Technical Report

Moreh vLLM Performance Evaluation: Llama 3.3 70B on AMD Instinct MI300X GPUs

Moreh, Inc.

August 2025

Contents

Overview 1
AMD Instinct MI300X GPU 1
Experimental Setup 3
Output TPS, TTFT, and TPOT 4
Trade-Off Between Latency and Throughput 6
Conclusion 7
Appendix: Raw Data 8

Moreh vLLM Performance Evaluation: Llama 3.3 70B on AMD Instinct MI300X GPUs

Overview

Moreh develops software to enable various AI workloads – from pretraining to inference – to run efficiently on non-NVIDIA accelerators, with a particular focus on AMD GPUs.

vLLM is one of the most widely adopted inference engines for running LLM services in research, enterprise, and production environments. It is developed by a strong open-source community with contributions from both academia and industry, and provides broad support for various models, hardware, and optimization techniques. AMD is also contributing to the project to make vLLM run on AMD GPUs and the ROCm software stack. Nevertheless, most optimizations in vLLM still target NVIDIA GPUs, and the performance of AMD GPU hardware has yet to be fully utilized.

Moreh vLLM is our optimized version of vLLM, designed to deliver superior LLM inference performance on AMD GPUs. It supports the same models and features as the original vLLM, while maximizing computational performance on the AMD CDNA architecture. This is achieved through Moreh's proprietary compute and communication libraries, along with model-level optimizations and vLLM engine-level modifications.

This technical report evaluates the inference performance of Meta's Llama 3.3 70B model on Moreh vLLM. We conduct comprehensive testing across various input/output lengths and concurrency levels. Compared to the original vLLM, Moreh vLLM delivers an average of **1.68x** higher throughput (total output tokens per second). Furthermore, it reduces latency metrics (time to first token and time per output token) by an average of **2.02x** and **1.59x**, respectively. In conclusion, adopting Moreh vLLM unlocks the full potential of AMD MI300 series GPUs, enabling them to serve as an efficient inference system.

AMD Instinct MI300X GPU

The AMD Instinct MI300X GPU presents a compelling alternative to NVIDIA's H100. It provides 1.32x higher theoretical compute performance, 2.4x larger memory capacity, and 1.58x higher peak memory bandwidth compared to the H100. In particular, its significantly larger memory capacity and bandwidth are a major advantage for optimizing LLM inference. Table 1 compares the detailed hardware specifications.

Table 1. Comparison between NVIDIA H100 and AMD MI300X

Items	H100 SXM	MI300X	Relative (MI300X/H100)
Basic facts			
Architecture	Hopper	CDNA3	
Form factor	SXM5 module	OAM module	
Lithography	TSMC 4 nm	TSMC 5 nm	
# SMs (compute units)	132	304	
# cores	16,896	19,456	
# tensor/matrix cores	528	1,216	
Peak engine clock	1,830 MHz	2,100 MHz	
TDP	700 W	750 W	
Peak theoretical performance (dense)			
FP32 vector	66.9 TFLOPS	163.4 TFLOPS	2.44x
TF32 matrix	494.7 TFLOPS	653.7 TFLOPS	1.32x
FP16/BF16 matrix	989.4 TFLOPS	1,307.4 TFLOPS	1.32x
FP8 matrix	1978.9 TFLOPS	2,614.9 TFLOPS	1.32x
INT8 matrix	1978.9 TOPS	2,614.9 TOPS	1.32x
GPU memory			
Technology	HBM3	HBM3	-
Capacity	80 GB	192 GB	2.40x
Peak bandwidth	3.35 TB/s	5.3 TB/s	1.58x
Cache and scratchpad			
L1D + scratchpad	256 KB per SM	32+64 KB per SM	0.38x
L2/L3	50 MB L2	32 MB L2, 256 MB L3	-
Connectivity (H2D: host to device, D2D: device to device within a server)			
H2D interface	PCIe Gen5 x16	PCIe Gen5 x16	-
H2D bandwidth	128 GB/s	128 GB/s	1.00x
D2D interface	NVLink Gen4	Infinity Fabric Gen4	-
D2D bandwidth	900 GB/s	896 GB/s	0.996x
# GPUs per server	8	8	1.00x

AMD has also released the MI325X and MI355X as successors to the MI300X, which are direct competitors to NVIDIA's H200 and B200 GPUs, respectively. Since these next-generation models are also based on the AMD CDNA3 architecture, all optimizations within Moreh vLLM will continue to apply seamlessly. We plan to publish performance evaluation results on the MI325X and MI355X in the near future and are always open to partners who can provide development and testing servers.

