vLLM serving features and answer flips: why unchanged accuracy can hide changes

27 November 2026 · 5 min read · blogs.iictp.org/vllm-serving-features-answer-flips-audit

Short answer: an audit of vLLM serving features by Sharven Rane, posted on arXiv on 22 November 2026, found that settings such as prefix caching, chunked prefill and speculative decoding can change which requests are answered correctly even when overall accuracy stays the same. With default kernels, those changes were no larger than the effect of reordering requests, except for an FP8 key-value cache. It matters for anyone approving a serving change on accuracy alone.

TL;DR The short version

- An arXiv audit of 22 November 2026 found that vLLM features such as prefix caching, chunked prefill and speculative decoding can change which requests are answered correctly while overall accuracy stays the same.
- With default kernels these flips were no larger than the effect of reordering requests, except for an FP8 key-value cache.
- Do not approve a serving change on accuracy alone: compare per-request answers against a baseline.

Read online, listen to it, or change the text size: https://blogs.iictp.org/vllm-serving-features-answer-flips-audit

What is an answer flip?

A flip is a request whose graded result changes from correct to wrong or the reverse between two serving configurations. Accuracy is an average, so equal numbers of flips in each direction leave it unchanged.

How the audit worked

1. Serve one instruction model of four billion parameters in vLLM.
2. Run nine configurations that each change one feature, plus five replicas of a reference setup.
3. Grade 3,380 requests from five benchmarks in every run.
4. Compare each candidate with a restart floor, from a fresh process, and a scheduling floor, from the same requests in a different order.
5. Repeat with batch-invariant kernels, which make results independent of how requests are batched.

What is new compared with accuracy gating

The paper measures paired differences and sets them against the engine's own run-to-run noise. A change is judged against that floor, not against an assumption that an exact feature reproduces text perfectly.

Results reported by the author

- Every configuration stayed within one point of accuracy with default kernels.
- Seven features documented as exact flipped 1.47% (95% interval 1.13 to 1.85) to 1.78% (1.39 to 2.20) of requests, none above the scheduling floor of 1.72%.
- An FP8 key-value cache flipped 3.36% (2.80 to 3.96), as many as FP8 weights.
- With batch-invariant kernels, all seven exact features, including speculative decoding, reproduced reference text on every request, while the FP8 cache still flipped 3.91% (3.28 to 4.53).
- A gate comparing against a single replica pair rejected an unchanged configuration.

This is an arXiv preprint or workshop paper. Treat its numbers as the authors' own results, not as independently confirmed findings.

Limitations and open questions

- One model, one engine version (vLLM 0.33.0) and one consumer GPU.
- One candidate run per feature and three restart replicas.
- Only request order was varied; real traffic changes arrival times and lengths too.
- The zero-noise result applies to those kernels in that version.
- Judged or preference-based tasks may behave differently.
- The author does not claim the features reduce accuracy.

Sources were read on 27 November 2026. Products, rules and data change, so check the linked originals before you rely on any detail.

How to certify a serving change

1. Save graded results per request, not only the average.
2. Where your hardware and version allow, test under batch-invariant kernels so the floor is zero.
3. Otherwise measure a scheduling floor from at least three replicas.
4. Report the paired flip rate with a confidence interval beside the accuracy change.
5. Treat an FP8 key-value cache as a lossy change that needs its own approval.
6. Repeat after engine upgrades.

What it means for inference engineers

Users experience individual answers, not averages. Record which configuration served a request so that a complaint can be reproduced. If you are moving between versions, check the vLLM 0.34 upgrade notes and rerun the audit, since this result was measured on 0.33.0.

Frequently asked questions

- **Does speculative decoding change answers in vLLM?** In this audit, only to the level of normal scheduling noise by default, and not at all under batch-invariant kernels.
- **Is prefix caching exact?** It behaved like other exact features in these tests.
- **What about an FP8 KV cache?** It flipped more requests and should be treated as lossy.
- **Is accuracy parity enough?** The author argues it is not.

Related reading

- vLLM 0.34 upgrade guide
- Speculative decoding tuning
- Local LLM Practitioner certification

> Related credential: Local LLM Practitioner. Explore the exam facts and syllabus.

Sources and further reading

1. Sharven Rane. Exact on Paper: Which LLM Serving Shortcuts Flip Graded Answers, Measured Against the Noise of Reordering the Same Requests. arXiv 2611.29663. 22 November 2026 (preprint). arxiv.org/abs/2611.29663

Published by IICTP, an independent technical credential body, on 27 November 2026. The current version of this post is at https://blogs.iictp.org/vllm-serving-features-answer-flips-audit. More guides: blogs.iictp.org.