Running 70B Models Locally — Exact VRAM by Quantization

February 14, 2026 · by Mark Bartlett

Download this guide as PDF

> **Quick Answer:** Llama 3.3 70B at Q4_K_M needs ~43-45GB of VRAM. That won't fit on any single consumer GPU. Your options: dual RTX 3090s (48GB total, ~16-21 tok/s, ~$1,700 for both), a Mac with 64-128GB unified memory (8-12 tok/s, $2,000-4,100), or a datacenter card like the A100 80GB. The RTX 5090's 32GB can squeeze Q3 but quality degrades noticeably. At Q4_K_M, 70B models retain excellent quality — larger models tolerate quantization better than small ones. The honest question: do you need 70B? Qwen 3 32B fits on a single 24GB GPU at Q4, runs 3-4x faster, and matches 70B on many benchmarks. Run 70B for complex reasoning and deep research. Run 32B for everything else.

📚 **More on this topic:** VRAM Requirements Guide · Quantization Explained · Multi-GPU Guide · Mac vs PC for Local AI · Used RTX 3090 Buying Guide

Running a 70B model locally is the line between “hobby” and “serious local AI.” On the other side of that line is reasoning that competes with GPT-4 and the ability to process complex problems without sending your data to the cloud.

The barrier is VRAM. A 70B model at full precision needs 141GB of memory. No consumer GPU comes close to that. Quantization brings it down to 43GB at Q4, which still won’t fit on a single RTX 4090 or 3090. You need either two GPUs, a Mac with enough unified memory, or a workstation-class card.

This guide gives you exact VRAM numbers at every quantization level, which hardware setups actually work, realistic speed expectations, and an honest assessment of when 70B is worth the investment versus running a 32B model instead.

The 70B Math

The formula is simple:

VRAM (GB) = Parameters (billions) × Bytes per parameter

At FP16 (2 bytes per parameter): 70B × 2 = **140GB**. That's the model weights alone. Context and framework overhead are extra.

Quantization compresses those weights:

Precision	Bytes per Param	Weight Size (70B)	With Overhead*
FP16	2.0	140 GB	~142 GB
Q8_0	1.0	70 GB	~75 GB
Q6_K	0.75	52.5 GB	~58 GB
Q5_K_M	0.625	43.75 GB	~50 GB
Q4_K_M	0.5	35 GB	~43 GB
Q3_K_M	0.375	26.25 GB	~35 GB
Q2_K	0.25	17.5 GB	~27 GB

*Overhead includes KV cache at 4K context, framework memory, and CUDA/Metal context. Real GGUF files are slightly larger than the theoretical minimum due to metadata and mixed-precision layers.

The theoretical calculation gets you in the ballpark. Real file sizes are what matter. See the next section.

Exact VRAM: Llama 3.3 70B and Qwen 2.5 72B

These are the two 70B-class models most people run locally. Numbers from actual GGUF builds on HuggingFace:

Llama 3.3 70B Instruct

Quantization	File Size	VRAM Needed (4K ctx)	VRAM Needed (8K ctx)
FP16	141.1 GB	~143 GB	~148 GB
Q8_0	75.0 GB	~77 GB	~82 GB
Q6_K	57.9 GB	~60 GB	~65 GB
Q5_K_M	50.0 GB	~52 GB	~57 GB

Quantization	File Size	VRAM Needed (4K ctx)	VRAM Needed (8K ctx)
Q4_K_M	42.5 GB	~45 GB	~50 GB
Q3_K_M	34.3 GB	~37 GB	~42 GB
Q2_K	26.4 GB	~29 GB	~34 GB

Qwen 2.5 72B Instruct

Quantization	File Size	VRAM Needed (4K ctx)	VRAM Needed (8K ctx)
Q8_0	77.3 GB	~79 GB	~84 GB
Q6_K	64.4 GB	~66 GB	~71 GB
Q5_K_M	54.5 GB	~57 GB	~62 GB
Q4_K_M	47.4 GB	~50 GB	~55 GB
Q3_K_M	37.7 GB	~40 GB	~45 GB
Q2_K	29.8 GB	~32 GB	~37 GB

Qwen 2.5 72B is about 10-15% larger than Llama 3.3 70B at the same quantization because it has 72 billion parameters versus 70.6 billion, plus slightly different architectural choices. Both produce similar quality at the same quant level.

