MARCH 2019

TEXAS
The University of Texas at Austin

METRICS FOR MACHINE LEARNING WORKLOAD BENCHMARKING

FastPath 2019, in conjunction with ISPASS 2019

SNEHIL VERMA, QINZHE WU, BAGUS HANINDHITO, GUNJAN JHA, EUGENE JOHN, RAMESH RADHAKRISHNAN, LIZY JOHN

The University of Texas at Austin, The University of Texas at San Antonio, Dell Inc.

Server industry trends

Machine learning/Deep learning emerging to provide better business insight

6x growth in AI

By 2020, 20% of the enterprise infrastructures deployed will be used for AI. Up from 3% in 2017.

40x growth in edge computing

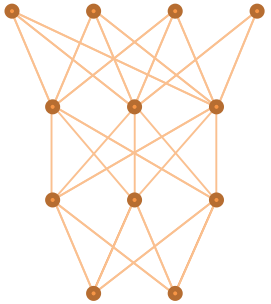
40% of large enterprises will be integrating edge computing principles into their IT projects by 2021. Up from less than 1% in 2017.

Stats from Gartner

Deep Learning

Untrained neural network model

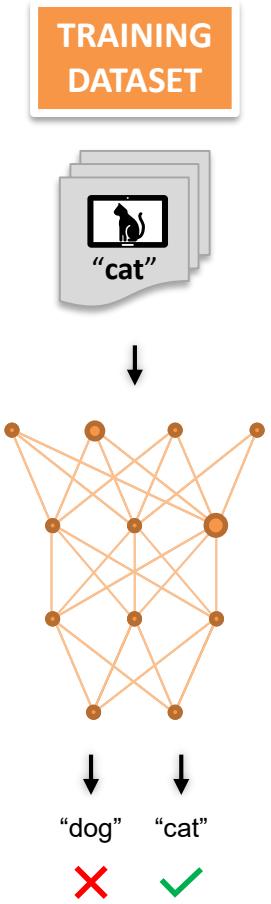
TRAINING



Deep Learning

TRAINING

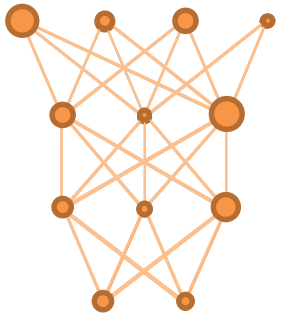
Learning a new capability from existing data.



Deep Learning

Trained Model
New Capability

INFERENCE

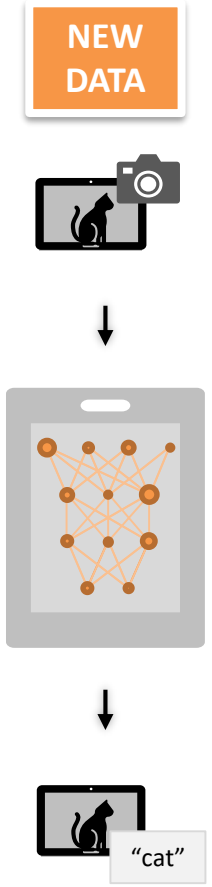


Deep Learning

INFERENCE

Applying this capability to new data.

Trained Model
Optimized for Performance



Importance of training hardware

- Flood of the data available
- Increasing computational power
- Competition (GPU, TPU, IPU, ASICs)

How to evaluate hardware for DL?

- Benchmarks?
- Metrics?

Benchmarks

Fathom	DAWNBench	DeepBench	TF CNN Bench	AI - PEP
Harvard	Stanford	Baidu	Google	Facebook

and many more…

PRIOR MEASUREMENT METHODOLOGIES

- Focused on one domain
- No standard ML suite maintained by a governing body

A broad ML benchmark suite for measuring performance of ML software frameworks, ML hardware accelerators, and ML cloud platforms.

