Best Local Alternatives to Claude Code in 2026

February 23, 2026 · by Mark Bartlett

Download this guide as PDF

> **Quick Answer:** For terminal-based agent coding with local models: Aider (40.6K stars, Apache 2.0) with Qwen 2.5 Coder 32B on a 24GB GPU gives you roughly GPT-4o-tier coding assistance. For IDE integration: Continue.dev (31.4K stars) for VS Code tab completion + chat. For an autonomous agent in VS Code: Cline (57K stars) with approval workflows. For the fastest-growing alternative: OpenCode (95K+ stars) is a Go-based terminal agent with multi-editor support. The practical setup: use local models (Qwen 2.5 Coder 32B via Ollama) for the 80% of routine coding, and Claude Code for the 20% of hard problems that require frontier reasoning.

📚 **Related:** Best Models for Coding Locally · Ollama vs LM Studio · The 5 Levels of AI Coding · VRAM Requirements

Claude Code with Opus is the best AI coding tool available. It's also $125/month for the Max plan, sends your code to Anthropic's servers, and requires an internet connection.

If you have a 24GB GPU and run Ollama, you can get surprisingly close with open-source tools and local models. Not all the way — frontier models still handle complex multi-file refactors better than anything running on consumer hardware. But for tab completion, single-file edits, bug fixes, and routine coding tasks, local is genuinely practical in February 2026.

Here's what works, what doesn't, and how to set up the best local coding stack.

The Best Local Coding Model: Qwen 2.5 Coder 32B

Before choosing tools, you need the right model. For local coding in 2026, Qwen 2.5 Coder 32B at Q4 quantization is the sweet spot:

Model	VRAM (Q4)	Aider Benchmark	Best For
Qwen 2.5 Coder 32B	~20GB	73.7% (matches GPT-4o)	Best all-around on 24GB
Qwen 2.5 Coder 14B	~9GB	Good	Mid-range editing on 16GB
Qwen 2.5 Coder 7B	~6GB	88.4% HumanEval	Tab completion, quick edits
DeepSeek Coder V2 Lite 16B	~10GB	Competitive	Fast inference (2.4B active MoE)

```
ollama pull qwen2.5-coder:32b-instruct-q4_K_M
```

The 32B model on a single RTX 3090 or 4090 is where local coding becomes genuinely useful.

Aider — Best Terminal Agent

Stars: 40,600 | **License:** Apache 2.0 | GitHub

Aider is the closest thing to Claude Code that runs with local models. You run it in a git repo, describe changes in natural language, and it edits your files directly with automatic git commits.

Setup

```
pip install aider-chat
ollama pull qwen2.5-coder:32b-instruct-q4_K_M
aider --model ollama/qwen2.5-coder:32b-instruct-q4_K_M
```

Strengths

- Builds a repo map of your entire codebase for context
- Automatic git integration — every change is a commit you can undo
- Voice coding support
- Mature (since 2023) with transparent model benchmarks
- Works with 90+ languages

Weaknesses

- Terminal-only (no GUI; third-party Aider Desk wrapper exists)
- Local models struggle with large multi-file refactors compared to Claude/GPT
- No browser automation or MCP support
- Repo map generation is slow on very large codebases

Verdict

The best terminal-based coding agent for local models. If you liked Claude Code’s terminal workflow but want to run on your own GPU, start here.

Continue.dev — Best IDE Tab Completion

Stars: 31,400 | **License:** Apache 2.0 | GitHub

An open-source VS Code / JetBrains extension that provides tab completion and inline chat. Think “open-source Copilot” wired to any backend.

Setup

1. Install “Continue” from VS Code marketplace
2. In Continue settings, add Ollama as a provider:
    - Chat model: `qwen2.5-coder:32b-instruct-q4_K_M`
    - Tab completion model: `qwen2.5-coder:7b` (smaller, faster)

Strengths

- Lives inside your editor — no context switching
- Tab completion feels like Copilot with a fast local model
- Codebase indexing for RAG-style context retrieval
- Can mix local and cloud models (local for autocomplete, Claude for complex tasks)

Weaknesses

- Not an agent — it doesn’t execute commands, create files, or run tests autonomously
- Tab completion quality with 7B models is noticeably below Copilot for complex completions
- Configuration can be fiddly (model parameters, prompt templates, context windows)

Verdict

The best way to get Copilot-style tab completion running entirely on your GPU. Pair it with Aider or Cline for agent-style tasks.

Cline — Best VS Code Agent

Stars: 57,000 | **Installs:** 5M+ | **License:** Apache 2.0 | GitHub

The most-installed open-source AI coding agent for VS Code. Originally "Claude Dev." Supports Plan/Act modes, MCP integration, file editing, terminal commands, and browser automation — all with explicit user approval at each step.

Setup

1. Install "Cline" from VS Code marketplace
2. Set provider to Ollama, select your model
3. Every action requires your approval (approve/deny per tool call)

Strengths

- Full agent capabilities (file create/edit, terminal, browser)
- Explicit approval workflow prevents surprises
- MCP integration for custom tools
- Cline CLI 2.0 brings it to the terminal with parallel agents

Weaknesses

- Agent loop with local models fails more often than with Claude/GPT
- Approval-per-action gets tedious on long tasks
- Heavy token consumption

Verdict

Best VS Code agent if you want autonomous capabilities with safety rails. Works with local models, but expect more iterations than with frontier models.

