GLM-4.6: An Open-Source AI for Coding vs. Sonnet & GPT-5

By InuitionLabs.ai • 10/17/2025 • 30 min read

glm-4.6 | open source ai | coding ai | claude sonnet | gpt-5 | mixture of experts | zhipu ai | llm benchmarks

Executive Summary

The GLM-4.6 model (General Language Model, version 4.6) is an ambitious open-source AI system developed by Zhipu AI (now Z.ai) that specifically targets complex reasoning and coding tasks. With **355 billion parameters** (32 billion active) and a **200K-token context window**, GLM-4.6 represents a major advance over earlier open models. Zhipu AI reports that GLM-4.6 outperforms its predecessor GLM-4.5 on numerous benchmarks and achieves *"state-of-the-art performance among open source models"* (blogs.novita.ai). In practical coding evaluations, GLM-4.6 reaches near parity with Anthropic's Claude *Sonnet 4* (a closed model) – human evaluators judged it about 48.6% as good as Sonnet 4 at real-world coding tasks (blogs.novita.ai). However, Anthropic's freshest closed model, Claude Sonnet 4.5, is widely reported to be the world's best coding AI (77.2% on the SWE-Bench Verified coding benchmark (www.theneuron.ai)). Thus, while GLM-4.6 is likely *the leading open-source coding model in late 2025*, it still trails the proprietary Sonnet 4.5 in raw coding performance (z.ai) (www.theneuron.ai).

In contrast, OpenAI's GPT-5 (rumored for release in 2025) is expected to unify and improve upon all prior GPT models. OpenAI executives hint GPT-5 will integrate multimodal and chain-of-thought reasoning and could even consist of multiple specialized sub-models (codenamed *Zenith, Summit, Lobster*, etc. (www.theneuron.ai) (www.techopedia.com)). Leaked reports suggest GPT-5 is being designed to *"crush coding tasks"* (www.theneuron.ai). However, GPT-5 remains unreleased and details are speculative. Unlike GLM-4.6, any GPT-5 system would be proprietary and likely require massive compute beyond hobbyist reach.

Running GLM-4.6 effectively requires cutting-edge hardware. Published deployment guides indicate GLM-4.5 (its predecessor) needed on the order of **16–32 NVIDIA H100 GPUs** (or equivalent) even for inference (openlm.ai). GLM-4.6's larger context likely demands roughly *twice* that. In practice, this means multi-GPU server clusters (with >>1 TB system RAM) or specialized superchips (e.g. NVIDIA's Hopper/Grace architecture) for real-time use. For reference, Zhipu's documentation shows GLM-4.5 needs 16×80GB H100 GPUs for BF16 inference at 128K context (openlm.ai). One can roughly extrapolate that GLM-4.6 at 200K context might require on the order of **32+** such GPUs. At present, GLM-4.6 is primarily run via cloud/AIs-as-a-service (Novita AI, Z.ai API) where the provider bulks together this hardware.

This report provides a comprehensive review of GLM-4.6: its architecture, training, performance benchmarks, and real-world usage. We compare it in detail to Anthropic's Claude Sonnet models and to the expected GPT-5. We analyze quantitative results on coding and reasoning tasks, examine hardware requirements, present case studies of GLM-4.6 in use, and discuss broader implications. Throughout, claims are supported by extensive citations to technical blogs, news reports, and official documentation.

Introduction and Background

The Rise of AI-Assisted Coding

AI-driven code generation and debugging have become a central focus of recent AI research. Models like OpenAI's Codex and Google DeepMind's AlphaCode showed in 2021–2022 that large language models (LLMs) can tackle programming problems. In practice, tools such as GitHub Copilot (based on Codex) have begun assisting millions of developers. As coding is a concrete, high-stakes task requiring logical reasoning, coding models are often seen as a litmus test for AI progress: they must handle precise, multi-step logic that traditional LLM conversations might not. Many experts believe advances in AI coding assistants are crucial stepping stones toward more general AI systems.

By mid-2025, a new wave of coding AI "arms race" has emerged. Major players include **Anthropic** (the 'Sonnet' series under the Claude brand), **OpenAI** (GPT and tools like GPT-4o/GPT-4o Opus), and in China, companies like **Zhipu AI** (GLM series) and **DeepSeek** (V series). These models advertise specialized "agentic" and "reasoning" modes to handle complex tasks beyond mere text completion. In open source, few models have matched the scale and performance of top proprietary systems on coding tasks. GLM-4.6 emerges in this context as a flagship open model aiming squarely at coding and multi-step reasoning.

GLM Series and Zhipu AI

Zhipu AI, now known as Z.ai, is a Chinese AI startup founded in 2019 that quickly became one of China's "AI Tigers" for LLM development (www.1950.ai). It has been positioned as a response to Western LLM leaders. The core of its technology is the **GLM** ("General Language Model") series. GLM models have historically focused on Chinese-language tasks — for example, GLM-130B (released in 2023) scored very highly on Chinese comprehension benchmarks (www.1950.ai).

Over 2024–25, Zhipu AI released increasingly capable GLM models. GLM-4.5 (July 2025) introduced "hybrid reasoning" modes and MoE architecture as the company's first focus on coding and agent tasks. It leveraged a *Mixture-of-Experts* design to pack 355B parameters (with only 32B active at any time) into the model (apxml.com). This allowed very high capacity while containing compute costs. By September 2025, Zhipu unveiled **GLM-4.6**. This new version retains the 355B/32B MoE architecture but expands the context window from 128K to **200K tokens** (z.ai). It additionally refines training with more code data and new "thinking mode" switches for deep reasoning (z.ai). As a result, GLM-4.6 shows markedly better performance than 4.5 on coding and reasoning benchmarks (openlm.ai) (blogs.novita.ai).

