Llama 4 Guide: Running Scout and Maverick Locally

February 16, 2026 · by Mark Bartlett

Download this guide as PDF

> **Quick Answer:** Llama 4 Scout has 109B total parameters but only activates 17B per token (MoE architecture). Q4 quantization needs ~55GB VRAM, so dual GPUs or a Mac with 64GB+ unified memory. The Unsloth 1.78-bit quant squeezes into 24GB VRAM at ~20 tok/s — usable but tight. Maverick (400B total, 17B active) needs 200GB+ at Q4 and is out of reach for most local setups. For budget builders with a single GPU, Llama 3.3 70B or Qwen 3 32B still deliver better bang per VRAM dollar. Scout's killer feature is native multimodal (text + images) with a 10M token context window — if you have the hardware for it.

📚 **More on this topic:** Llama 3 Guide · VRAM Requirements · GPU Buying Guide · What Can You Run on 24GB

Llama 4 Scout has 109 billion parameters but only uses 17 billion per token. Sounds amazing — until you check what it actually needs to run locally.

Meta's first MoE (Mixture of Experts) models are a genuine architecture shift from the dense Llama 3 family. Scout and Maverick activate only a fraction of their total parameters on each forward pass, which means faster inference than you'd expect from a 109B or 400B model. But MoE has a catch: you still need to load all the weights into memory, even though most of them sit idle on any given token.

This guide covers what you can actually run, what hardware you need, and when you should stick with Llama 3 instead.

The Llama 4 Family

Model	Total Params	Active Params	Experts	Context	Ollama
Scout	109B	17B	16	10M tokens	`ollama run llama4:scout`
Maverick	400B	17B	128	1M tokens	`ollama run llama4:maverick`
Behemoth	~2T+	TBD	TBD	TBD	Not released

All three use Mixture of Experts. Only Scout and Maverick are available. Behemoth has been teased but remains unreleased.

What Changed from Llama 3

	Llama 3	Llama 4
Architecture	Dense (all params active)	MoE (17B active per token)
Vision	Text-only (needed LLaVA adapter)	Native multimodal — early fusion
Max context	128K tokens	10M tokens (Scout)
Languages	8	12 (added Arabic, Hindi, Thai, Vietnamese)
Training data	15T tokens	40T tokens (Scout)
Smallest useful	8B	109B total (17B active)

The jump from dense to MoE is the headline. Llama 3.3 70B activates all 70B parameters on every token. Scout activates 17B out of 109B — faster inference per token, but you still carry the full 109B in memory.

VRAM Requirements

Here's the reality check. MoE models need VRAM proportional to total parameters, not active parameters.

Scout (109B total, 17B active)

Precision	VRAM Needed	Hardware	Practical?
BF16 (full)	~216GB	4x H100	Datacenter only
Q8	~109GB	2x H100 or Mac Studio 192GB	Prosumer
Q4	~55GB	2x RTX 3090, Mac M4 Max 128GB	Enthusiast
1.78-bit (Unsloth)	~24GB	Single RTX 3090/4090	Tight but works

Maverick (400B total, 17B active)

Precision	VRAM Needed	Hardware	Practical?
BF16 (full)	~800GB	7x H200	Datacenter
Q4	~200GB	3x H100 or exotic multi-GPU	Not for hobbyists
1.78-bit (Unsloth)	~100GB	2x 48GB GPUs	Extreme enthusiast

The bottom line: Scout at aggressive quantization is the only Llama 4 model that fits on consumer hardware. Maverick is a datacenter model for local purposes.

Hardware Reality Check

This is where it gets honest. If you're reading InsiderLLM, you probably have a single GPU, maybe two. Here's what actually works:

Your Hardware	Can You Run Scout?	Notes
RTX 3060 12GB	No	Not even close. Stick with 7-8B models
RTX 4060 Ti 16GB	No	Still too small. Qwen 3 14B is your sweet spot
RTX 3090 24GB	1.78-bit only	~20 tok/s. Usable for chat, degraded quality
RTX 4090 24GB	1.78-bit only	~20-25 tok/s. Same constraint, slightly faster
2x RTX 3090 (48GB)	Q4 works	Good experience. ~15-20 tok/s with tensor parallel
Mac M4 Max 128GB	Q4 works great	Unified memory shines here. MLX support.
Mac M4 Pro 48GB	1.78-bit only	Similar to single 24GB GPU
CPU only / mini PC	No	These models need real hardware

For most budget builders: Llama 3.3 70B or Qwen 3 32B still deliver better results per dollar of VRAM. Scout only makes sense if you have 24GB+ AND specifically want native multimodal or the massive context window.

→ Check what fits your hardware with our Planning Tool.

Running Scout with Ollama

If you have the hardware:

```
# Pull and run Scout (default quantization)
ollama run llama4:scout

# Pull a specific quantization
ollama pull llama4:scout:q4_K_M    # ~55GB, needs 2+ GPUs
```

For the 1.78-bit quantization that fits on 24GB, check Unsloth's Llama 4 page — they provide GGUF files optimized for single-GPU inference.

