Qwen 35B-A3B on RTX 3090: 157 tok/s With No Offload (2026)

July 28, 2026 · by Mark Bartlett

Download this post as PDF

InsiderLLM Weekly issue 15 – July 28, 2026

Three things this week and they're the same thing: a number everybody repeats, us included, that nobody actually measured. I ran the sweep on my own 3090 and the headline moved 56%. Then the used-GPU market rewrote the reasoning behind advice we'd given for a year.

2,471 MiB to Spare on a 24GB Card

Our Qwen3.6-35B-A3B guide carried ~101 tok/s for a 3090, a figure from the Unsloth model card and Amine Raji's writeup that we'd repeated for months. I ran it on Miu, my own 3090, headless: **157.66 tok/s**. Fifty-six percent above what we'd been citing.

The bigger finding is the one nobody states plainly. **The whole model fits on the card.** No `--cpu-moe`, no `-ncmoe`, no offload at all. Peak VRAM was 21,652 MiB against the 24,123 a headless 3090 reports free — 2,471 MiB spare, with 8K of context already in the cache. And it barely moves with depth: 156.25 at 4K, 153.24 at 8K.

Before you set that 157.66 next to anything else, know what it is. It's a `llama-bench` tg128 figure at up to 8K depth with default KV cache. The community numbers came from llama-server at 65K context with q8_0 KV quantization — heavier, more realistic work. Some of the 56% is my rig; some of it is that a synthetic decode benchmark flatters everyone. Both are honest; they answer different questions. I'd rather say so than let you quote mine into an argument it can't win.

📖 **The full sweep, every `-ncmoe` value, is here.**

The $1,200 Card That Loses to the $300 One

Now the other end. At `-ncmoe 40`, every expert pushed out to system RAM, the 3090 does **36.11 tok/s**. My 12GB RTX 3060, running its recommended `-ncmoe 24`, does **38.9**. A used

3090 runs about $1,200 and a used 3060 about $300. Configured badly, the expensive card loses to the cheap one — while sitting on 24GB of VRAM with 3GB of it in use.

The honest caveat, because these are two different boxes: the 3090 rig runs DDR4-2667 against the 3060's DDR4-2133, a 1.25x memory ratio, and at matched full offload the gap is 1.28x. At max offload you're measuring RAM, not the GPU — the residual is the faster DIMMs, not the faster card.

And one you can act on tonight: **if that 3090 is driving your monitor, the two fastest configs won't load at all.** A desktop session holds about 4.4 GiB, so `-ncmoe 0` never loads and `-ncmoe 2` dies creating the context. `-ncmoe 4` is the practical top of the table with a display attached, at about a 10% cost — or SSH in with the session stopped and get the fully resident config back.

The Cheaper Card Was Never Cheaper

Separately, I repriced the catalog against firsthand eBay sold listings. Two comparisons need reading again — not because the recommendation changed, but because the reason behind it did.

The used 3090 now runs about **$450 more** than a 4070 Ti Super, not less — a reversal of the gap those pages were built on. It doesn't move the pick: 24GB is still the card that decides what you can load. Want 32B+ models or long contexts, buy the 3090 anyway and know you're paying $450 for the headroom instead of pocketing it. If 14B is enough, take the 4070 Ti Super. Same one tier down — the 3060 12GB costs about **$90 more** than the faster 3060 Ti and is still the right call, because 12GB runs 14B at Q4 while 8GB caps you at 7-8B.

The number that explains both: price per gigabyte is nearly flat. $50/GB against $47/GB on the first pair, $25/GB against $26/GB on the second. The used market stopped paying for frame rates and started paying for VRAM, so the 3060 Ti isn't overpriced anymore; it's correctly priced as a card with 8GB. What changed isn't which card to buy — it's that the VRAM premium is explicit now.

📖 **The 3090 vs 4070 Ti Super rewrite is here.**

Every Number Above Is Downloadable

All of it lives in one file. The open benchmark dataset is at 50 rows — two rigs, every flag, error bars, repetition counts, whether a display was attached, and the two configs that failed to load.

It's CC BY 4.0: use it in a paper, a model card, a spreadsheet, or sell something built on it. All I ask is attribution.

📖 **The dataset, and the raw JSON, are here.**

Quick Hits

- **Kimi K3's weights landed July 27, the day Moonshot promised.** Issue 14 said they were due that date and that you still wouldn't be able to run them. Both held. 2.8T total, 104B active per token, and Moonshot's own deployment guidance starts at 64 accelerators. The license is its own document — MIT-derived, effectively unrestricted for small teams, and not the Apache 2.0 several outlets printed. 📖 The K3 breakdown.
- **The Hugging Face intruder has a name, and it's OpenAI.** A joint disclosure on July 21 named the agent behind issue 14's breach story: two OpenAI models — GPT-5.6 Sol and an unreleased, more capable one — running an ExploitGym eval with cyber refusals dialed down. They broke out of the sandbox to go after the benchmark's own answers. 📖 The breach piece, with the attribution update.
- **Use `-ngl 99`, not your model's layer count.** llama.cpp counts one more offloadable layer than there are blocks, so `-ngl 40` on a 40-block model strands one in RAM. On our 3060 that cost Qwen3-14B 20%: 28.5 tok/s against 35.9, with nothing in the logs to flag it. 📖 Why your local LLM is slow.

That's the week. If you run a 24GB card and you've been offloading experts out of habit, go check what your card is actually holding. That's the issue in one sentence.

New here, reading this on the web? Subscribe and the next one lands in your inbox.

— Mark, InsiderLLM

Ran the sweep on your own card and got something different? Reply, or hit me at hello@insiderllm.com. I read everything.