Crucially, GLM-4.x models are **open source**. Zhipu released GLM-4.5 and its smaller variant GLM-4.5-Air under permissive licenses (MIT/Apache) and published weights and code (openlm.ai) (partimus.com). GLM-4.6 likewise has its base and chat weights publicly available on HuggingFace and other platforms (z.ai). This contrasts with Anthropic and OpenAI, which keep their flagship models proprietary. Thus GLM-4.6 has democratized access: anyone can download and run it (given sufficient hardware) or try it via free tiers on services like Novita.ai.

Sonnet (Anthropic Claude) Models

Anthropic's **Claude** has emerged as the main Western rival. In 2024–2025, Anthropic released a series of progressively more powerful models. They branded their coding-focused variants as *Claude Sonnet*, following *Opus* for reasoning and *Claude 4o/m* for multimodal. Sonnet models have hybrid "chain-of-thought" capabilities engineered for long-horizon tasks. Most notably:

- **Sonnet 4** (May 2025) introduced significant coding performance improvements, with a 200K context window (www.anthropic.com) (matching GLM) and integrated tool use. Anthropic touted it as offering *"frontier performance"* for coding and agents (www.anthropic.com).
- **Sonnet 4.5** (Sept 2025) delivered further gains.Anthropic announced that Sonnet 4.5 achieved *77.2%* on SWE-Bench Verified, the highest ever on this real-world coding benchmark (www.theneuron.ai). They claimed it was *"the best model in the world for agents, coding, and computer use"* (www.anthropic.com). Sonnet 4.5 can carry out multi-day coding projects without losing context ("30+ hours" workflows) (www.theneuron.ai). It also introduced features like native code execution, extended memory, and VS Code integration (www.theneuron.ai).

These Claude Sonnet models are cloud-only (proprietary) and primarily targeted at enterprise dev environments. Anthropic's goals for Sonnet are to enable truly autonomous coding agents. Indeed, the 77.2% SWE-bench performance of Sonnet 4.5 stands significantly above any known open model. (By comparison, GPT-4 scored around the 50–60% range on earlier CodeBench tasks.)

GPT-5 and the OpenAI Roadmap

OpenAI's current generation is GPT-4 (plus specialized variants like GPT-4o). They also provide a code-writing agent via ChatGPT with Code Interpreter and Copilot capabilities. Looking ahead, OpenAI leadership has publicly signaled that **GPT-5** is on the horizon. In early 2025, Sam Altman announced a "soonish" schedule: GPT-4.5 in weeks and GPT-5 in *months* (www.techopedia.com).

Few official details are available, but hints suggest GPT-5 will *"merge their scattered models into one"* (www.techopedia.com). In practice this likely means a unified agent architecture rather than separate GPT-4/GPT-4o/"o-series" models. Key features rumored for GPT-5 include:

- **Integrated Chain-of-Thought**: Deep reasoning built-in rather than optional (www.techopedia.com).
- **Persistent Memory**: The kind of long-term, cross-session memory not present in GPT-4 (www.toolcradle.com).
- **Multimodal & Video**: Even more advanced image/video understanding.
- **Auto-circuit**: "Smart routing" of tasks (so GPT-5 handles job selection behind the scenes) (www.techopedia.com).

From coding benchmarks angle, leaks reported via The Information say *"GPT-5 is specifically designed to crush coding tasks"* (www.theneuron.ai). Enthusiasts found mysterious GPT-5-like models ("Zenith, Summit, Lobster" etc.) on the LM Arena leaderboard, with the top unnamed models allegedly beating Claude Sonnet 4 on coding problems (www.theneuron.ai) (www.theneuron.ai). If true, this would put GPT-5's coding capability even beyond Sonnet 4.5. However, all GPT-5 details remain rumors until OpenAI releases it (likely in late 2025). Significantly, like previous GPTs, GPT-5 will be **closed-source** and only accessible via API on OpenAI's platform.

Given this context, GLM-4.6 occupies an interesting niche: as of late 2025, it is **the leading open-source solution for coding tasks**, whereas Sonnet 4.5 is the (openly acknowledged) leader overall, and GPT-5 is undisclosed. The rest of this report examines GLM-4.6's capabilities and benchmarks in depth, then directly compares it against Sonnet and the expected trajectory of GPT-5.

GLM-4.6: Architecture and Innovations

Core Architecture

GLM-4.6 is built on Zhipu AI's "GLM-4.x" foundation. It is a **Mixture-of-Experts (MoE)** Transformer, totaling 355 billion parameters. Critically, it uses MoE sparsity: **only about 32 billion parameters are active on any given forward pass**, yielding considerable efficiency (apxml.com). This design lets the model scale up massively without prohibitive compute on each token. The model uses grouped query attention (MQA) and a large number of attention heads (96 heads) for deeper context interaction (openlm.ai).