For a deeper understanding of what these quantization levels mean and how they affect output quality, see our quantization explainer.

→ Use our Planning Tool to check exact VRAM for your setup.

Context Length Eats Your VRAM

The tables above assume 4K or 8K context. But Llama 3.3 supports 128K tokens and Qwen 2.5 72B supports 128K too. The KV cache (where the model stores attention state for the conversation) grows linearly with context length.

KV Cache VRAM at 70B Scale

Context Length	KV Cache (FP16)	KV Cache (Q8)	KV Cache (Q4)
4K tokens	~2.4 GB	~1.2 GB	~0.6 GB

Context Length	KV Cache (FP16)	KV Cache (Q8)	KV Cache (Q4)
8K tokens	~4.9 GB	~2.4 GB	~1.2 GB
16K tokens	~9.8 GB	~4.9 GB	~2.4 GB
32K tokens	~14 GB	~7 GB	~3.5 GB
64K tokens	~28 GB	~14 GB	~7 GB
128K tokens	~39 GB	~20 GB	~11 GB

At 32K context with FP16 KV cache, you're adding 14GB on top of the model weights. On a dual RTX 3090 setup (48GB total) running Llama 70B Q4_K_M (42.5GB file), that leaves about 5.5GB for everything. With a 14GB KV cache, you're already over.

This is why most 70B setups run with 4K-8K context and why the 128K advertised context length is mostly theoretical for consumer hardware. You can extend it with quantized KV cache (Ollama and llama.cpp both support this), but even then, 32K+ context on 48GB total VRAM is tight.

Hardware That Can Actually Run 70B

Single Consumer GPUs: Mostly Can't

GPU	VRAM	Best 70B Quant	Context	Verdict
RTX 4060 (8GB)	8 GB	None	—	Not happening
RTX 3060 12GB	12 GB	None	—	Not happening
RTX 4060 Ti 16GB	16 GB	None	—	Not happening
RTX 3090 / 4090	24 GB	None (Q2_K is 26.4GB)	—	Doesn't fit even at Q2
RTX 5090	32 GB	Q2_K or Q3_K_M	~4K tokens	Technically works. Quality is poor at Q2, marginal at Q3.

The RTX 5090 is the only consumer GPU that can load a 70B model at all. Q3_K_M (34.3GB file) fits with about 4K context, but quality degrades noticeably at Q3 and you have zero headroom. It's a proof-of-concept, not a daily driver.

Dual GPU Setups

This is where 70B becomes practical on consumer hardware. Two GPUs pool their VRAM.

Setup	Total VRAM	Best Quant	Context	Speed	Cost (Feb 2026)
2× RTX 3090	48 GB	Q4_K_M	~4-8K	16-21 tok/s	~$1,700
2× RTX 4090	48 GB	Q4_K_M	~4-8K	20-25 tok/s	~$3,200+
2× RTX 5090	64 GB	Q4_K_M	~16-32K	25-30 tok/s	~$4,000+

Dual RTX 3090s ($1,700 total) is the budget path. 48GB runs Llama 3.3 70B at Q4_K_M with 4-8K context. You get 16-21 tokens per second, which is readable but noticeably slower than the 40+ tok/s you'd get from a 32B model on a single card. See our multi-GPU guide for setup instructions.

Dual RTX 5090s ($4,000+) with 64GB total opens up longer context. Q4_K_M with 16-32K tokens is comfortable, and Q5_K_M becomes viable for better quality.

Both setups require a motherboard with two PCIe x16 slots (or at least x16 + x8), a 1000W+ power supply, and good airflow. Two 3090s at full inference draw 700+ watts combined.