- Coverage of different DL domains
- Reproducibility of results
- Accelerate innovation in DL hardware, systems, software & algorithms

MLPerf Training benchmark suite [v0.5]

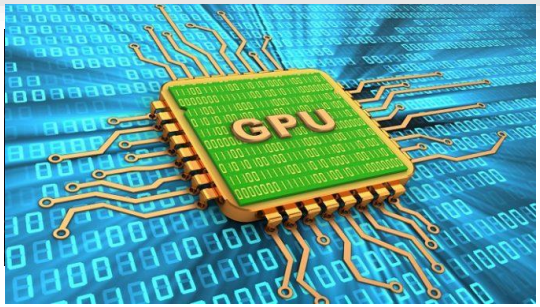
Image Classification	TensorFlow	ResNet v1.5	Cisco, BNL
Object Detection*	Caffe2	Mask R-CNN with ResNet50	Stanford, Alibaba
Recommendation	PyTorch	Neural Collaborative Filtering	Stanford, Google
Reinforcement Learning	TensorFlow	Fork of the Mini Go project	N/A
Speech Recognition	PyTorch	DeepSpeech2	Baidu
Translation*	TensorFlow	Transformer	Cisco

* One more variation of the benchmark is available.

Metrics

Traditionally:

- Execution time
- IPC or IPC / Watt



With the advent of GPUs:

- Throughput: $\frac{\#images}{sec}$

Metrics

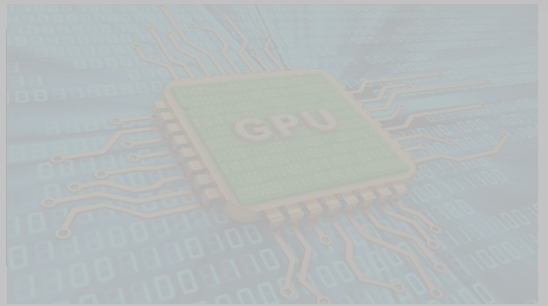
Traditionally:

- Execution time
- IPC or IPC / Watt

With the advent of GPUs:

- Throughput: $\frac{\#images}{sec}$

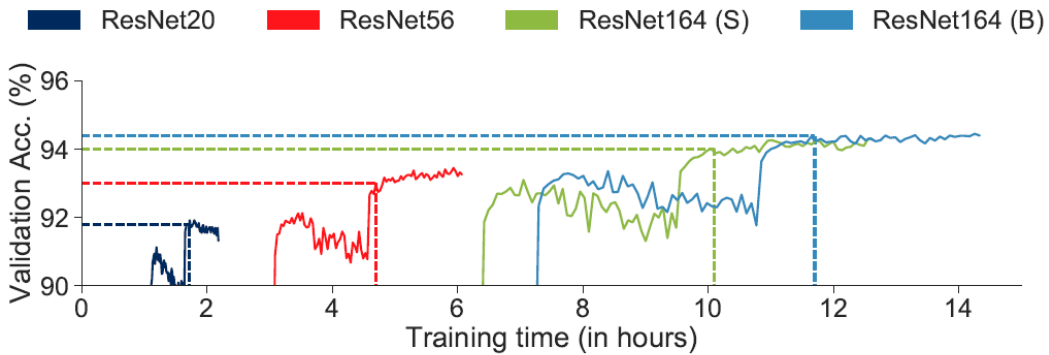
Issues?



Deep Learning needs a new metric!

Objective: Propose an appropriate metric for Deep Learning.

DAWNBench – Time to Accuracy (TTA)



Accuracy threshold	Time to threshold	Fastest model
91.8%	1h 43m	ResNet20
93%	4h 42m	ResNet56
94%	10h 6m	ResNet164 (simple)
94.4%	11h 42m	ResNet164 (bottleneck)

Figure 6: Validation accuracy vs. training time for different ResNet architectures on CIFAR10. Horizontal lines indicate accuracy thresholds of 91.8%, 93%, 94%, and 94.4%. ResNet20, ResNet56, ResNet164 (with simple building blocks), and ResNet164 (with bottleneck building blocks) are fastest to the corresponding accuracy thresholds.