A key innovation in GLM-4.6 is the **200K-token context window** (z.ai). This is one of the largest contexts of any publicly known model, enabling the AI to ingest entire codebases or long documents at once. GLM-4.5 had 128K; GLM-4.6 boosted this further. Longer context directly addresses coding scenarios where projects span many files or discussions. In practice, developers can ask GLM-4.6 about a multi-hundred-page spec or a large code repository in one

go. (By contrast, GPT-4 is limited to 32K, and even Claude Sonnet 4.5 is quoted with 64K output, though its context is 200K (www.anthropic.com).)

GLM-4.6's **"Hybrid Reasoning" mode** is another distinctive feature. Like GLM-4.5, it has two modes of operation: a fast "non-thinking" mode for simple queries, and a slower "thinking" mode for complex, multi-step reasoning (openlm.ai). This toggle can be controlled via a parameter (e.g. `thinking.type`). In thinking mode, the model deliberately slows down its output, internally using more computation (multi-token lookahead with a "Multi-Token Prediction" head) to plan multi-step solutions (openlm.ai). This resembles Anthropic's approach in Claude (chain-of-thought introspection made explicit) and ensures the model doesn't simply spill tokens but can reflect on a plan. The GLM-4.6 doings these rigorous steps to tackle logic puzzles, multi-part code generation, or extended agent interactions.

Training Data and Process

According to Zhipu's documentation, GLM-4.5 (and by extension 4.6) received extremely large-scale training. GLM-4.5 was pretrained on about **15 trillion tokens** of broad-domain data, then fine-tuned on an extra **7 trillion tokens** of data focused on code and reasoning tasks (apxml.com). Zhipu also applied reinforcement learning ("{slime} engine") to align GLM-4.5 with human preferences and strengthen coding/ reasoning ability (apxml.com). We can infer GLM-4.6 used a similar pipeline, likely adding even more diverse code-related examples and RL training focusing on multi-turn coding use cases. The model's explicit support for "tool use during inference" suggests that some RL tasks involved integrating AWS APIs, search tools, or code compilers to train the agentic behavior (z.ai).

Notably, the GLM-4 series emphasizes **China-specific content** in training. Zhipu's earlier GLMs showed superlative performance on Chinese benchmarks (www.1950.ai). For GLM-4.6, publicly released artifacts and API examples are bilingual, but it likely has extra Chinese-language training, complementary to the mostly English code benchmarks like SWE-Bench. This broad training yields the model both fluency in Chinese (common in Zhipu's user base) and strong coding ability from global coding corpora.

Open-Source Availability

Unlike Anthropic and OpenAI, Zhipu AI *open-sourced* GLM-4.x. All GLM-4.5 variants and 4.6 base weights are released under permissive license. GLM-4.5 (and 4.5-Air) was published under an **MIT license** (openlm.ai). GLM-4.6 is released under **Apache 2.0** (partimus.com), also allowing commercial and research use. The base TPU and chat models for GLM-4.6 are downloadable from HuggingFace and ModelScope (z.ai). This democratizes access: any organization or individual with adequate hardware can run GLM-4.6 locally or on their own servers.

Zhipu also published open-source code and tools to support GLM. For example, they provide recipes for SGLang by THUNLP, vLLM, and HuggingFace Transformers integration (openlm.ai). This means users can deploy GLM-4.6 with popular LLM-serving frameworks (as opposed to it being locked behind a single API). For commercial crowd or small startups, this is huge: they can build GPT-like coding assistants without licensing fees to OpenAI or Anthropic. LucidGraph's Zyai and Novita AI platforms essentially act as managed huggingface-like services on GLM-4.6.

Model Capabilities

According to benchmark reports, GLM-4.6 exhibits a strong overall capability profile. In general reasoning, it outperforms GLM-4.5, scoring higher on tasks like mathematics (MATH), commonsense (MMLU), and logical puzzles. In coding-specific tests, it likewise does better than GLM-4.5 but not quite as high as Sonnet 4.5. Zhipu's own evaluation (eight public benchmarks) shows GLM-4.6 achieving *"clear gains over GLM-4.5"*, including on coding metrics where it narrowly trails the best models (z.ai).

Zhipu claims that GLM-4.6 "achieves higher scores on code benchmarks" and in practical agent-based coding tasks, producing more polished components (e.g. nicer front-end pages) (openlm.ai). In the CC-Bench real-world coding challenge (multi-turn tasks with human oversight), GLM-4.6 won about 48.6% of evaluations against Claude Sonnet 4, which is roughly "near parity" (z.ai). This indicates that while GLM-4.6 is not identically powerful as Sonnet 4 in coding-specific contexts, it is not far behind – and significantly outperforms all other open-source alternatives on those tasks.

Other specialized aptitudes include multi-step reasoning and structured output. GLM-4.6 incorporates a "tool use during inference" ability, meaning it can call external functions or APIs when generating. This aligns it with others like Claude (tool-calling Claude Code) or OpenAI Codex (via plugins). When deployed in coding agents (Claude Code, Cline, Kilo Code, etc.), GLM-4.6 reportedly handles end-to-end development flows (designing algorithms, writing code, running tests) more robustly than GLM-4.5 (z.ai) (blogs.novita.ai).

In terms of language and content style, GLM-4.6 has been fine-tuned for **fluent writing and preferences**. Zhipu notes it is better aligned to human style and more "natural" in role-play scenarios (openlm.ai). This indicates they did additional reinforcement learning or supervised data to shape its conversational tone. (Anthropic's Claude models have a similar emphasis on politeness and safety through RLHF.) In sum, GLM-4.6 aims to be a versatile general agent with special edges in code and reasoning, while still conversing clearly.

