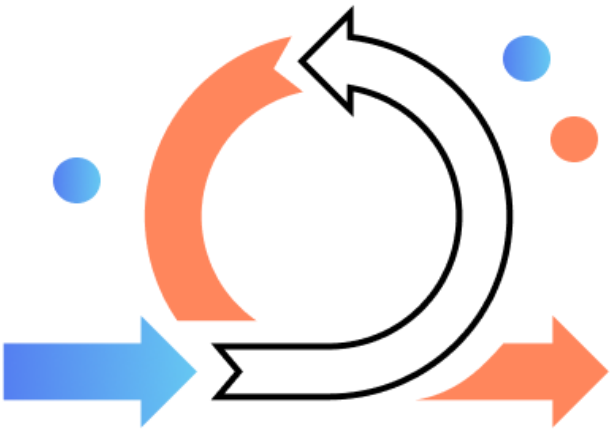
LuxiEdge Performance validation report

Independent Validation of LuxiEdge Deterministic Packed Prefill Execution Versus vLLM on NVIDIA H100



To: Eric Waller, LuxiEdge
e@ewaller.com

Prepared by: TestFort

Date: July 2026

Contact Person: Nora Laievska, Director of Partnerships & Growth
laievska@qarea.us

Table of Contents

Executive Summary

Test Environment

Performance vs vLLM Default Mode

Performance vs vLLM Batch-Invariant Mode

Determinism Receipts

Stability Assessment (10-minute Soak)

Limitations and Non-Claims

Recorded Anomalies and Deviations

Conclusion

Appendix

Contacts

Executive Summary

TestFort conducted an independent validation of LuxiEdge, a deterministic packed prefill executor, against vLLM on a single NVIDIA H100 80GB HBM3 GPU. The workload was a prefill-heavy comparison at Qwen2-7B, FP16, batch size 16, sequence length 128, tensor parallelism 1, with prefix caching disabled.

Key Findings

- Against vLLM default mode: LuxiEdge median throughput ratio 0.806 (28,374.7 vs 35,203.1 positions/s), 3.10% lower GPU-board energy per position.
- Against vLLM batch-invariant mode: LuxiEdge median throughput ratio 0.918 (28,374.7 vs 30,914.3 positions/s), 9.15% lower GPU-board energy per position.
- LuxiEdge is slower than vLLM in both comparisons measured. No claim is made that LuxiEdge is faster than vLLM under any configuration tested.
- All scoped determinism receipts passed for both engines, with one exception reported as N/A (LuxiEdge request-order invariance — the required interface is not exposed at this commit).
- 10-minute soak test: all four throughput/energy drift figures within ±3% (PASS).
- Delivery package: testfort_delivery_20260723_tar.gz, SHA256 64bbae09b3e94f4bc42f97f75346c7b628f15af1f8842d90a3b28423915638e9 — contains full raw logs, per-run harness JSON, and receipts underlying every figure in this report.

Energy figures are cumulative NVML GPU-board energy for a prefill executor workload only; they are not wall-plug, facility, rack, or full-serving measurements. Full scope limitations are listed in Section 7.

Test Environment

Parameter	Value
Repository	github.com/RegularJoe-CEO/attention-transformer-v2
Commit pin	d9ad7e06ff727440a1237e9fbb709d8bad0ba452
Working tree	Clean, detached HEAD
Built binary SHA256	ed828a908a6f20b5e4645c19497a1948ffd805d546dc5a b4a4a3e10166287c92
Model	qwen2-7b-instruct-luxi
GPU	NVIDIA H100 80GB HBM3
GPU UUID	GPU-0fab7d65-c9c7-f2dd-0e7b-4a9efc010912
Driver	580.126.09
Enforced power limit	700.00 W
Max SM clock	1980 MHz
MIG state	Disabled
OS / Kernel	Ubuntu 22.04.5 LTS, kernel 6.17.0-1008-nvidia
CUDA toolkit (nvcc)	12.4, V12.4.131
Rust / Cargo	rustc 1.97.1 / cargo 1.97.1
LuxiEdge stack	Python 3.11.10, torch 2.4.1+cu124, FlashAttention 2.8.3.post1
vLLM stack	Python 3.11.10, torch 2.11.0+cu130, vLLM 0.25.1
Attention backend (both engines)	FLASH_ATTN, FlashAttention version 3

The two engines ran in separate staged Python environments as required by protocol §3.2. Neither environment was modified during testing; both loaded an identical staged model directory and selected an FA3-class attention backend.