MLPerf quality target references

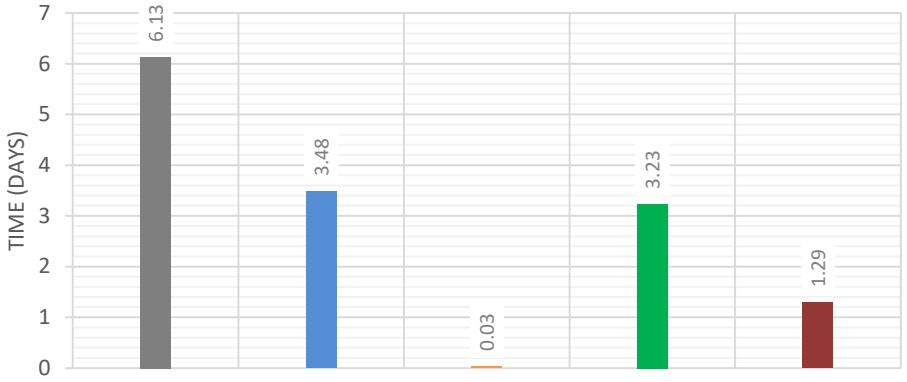


Image Classification
Accuracy: 74.9%

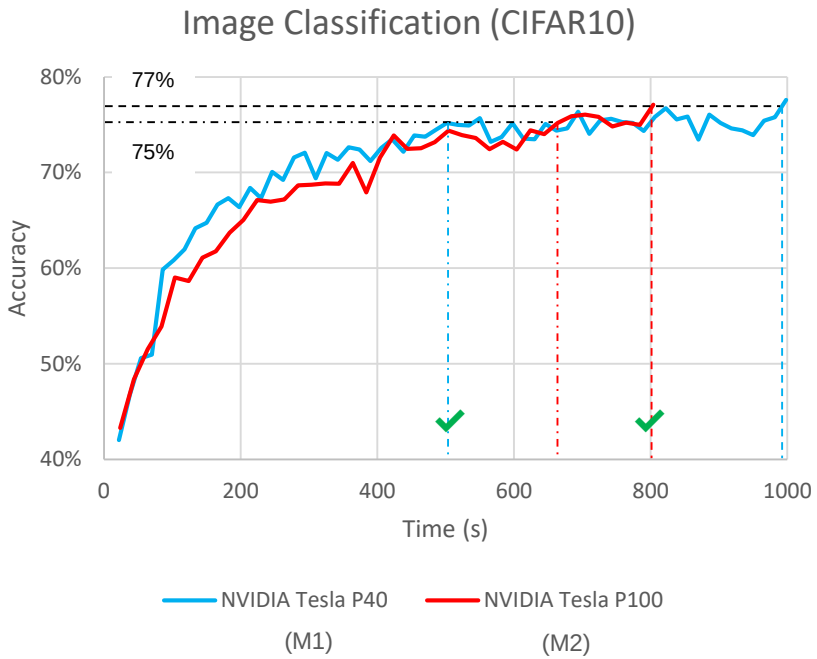
Recommendation
Hit Ratio @ 10: 0.635

Translation (transformer)
BLEU score (uncased): 25

Object Detection (R-CNN)
Box mAP: 0.377
Mask mAP: 0.339

Reinforcement Learning
Pro move prediction: 40%

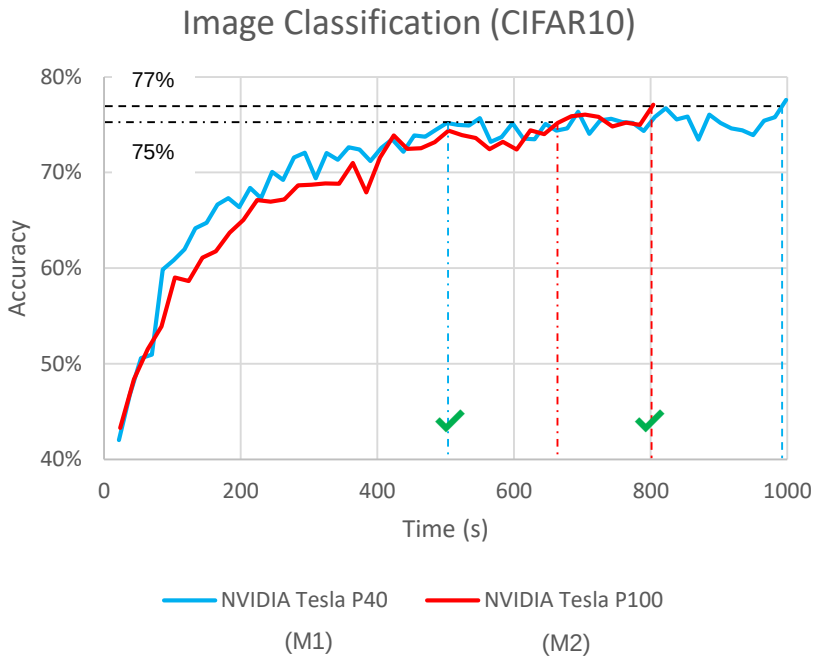
Time to Accuracy - Issues



Time to Accuracy - Issues

Sensitivity to threshold

Acc. (%)	TTA (M1)	TTA (M2)	Ratio (M2/M1)