Comparison to Claude/Sonnet Models

Overview of Claude Sonnet 4 and 4.5

Anthropic's Claude Sonnet lineup is explicitly engineered toward coding and long-horizon reasoning. Key facts:

- **Claude Sonnet 4** (outright release May 2025): Hybrid model (likely over 100B parameters, internal architecture undisclosed). It has a 200K context window and advanced chain-of-thought prompting. Anthropic states Sonnet 4 "improves on Sonnet 3.7 especially in coding" and offers "frontier performance" for AI assistants (www.anthropic.com). In practice, Sonnet 4 achieved roughly the top performance at its time on benchmarks like Terminal-Bench and coding tasks, surpassing GPT-4.

- **Claude Sonnet 4.5** (release Sep 29, 2025): Anthropic's caption declares Sonnet 4.5 *"the best model in the world for agents, coding, and computer use"* (www.anthropic.com). This model raised the bar dramatically: it hit **77.2%** on SWE-Bench Verified (www.theneuron.ai), an unprecedented score. It also dramatically improved on OSWorld (human-level computer interaction), jumping to 61.4% (www.theneuron.ai). Sonnet 4.5 can sustain multi-day coding projects (30+ hours) without context loss (www.theneuron.ai), and supports up to 64K output tokens (www.anthropic.com). In side-by-side developer experience, Sonnet 4.5 in Claude Code (their coding IDE) became markedly more autonomous with checkpointing and execution.

Context length: Both Sonnet 4 and 4.5 star a **200K token context window** (www.anthropic.com). This is on par with GLM-4.6, exceeding that of any previous GPT.

Availability: Unlike open-source GLM, Claude Sonnet models are only CPU/GPU clusters in cloud APIs. Sonnet 4.5 is accessible via Claude.ai web or as an agent via Anthropic's SDKs and through certain cloud platforms (AWS Bedrock, Google Vertex) (www.anthropic.com). They are proprietary and paid services, with pricing at about $3 input/$15 output per million tokens (www.anthropic.com) (with caching/batching offsets).

Architectural Differences: Details are scarce, but Sonnet models are not MoE. Claude has referenced a "Hybrid reasoning model" architecture that blends different layers (and likely heavy RLHF) (www.anthropic.com) (openlm.ai). They also emphasize having a system prompt orchestrating "thinking" vs "speaking" modes. GLM's MoE approach is a distinctly different technical path for maximizing parameters.

Benchmarks and Performance

When directly comparing GLM-4.6 and Claude Sonnet on benchmarks:

- **Coding Tasks (SWE-bench, etc.):** Sonnet 4.5 leads. 77.2% on SWE-bench Verified (www.theneuron.ai) is far above any publicly known open model. GLM-4.6 has no published SWE score, but indirect evidence suggests it is lower. For example, Zhipu notes GLM-4.6 "still lags behind Claude Sonnet 4.5 in coding ability" (z.ai). In CC-Bench (human-coded coding tasks), GLM-4.6's 48.6% win rate against Sonnet 4 means it loses slightly more than it wins. We infer GLM-4.6's raw code-fixing and generation success rate is roughly half to two-thirds of Sonnet 4.5's, given Sonnet 4.5's dominance. (No numeric table exists for exact comparison, but the pattern is clear: Sonnet 4.5 ≫ GLM-4.6 ≈ Sonnet 4 >> other models.)

- **General Reasoning:** Sonnet 4.5 reportedly improved on math and logic, but GLM-4.6 similarly boosts reasoning over GLM-4.5 (z.ai). On benchmarks like MATH or MMLU, Sonnet 4.5 likely has the edge due to larger parameter count and more extensive fine-tuning on English data. However, GLM-4.6's multi-layer, deep attention architecture assures it remains very competitive. In fact, Zhipu claims GLM-4.6 even surpasses Sonnet 4 on certain agentic benchmarks like BFCL-v3 (openlm.ai). Precise head-to-head numbers are not public, but we can say performance is roughly comparable on reasoning when context length is accounted for.

- **Agentic and Tool Use:** Both GLM-4.6 and Sonnet models integrate tool/model chaining. Anthropic's Claude has the **Agentic Claude** framework and explicit support for function calls. GLM-4.6 supports similar "function calling" natively (partimus.com) and has been placed inside coding agents (Roo, Kilo, etc.). Zhipu reports GLM-4.6 shows *stronger* performance specifically in tool-using and search-based agents compared to GLM-4.5 (openlm.ai). Anecdotally, GLM-based agents (via Claude Code or Novita) can navigate APIs and write correct code sets, though Sonnet's longer sustained focus might give it a slight advantage on very extended agent workflows.

- **Content Quality:** In non-code tasks like writing or role-play, GLM-4.6 is also strong but not specifically top-tier. Anthropic's Claude systems are often praised for coherent tone and creativity. GLM-4.6 uses RL alignments for style, but it was optimized more for logic. Nevertheless, Zhipu notes GLM-4.6 outputs more "refined" writing and more human-aligned style (openlm.ai). We lack a direct qualitative score to compare, but expert evaluations often find few cracks in modern models' prose. One practical difference: GLM-4.6 may incorporate more Chinese cultural data than Claude, while Claude might have more training on diverse English codebases.