Workstation / Datacenter GPUs

GPU	VRAM	Best Quant	Context	Speed	Price
A6000	48 GB	Q4_K_M	~4-8K	12-16 tok/s	~$2,200 used
A100 80GB	80 GB	Q5_K_M	~16K+	19-22 tok/s	~$8,000+ used

The A6000 at $2,200 used gives you the same 48GB as dual 3090s in a single card, no multi-GPU hassle. But it's slower for inference (smaller memory bandwidth) and costs $500 more.

Mac (Unified Memory)

This is where Macs win. Unified memory lets the entire RAM pool serve as model memory.

Config	Unified Memory	Best Quant	Context	Speed	Price
Mac Mini M4 Pro 48GB	48 GB	Q4_K_M	~4-8K	6-8 tok/s	~$1,900
Mac Studio M4 Max 64GB	64 GB	Q4_K_M	~16K	8-10 tok/s	~$2,500
Mac Studio M4 Max 128GB	128 GB	Q6_K	~32K+	10-12 tok/s	~$4,100

Mac speeds are slower than dual NVIDIA GPUs because unified memory bandwidth (273-546 GB/s) is lower than GDDR6X (936 GB/s per 3090). But the Mac loads the model at all, which a single 24GB GPU can't. And it does it silently, at 15 watts idle.

The M4 Max 128GB at $4,100 is the most comfortable 70B experience. Q6_K with long context, no fan noise, no multi-GPU setup. The tradeoff is speed. See our Mac vs PC comparison for the full breakdown.

Speed Expectations

70B models are slow. Set your expectations accordingly.

Hardware	Llama 3.3 70B Q4_K_M	Context
2× RTX 3090 (48GB)	16-21 tok/s	4-8K
2× RTX 4090 (48GB)	20-25 tok/s	4-8K
2× RTX 5090 (64GB)	25-30 tok/s	16K
Mac M4 Max 128GB	10-12 tok/s	32K
Mac M4 Max 64GB	8-10 tok/s	8K
A100 80GB	19-22 tok/s	16K
Single 24GB GPU + CPU offload	1-5 tok/s	4K

For comparison, Qwen 3 32B at Q4_K_M on a single RTX 3090 runs at 35-45 tok/s. A 70B model on dual 3090s runs at about half that speed while costing twice the hardware.

CPU offloading (splitting the model between GPU and system RAM) technically works but is painfully slow. The PCIe bus becomes the bottleneck, dropping generation to 1-5 tok/s. At that speed, you're waiting 10-20 seconds for a single sentence. It's fine for testing. It's not usable for daily work.

Quality vs Quantization at 70B

Good news: 70B models tolerate quantization better than smaller models. Research confirms that models above 30B parameters retain ~99% of FP16 accuracy at 4-bit quantization, while 7B models lose 2-5%.

Quality by Quant Level

Quantization	Quality Retention	Best For
Q8_0	~99.5% of FP16	Maximum quality when VRAM allows
Q6_K	~99% of FP16	Excellent. Hard to distinguish from Q8 in practice
Q5_K_M	~97-99% of FP16	Great balance. Most users won't notice the difference
Q4_K_M	~95-97% of FP16	The sweet spot. Minor degradation on complex reasoning
Q3_K_M	~90-93% of FP16	Noticeable. Reasoning and math tasks suffer first
Q2_K	~80-85% of FP16	Severe. Unpredictable behavior on hard problems. Skip this.

Q4_K_M is the recommendation for almost everyone running 70B locally. The 3-5% quality loss versus FP16 is barely perceptible in normal use. You'd need benchmark suites to measure the difference reliably. The VRAM savings (142GB down to 43GB) make it the only practical option on consumer hardware.

Q3_K_M is where you start noticing. Math problems that Q4 handles cleanly will occasionally fail at Q3. Multi-step reasoning chains break more often. If you're running on an RTX 5090 and Q3 is your only option, it works. Just know you're leaving quality on the table.