70	246	364	1.479
71	278	364	1.309
72	294	424	1.442
73	422	424	1.004
74	486	504	1.037
75	502	664	1.323
76	694	704	1.014
77	998	804	**0.805**



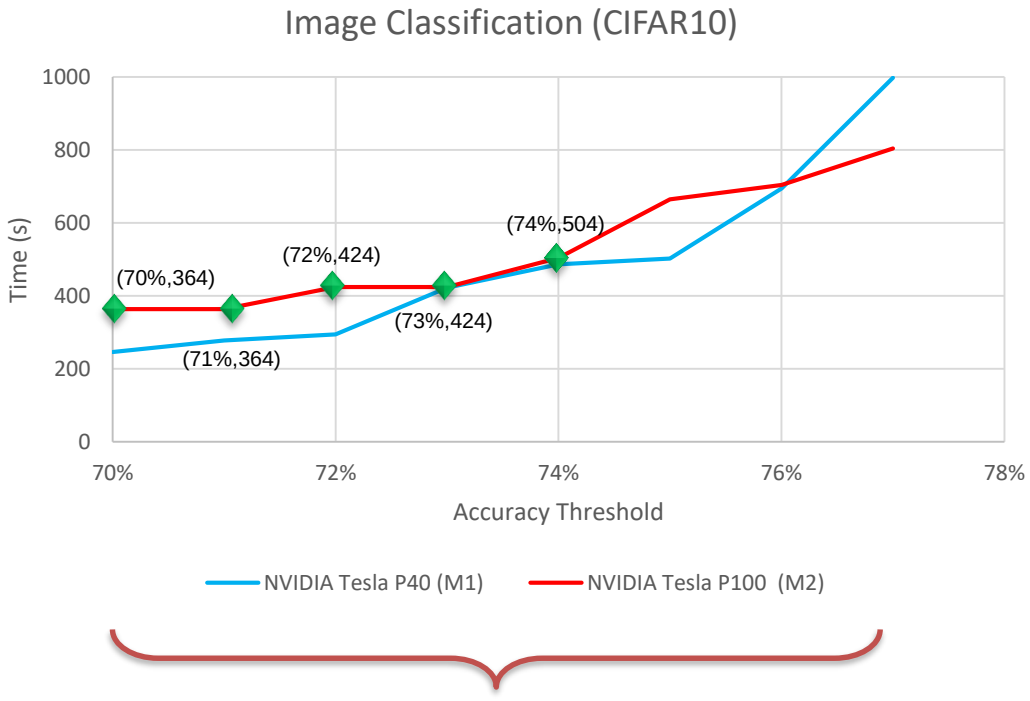
Proposed Metric

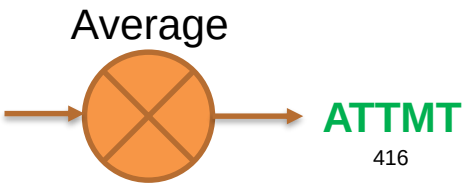
- Average Time to Multiple Thresholds (ATTMT)

Ideally, a metric should be a function of both time and accuracy that rewards for:

- *hitting the target accuracy*
- *reducing the time to achieve whatever accuracy have been achieved*