Summary of Comparison

Overall, the **head-to-head comparison** can be summarized:

- *Parameter and Architecture:* GLM-4.6 (355B MoE open-source) vs. Sonnet 4.5 (unknown param, closed). Both have massive scale; Sonnet's exact count isn't public, but it likely exceeds 100B. Sonnet instead uses a dense Transformer with heavy RLHF, whereas GLM uses experts and deeper stacking.

- *Context Window:* Both boast ~200,000 tokens of context, far above GPT-4. Sonnet claims native 200K window (www.anthropic.com), and GLM-4.6 likewise expanded to 200K (z.ai).

- *Coding Performance:* Sonnet 4.5 leads (77.2% on SWE-bench) vs. GLM-4.6 at lower (around 50% as strong). GLM-4.6 outshines all other open models and is competitive with Sonnet 4 (z.ai) (blogs.novita.ai).
- *Open vs Closed:* GLM-4.6 is MIT/Apache-licensed (developers can run it locally) (openlm.ai) (partimus.com). Sonnet 4.5 is proprietary (accessible only via Claude APIs) (www.anthropic.com).
- *Availability:* Sonnet 4.5 is already in Claude web/API with developer SDK. GLM-4.6 is accessible via select services (Novita AI, Z.ai API) and downloadable for private use (z.ai).
- *Agentic Capabilities:* Both target tool-learning, but Sonnet 4.5 is often optimized for user-facing agents (customer support, autonomous coding bots) (www.anthropic.com). GLM-4.6 similarly targets agents and coding workflows (z.ai) (blogs.novita.ai), with demonstrated improvements in integrated agents.
- *Special Features:* Sonnet 4.5 adds things like dynamic "thinking" check-points, extensions (VS Code, Chrome plugin) and tightly integrated execution. GLM-4.6 has a unique integrated multi-token prefix cache (MTP) and explicit thinking mode switching.

In short, GLM-4.6 is the **strongest open-source code model** known, but if raw coding power is the only metric, Claude Sonnet 4.5 currently outperforms it. For many users, the decision will hinge on openness and cost: GLM-4.6 can be self-hosted, whereas Sonnet requires Anthropic's paid service.

Performance and Benchmarks

Industry Benchmarks

Both developers and independent evaluators usually benchmark coding models on standardized tasks. Relevant metrics include:

- **SWE-Bench Verified:** A benchmark that tests fixing real GitHub issues. Sonnet 4.5 scored **77.2%** on SWE-Bench Verified (www.theneuron.ai), substantially better than any published model. OpenAI's GPT-4 had lower scores (around 50%), and GLM-4.x has not published a direct score.
- **OSWorld:** AI's ability to operate in a simulated OS environment. Sonnet 4.5 achieved **61.4%** on OSWorld (www.theneuron.ai), up from 42.2% pre-upgrade. This demonstrates autonomous use of tools. No open-model comparison data is available for OSWorld, but GLM-4.x was optimized for tool use, so it likely performs well (GLM-4.5 topped earlier browsing benchmarks (openlm.ai)).
- **Agent Benchmarks:** Y et another category is agentic benchmarks (things like web browsing tasks). Zhipu reports GLM-4.5 matched Claude Sonnet 4 on certain agent benchmarks (τ-bench, BFCL-v3) and even exceeded GPT-4 mini on web browsing (BrowseComp) (openlm.ai). We expect GLM-4.6 to similarly be competitive in agent tasks. Anthropic has not published similar head-to-heads aside from shareholder docs, but Sonnet's claim to be "best for agents" suggests top-tier results too.

- **General Reasoning:** On academic tests (MATH, MMLU, etc.), GLM-4.5 scored mid-80s (e.g. MMLU-Pro: 84.6, MAT H500: 98.2) (openlm.ai), reflecting strong base reasoning. Sonnet 4 likely scores higher but typically such benchmarks are all near ceiling (50–90% in these). These numbers suggest GLM-4.x is at least competitive with GPT-4-level reasoning.

To date, no single authoritative leaderboard includes all these new models. Anecdotally, GLM-4.5 was ranked 3rd across a dozen mixed benchmarks (openlm.ai), behind only a couple proprietary giants. GLM-4.6's improvements likely push it into the top 2 or 3 open models in aggregate performance.

Real-World Coding Tasks (CC-Bench)

Beyond static tests, the **CC-Bench** represents a real-world coding evaluation framework. In this test, human overseers give the model real coding tasks (front-end development, data analysis, testing, etc.) within an isolated environment. The latest CC-Bench results (with GLM-4.6 vs others) reveal:

- GLM-4.6 achieved a **48.6% win rate** against Claude Sonnet 4 in multi-turn scenarios (z.ai). In these head-to-head trials, GLM-4.6 solved about half the tasks as well as Sonnet 4 (i.e. Sonnet 4 "won" 51.4% of the time). By contrast, GLM-4.5 (the previous generation) was clearly worse. This indicates GLM-4.6 has nearly closed the gap for *practical development work*.
- Moreover, against all other *open-source* baselines (DeepSeek, K2, MoonMonkey etc.), GLM-4.6 "clearly outperformed" them in CC-Bench (z.ai). This underscores that it is the new king of open code models.
- In terms of **token efficiency** (a measure of how succinctly the model solves tasks), GLM-4.6 is about **15% more efficient** than GLM-4.5 (z.ai). This means it often generates shorter solutions, which yields faster execution and lower cost. (By contrast, no numbers are given for Sonnet on token use, but Sonnet's sustained focus suggests it also solves problems with moderate succinctness.)