Experimental Setup

All experiments were conducted on an MI300X server configured as follows:

- **Server:** Lenovo ThinkSystem SR685a V3
- **CPU:** 2x AMD EPYC 9534 (128 cores in total, 2.45 GHz)
- **GPU:** 8x AMD Instinct MI300X OAM
- **Main Memory:** 2,304 GB (24x 96 GB)
- **Operating System:** Ubuntu 22.04.4 (Linux kernel 5.15.0-25-generic)
- **ROCm Version:** 6.8.5

We used the open-source vLLM 0.9.2 (tag v0.9.2 of https://github.com/ROCm/vllm) as a baseline for comparison. This was the latest versions available at the time of testing.

The Llama 3.3 70B model was executed in parallel across 2 GPUs of the server with a tensor parallelism (TP) of 2. Performance was measured using vLLM's benchmark_serving tool. We chose 64 different combinations of input sequence length (ISL), output sequence length (OSL), and concurrency, as shown in Table 2.

The experimental setup was determined through discussions with one of our customers in Korea.

Table 2. Various request patterns used for performance measurement

Input sequence length (ISL)	Output sequence length (OSL)	Concurrencies
1024	1024	1, 2, 4, 8, 16, 32, 64, 128, 256, 512
1024	4096	1, 2, 4, 8, 16, 32, 64, 128, 256, 512
4096	1024	1, 2, 4, 8, 16, 32, 64, 128, 256
4096	4096	1, 2, 4, 8, 16, 32, 64, 128, 256
16384	1024	1, 2, 4, 8, 16, 32, 64
16384	4096	1, 2, 4, 8, 16, 32, 64
32768	1024	1, 2, 4, 8, 16, 32
32768	4096	1, 2, 4, 8, 16, 32

Output TPS, TTFT, and TPOT

Output tokens per second (TPS), time to first token (TTFT), and time per output token (TPOT) are three key metrics for evaluating the performance of LLM inference.

- *Output tokens per second* measures the overall throughput of the system, indicating how many tokens the model can generate in one second across all concurrent requests.
- *Time to first token* captures the initial latency – the time from when a request is sent until the very first token is produced.
- *Time per output token* indicates the average time taken to generate each subsequent token after the first one.

Output tokens per second is directly tied to service cost (dollar per token). The latter two metrics are important for user-perceived responsiveness. Together, measuring these three metrics provides a comprehensive view of inference performance, balancing cost and user experience.

Figure 1 shows a graph comparing output tokens per second. Figure 2 and Figure 3 present graphs comparing the mean time to first token and the mean time per output token, respectively. The raw data can be found in the appendix. Moreh vLLM achieves 1.68x higher total output tokens per second, 2.02x lower time to first token, and 1.59x lower time per output token compared to the original vLLM. Especially, it can be seen that the time to first token for long input sequences is reduced by about 3-4x. This demonstrates that simply replacing the software with Moreh vLLM on the same AMD MI300 series GPU system can reduce costs while improving user experience.