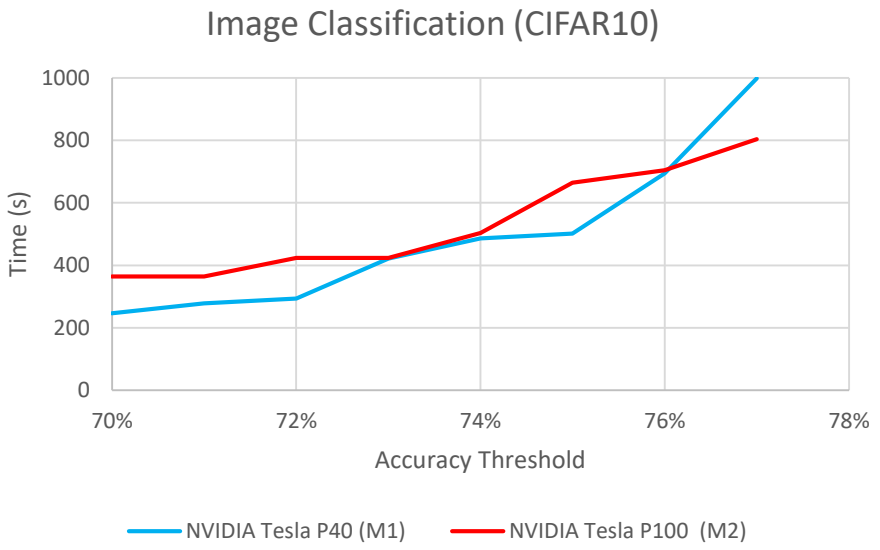
Time to Multiple Threshold (TTMT) curves





ATTMT

Acc. (%)	ATTMT (M1)	ATTMT (M2)	Ratio (M2/M1)	
			TTA	ATTMT
70	246	364	1.479	1.480
71	262	364	1.309	1.389
72	272.6	384	1.442	1.409
73	310	394	1.004	1.271
74	345.2	416	1.037	1.205
75	371.3	457.3	1.323	1.232
76	417.4	492.6	1.014	1.180
77	490	531.5	**0.805**	**1.085**



Methodology

Benchmarks – MLPerf Training v0.5

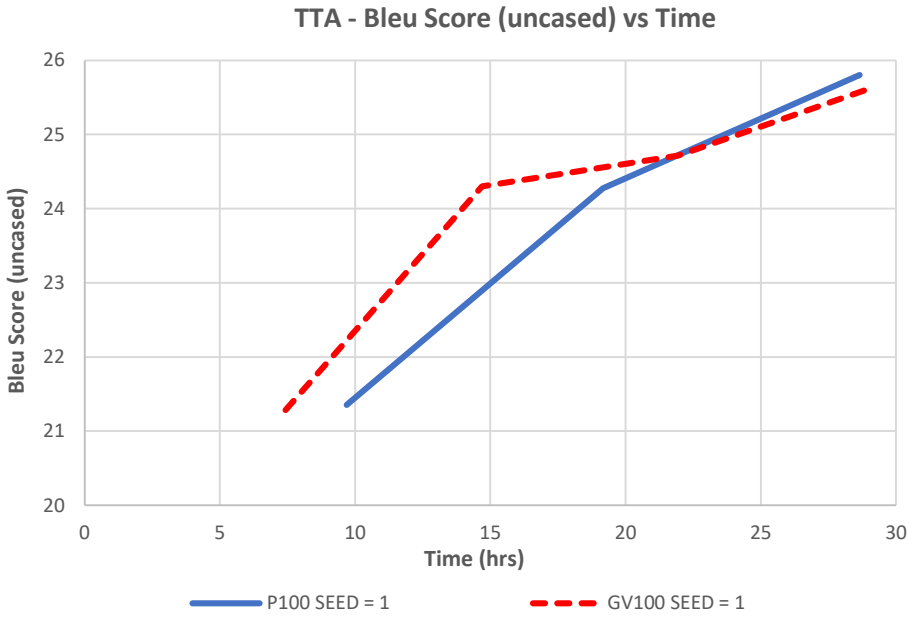
Image Classification	ResNet v1.5
Object Detection	Mask R-CNN with ResNet50
Recommendation	Neural Collaborative Filtering
Reinforcement Learning	Fork of the Mini Go project
Translation	Transformer

