SCORED.TOOLS · ROUNDUP

Best Open Source LLM Fine-Tuning Tools 2026

Seven open source LLM fine-tuning tools ranked by actual usability, GPU efficiency, and training quality. No hype, just what works in production.

Fine-tuning an open source LLM in 2026 is no longer the dark art it was two years ago. The frameworks have matured, QLoRA is table stakes, and you can fine-tune a 70B model on a single consumer GPU if you pick the right tool. The bad news: there are now a dozen frameworks claiming to be the easiest, fastest, or most memory-efficient. Most of them aren't.

This roundup ranks the seven tools we'd actually reach for, based on running real fine-tuning jobs (Llama 3.x, Qwen 2.5, Mistral, Gemma 2) on hardware ranging from a single RTX 4090 to multi-node H100 clusters. Rankings reflect ergonomics, training throughput, memory efficiency, and how often the tool silently produces a broken checkpoint.

1. Axolotl — Score: 9.4/10

Axolotl is the framework most production teams converge on once they've outgrown notebook tinkering. It wraps HuggingFace transformers, PEFT, TRL, and DeepSpeed behind a single YAML config, so you describe what you want and Axolotl figures out the plumbing. Multi-GPU, FSDP, DeepSpeed ZeRO-3, QLoRA, ReLoRA, DPO, ORPO, KTO — all configured the same way. The defaults are sane, the example configs in the repo cover most modern architectures, and the maintainer community is responsive on Discord when something breaks.

Best for

Teams running multi-GPU SFT or preference-tuning jobs who want one consistent interface across model families and training methods.

Pricing

Free, Apache 2.0. You pay for the GPUs.

2. Unsloth — Score: 9.2/10

Unsloth is the right answer when you have one GPU and you want to fine-tune the biggest model that will fit on it. Through hand-written Triton kernels and aggressive memory optimization, Unsloth claims 2x faster training and ~70% less VRAM than vanilla HuggingFace, and in our benchmarks those numbers hold up. We've fine-tuned Llama 3.1 70B with QLoRA on a single 48GB GPU using Unsloth where the same job OOMs in stock transformers. The catch: the free open source version is single-GPU only; multi-GPU is gated behind their Pro tier.

Best for

Solo developers, researchers on a budget, and anyone fine-tuning on a single 24GB or 48GB card.

Pricing

Open source version free (single GPU). Unsloth Pro for multi-GPU starts around $50/mo per seat.

3. LLaMA-Factory — Score: 9.0/10

Llama Factory earns its spot through breadth. It supports more model architectures out of the box than anything else on this list (100+ at last count, including most Chinese-origin models like Qwen, DeepSeek, Yi, ChatGLM), and ships with a clean Gradio web UI for people who'd rather click than YAML. It covers SFT, RLHF (PPO, DPO, ORPO, KTO, SimPO), pretraining, and reward modeling. The codebase is dense but well-organized, and the documentation is bilingual (English / Chinese) which matters if you're working with the Qwen or DeepSeek ecosystems.

Best for

Anyone fine-tuning Qwen, DeepSeek, or other models with strong Asian-market support, or teams that want a no-code UI option.

Pricing

Free, Apache 2.0.

4. TorchTune — Score: 8.7/10

Torchtune is PyTorch's official, in-house fine-tuning library, and it shows in the code quality. Everything is native PyTorch — no transformers dependency, no PEFT abstraction layer, no hidden magic. Configs are Python dataclasses, recipes are readable from top to bottom, and the FSDP integration is the cleanest of any tool in this list. The trade-off: model coverage is narrower (Llama, Mistral, Gemma, Qwen, Phi, and a handful of others) and you'll write more code to do non-standard things. If you want to understand exactly what your training loop is doing, TorchTune is the right choice.

Best for

Researchers, engineers who want hackable PyTorch-native code, and teams already standardized on torch tooling.

Pricing

Free, BSD-3.

5. TRL (Transformer Reinforcement Learning) — Score: 8.5/10

Trl is HuggingFace's official library for everything beyond plain SFT — DPO, PPO, GRPO, KTO, ORPO, reward modeling, online RL. It's the substrate that Axolotl and several others wrap, and if you're doing serious preference-tuning research you'll end up using it directly. The 2025-2026 releases added solid GRPO support (the technique DeepSeek-R1 used) and online DPO. It's not the prettiest framework to use raw — the trainer APIs require boilerplate — but for anything experimental in the RLHF or reasoning space, TRL is where the cutting edge lands first.

Best for

RLHF research, reasoning model training (GRPO/PPO), and anyone implementing a method that hasn't been wrapped by a higher-level tool yet.

Pricing

Free, Apache 2.0.

6. Lit-GPT — Score: 8.2/10

Lit Gpt from Lightning AI takes the opposite philosophy from Axolotl: instead of abstracting everything into YAML, it gives you a hackable single-file implementation per model family. The code is genuinely readable — you can trace a forward pass top to bottom in an afternoon. It's also one of the few tools with first-class support for from-scratch pretraining, not just fine-tuning. The trade-off is less breadth and fewer batteries-included features compared to Axolotl or LLaMA-Factory.

Best for

Education, custom architecture experiments, and small-team pretraining runs.

Pricing

Free, Apache 2.0.

7. DeepSpeed — Score: 8.0/10

Deepspeed isn't a fine-tuning framework per se — it's the distributed training engine that powers most of the others. We include it here because for very large jobs (multi-node training of 70B+ models, anything pushing the limits of available VRAM) you'll eventually configure DeepSpeed directly: ZeRO-3, parameter offloading, mixed-precision optimizer states. The learning curve is steep, the config files are notoriously fussy, and debugging a stuck job at 3 AM is a rite of passage. But nothing else gets you the same throughput at scale.

Best for

Multi-node training, 70B+ models, and teams whose bottleneck is GPU memory rather than developer time.

Pricing

Free, Apache 2.0 (Microsoft).

Comparison Table

Tool	Score	Best For	Multi-GPU	Ease of Use	Pricing
Axolotl	9.4	Production multi-GPU SFT/DPO	Excellent	High (YAML)	Free
Unsloth	9.2	Single-GPU max efficiency	Pro tier only	High	Free / $50+ mo
LLaMA-Factory	9.0	Broad model support + UI	Good	Very high (UI)	Free
TorchTune	8.7	PyTorch-native research	Excellent (FSDP)	Medium	Free
TRL	8.5	RLHF, DPO, GRPO research	Good	Low (boilerplate)	Free
Lit-GPT	8.2	Hackable code, pretraining	Good	Medium	Free
DeepSpeed	8.0	Multi-node 70B+ training	Best-in-class	Low	Free

Final Picks

If you're fine-tuning on a single GPU: Unsloth. Nothing else gets you the same VRAM efficiency, and the single-GPU version is free.

If you're running production training jobs on multi-GPU: Axolotl. The YAML-config model scales from one engineer to a whole team without anyone having to learn a new abstraction.

If you're fine-tuning Qwen, DeepSeek, or other non-Western models: Llama Factory. Broadest model coverage and the Chinese-language ecosystem support is unmatched.

If you're doing RLHF or reasoning research: Trl directly. New methods land there first.

If you want to actually understand your training loop: Torchtune or Lit Gpt. Both have code you can read.

The honest meta-take: most teams in 2026 should default to Axolotl unless they have a specific reason not to, and reach for Unsloth when GPU memory becomes the binding constraint. Everything else on this list is excellent at what it does, but those two cover roughly 80% of real-world fine-tuning needs.
