A Study of Machine Learning Inference Benchmarks

Oliver Alvarado Rodriguez[†]
Department of Computer Science, William Paterson University, Wayne, NJ, USA

Dev Dave[†]
Department of Computer Science, William Paterson University, Wayne, NJ, USA

Weihua Liu[*]
Department of Computer Science, William Paterson University, Wayne, NJ, USA

Bogong Su
Department of Computer Science, William Paterson University, Wayne, NJ, USA

ABSTRACT

Machine learning (ML) is becoming a powerful tool for a variety of applications where artificial intelligence solutions are required. A ML benchmark is a standard suite to measure, evaluate and compare the performance and efficiency of ML systems. This study analyzes the benchmark results from two famous benchmarks MLMark and MLPerf to provide a basis of comparison between both benchmarks as well as to provide recommendations on computer architectures to utilize for ML inferencing. Lastly, special emphasis is placed on the performance of edge AI devices.

CCS CONCEPTS

• **Computer systems organization**; • **Architectures**; • **Other architectures**; • **Neural networks**;

KEYWORDS

Machine learning, benchmarks, inference, performance, computer architectures

ACM Reference Format:
Oliver Alvarado Rodriguez[†], Dev Dave[†], Weihua Liu, and Bogong Su. 2020. A Study of Machine Learning Inference Benchmarks. In *2020 4th International Conference on Advances in Image Processing (ICAIP 2020), November 13–15, 2020, Chengdu, China.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3441250.3441277

1 INTRODUCTION

As there is a growing demand of ML applications, numbers of different ML inference systems have been developed ranging from embedded devices to data-center solutions. Many ML benchmarks [1] were developed both by academic and industrial organizations such as AI Benchmark [2], MLMark [3, 4], MLPerf [5, 6] and so on. The two benchmarks focused on in this article are MLMark and MLPerf that are used by various benchmark submitters.

[*†]Graduated CS major students
*Corresponding author: liuw3@wpunj.edu

ICAIP 2020, November 13–15, 2020, Chengdu, China

ACM ISBN 978-1-4503-8836-8/20/11…$15.00
https://doi.org/10.1145/3441250.3441277

Table 1: Basic Information of MLPerf and MLMark

		MLPerf	MLMark
# of results		165 (closed division)	114
# of platforms		23 (closed division)	8
Tasks & models	image classification	ResNet-50 1.5	ResNet-50 1.0
		MobileNet 1.0	MobileNet 1.0
	object detection	SSDResNet-34	/
		SSDMobileNet-v1	SSDMobileNet 1.0
	machine translation	NMT	/
Scenarios & metrics	single stream	90th-percentile latency (ms)	95th-percentile latency (ms)
	multistream	no. of streams	/
	server	queries/sec	/
	offline	throughput (samples/second)	throughput (samples/second)

MLMark is a ML inference benchmarking solution for edge devices created by the Embedded Microprocessors Benchmark Consortium (EEMBC) in 2019. MLPerf was founded in February of 2018 as a collaborative system by both companies and academic researchers to provide an all-inclusive benchmarking tool for neural network training and inference. The results from MLPerf contain both open and closed divisions. Only the closed division data was utilized as its testing restrictions allow for a fair comparison between devices. Both benchmarks use similar tasks, inference models, scenarios, and other metrics as shown in Table 1

MLPerf and MLMark were analyzed using the image classification task with the ResNet ad MobileNet models and, in later sections, the object task with the SSDMobileNet model. The data was gathered from the datasheets on the web sites of MLPerf and MLMark; however, MLPerf data types were extracted from their GitHub as they are not provided in the MLPerf datasheet. The latency data of MLPerf was updated to the 95th percentile latency found within MLPerf's extensive documentation.

Since the edge AI device market grows fast, the performance of edge AI devices was analyzed with respects to the power consumption and cost. This allows for a comparison between the edge AI devices that can then be utilized for device selection.

In the following sections, first presented is the methodology in section 2 alongside related performance factors. Section 3 and 4 present the performance analysis and comparison of edge AI devices respectively. Section 5 is the conclusion.