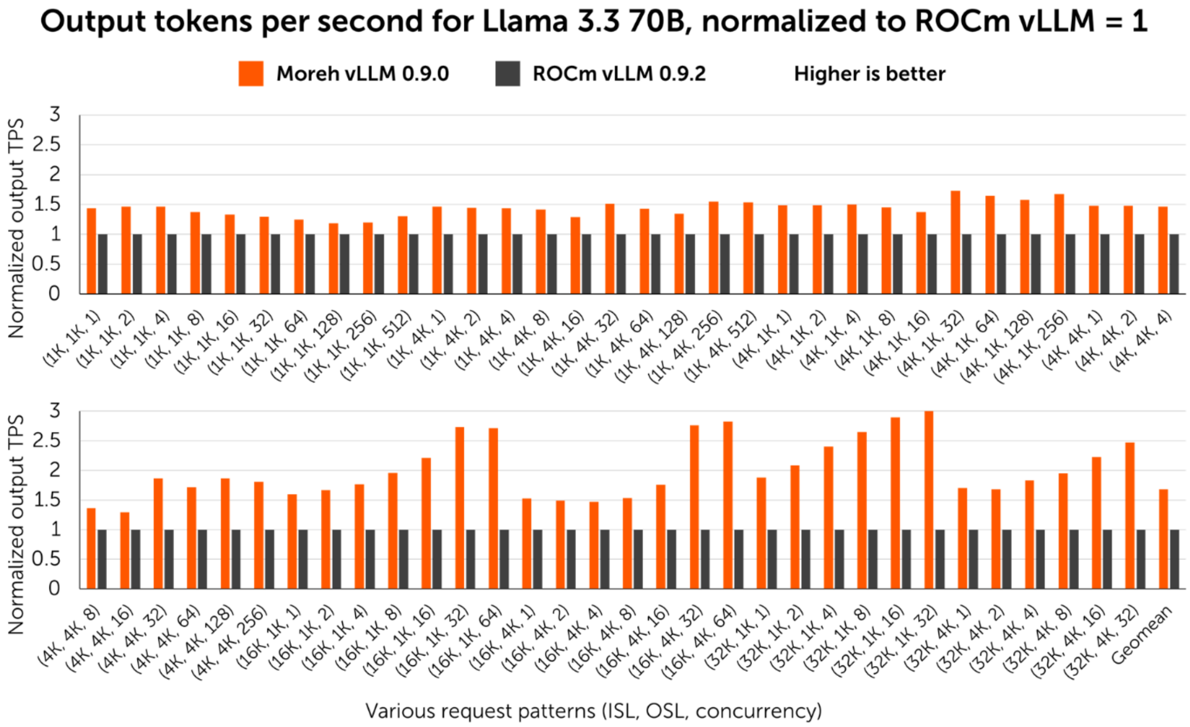

Figure 1. Output tokens per second for various request patterns. Higher is better. Moreh vLLM shows an average of 1.68x higher performance.

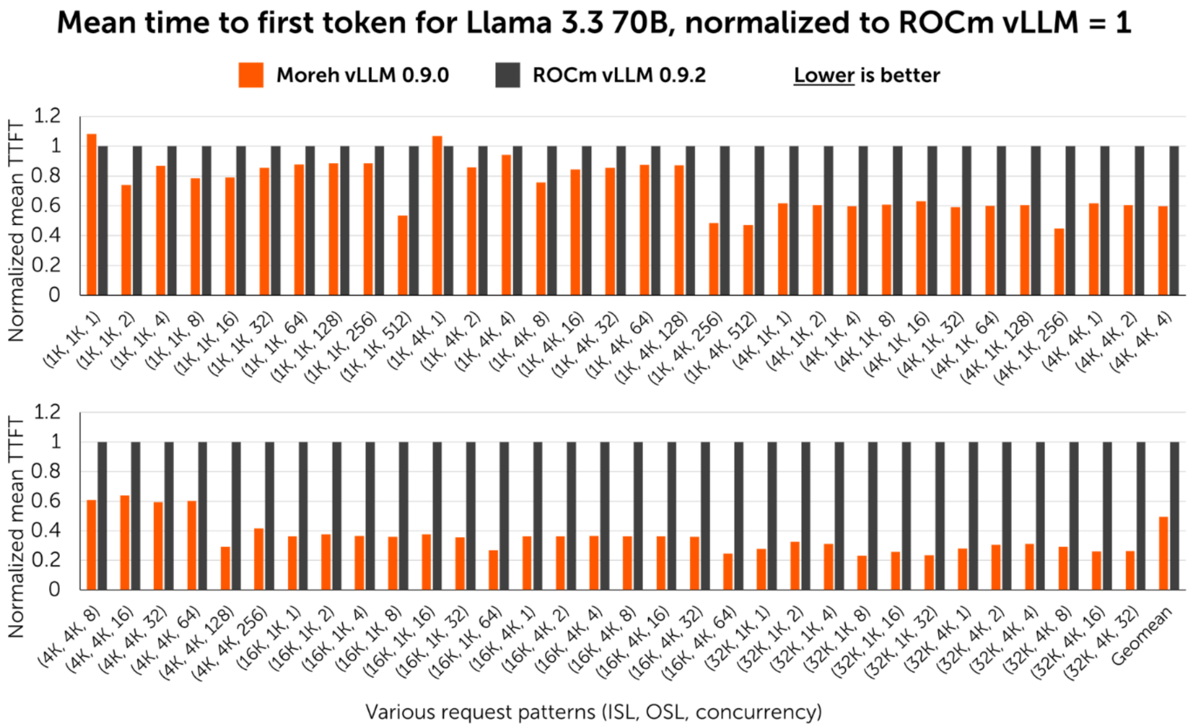

Figure 2. Mean time to first token for various request patterns. Lower is better. Moreh vLLM shows an average of 2.02x lower latency.