Coding Benchmark Scores

We compile some key figures in the table below:

Model	SWE-Bench (Verified)	OSWorld	CC-Bench (win% vs others)	Remarks
Claude Sonnet 4.5	77.2% (www.theneuron.ai)	61.4% (www.theneuron.ai)	best-of-class on agentic tasks	"Best coding model" (www.anthropic.com)
Claude Sonnet 4	–	–	~51.4% targets (vs GLM-4.6) (z.ai)	Lower than 4.5; strong in agents
GLM-4.6	– (not reported)	–	48.6% win vs Sonnet 4 (z.ai)	State-of-art open-source

Model	SWE-Bench (Verified)	OSWorld	CC-Bench (win% vs others)	Remarks
GLM-4.5	(aggregate 12-task score 63.2) (openlm.ai)	–	–	3rd among models on mixed tasks
GPT-4	~40-60% (codex-level)	–	–	(Scored similarly to GLM-4.5 on code)
GPT-5 (rumor)	–	–	–	Leaked to *crush* coding (www.theneuron.ai)

Note: "–" indicates unpublished. GPT-5 data is speculative. SWE-bench and OSWorld are the most comparable quantitative metrics reported by Anthropic. CC-Bench figures come from Zhipu's GLM-4.6 evaluation with human testers (z.ai).

The table highlights that **Sonnet 4.5 is clearly the leader on coding benchmarks**, with GLM-4.6 second (of open models) and Sonnet 4 close behind. GLM-4.6's near-equal performance vs Sonnet 4 in realistic tasks indicates industry-grade capability. GPT-4 (though not shown) is known to solve many code challenges but generally below Sonnet 4.5.

In summary, GLM-4.6 is demonstrably the *best open model* for coding problems (state-of-art in open-source), whereas the Claude Sonnet 4.5 holds the overall top spot.

Hardware Requirements to Run GLM-4.6

Running a 355-billion-parameter model like GLM-4.6—especially with a 200K context window—requires substantial computational resources. Both training and inference are extremely hardware-intensive:

Training Hardware (Context)

- While Zhipu AI has not fully detailed the training cluster, it is reasonable to assume **thousands of top-tier GPUs** were used over months. For instance, training GPT-4 famously used tens of thousands of Nvidia GPUs over months, and a similarly scaled Chinese cluster likely powered GLM-4.5/4.6. Given geopolitical constraints (U.S. chip export bans), Chinese companies often rely on domestic hardware or scarce Channel chips (e.g. Huawei, or a limited supply of NVIDIA via indirect channels). Thus, GLM-4.x training likely used clusters of **8,192+ GPUs** (80GB-class) running continuously, akin to how DeepMind trained AlphaCode on thousands of TPUs.
- The timing suggests Zhipu likely used NVIDIA H100 (or soon H200) GPUs. Reports noted "H100 x16 or H200 x8" configurations for inference; by analogy, training may have involved large pools of H100s (TensorCore performance is vital). Alternatively, China's own Ascend or new AI chips (such as MiCS by Horizon or others) might have supplemented the cluster.

Inference (Running the Model)

For the end user or developer wanting to **run inference on GLM-4.6**, the requirements are very high. Going off the published guidelines for GLM-4.5 inference (openlm.ai) (which is slightly smaller context), one can estimate the needs for 4.6:

- **GPUs:** Zhipu provides recommended configurations for GLM-4.5 inference. For the full 128K context, GLM-4.5 needed 32 × NVIDIA H100 GPUs when using BF16 precision (16 × H100 for 128K context in partial modes) (openlm.ai). Even the "smaller" GLM-4.5-Air (106B params) needed 4×H100 BF16 just for baseline. Extrapolating to GLM-4.6 (200K context), one likely needs **on the order of 32–40 H100 GPUs** (80GB each) for BF16 inference. Zhipu's docs also mention HG200 (likely NVIDIA's next-gen Hopper/Grace hybrid) which could halve the GPU count (e.g. 16×H200) (openlm.ai). Running FP8 quantized weights can halve these numbers (16 H100s for GLM-4.5; so perhaps ~32×H100 FP8 for GLM-4.6).
- **GPU Memory:** The context length is key. A 200K token context requires storing huge activation buffers. Even with 80GB VRAM cards, a single inference pass at 200K tokens likely spans multiple GPUs. The server memory should exceed **1 TB** (as Zhipu recommends) to even load all weights and cache (openlm.ai).
- **Cluster Setup:** The official note says inference generally assumes no CPU offloading and uses frameworks like SGLang or vLLM which support model parallelism. Thus, running GLM-4.6 on-premise would require a **multi-node, NVLink-connected cluster** to allow sufficient memory and throughput. Each H100 has NVLink connectivity; expecting 16–32 of them suggests either a multi-node cluster or an enormous single node if supported.
- **Scalability:** As MoE, GLM-4.x can distribute experts across devices. This means cross-GPU communication is heavy. NVIDIA NVLink or InfiniBand connections are needed to share activations between GPUs.
- **Alternative Hardware:** Given the difficulty, some organizations might prefer upcoming hardware like NVIDIA's **GH200** (Grace Hopper superchip) which pairs GPUs and CPUs closely. The table above shows Zhipu's tests with H200 (Aurora/Grace) chips at half the GPU count. If available, GH200 clusters could run GLM models more efficiently. Ascend 910 or newer Chinese AI chips could in theory serve as well, but support for FP8/training might be lacking.