Q2_K is not worth running. At 70B, even the higher quantization tolerance can't save Q2 from significant output degradation. If Q2 is your only option, run a 32B model at Q4 instead. You'll get better results.

When 70B Is Worth It

Run 70B For:

Complex reasoning. Multi-step logic problems, mathematical proofs, scientific analysis. The gap between 32B and 70B is widest here. A 70B model at Q4 catches errors and follows chains of reasoning that a 32B model misses.

Deep research and analysis. Summarizing long documents, comparing multiple sources, identifying inconsistencies. 70B models have broader knowledge and make fewer factual errors.

Nuanced writing. When you need precise tone control, subtle arguments, or professional-grade output. 70B models handle ambiguity and subtext better.

Skip 70B For:

Quick chat and Q&A. A 32B model answers "what's the capital of France" just as correctly, 3-4x faster.

Simple code generation. For boilerplate, function scaffolding, and straightforward coding tasks, 32B coding models (Qwen 2.5 Coder 32B, DeepSeek-Coder-V2) are more than sufficient and much faster.

Anything speed-sensitive. If you need responses in under 2 seconds, 70B won't deliver. A 32B model at 40 tok/s starts generating immediately. A 70B model at 15 tok/s has noticeable latency.

The 32B Alternative

This is the honest question: in 2026, do you need 70B?

	Qwen 3 32B (Q4_K_M)	**Llama 3.3 70B (Q4_K_M)**
VRAM needed	~20 GB	~43 GB
Hardware	Single RTX 3090 ($850)	Dual RTX 3090 ($1,700)
Speed	35-45 tok/s	16-21 tok/s
Benchmark quality	~85-90% of 70B	Baseline
Complex reasoning	Good	Better
Creative writing	Competitive (85% human preference)	Good
Coding	Strong (DeepSeek R1 Distill 32B leads some benchmarks)	Strong

The gap has narrowed. Qwen 3 32B and DeepSeek-R1-Distill-Qwen-32B compete with 70B models on many benchmarks while using less than half the VRAM. On creative writing, Qwen 3 32B actually gets 85% human preference over larger models. On coding, DeepSeek R1 Distill 32B leads Llama 3.3 70B on several benchmarks.

70B still wins on complex multi-step reasoning and factual depth. If that's your primary use case, the hardware investment is justified. For everything else, a 32B model on a single GPU is faster and cheaper, with nearly the same quality.

Bottom Line

Running 70B locally requires either dual GPUs (2× RTX 3090 at $1,700), a Mac with 64GB+ unified memory ($2,000+), or a datacenter card. Q4_K_M is the quantization sweet spot: 43GB for Llama 3.3 70B, excellent quality retention. Below Q4, quality drops noticeably. Below Q3, don't bother.

The practical setup for most people: dual RTX 3090s with Llama 3.3 70B at Q4_K_M. You get 16-21 tok/s with 4-8K context. It's slower than a 32B model and costs twice the hardware. But for complex reasoning and research, the quality difference is real.

If you're not sure whether you need 70B, start with Qwen 3 32B on a single 24GB GPU. It handles 80-90% of tasks just as well. Upgrade to 70B when you consistently hit the quality ceiling on reasoning-heavy work.

Related Guides

- How Much VRAM Do You Need? — full VRAM chart for every model size
- What Quantization Actually Means — how Q4, Q6, Q8 affect quality
- Multi-GPU Local AI — tensor vs pipeline parallelism, setup guides
- Used RTX 3090 Buying Guide — the best 24GB GPU for budget AI
- Mac vs PC for Local AI — unified memory vs discrete VRAM
- Llama 3 Guide: Every Size — which Llama to pick
- Building a Distributed AI Swarm — multi-node alternative
- What Can You Run on 24GB VRAM? — the 32B sweet spot

Get notified when we publish new guides.

Subscribe — free, no spam

Source: https://insiderllm.com/guides/running-70b-models-locally-vram-guide/