2 METHODOLOGY

To carry out the comparison between the MLMark and MLPerf benchmarks, firstly, platforms were selected from both benchmarks that utilized their own software framework. Secondly, three trained ML models were utilized: two concerning image classification, MobileNet and ResNet, and the last one concerning object detection, SSDMobileNet. Thirdly, the most popular performance metrics used in both academia and industry were used for comparison: latency and throughput. Lastly, the cost and power consumption information of the devices, gathered from manufacturer websites, were used to study the performances of the edge AI devices. In summary, a listed version of the methodology follows:

- Selected platforms with their own software framework from two major ML benchmarks MLPerf and MLMark for comparison.
- Used three trained ML models MobileNet, ResNet, and SSDMobileNet.
- Used latency and throughput for comparison.
- Included the cost and power consumption information of devices.

2.1 Data and Performance Related Factors

There are 8 platforms in the MLMark datasheet and 23 platforms in the closed division MLPerf datasheet. They are listed with their core, accelerators and type in Table 2. Notice that some of the platforms may have different accelerators.

2.1.1 Software Framework. Also notice that some platforms have several results with different software frameworks in MLPerf as shown in Table 3. By using their own framework, the latency is 1.13 to 3.83 times shorter. This is equivalent to an increase in speed as the systems are more responsive.

2.1.2 Accelerator. The MLPerf datasheet shows that some platforms' results have both CPU only and CPU with accelerators as shown in Table 4. Latency can be reduced 1.5 to 2.48 times by using accelerators. Same as *2.1.1*, This is equivalent to an increase in speed as the systems are more responsive.

2.1.3 Data Types. Table 5 presents the relationship between performance and data types from MLMark results. This shows that FP16 can have 13% to 130% better performance than FP32. INT8 can have 500% to 2000% better performance; Coral Dev board's INT performances are very high due to its 8-bit TPU accelerator [7].

3 PERFORMANCE ANALYSIS

Table 6 presents 22 (5 from MLMark and 17 from MLPerf) platforms' best performances. Some notes on performance analysis follow:

1. Table 6 ignores MLPerf's results in the translation task and results in multistream and server scenarios. Three models were selected: MobileNet, ResNet, and SSDMobileNet that match the models used in MLMark.
2. Results in MLMark are based on batch = 1 and concurrency = 1.
3. There is no data type presented in the MLPerf datasheet. Data types were extracted from GitHub and are presented in Table 6
4. All platforms use their accelerators and own frameworks to represent their highest performance.
5. Both benchmarks use the same inference models as workload except MLPerf which uses the ResNet model v1.5 which has a slight difference from v1.0 used by MLMark.
6. Latency is measured as the 95th percentile of one inference to completion in MLMark; however, MLPerf uses the 90th percentile in their datasheet. For comparison, the 95th percentile was extracted for single stream from the MLPerf documentation and is presented in Table 6

MLMark's two metrics latency and throughput are analogous to the MLPerf single-stream and offline metrics [8]. But the inference models of workloads and testing conditions of those two ML benchmarks are different. Therefore, conducting precise comparisons across benchmarks is not suitable. Further, two more problems are listed:

1. The latency of platform Raspberry Pi 4 using TF Lite in MLPerf is much higher than the same platform using TF in MLMark due to difference between those two frameworks.
2. NVDIA submitted its Edge AI device Jetson AGX Xavier to both ML benchmarks. However, Table 6 shows the performances have one order of magnitude difference between those two benchmarks because its platform in MLPerf uses both GPU and DLA which is NVIDIA's new deep learning accelerator including a special convolution core with 2048 MAC units [9].

The performances from MLMark shown in Table 6 will be discussed in Section 4. From the MLPerf part in Table 6 some observations follow:

1. There are two groups of platforms, one consists of platforms for edge devices using the FP32 data type, and another group for servers with higher performance using the INT8 data type.
2. The throughput of Cloud TPU v3 platform [10] is proportional to the number of TPUs in MLPerf datasheet and its peak throughput can reach over one million fps for 128 TPUs. The following equation is used to calculate the scalability when the number of accelerators expands from n to m:

$$Scalability = \frac{\frac{Performance_m}{Performance_n}}{\frac{m}{n}}$$

   The scalability of Cloud TPU v3 platform is 99.2% when the number of TPUs is expanded from 4 to 128.
3. Three platforms: Supermicro 4029 and 6049, and SCAN 3XS, submitted by NVIDIA, also have large throughputs by using many GPU accelerators [11] with a slightly weaker scalability of 92% when the number of GPUs is expanded from 4 to 20.
4. Intel Xeon Platinum 9200 and Tencent Cloud have very low latency and large throughput because they use the highest-end CPUs available: the Intel Xeon Platinum 9200 and the Intel Xeon 8255C which have 56 and 24 cores respectively and their max turbo clock rate can reach 3.9 GHz; further, they have special instructions for deep learning [12].