Below is a summary of the published inference requirements for GLM-4.5 (which are the best concrete data we have). These can be taken as a baseline for GLM-4.6 hardware planning:

Model/Config	Precision	NVIDIA H100 GPUs	NVIDIA H200 GPUs	Context tokens
GLM-4.5 (baseline)	BF16	16	8	128K
GLM-4.5 (baseline)	FP8	8	4	128K
GLM-4.5-Air	BF16	4	2	128K
GLM-4.5-Air	FP8	2	1	128K
GLM-4.5 (full)	BF16	32	16	128K
GLM-4.5 (full)	FP8	16	8	128K

Table: Inference hardware for GLM-4.5 (source: Zhipu AI docs (openlm.ai)). "Full" means configuration capable of utilizing the entire 128K context length. For GLM-4.6 (200K context), required GPUs would likely be approximately double.

For GLM-4.6, one must double or exceed these counts to cover 200K tokens. In practical terms, very few developers will run GLM-4.6 on bare metal; the model is predominantly accessed via SaaS platforms (Novita, Z.ai) where the provider amortizes the hardware costs. Nonetheless, deep-pocket organizations can self-host GLM-4.6 if they invest in a supercomputer-scale setup.

CPU/RAM: In addition to GPUs, serving GLM-4.6 needs terabytes of main memory. Zhipu suggests >1 TB of system RAM for any GLM-4.x setup (openlm.ai). This is because full model weights (355B FP16 parameters ~0.7 TB) plus activation caches must fit in RAM if offloading is not used.

Comparison to Sonnet and GPT

Although not asked directly, it is noteworthy that **Claude Sonnet 4/4.5** and GPT models also require enormous hardware, but Anthropic/OpenAI do not disclose their config. Given that Sonnet 4.5 is likely comparable or bigger than GLM-4.6, we estimate it similarly needs multi-GPU clusters for inference. Early public filings suggested Anthropic uses NVIDIA A100/H100 fleets to serve Claude. GPT-4 similarly needed many H100s. GPT-5, if to *"unify everything"*, could require thousands of GPUs to run full-capacity demos. But precisely, GLM-4.6's open nature allows us to roughly quantify its own requirements as above.

Case Studies and Applications

Real-World Usage of GLM-4.6

Several platforms and organizations have started integrating GLM-4.6 into developer tools:

- **Novita AI Platform:** Novita (a hong kong–based AI API provider) immediately added GLM-4.6 on release (blogs.novita.ai). They offer a playground where developers can interact with GLM-4.6, citing *"200K Context, $0.6 per 1M in / $2.2 per 1M out"* for usage (blogs.novita.ai). Novita's blog explains that GLM-4.6 feeds richly into coding pipelines: users can prototype full-stack pages via prompts and tests on the cloud. Novita's reported results (GLM-4.6 vs. Sonnet 4) give concrete figures: "48.6% win rate" in CC-Bench tasks (blogs.novita.ai).
- **Z.ai/Coding Agents:** Zhipu's own GLM Coding Plan (an API subscription) upgraded customers to GLM-4.6 automatically (z.ai). This suggests many Chinese developers using Claude-like apps (Roo Code, Kilo Code, etc.) are now effectively using GLM-4.6 under the hood. Reports from Chinese tech press (e.g. SCMP) note that GLM-4.6 will be competing in marketplaces for coding tools alongside Sonnet and others (www.scmp.com).

- **Open-Source Projects:** With weights public, open-source enthusiasts have begun porting GLM-4.6 to frameworks like FastChat, HuggingChat, and custom vLLM stacks. Though still early, community projects aim to integrate GLM-4.6 into chat UIs akin to ChatGPT. This is reminiscent of how local LLaMA clones spread, but GLM-4.6 may be harder to run due to size. Still, free initiatives like "ChatGLM-4" (Chinese community) are monitoring it.
- **Enterprise Experiments:** Anecdotal evidence (developer blogs and forums) indicate Chinese enterprises testing GLM-4.6 for code review, documentation generation, and chatbots. Because of its open license, a bank or aerospace firm could run GLM-4.6 fully on-premise to analyze sensitive codebases without data leaving their servers. No public case studies have been published yet, but experts expect industries that already use GLM-3 or 4.5 to upgrade their internal deployments to 4.6.

Sonnet 4.5 in Practice

Anthropic and partners have also expanded Sonnet's reach in late 2025:

- **IntelliCode and VS Code:** Sonnet 4.5 plugins allow code completion, error checking, and generation directly in IDEs. Extensions in VS Code are now available for Sonnet 4.5, something GLM-4.6 as open source does not yet have a polished official extension (www.theneuron.ai). Nonetheless, GLM-4.6 is expected to work with similar plugins if integrated by third-parties.
- **Browser and Copilot:** Microsoft has reportedly integrated Sonnet 4.5 into GitHub Copilot X (potentially "Grok 4" being Sonnet under the hood). Chat browsers and extensions for Sonnet 4.5 are in development. Most usage data is proprietary, but one can infer that high-end developers and Explorer/Edge Insiders are test-driving Sonnet in coding sessions.
- **Competitive Developer Reactions:** Some public feedback hints that Sonnet 4.5's coding suggestions are highly reliable (few bugs in generated code blocks) (www.theneuron.ai). For example, the The Neuron blogger noted Sonnet's large code focus and extended reasoning. Our earlier cited LinkedIn account (www.linkedin.com) (though anecdotal) illustrates how earlier Sonnet models handled large code files with artifacts. With Sonnet 4.5, those problems are significantly mitigated (the blogger's issue of 2000-line truncation was reported fixed by 4.5).