The clean-clone rebuild produced a binary SHA256 that differs from the two protocol reference values. This is expected and explicitly permitted: the source commit pin is authoritative, and compiler/path timestamps routinely change the resulting hash. The commit pin was exact, the working tree clean, the build successful, the Flash backend active, and the fallback count zero on every run.

Performance vs vLLM Default Mode

Five 60-second runs per engine (B=16, S=128, FP16).

Metric	vLLM default	LuxiEdge
Median positions/s	35,203.1	28,374.7
Median J/position	0.019316	0.018718
Median positions/J	51.8	53.4

Comparison	Value	Threshold	Verdict
LuxiEdge / vLLM default speed ratio	0.8060	≥ 0.75	PASS
LuxiEdge median J/position ≤ vLLM default	0.018718 vs 0.019316	≤	PASS
LuxiEdge energy advantage vs default vLLM	3.10%	—	reported

Primary gate: PASS. LuxiEdge is slower than default-mode vLLM by a factor of 0.806 while consuming 3.10% less GPU-board energy per prefill position.

Performance vs vLLM Batch-Invariant Mode

Three 30-second runs per engine with VLLM_BATCH_INVARIANT=1 (confirmed active at preflight and via kernel registration logs).

Metric	Default vLLM	Batch-invariant vLLM	LuxiEdge
Median positions/s	35,203.1	30,914.3	28,374.7
Median J/position	0.019316	0.020604	0.018718

Threshold	Measured	Result
vLLM batch-invariant throughput loss ≥ 10%	12.18%	PASS
vLLM batch-invariant J/position increase ≥ 5%	6.67%	PASS
LuxiEdge speed ratio vs batch-invariant vLLM ≥ 0.85	0.9178	PASS

Threshold	Measured	Result
LuxiEdge J/position ≥ 5% lower than batch-invariant vLLM	9.15%	PASS

LuxiEdge remains slower than vLLM in batch-invariant mode (ratio 0.918), while using materially less energy per position in this comparison as well.

Determinism Receipts

Receipts are reported separately per engine; LuxiEdge and vLLM cover different determinism properties and are not directly equivalent.

#	Receipt	Result
1	LuxiEdge Flash vs serial numerical error (B1, B32)	PASS
2	LuxiEdge cross-sequence contamination	PASS — zero deviation
3	LuxiEdge Flash fixed-input hashes (3 repeats)	PASS — stable
4	LuxiEdge full-stack fingerprint (all layers/rows, B1/B16/B32)	PASS
5	LuxiEdge three cold-process generations	PASS — identical output
6	vLLM batch-invariant repeated-output hash (3 runs)	PASS — identical
7	vLLM target alone vs target with companions	PASS — identical
8	vLLM original / reversed / shuffled batch order	PASS — identical
9	LuxiEdge request-order invariance	N/A — interface not exposed at this commit

Receipt 9 is reported as not-applicable, not as a pass or a failure: the packed executor API at the tested commit does not expose the stable request-ID shuffle interface this test requires. No request-order invariance claim is made for LuxiEdge on this basis.

Stability Assessment (10-minute Soak)

Soak order was reversed relative to the main test (LuxiEdge first, then vLLM default) to guard against order and temperature effects.

Engine	Metric	Main-test median	Soak aggregate	Drift	Gate ±3%
LuxiEdge	positions/s	28,374.7	28,145.1	-0.81%	PASS
LuxiEdge	J/position	0.018718	0.018851	+0.71%	PASS
vLLM default	positions/s	35,203.1	35,533.9	+0.94%	PASS
vLLM default	J/position	0.019316	0.019180	-0.71%	PASS

Metric	Value
Failed requests	0
System crashes	0
LuxiEdge median power (T1 + soak)	477–491 W
vLLM default median power (T1 + soak)	676–682 W (enforced limit 700 W)

Stability verdict: PASS. All four drift figures are well inside the ±3% gate. LuxiEdge performs the same counted work more slowly at substantially lower instantaneous power, which is the mechanism behind its lower energy per position.

