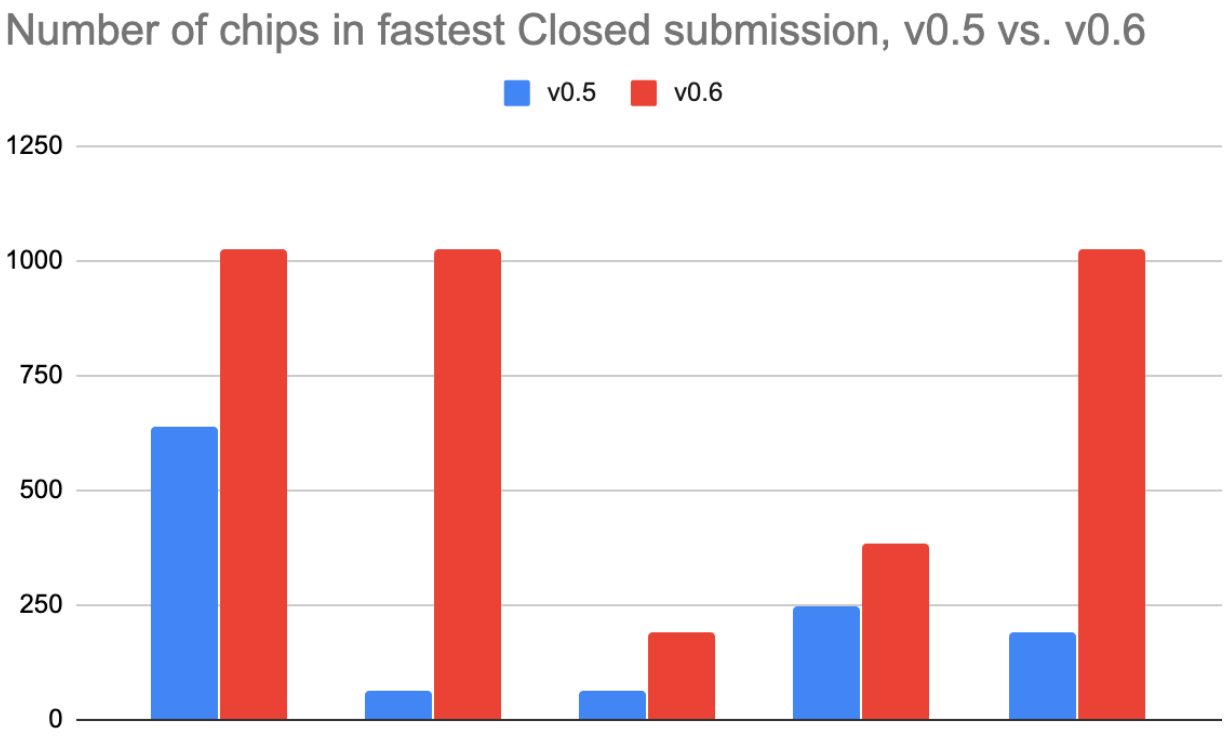
Same hardware platforms

Higher quality targets

Quality Targets		
	v0.5	**v0.6**
ResNet	74.9	75.9
SSD	21.2	23
Mask R-CNN	0.377/0.399	same
GNMT	21.8	24
Transformer	25	same

MLPerf

MLPerf drives scaling improvements



Over 6 months

Same hardware platforms

MLPerf

MLPerf makes market choices more transparent

- "...Microsoft is excited to participate in MLPerf to support an open and standard set of performance benchmarks to **drive transparency** and innovation in the industry." – **Eric Boyd, CVP of AI Platform, Microsoft**

- "MLPerf can **help people choose** the right ML infrastructure for their applications..." – **Urs Hölzle, SVP of Technical Infrastructure, Google**

- "You should factor [MLPerf] into your **evaluations of commercial offerings** and insist that providers include their AI-optimized solutions in the benchmark competitions." - **James Kobelius, Silicon Angle**

Agenda

Introduction

Design

Training

Inference

Presentation

Results

Future plans

MLPerf

Future plans: develop a benchmark framework

- What areas do we want to cover
- What benchmarks do we want in each area
- What application should drive each benchmark
- Identify advisors from industry and research to help guide direction

Area	Benchmark	Application	Industry software advisors	Research advisors
Vision	...			
	Object segmentation	Automotive vision	Carl at Cruise Teresa at Tesla,	Harry at Harvard, Stacey at Stanford

Possible benchmark framework

Area	Benchmark	Application	Advisors	Training status	Inference status
Vision	Image classification			v0.6	v0.5
	Object segmentation			v0.6	v0.5
Speech	Speech-to-text			v0.7	
	Text-to-speech				
Language	Translation			v0.6	v0.5
	NLP				
Commerce	Recommendation			v0.6 (revising)	
	Time series			v0.7	
Research (training only)	Reinforcement learning			v0.6 (revising)	
	GAN				
Mobile vision (inference only)	Image classification				v0.5
	Object segmentation				v0.5

Future plans: improve rules and reference code

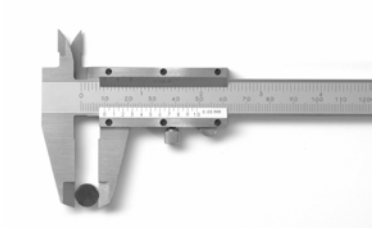
- Training rules challenges
  - Hyperparameter determination
  - Optimizer equivalence
  - Variance reduction

- Inference rules challenges
  - Quantization and retraining
  - Power measurement

- Make reference implementations faster and more readable

Future home of MLPerf: MLCommons

We are creating a non-profit called MLCommons to “accelerate ML innovation and increase its positive impact on society.”



Benchmarks + Large public datasets + Best practices + Outreach

Photo credits (left to right): Simon A. Eugster CC BY-SA 3.0, Riksantikvarieämbetet / Pål-Nils Nilsson CC BY 2.5 se, Public Domain, Public Domain

MLPerf

We need your help!

mlperf.org/get_involved

Join a working group

Submit results

Become a founding member of MLCommons, email **info@mlperf.org**

MLPerf