OpenCode — Fastest Growing Alternative

Stars: 95,000+ | **License:** MIT | GitHub

The explosive newcomer of early 2026. An open-source terminal coding agent written in Go with a TUI, multi-session support, LSP integration, and compatibility with 75+ models including local ones.

Setup

```
# Install via Go
go install github.com/opencode-ai/opencode@latest

# Or download binary from GitHub releases
opencode --provider ollama --model qwen2.5-coder:32b
```

Strengths

- Fast (Go binary, not Python)
- Multi-session support
- IDE extensions for VS Code, JetBrains, Neovim, Zed, and Emacs
- GitHub Actions integration via `/opencode` comments
- Agent Client Protocol (ACP) for editor communication

Weaknesses

- Primarily designed around cloud models — local model support works but isn't the primary focus
- Gained stars partly from a workaround to route Claude Max subscriptions through it (which Anthropic blocked)

Verdict

Worth trying if you want a fast, Go-based terminal agent. The multi-editor support via ACP is a genuine differentiator.

Void — Open-Source Cursor

Stars: 28,000 | **License:** MIT | GitHub | Y Combinator backed

An open-source VS Code fork that aims to replicate Cursor's feature set. Agent mode, inline editing, contextual chat.

Setup

Download from voideditor.com. It auto-detects Ollama at `http://127.0.0.1:11434` and transfers your VS Code themes, keybinds, and settings in one click.

Strengths

- Closest open-source equivalent to Cursor
- Full VS Code extension compatibility
- No middleman server — connects directly to Ollama
- You can view and edit the prompts sent to the AI

Weaknesses

- Still in beta — expect bugs and rough edges
- Agent mode with local models is significantly less capable than with Claude/GPT
- Smaller team than Cursor

Verdict

Best option if you want a Cursor-like experience with local models. Still maturing.

Tabby — Best for Teams

Stars: 32,900 | **License:** Apache 2.0 | GitHub

Self-hosted coding assistant server. Run a Tabby server on your hardware, get code completion and chat in VS Code, JetBrains, or Vim.

Setup

```
docker run -it --gpus all -p 8080:8080 \
  tabbyml/tabby serve --model StarCoder-1B --chat-model Qwen2-1.5B-Instruct
```

Strengths

- Purpose-built for self-hosting and enterprise deployment
- Repository-level code indexing (connects to GitHub, GitLab, local repos)
- Multi-user support with admin dashboard
- Clean REST API

Weaknesses

- Primarily completion and chat, not an autonomous agent
- No terminal/CLI agent mode
- Smaller model ecosystem than Continue

Verdict

Best choice for teams that want a self-hosted Copilot with admin controls and repository indexing.

Quick Comparison

Tool	Type	Stars	Local Models	Agent Mode	Best For
Aider	Terminal	40.6K	Excellent	Yes	Git-integrated editing
Continue	IDE extension	31.4K	Excellent	No	Tab completion + chat
Cline	VS Code agent	57K	Good	Yes (with approval)	Autonomous coding in VS Code
OpenCode	Terminal	95K+	Good	Yes	Multi-editor terminal agent
Void	VS Code fork	28K	Good	Yes	Open-source Cursor
Tabby	Self-hosted server	32.9K	Built-in	No	Team/enterprise self-hosting
Roo Code	VS Code agent	22K	Good	Yes	Multi-agent workflows

Where Local Closes the Gap (and Where It Doesn't)

Local models are competitive for:

- **Tab completion**: Qwen 2.5 Coder 7B running locally feels snappier than cloud Copilot due to zero latency
- **Single-file edits**: 32B models handle "add a function" and "fix this bug" about as well as GPT-4o
- **Privacy**: The only option for air-gapped environments and proprietary codebases
- **Cost at scale**: Free after GPU investment vs $125/month for Claude Code

The gap remains significant for:

- **Multi-file refactoring**: Claude Opus with 200K context can coordinate changes across dozens of files. Local 32B models degrade past 32K tokens in practice
- **Complex architectural reasoning**: Frontier models suggest patterns and trade-offs that 32B models cannot
- **Agent loop reliability**: Cloud models complete agent tasks in fewer iterations with fewer failures
- **Token efficiency**: Claude Code uses 5.5x fewer tokens than Cursor for identical tasks. Local models are even less efficient

---

The Recommended Setup

```
Hardware: RTX 4090 or used RTX 3090 (24GB)
Model: Qwen 2.5 Coder 32B Q4 via Ollama

Tools:
  - Continue.dev for tab completion (Qwen 2.5 Coder 7B — fast)
  - Aider or Cline for agent editing (Qwen 2.5 Coder 32B — capable)
  - Claude Code for hard problems (pay per use when local falls short)
```

→ Not sure what fits? Try our Planning Tool.

This hybrid approach — local for the 80% of routine work, cloud for the 20% of hard problems — is the most cost-effective setup in February 2026. The local tools are genuinely good enough for daily coding. They're just not quite good enough to replace frontier models on the tasks where you most want help.

That gap is narrowing. Check back in six months.

Source: https://insiderllm.com/guides/local-alternatives-claude-code-2026/