Table 2: Platforms in MLMark and MLPerf

	Platform	CPU	Accelerator	Application	
1	Coral Dev Board	i.MX8M Cortex-A53		edge AI	MLMark
2	Coral Dev Board	i.MX8M Cortex-A54	Edge TPU	edge AI	
3	HiKey970	ARM Cortex-A73 (4 cores)	ARM Mali G72 M12 GPU	mobile	
4	Jetson AGX Xavier	ARM v8.2 64-Bit (8-Core)	Volta GPU	edge AI	
5	Jetson Nano	ARM A57 (4 cores)	Maxwell GPU	edge AI	
6	GEFORCE RTX 2080 Ti	Turing GPU		PC	
7	Neural Compute Stick 2	Myriad X		edge AI	
8	Raspberry Pi 4 Model B	Quad coreCortex-A72		edge AI	
1	Raspberry Pi 4 (rpi4)	Quad coreCortex-A72		edge AI	MLPerf
2	Jetson AGX Xavier	NVIDIA Carmel (ARMv8.2)	NVIDIA Xavier	edge AI	
3	Alibaba Cloud T4	Intel Xeon Platinum 8163	Nvidia Tesla T4	server	
4	Dell EMC R740	Intel(R) Xeon(R) Gold 6154 x2	Nvidia T4	server	
5	Dell EMC R740xd with 2nd generation Intel® Xeon® Scalable Processor	Intel(R) Xeon(R) Gold 6248 CPU x2		server	
6		Intel(R) Xeon(R) Platinum 8276 CPU x2		server	
7	Linaro HiKey960 (hikey960)	HiSilicon Kirin959		mobile	
		HiSilicon Kirin960	Arm Mali-G71 MP8	mobile	
8	Huawei Mate 10 Pro (mate10pro)	HiSilicon Kirin969		mobile	
		HiSilicon Kirin970	Arm Mali-G72 MP12	mobile	
9	Firefly-RK3399 (firefly)	Rockchip RK3398		edge AI	
		Rockchip RK3399	Arm Mali-T860 MP4	edge AI	
10	HL-102-Goya PCI-board	Intel(R) Xeon(R) CPU E5-2630 v4	Synapse-V0.2.0	server	
11	Cloud TPU v3	Intel Skylake x 2	TPU v3 x 4	server	
12	Intel® Xeon® Platinum 92	9282 Processor x 2		server	
13	DELL ICL i3 1005Gl	Intel® Core™ i3-1005Gl	Intel® UHD Graphics	lap top	
14	Supermicro 4029 8xT4	Intel(R) Xeon(R) Platinum 8280	NVIDIA Tesla T4 x8	server	
15	Supermicro 6049 20xT4		NVIDIA Tesla T4 x20	server	
16	SCAN 3XS DBP T496X2	Intel(R) Xeon(R) 8268 x2	NVIDIA TITAN RTX x4	server	
17	SDM855 QRD	Qualcomm Kryo485	Hexagon 690 Processor	mobile	
18	AlibabaHanGuang	IntelXeon Platinm 8163	Hanguang 800	server	
19	Centaur Technology Reference Design v1.0	Centaur Integrated x86 CPUs	Centaur Integrated AI Coprocessor	edge AI	
20	2x NNP-I 1000	Intel(R) Xeon(R) Silver 4116 Processor	Intel® Nervana™ Neural Network Processor	server	
21	Hailo-8	Intel(R) Core(TM) i7-7820HQ	Hailo8	edge AI	
22	Tencent Cloud	Intel Xeon 8255C CPU x4		server	
23	furiosa-single-fpga	Intel(R) Core(TM) i3-7100	Renegade	server	
24		Intel(R) Xeon(R) CPU E5-2620 v4	Renegade	server	