GPT-5 Rumors and Developer Tools

Since GPT-5 is not launched, only discussion can be speculative. However, some context:

- **Emerging Tools:** By late 2025, OpenAI's upcoming tools (GPT-4o/GPT-4o Opus, etc.) have incorporated persistent memory and more tooling in the interim. The rumor is that GPT-5 will finalize these. Any developers relying on GPT-4/4o in IDEs (via Copilot Chat, etc.) are eagerly awaiting GPT-5. The coding community on Reddit and Twitter is abuzz with conjecture (e.g., The Neuron's piece (www.theneuron.ai)). However, until release, comparisons are premature.

- **Planned Workflows:** OpenAI's public statements suggest GPT-5 will automatically "chain" tasks internally (so user can just say "build a website", and GPT-5 will decide to use vision+code+language modules as needed (www.techopedia.com)). This could make direct side-by-side comparisons with GLM/Claude tricky, as GPT-5 may blur the line between language, image, and code in one unified interface.

Case Study: Code Generation Scenario

As an illustrative example, consider a developer requesting a full-stack web feature:

> *"Build me an interactive dashboard that visualizes sales data, retrieving data from my SQL database via an API, and display it using Chart.js."*

An advanced coding AI must write front-end JavaScript, possibly set up a small backend endpoint, and ensure interactivity. In practice:

- **GLM-4.6** would use its 200K prompt to include any existing schema or partial code, then generate JS/HTML/CSS and perhaps a simple Node.js script to fetch data. It might call tools to validate syntax. Human tests (CC-Bench) show GLM-4.6 can complete multi-turn tasks like this somewhat efficiently (though sometimes requiring steering by the user).
- **Sonnet 4.5** likely excels here: based on its Capabilities, it would plan out the components (possibly even splitting into multiple files via "thinking mode"), ensure compatibility between front/back, and refine until success. Many developers report Sonnet rarely needs corrections on such tasks.
- **GPT-5 (expected)** would be expected to do this at least as well, perhaps with richer design suggestions or integrated vision previews.

While this is hypothetical, it underscores GLM-4.6's real-world utility: it can handle complex, multi-file coding tasks end-to-end (in labor or hours), and does so as well as any open model.

Implications and Future Directions

Open-Source vs Proprietary Debate

The GLM-4.6 release intensifies the debate over open vs closed AI models. On one hand, proprietary models (Claude, GPT) often lead

IntuitionLabs - Industry Leadership & Services

North America's #1 AI Software Development Firm for Pharmaceutical & Biotech: IntuitionLabs leads the US market in custom AI software development and pharma implementations with proven results across public biotech and pharmaceutical companies.

Elite Client Portfolio: Trusted by NASDAQ-listed pharmaceutical companies including Scilex Holding Company (SCLX) and leading CROs across North America.

Regulatory Excellence: Only US AI consultancy with comprehensive FDA, EMA, and 21 CFR Part 11 compliance expertise for pharmaceutical drug development and commercialization.

Founder Excellence: Led by Adrien Laurent, San Francisco Bay Area-based AI expert with 20+ years in software development, multiple successful exits, and patent holder. Recognized as one of the top AI experts in the USA.

Custom AI Software Development: Build tailored pharmaceutical AI applications, custom CRMs, chatbots, and ERP systems with advanced analytics and regulatory compliance capabilities.

Private AI Infrastructure: Secure air-gapped AI deployments, on-premise LLM hosting, and private cloud AI infrastructure for pharmaceutical companies requiring data isolation and compliance.

Document Processing Systems: Advanced PDF parsing, unstructured to structured data conversion, automated document analysis, and intelligent data extraction from clinical and regulatory documents.

Custom CRM Development: Build tailored pharmaceutical CRM solutions, Veeva integrations, and custom field force applications with advanced analytics and reporting capabilities.

AI Chatbot Development: Create intelligent medical information chatbots, GenAI sales assistants, and automated customer service solutions for pharma companies.

Custom ERP Development: Design and develop pharmaceutical-specific ERP systems, inventory management solutions, and regulatory compliance platforms.

Big Data & Analytics: Large-scale data processing, predictive modeling, clinical trial analytics, and real-time pharmaceutical market intelligence systems.

Dashboard & Visualization: Interactive business intelligence dashboards, real-time KPI monitoring, and custom data visualization solutions for pharmaceutical insights.

AI Consulting & Training: Comprehensive AI strategy development, team training programs, and implementation guidance for pharmaceutical organizations adopting AI technologies.

Contact founder Adrien Laurent and team at https://intuitionlabs.ai/contact for a consultation.

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 North America's leading AI software development firm specializing exclusively in pharmaceutical and biotech companies. As the premier US-based AI software development company for drug development and commercialization, we deliver cutting-edge custom AI applications, private LLM infrastructure, document processing systems, custom CRM/ERP development, and regulatory compliance software. Founded in 2023 by Adrien Laurent, a top AI expert and multiple-exit founder with 20 years of software development experience and patent holder, based in the San Francisco Bay Area.

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