

We published someone else's benchmark. Here's the fix.

October 5, 2026 · by Mark Bartlett

[Download this post as PDF](#)

InsiderLLM Weekly issue 24 – October 5, 2026

No new article this week. I spent it checking our own numbers. One benchmark on this site turned out to be someone else's, and four corrections from the cleanup change what you'd pay for a card. The main piece is the benchmark. First, the corrections.

New This Week

- No new article this week (September 28 to October 5).

Updated:

- **How Much Does It Cost to Run LLMs Locally?** "Pays for itself in 3-6 months" is now 17-25 months for a used 3060 setup and 7-12 years for a 3090. Buy a 3090 for privacy and no limits, not to save money. 📖 [The guide](#).
 - **Used RTX 4090 prices are up about \$470 since late July**, to \$2,575-3,010 on eBay sold listings, and 28 pages are corrected. 📖 [The buying guide](#).
 - **The budget build is now "\$500", not "under \$450"**, after the 3060 reprice, with a 24 GB Tesla P40 build at about the same money as the alternative. 📖 [The build](#).
 - **NVFP4 is about 7 percent smaller than Q4_K_M, not 25 percent.** Checked against llama.cpp's own block layout: 4.5 bits per weight against about 4.8. 📖 [The FP4 explainer](#).
 - **33 pages in all** carry a dated correction from this week, each saying what it used to say.
-

Quick Hits

- **A 2.5-bit Qwen3.8-Flash-Next with "32GB" in the name.** That's a 32 GB GPU on a 64 GB Strix Halo, not a 32 GB box: the file is 53 GiB, with 352 of 512 experts kept per layer and no retraining after the cut. Its 37.6 tok/s needs a fork for speculation. On a 12 GB card with 32 GB of RAM, the 3-bit measured 8.7 tok/s here. 📖 [The GGUF](#) · [our Flash-Next test](#).
- **aa-agentperf-local.** Artificial Analysis replays eight recorded agent tasks, 168 turns with full history each time, against your llama.cpp, vLLM, SGLang, LM Studio or Ollama server. It

reports p95 turn time, which is what an agent feels and empty-context tok/s hides. Results only compare at 64K context, and Ollama's are estimates. Apache 2.0. 📖 [The repo](#).

- **Apple no longer sells the M4 Mac mini.** The US store lists an M6 from \$899 and M5 Pro models from \$1,699 (apple.com, checked October 2). Any "\$599 Mac mini" advice is now used-market advice, ours included; the OpenClaw hardware page carries a dated note. 📖 [Apple](#) · [ours](#).
- **Three specialists don't make one generalist by default.** DN-MOPD distills math, code and instruction-following teachers into one Qwen3.5 student. Untreated, instruction-following feedback is 2.3 to 4.4 times as spread out and supplies 94 percent of the gradient at 4B, so the math barely transfers. Rescaling each teacher's feedback by its spread recovers most of it: +1.2 to +2.4 points on average at 9B, 4B and 2B. It's distillation, not weight merging, and the same lesson as our ten LoRA seeds: skills put into weights don't add up on their own. 📖 [The paper](#) · [our ten seeds](#).
- **vLLM 0.31.** Adds Qwen3.8-Flash-Next and GLM-5.3-Flash, and can quantize to NVFP4 or MXFP4 at load time. Before upgrading, per-request multimodal kwargs now need `--trust-request-mm-kwargs`. 📖 [Release notes](#).
- **GLM-5.3-Flash in NVFP4, from NVIDIA.** Blackwell only, tested on GB200, served by vLLM or SGLang. It's a 320B MoE with 18B active, and the repo is about 205 GB. FP4 can't shrink a total parameter count that size onto a desktop card, 5090 included. 📖 [The model card](#).
- **Gemini 4 Argon.** Closed, and open only to Google's cyber partners for now, with no date for anyone else. 📖 [TechCrunch](#).

The 11 tok/s That Wasn't Ours

Since July, the Qwen 3.6 35B guide said a small drafter model dragged the 35B down to 11 tok/s on our RTX 3060. It was never our 3060. The 11 tok/s was Codacus's figure from his GTX 1060 6 GB rig, and in May we quoted it with his name. A July edit moved the sentence into the 3060 section and the name fell off. Our benchmark dataset then logged it as our own measurement and divided it by our 3060's baseline to get 0.28x.

So I ran the test. Same 3060, same file, nine prompts, three reps per config. A Qwen 3.5 0.8B drafter runs at 0.42x of plain `-ncmoe 24`. z-lab's DFlash drafter doesn't fit at `-ncmoe 24`; at `-ncmoe 26`, the closest config it fits, it lands at 0.58x. Drafters still lose on this card, just not by the 0.28x the old row claimed.

The explanation was wrong too. We'd blamed PCIe, experts streaming across the bus. llama.cpp only sends offloaded expert work to the card for batches of 32 tokens or more, so every 16-token

verification batch is computed on the CPU, reading experts out of DDR4. That verify step costs 7.3x a normal decode step on this card. PCIe sits near 165 MB/s the whole time.

If one row could lose its source, others could too. The audit found the July 3060 figures with no record behind them, and I re-measured every one against a rule written before the first run: within 3 percent keeps the published value. All but one held, the largest move 1.8 percent. The exception, `-ncmoe 20` at 42.6 tok/s, no longer loads on this card on either build, so it's withdrawn. Every re-measured figure now links a public record. One older 3060 number, 28.5 tok/s at `-ngl 40`, still has none, and it's flagged.

On a 12 GB card: skip the drafter and run `-ncmoe 24`. In the re-measure it served a 16K-token prompt and still decoded at 35.8 tok/s.

Codacus's video is where the test started, and his number is back under his name.

📖 **The corrected guide is [here](#).** 📖 **The records, public, are [here](#).**

That's the week. If you have a 12 GB card and a 35B MoE, run `-ncmoe 24` with no drafter. If you're pricing a build, re-check any number you saved before October. And if a figure on this site looks like it came from somewhere else, tell me. This one did.

New here, reading this on the web? [Subscribe](#) and the next one lands in your inbox.

— Mark, InsiderLLM

Run a drafter under offload on your own card? Seen a number here you couldn't reproduce? Reply, or hit me at hello@insiderllm.com. I read everything.

A weekly email with every new guide and measured benchmark.

[Subscribe — free, no spam](#)

Source: <https://insiderllm.com/blog/newsletter-2026-10-05/>

Free guides for running AI locally