Local LLM Deployment on 24GB GPUs: Models & Optimizations

By InuitionLabs.ai • 5/27/2025 • 5 min read

large language models | local llms | gpu inference | llm deployment | vram optimization | quantization | gguf | inference frameworks

Running Large Language Models (LLMs) Locally on a 24GB GPU (RTX 5090)

Running large language models (LLMs) on local hardware has become increasingly feasible. This report explores the top LLMs that can be deployed on a single high-end GPU (NVIDIA RTX 5090 with 24 GB VRAM) – focusing on open-source models, with notes on a few closed models – and covers their architectures, VRAM requirements, speeds, context lengths, and use cases. We also discuss popular local inference frameworks (like **llama.cpp**, **vLLM**, **LM Studio**, **Ollama**) and optimization techniques (quantization, RoPE scaling, GGUF format) to maximize performance.

Hardware Considerations: 24 GB VRAM and Model Size

GPU VRAM and Model Parameters: The capacity of your GPU's VRAM primarily determines which models you can run. LLMs are often categorized by parameter count (e.g. 7B, 13B, 70B for 7 billion, 13 billion, 70 billion parameters). VRAM usage scales roughly linearly with model size and precision: for example, a 7B model in half-precision (FP16) may require ~14 GB VRAM, whereas a 13B model is ~26 GB (too large to fit 24 GB without compression) apxml.com. An RTX 5090 with 24 GB can handle *smaller models at full precision* or *larger models with quantization (compression)* apxml.com apxml.com. The RTX 40/50 series GPUs also offer high memory bandwidth and tensor core performance, which improve throughput for lower precision inference apxml.com apxml.com.

Quantization: Quantization reduces memory usage by using lower precision for model weights (and sometimes activations). Common formats include 8-bit (INT8) and 4-bit (INT4) weight compression. For instance, 4-bit quantization cuts memory roughly to one-quarter: a 7B model that needs ~14 GB in FP16 might use only ~4–5 GB in 4-bit form apxml.com. Popular quantization methods are **GPTQ** (post-training quantization for GPU), **bitsandbytes** (8-bit loader), and the **GGUF**/GGML 4-bit quant formats used by llama.cpp apxml.com apxml.com. These enable fitting larger models on 24 GB – with some quality trade-off. *Example:* LLaMA-2 70B in FP16 needs ~140 GB of memory, but in 4-bit (INT4) it's about 35 GB medium.com. This still exceeds 24 GB, but with **3-bit mixed-precision** quantization (ExLlama), it can be reduced to ~26 GB – just at the edge of a 24 GB GPU medium.com medium.com. Some layers can be kept at higher precision while others are lower, to balance performance and memory medium.com medium.com. In practice, many 70B models require splitting across two 24 GB GPUs or offloading some layers to CPU RAM, but advanced quantization (down to 3-bit or even 2-bit) and partial loading can make single-GPU operation *almost* possible medium.com medium.com.

Inference Speed: Speed is typically measured in tokens generated per second (tok/s). It depends on model size, quantization level, and the efficiency of the software. A larger model means more computation per token, so throughput drops as model size increases. On an RTX 4090/5090-class GPU, a 7B model might generate on the order of ~100–140 tokens/s, whereas a 30B+ model might do ~30–40 tokens/s under similar conditions news.ycombinator.com. For example, using the optimized *exllama* GPU backend, users reported ~140 tok/s for a 7B model and ~40 tok/s for a 33B model on a 24 GB GPU news.ycombinator.com. An independent benchmark with the Ollama engine (using 4-bit quantized models) showed LLaMA-2 13B-chat at ~71 tok/s and LLaMA-3.1 8B at ~95 tok/s on a 4090 databasemart.com. In the same test, a 32B model (Qwen-32B) reached ~34 tok/s and a 34B model ~37 tok/s databasemart.com. This illustrates the **inverse scaling** of speed with size. Quantization can improve speed slightly (by reducing memory bandwidth pressure) but sometimes at the cost of some GPU compute efficiency. The chart below shows an example of running a 32B model on a 24 GB GPU, reaching about 34 tok/s with ~90% VRAM utilization:

! https://www.databasemart.com/blog/ollama-gpu-benchmark-rtx4090?srsltid=AfmBOopzHPvUpo38QJ2k5oZrj_XyoTsTiYA94YGm69975KXojvo9SE78

DISCLAIMER

The information contained in this document is provided for educational and informational purposes only. We make no representations or warranties of any kind, express or implied, about the completeness, accuracy, reliability, suitability, or availability of the information contained herein.

Any reliance you place on such information is strictly at your own risk. In no event will IntuitionLabs.ai or its representatives be liable for any loss or damage including without limitation, indirect or consequential loss or damage, or any loss or damage whatsoever arising from the use of information presented in this document.

This document may contain content generated with the assistance of artificial intelligence technologies. AI-generated content may contain errors, omissions, or inaccuracies. Readers are advised to independently verify any critical information before acting upon it.

All product names, logos, brands, trademarks, and registered trademarks mentioned in this document are the property of their respective owners. All company, product, and service names used in this document are for identification purposes only. Use of these names, logos, trademarks, and brands does not imply endorsement by the respective trademark holders.

IntuitionLabs.ai is an AI software development company specializing in helping life-science companies implement and leverage artificial intelligence solutions. Founded in 2023 by Adrien Laurent and based in San Jose, California.

This document does not constitute professional or legal advice. For specific guidance related to your business needs, please consult with appropriate qualified professionals.