Table 3: Latency (ms) vs. Software Frameworks

	TF lite	own software framwork	increase
Raspberry Pi 4 (rpi4)	394	103	3.83
Linaro HiKey960 (hikey960)	143	121	1.18
Firefly-RK3399 (firefly)	120.56	106.5	1.13

Table 4: Latency (ms) of MobileNet FP32 Using Accelerators

	CPU	w/h accelerator	increase
HiKey970	55.1	22.2	2.48
Linaro HiKey960 (hikey960)	121	50.77	2.38
Huawei Mate 10 Pro (mate10pr	111.6	74.2	1.50

Table 5: Performance vs. Data Type from MLMark Results

Platform	Workload	Latency (ms)					Throughput (fps)				
		INT8	FP16	FP32	INT8 vs FP32	FP16 vs FP32	INT8	FP16	FP32	INT8 vs FP32	FP16 vs FP32
Coral Dev Board	MobileNet 1.0	3.3		88.6	2585%		339		11.4	2874%	
	ResNet-50 1.0	51.7		318	515%		19.4		3.2	506%	
	SSDMobileNet 1.0	33.9		201	493%		34		5	580%	
HiKey970	MobileNet 1.0		17	22.2		31%		68	45.6		49%
	ResNet-50 1.0		66.8	96.8		45%		16.7	10.5		59%
	SSDMobileNet 1.0										
Jetson Nano	MobileNet 1.0		16.2	18.4		14%		62.7	55.3		13%
	ResNet-50 1.0		26.3	48		83%		38.4	21.2		81%
	SSDMobileNet 1.0		40.4	45.7		13%		25.3	22.4		13%
Jetson AGX Xavier	MobileNet 1.0	0.7	2.3	2.8	300%	22%	1600	547	367	336%	49%
	ResNet-50 1.0	1.9	3.5	7.9	316%	126%	540	291	128	322%	127%
	SSDMobileNet 1.0	1.8	6	8	344%	33%	642	171	128	402%	34%

Table 6: Performance of Major Platforms in MLMark and MLPerf Results

	Platform	Framework	Data type	Latency (ms)			Thruput (fps)			Bechmark
				MobileNet 1.0	ResNet 1.0 or 1.5	SSDMobileNet 1.0	MobileNet 1.0	ResNet 1.0 or 1.5	SSDMobileNet 1.0	
1	Coral Dev Board	ArmNN/TFLite	FP32	88.6	318	201	11.4	3.2	5	MLMark
2	HiKey970	ArmNN/TF	FP32	22.2	96.8		45.6	10.6		
3	Jetson AGX Xavier	TensorRT	FP32	2.8	7.9	8	367	128	128	
4	Jetson Nano	TensorRT	FP32	18.4	48	45.7	55.3	21.2	22.4	
5	Raspberry Pi 4 Model B	TF	FP32	219	1080	414	5	1.1	2.7	
1	Raspberry Pi 4 (rpi4)	ArmNN v19.08 (Neon)	FP32	108.3	453.1					MLPerf
2	Jetson AGX Xavier	TensorRT	INT8	0.58	2.04	1.5	6520	2158	2485	
3	Linaro HiKey960 (hikey960)	ArmNN v19.08 (OpenCL)	FP32	52.9	206					
4	Huawei Mate 10 Pro (mate10pro)	ArmNN v19.08 (OpenCL)	FP32	79.29	355.4					
5	Firefly-RK3399 (firefly)	ArmNN v19.08 (OpenCL)	FP32	108	449.3					
6	Cloud TPU v3	TensorFlow, TPU 1.15.dev	bfloat16					1,038,510		
7	HL-102-Goya PCI-board	Synapse-V0.2.0	INT8		0.25			14,451		
8	Intel® Xeon® Platinum 9200	PyTorch Caffe2	INT8	0.49	1.35		29,203	5,966		
9	DELL ICL i3 1005G1	OpenVINO	INT8	3.72	13.58	6.67	507	101	218	
10	Supermicro 4029 8xT4	TensorRT 6.0, CUDA 10.1, cuDNN 7.6.3	INT8				141,807	44,978	60,872	
11	Supermicro 6049 20xT4		INT8					113,592	143,084	
12	SCAN 3XS (TitanRTXx4)		INT8				222,388	66,250	91,780	
13	SDM855 QRD	SNPE V1.30	INT8	3.02	8.95					
14	Centaur Technology Reference Design v1.0	TF + Centaur ML Library	INT8 /UINT8	0.33	1.05	1.54	6,042	1,218	652	
15	Hailo-8	Hailo SDK	INT8	7.98	12.48	13.3	645	545	373	
16	Tencent Cloud	PyTorch Caffe2	INT8	0.52	1.49		24,866	5,169		
17	furiosa-single-fpga	FuriosaRuntime/TFLite	INT8	3.29		11.55				