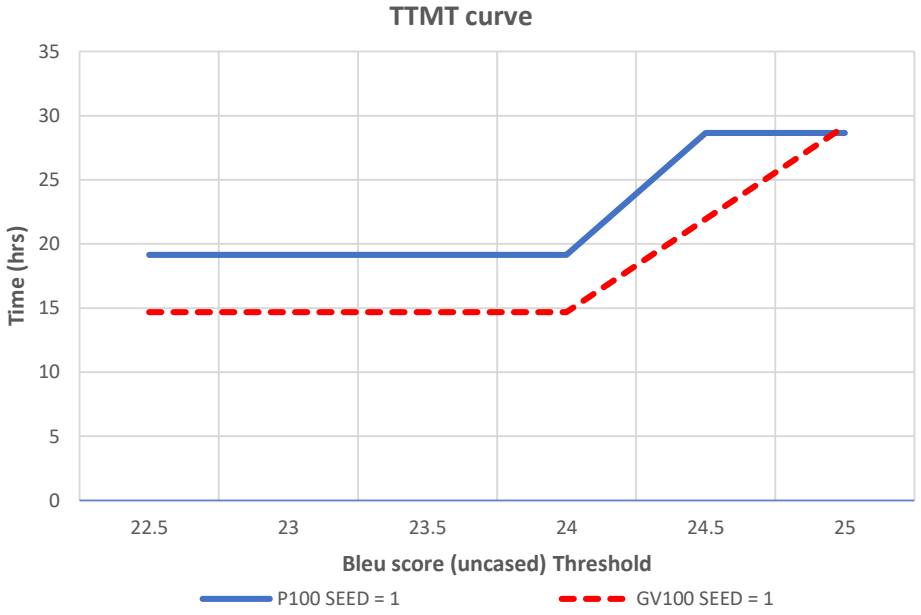
Platforms

Parameters	Platform 1 – P100	Platform 2 – GV100
CPU	Intel Xeon E5-2660v4	Intel Xeon W-2195
Architecture | Base Freq.	Broadwell | 2.00 GHz	Skylake | 2.30 GHz
#CPU | #Cores | #Threads	2 | 28 | 56	1 | 18 | 36
Physical Memory	4-Channel 256 GB DDR4	4-Channel 256 GB DDR4
GPU (architecture)	NVIDIA Tesla P100 (Pascal)	NVIDIA Quadro GV100 (Volta)
CUDA cores | **Tensor cores**	3584 | -	5120 | **640**
Mem. Size | BW	16 GB HBM2 | 720 GBps	32 GB HBM2 ECC | 870 GBps