Multimodal (Images)

Scout handles images natively — no adapter needed:

```
# In Ollama chat, reference an image
ollama run llama4:scout
>>> Describe this image: /path/to/photo.jpg
```

This is a genuine upgrade over Llama 3.2 Vision 11B, which was a bolted-on vision adapter. Scout's early fusion architecture means the model reasons about text and images together from the ground up.

Benchmarks: Scout vs the Field

Scout's active parameter count (17B) puts it in an interesting spot — competing against much smaller models on efficiency while targeting larger models on quality.

Quality Benchmarks

Model	Active Params	MMLU	MATH	IFEval	GPQA
Llama 4 Scout	17B	79.6	50.3	—	—
Llama 3.3 70B	70B	79.3	41.6	92.1	—

Model	Active Params	MMLU	MATH	IFEval	GPQA
Llama 3.1 8B	8B	69.4	30.6	80.4	—
Qwen 3 32B	32B	83.1	—	—	—

Scout matches Llama 3.3 70B on MMLU (79.6 vs 79.3) while activating only 17B parameters per token — a 4x efficiency gain. It crushes Llama 3.3 on math (50.3 vs 41.6 MATH).

But Qwen 3 has overtaken both on raw benchmarks. If you're choosing purely on text quality and don't need vision, Qwen 3 32B fits in 24GB at Q4 and benchmarks higher.

Inference Speed (Estimated, Consumer Hardware)

Setup	Model	Quant	Speed
RTX 3090 24GB	Scout	1.78-bit	~20 tok/s
RTX 4090 24GB	Scout	1.78-bit	~20-25 tok/s
2x RTX 3090 48GB	Scout	Q4_K_M	~15-20 tok/s
RTX 3090 24GB	Llama 3.1 8B	Q4_K_M	~67 tok/s
RTX 3090 24GB	Llama 3.3 70B	Q4 (offload)	~8 tok/s

The MoE architecture helps here: despite being a 109B model, Scout generates tokens at speeds closer to a 17B dense model. You still pay the VRAM tax for all 109B weights, but inference is fast once they're loaded.

When to Use Llama 4 vs Stay on Llama 3

Use Scout if:

- You need vision + text in one model (not a bolted-on adapter)
- You have 24GB+ VRAM and accept 1.78-bit quality, OR 48GB+ for Q4
- You want the 10M token context window for massive document analysis
- You're on a Mac with 64GB+ unified memory

Stay on Llama 3.1 8B if:

- You have less than 16GB VRAM
- You don't need vision

- You want the fastest possible tok/s on modest hardware

Stay on Llama 3.3 70B if:

- You need the best Llama text quality and have the VRAM
- You're already running it and it works for your use case
- You need the massive fine-tune ecosystem (thousands of community models)

Consider Qwen 3 32B instead if:

- You want top-tier text quality on 24GB VRAM
- You don't need multimodal
- You care more about benchmark scores than the Llama ecosystem

---

The MoE Tradeoff, Explained

Mixture of Experts is not free magic. Here's the deal:

The win: Only 17B parameters activate per token, so inference (token generation) is as fast as a 17B dense model. You get 109B-level quality at 17B-level speed.

The cost: All 109B parameters must be loaded into VRAM. The model is huge on disk and in memory, even though most weights sit idle on any given token. This is why Scout needs ~55GB at Q4 despite "only" using 17B params.

The implication for local users: MoE models are bandwidth-bound, not compute-bound. The bottleneck is loading all those expert weights, not doing math on them. This means VRAM capacity matters more than GPU compute speed — which is why a Mac with 128GB unified memory (lower bandwidth but massive capacity) can outperform a single RTX 4090 (fast but only 24GB) for these models.

What About Maverick?

Maverick uses 128 experts instead of Scout's 16, with the same 17B active parameters. At Q4, it needs ~200GB VRAM. At 1.78-bit, it's ~100GB.

```
ollama run llama4:maverick    # Only if you have serious hardware
```

For local users, Maverick is realistically out of reach unless you have a multi-GPU server or a maxed-out Mac Studio. If you're in that territory, it exists and it's good — but that's not most people reading this guide.

Getting Started

If you have a 24GB GPU and want to try Scout:

1. Update Ollama: `ollama --version` (needs a recent version for Llama 4 support)
2. Pull Scout: `ollama run llama4:scout`
3. For 1.78-bit on 24GB: grab the Unsloth GGUF from HuggingFace and load manually
4. Test with an image to see native multimodal in action

If you're on a single GPU under 24GB, skip Llama 4 entirely. Llama 3.1 8B, Qwen 3 14B, or Gemma 3 27B will give you a better experience on your hardware. Don't chase parameter counts — chase what actually fits.