4 COMPARISON OF EDGE DEVICES

Edge AI signifies that AI algorithms are processed locally on a hardware device. The global edge AI hardware market is expected to grow from USD 423.34 Million in 2018 to USD 1,929.21 Million by 2026 at a CAGR of 20.9% during the forecast period 2019-2026 [13]. The MLMark datasheet provides complete performance information of five Edge AI platforms: Jetson AGX [14], Jetson Nano [15], Raspberry Pi 4 [16], HiKey970 [17] and Coral Dev Board [7]. Additional information is added about energy consumption and cost for comparison. In Table 7, latency and throughput of the workloads MobileNet and SSDMobileNet are used to represent performance. To compare the platforms, the following metrics are used: (1) Latency (MS), (2) Latency Per Power Unit (MS/W), and (3) Latency Per USD (MS/USD).

Latency results for each platform are shown in Figure 1. A lower latency value indicates better performance as the system is more responsive. Jetson AGX has the lowest latency value whereas the Raspberry Pi 4 has a high latency value. The Jetson AGX is designed

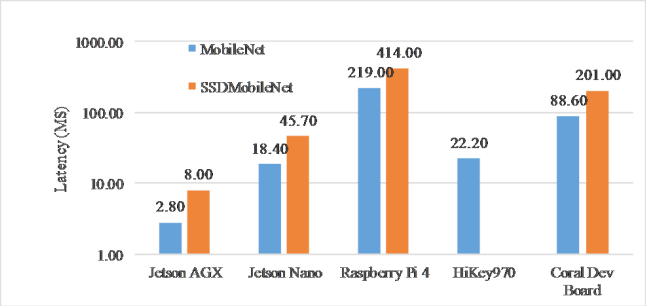

Figure 1: Latency (ms)

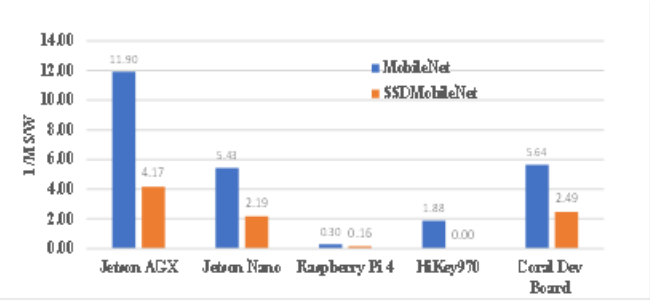

Figure 2: (1/Latency) per power unit

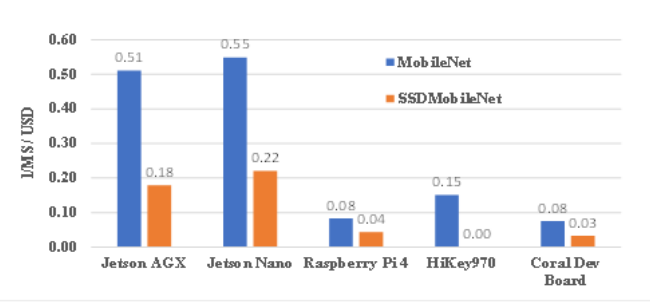

Figure 3: (1/Latency) per USD

for AI inferencing capabilities on edge devices and is focused on computing density. The Raspberry Pi 4 is designed to be a low power and low performance device for general computing tasks.
Figure 2 compares the power efficiency of each platform. The power efficiency of each platform is evaluated by comparing how much performance is gained per power unit (Watts). For a more intuitive comparison the inverse of the latency, $\frac{1}{latency}$, is used to describe the performance. The Jetson AGX has the best performance gain per wattage whereas the Raspberry Pi 4 has the lowest.
Figure 3 draws out the cost comparison in relation to each platform's performance. Jetson Nano and Jetson AGX are cost effective as they have good performance per USD due to being designed specifically for deep learning. As is apparent in the above figures, using GPUs as accelerators results in overall better latency and throughput results which means better cost and power efficiency.