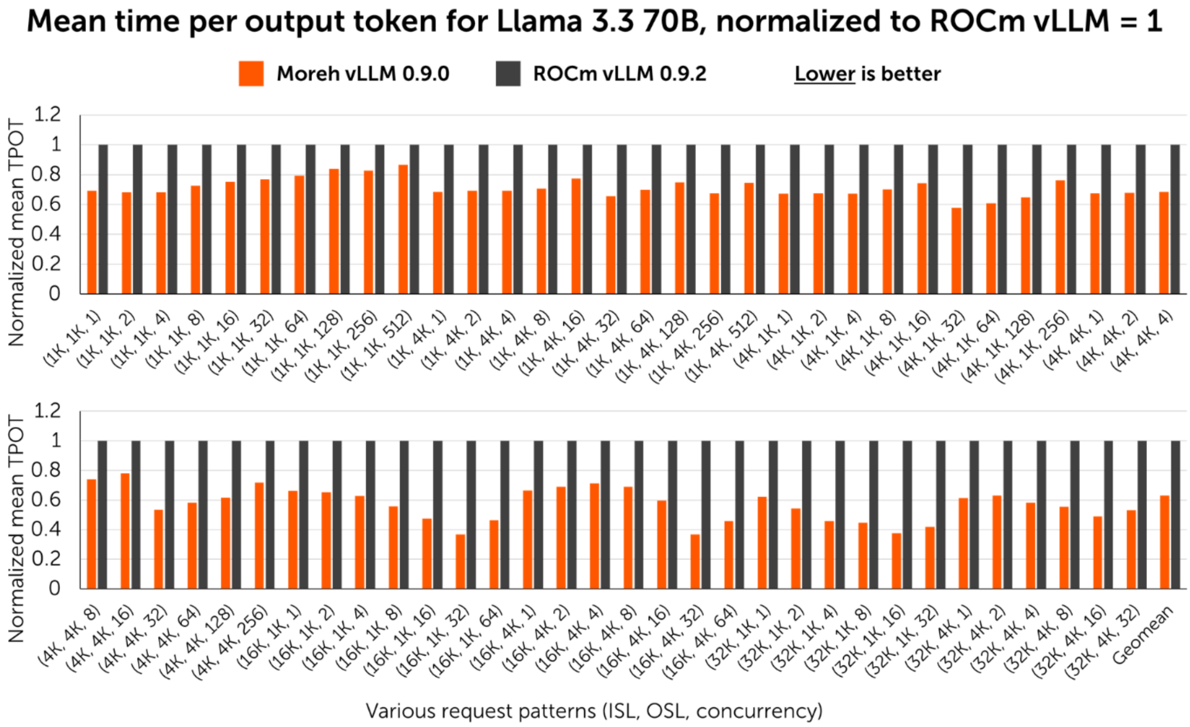

Figure 3. Mean time per output token for various request patterns. Lower is better. Moreh vLLM shows an average of 1.59x lower latency.

Trade-Off Between Latency and Throughput

LLM inference involves an inherent trade-off between latency and throughput. Increasing the maximum concurrency of a vLLM instance improves throughput but also increases latency, while decreasing concurrency improves latency but lowers throughput.

Figure 4 illustrates these latency-throughput trade-off curves for the original vLLM and Moreh vLLM across various request patterns (input/output sequence lengths). Overall, the closer the graph shifts toward the upper left, the better the performance characteristics.