Limitations and Non-Claims

This validation is deliberately narrow. The following are explicitly outside its scope:

- Measures prefill positions, not generated serving tokens. max_new_tokens=1; the generated token is excluded from all counted work.
- No decode or online serving throughput; no continuous-batch serving comparison was measured.
- Only B=16 was tested. No B32 performance claim is made.
- Not equal-power: the pod did not permit a 500 W power cap. The enforced 700 W limit is recorded as measured; no equal-power result is reported.
- Energy figures are cumulative NVML GPU-board energy only — not wall-plug, rack, cooling, facility, PUE, carbon, or electricity-cost measurements.
- Determinism receipts are scoped to this GPU, driver, and engine versions. No cross-GPU, cross-driver, or full-server determinism is claimed.

- This is not a claim that default-mode vLLM is nondeterministic, and not a claim of identical determinism semantics between engines.
- This is not an MLPerf benchmark, latency test, training test, or full serving test. No batch-1 latency, time-to-first-token, or tail-latency figures are reported.

LuxiEdge is slower than vLLM in both comparisons measured here — ratio 0.806 against default mode and 0.918 against batch-invariant mode. No claim is made that LuxiEdge is faster than vLLM under any configuration.

Recorded Anomalies and Deviations

Three entries were logged during this engagement. None is a protocol blocker.

1. nvidia-smi -q -d MIG returned a non-zero exit code because MIG is not supported on this device. MIG state was independently confirmed as Disabled via a separate identity query. No data was lost.

2. The clean-rebuild binary SHA256 differed from both protocol reference values — explicitly permitted, since the commit pin, clean tree, successful build, active Flash backend, and zero fallback count were all confirmed.

3. A RUNPOD_API_KEY value was inadvertently captured by an environment dump into log files and the session transcript. The credential value was redacted before packaging; no measurement data or result was altered.

One further item was observed but not classified as a deviation: the batch-invariant vLLM arm logged a single JIT kernel compilation during the smoke phase, outside all timed measurement windows.

Conclusion

1. LuxiEdge achieved a median throughput ratio of 0.806 versus default-mode vLLM (28,374.7 vs 35,203.1 positions/s), with 3.10% lower GPU-board energy per position.

2. Against vLLM batch-invariant mode, LuxiEdge achieved a ratio of 0.918, with 9.15% lower energy per position.

3. LuxiEdge is slower than vLLM in both measured comparisons. No configuration tested showed LuxiEdge faster than vLLM.

4. All applicable determinism receipts passed (8 of 9); one property (request-order invariance) is not testable on the current LuxiEdge API and is reported as N/A, not PASS.

5. A 10-minute soak test confirmed stability, with all throughput and energy drift figures within ±3% of the main-test medians.

6. 6. Results are scoped to a single H100 GPU, FP16 prefill at B=16/S=128, and do not extend to decode, online serving, or cross-hardware claims.

Appendix

A. Delivery Package

File: testfort_delivery_20260723_tar.gz

SHA256: 64bbae09b3e94f4bc42f97f75346c7b628f15af1f8842d90a3b28423915638e9

Contains: raw per-run harness JSON and logs, GPU identity captures, determinism receipts, arithmetic audit, and anomaly log underlying every figure in this report.

B. Repository and Build

Repository: github.com/RegularJoe-CEO/attention-transformer-v2

Commit pin: d9ad7e06ff727440a1237e9fbb709d8bad0ba452

Built binary SHA256:
ed828a908a6f20b5e4645c19497a1948ffd805d546dc5ab4a4a3e10166287c92

Contacts



Nora Laievska

Director of Partnerships & Growth

laievska@qarea.us

Tel: +1 310 388 93 34 USA, UK

testfort.com