Evaluation

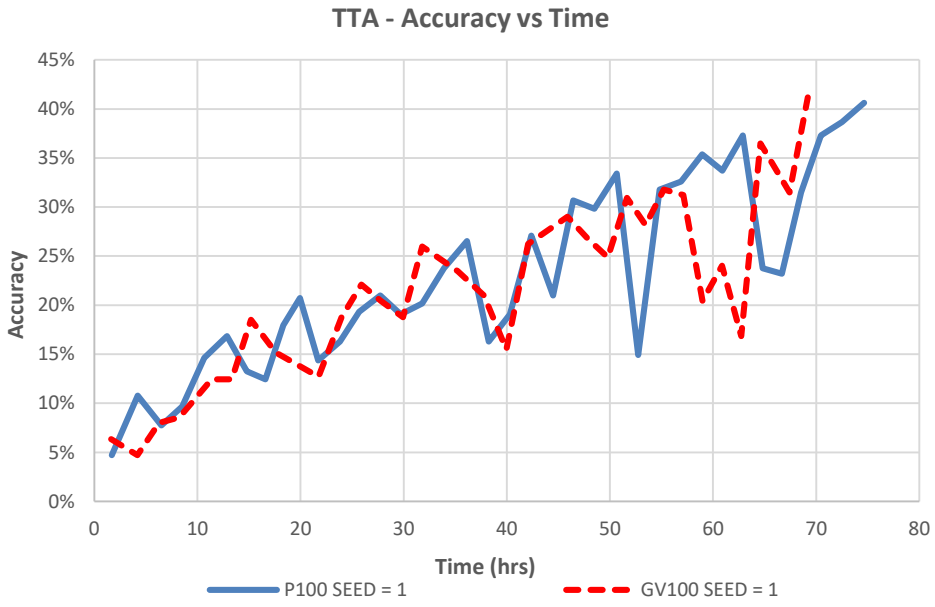
TTA – TTMT

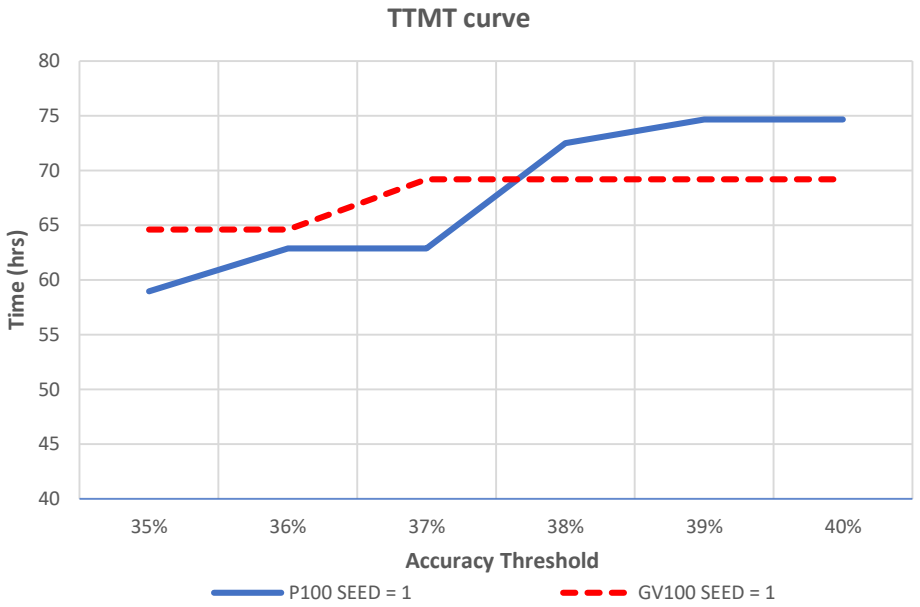




Translation: non-recurrent

TTA – TTMT





Reinforcement Learning

Sensitivity

Quality Target (%)	TTA (hrs)		ATTMT (hrs)		%improvement wrt	
	GV100	P100	GV100	P100	TTA	ATTMT
35	64.61	58.95	60.88	52.72	-9.60%	-15.48%
36	64.61	62.89	63.04	55.47	-2.73%	-13.65%
37	69.18	62.89	65.37	57.50	-10.00%	-13.69%
38	69.18	72.51	66.13	61.14	4.59%	-8.16%
39	69.18	74.65	66.90	65.14	7.33%	-2.70%
40	69.18	74.65	67.66	67.75	7.33%	0.13%

Sensitivity of metrics to the quality target for Reinforcement Learning Benchmark

Variability of metric

Quality Target (HR@10)

Quality target range, δ

Seed Value	TTA (0.635)	ATTMT (0.585, 0.635, +0.01)
1	59.70	26.34
2	66.35	28.44
3	64.68	30.80
4	77.74	31.64
5	95.57	36.25
Sample Variance	**205.45**	**13.94**
Standard Deviation	**14.33**	**3.73**

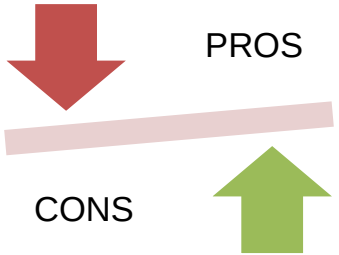
Variability in Recommendation Benchmark for different metrics (in minutes)

Single vs Multiple Threshold(s)

Benchmark	Speedup if TTA is used	Speedup if ATTMT is used
Image Classification (Full-Precision)	1.46	1.41
Image Classification (Mixed-Precision)	2.65	2.75
Object Detection	1.60	1.68
Recommendation	0.68	0.61
Reinforcement Learning	1.08	1.00
Translation	0.98	1.22

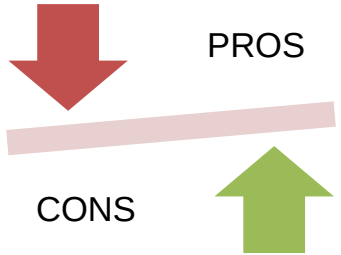
Speedup of GV100 over P100

Conclusion



- TTA
  - Sensitive to threshold and seed values (requires more number of runs)
  - Only one data point is used from a long run, resulting in wastage of many data points
  - Encourages risk taking

Conclusion



- ATTMT
  - Lower sensitivity to the chosen threshold and variability to the seed values
  - Tracks overall behavior of the system using multiple points from the same run
  - Slightly complex to calculate

Thank You!

Questions?

To know more about our work please visit:

http://lca.ece.utexas.edu/