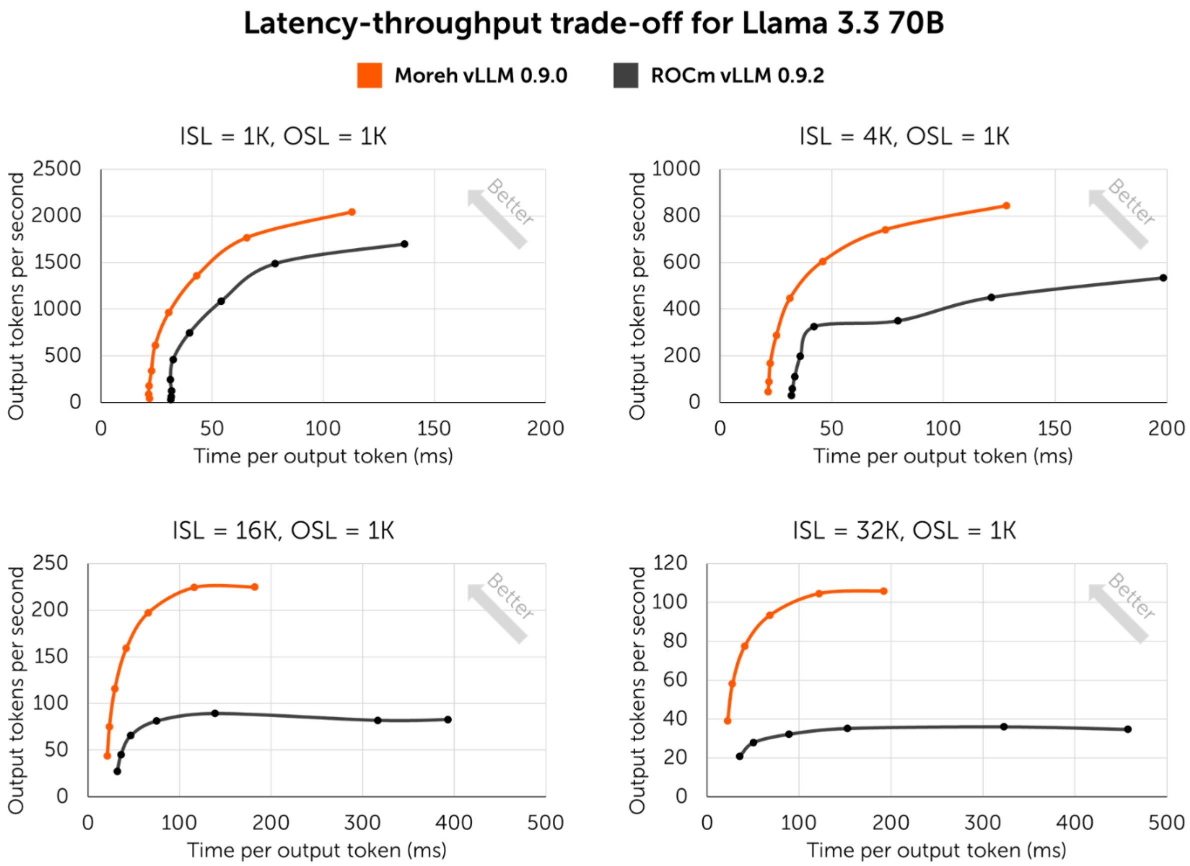

Figure 4. Trade-off curves between time per output token (latency) and output tokens per second (throughput), for different input/output sequence lengths.

Conclusion

Moreh vLLM incorporates various techniques to optimize inference for the Llama 3.3 70B model, including proprietary GPU libraries, model-level optimizations, and modifications to the vLLM engine. As a result, Moreh vLLM delivers substantial performance improvements over the original open-source vLLM across various inference metrics. By adopting Moreh vLLM on AMD MI300 series GPU servers, LLM services can reduce costs while simultaneously improving latency. Moreh also provides a service that optimizes a customer's proprietary AI model on AMD GPUs and delivers on-demand vLLM for it.