5 CONCLUSION

Based on the results of MLPerf and MLMark and information from manufactures' websites, the gathered data is studied, factors relating to performance are analyzed, and performance of some edge AI devices is compared.

Due to the limitation of the number of results and unclear information, the analysis of the performance of language translation and the comparison in multistream and server scenarios is not possible. Later, the study will be expanded to include these scenarios.

From the point-of-view of end users, the following is required:

1. A well-organized category based on various applications and/or different scenarios. The current closed division of MLPerf shows a 5-order of magnitude difference in performance [18] which makes it difficult for users to make comparisons.
2. Clear conditions such as data type for each individual result.
3. Power consumption information.

ACKNOWLEDGMENTS

Liu would like to thank the CfR award of College of Science and Health of William Paterson University. Su would like to thank the ART awards of William Paterson University.

REFERENCES

[1] Q. Zhang *et al.*, A Survey on Deep Learning Benchmarks: Do We Still Need New Ones? International Symposium on Benchmarking, Measuring and Optimization, 2018
[2] A. Ignatov *et al.*, AI Benchmark: All about Deep Learning on Smartphones in 2019, http://ai-benchmark.com
[3] MLMark A EEMBC benchmark, retrieved from https://www.eembc.org/mlmark/scores.php
[4] P. Torelli and M. Bangale, Measuring Inference Performance of Machine-Learning Frameworks on Edge class Devices with the MLMark™ Benchmark, MLMark white paper, retrieved from https://www.eembc.org/techlit/articles/MLMARK-WHITEPAPER-FINAL-1.pdf
[5] MLPerf Inference v0.5 Results, retrieved from https://mlperf.org/inference-results/
[6] P. Matson *et al.*, MLPerf: An Industry Standard Benchmark Suite for Machine Learning Performance, IEEE Micro, Volume: 40 , Issue: 2 , March-April 1 2020
[7] Dev Board data sheet, retrieved from https://coral.ai/docs/dev-board/datasheet/
[8] V. Reddi *et al.*, MLPerf Inference Benchmark, 2020 ACM/IEEE 47th Annual International Symposium on Computer Architecture (ISCA), Nov. 2019
[9] NVDLA, retrieved from http://nvdla.org/
[10] P. Teich, Tearing Apart Google's TPU 3.0 AI Coprocessor, May 2018, retrieved from https://www.nextplatform.com/2018/05/10/tearing-apart-googles-tpu-3-0-ai-coprocessor/
[11] Supermicro specification, 2020, retrieved from https://www.supermicro.com/products/system/4U/6049/SYS-6049GP-TRT.cfm
[12] Intel®Xeon®Platinum 9282 Processor, retrieved from https://www.intel.com/content/www/us/en/products/processors/xeon/scalable/platinum-processors/platinum-9282.html
[13] GLOBE NEWSWIRE, Global Edge AI Hardware Market, retrieved July 17, 2020 from https://www.globenewswire.com/news-release/2020/
[14] JETSON AGX XAVIER, retrieved from https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-agx-xavier/
[15] JETSON NANO Bringing the Power of AI to Millions of Devices, retrieved from https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-nano/
[16] Raspberry Pi 4 Tech Spec, retrieved from https://www.raspberryPi.org/products/raspberry-Pi-4-model-b/
[17] HiKey 970 specification, retrieved from https://www.hackerboards.com/product/253/
[18] S. Ward, Benchmark scores highlight broad range of machine-learning inference performance, Embedded Blog, Nov. 2019, retrieved from https://www.embedded.com/benchmark-scores-highlight-broad-range-of-machine-learning-inference-performance/