Appendix: Raw Data

Request patterns			Moreh vLLM 0.9.0			ROCm vLLM 0.9.2		
ISL	OSL	Conc.	Output TPS	Mean TTFT (ms)	Mean TPOT (ms)	Output TPS	Mean TTFT (ms)	Mean TPOT (ms)
1024	1024	1	45.78	135.29	21.73	31.83	124.99	31.33
1024	1024	2	91.86	284.04	21.51	62.69	384.49	31.55
1024	1024	4	181.14	483.21	21.63	123.70	555.86	31.82
1024	1024	8	338.48	1006.69	22.67	246.33	1283.47	31.25
1024	1024	16	614.11	1763.25	24.35	462.08	2231.31	32.45
1024	1024	32	965.95	2671.73	30.52	746.72	3130.89	39.79
1024	1024	64	1358.69	4160.03	43.03	1087.77	4740.10	54.17
1024	1024	128	1767.02	6990.31	65.53	1487.50	7888.65	78.23
1024	1024	256	2041.74	12583.04	112.81	1698.16	14228.23	136.48
1024	1024	512	1943.38	35083.98	214.87	1486.67	65643.93	247.76
1024	4096	1	45.02	135.45	22.19	30.79	126.91	32.46
1024	4096	2	89.05	244.80	22.40	61.57	285.52	32.42
1024	4096	4	176.02	527.92	22.60	122.28	560.70	32.58
1024	4096	8	341.79	977.62	23.17	241.49	1289.85	32.82
1024	4096	16	615.38	1779.26	25.57	476.26	2108.47	33.08
1024	4096	32	982.47	2694.00	31.91	647.83	3153.45	48.63
1024	4096	64	1410.14	4196.48	44.35	988.11	4793.58	63.59
1024	4096	128	1918.01	6975.04	65.00	1428.61	8008.78	86.75
1024	4096	256	1876.53	30423.74	113.7	1213.31	62773.22	168.56
1024	4096	512	1785.69	185865.73	200.69	1165.71	394792.00	269.59
4096	1024	1	45.83	383.69	21.47	30.79	621.03	31.90
4096	1024	2	88.80	728.27	21.83	59.65	1200.94	32.38
4096	1024	4	167.57	1454.55	22.45	111.89	2436.41	33.38
4096	1024	8	287.21	2770.95	25.15	198.28	4557.93	35.89
4096	1024	16	447.28	4614.56	31.25	325.10	7300.47	42.06
4096	1024	32	605.38	7121.00	45.85	350.18	12025.56	79.58
4096	1024	64	741.32	12418.42	74.03	450.90	20668.38	121.50
4096	1024	128	844.25	23456.94	128.18	534.36	38741.89	198.31
4096	1024	256	807.87	101856.85	195.04	481.90	227455.60	255.53
4096	4096	1	46.22	383.97	21.55	31.23	622.55	31.88
4096	4096	2	91.69	727.46	21.64	62.02	1203.84	31.96
4096	4096	4	178.29	1462.17	22.08	121.58	2444.09	32.31
4096	4096	8	318.11	2787.25	24.47	233.75	4583.57	33.10
4096	4096	16	534.72	4686.07	28.77	413.81	7323.32	36.87
4096	4096	32	782.38	7193.24	39.13	419.74	12098.96	73.26
4096	4096	64	1036.77	12493.28	58.62	603.81	20709.96	100.85
4096	4096	128	1097.68	34312.02	98.58	588.83	117844.43	160.31
4096	4096	256	1080.39	292503.63	137.00	596.32	703045.41	190.83
16384	1024	1	44.01	1599.19	21.18	27.51	4403.14	32.08
16384	1024	2	75.13	3230.73	23.48	45.03	8601.76	36.03
16384	1024	4	115.89	5321.31	29.31	65.60	14580.84	46.74
16384	1024	8	159.02	8644.20	41.82	81.22	24015.67	75.00
16384	1024	16	197.34	15480.16	65.83	89.32	41180.99	138.77
16384	1024	32	224.46	26841.86	115.97	82.01	75121.89	316.33

16384	1024	64	224.80	88571.25	181.91	82.85	330196.48	393.05
16384	4096	1	46.22	1603.89	21.25	30.29	4413.81	31.94
16384	4096	2	85.45	3127.08	22.64	57.33	8617.41	32.79
16384	4096	4	145.94	5356.25	26.10	99.46	14670.96	36.63
16384	4096	8	229.04	8735.11	32.78	149.67	24081.44	47.56
16384	4096	16	326.67	14930.43	45.29	185.99	41306.99	75.89
16384	4096	32	412.69	26941.34	70.85	149.45	75092.75	192.97
16384	4096	64	413.45	161154.35	100.85	146.27	652484.39	219.77
32768	1024	1	39.18	3567.35	22.06	20.85	12823.28	35.48
32768	1024	2	58.27	7041.32	27.45	27.92	21551.50	50.59
32768	1024	4	77.53	11117.55	40.71	32.23	35874.57	89.05
32768	1024	8	93.42	17772.43	68.16	35.23	76185.88	152.58
32768	1024	16	104.54	31838.01	121.61	36.08	123289.8	322.63
32768	1024	32	105.87	96997.96	191.96	34.72	413077.39	457.57
32768	4096	1	43.29	3581.72	22.23	25.44	12827.20	36.18
32768	4096	2	73.84	6706.98	25.45	43.81	21866.27	40.31
32768	4096	4	116.03	11247.87	31.72	63.35	36042.72	54.33
32768	4096	8	163.65	17891.77	44.48	83.98	61660.08	80.16
32768	4096	16	208.70	32039.47	68.74	93.79	123571.61	140.27
32768	4096	32	213.97	159336.37	96.77	86.57	603899.48	181.96

To learn more, please visit our website (https://moreh.io) or contact us (contact@moreh.io).