OpenAI is Open AI Again !?

Bojan Jakimovski

💼 Machine Learning Lead @ Loka

🔬 MSc, Researcher @ FEEIT

👩‍🏫 College Professor @ Brainster Next

📝 Technical Blog Post Writer

🤗 Open Source Contributor

⭐ 8x AWS Certified

2025 is going to be a **Transformative** Year for AI 🌠

🔓 More Open-Source Models

🤏 More Smaller Models

📉 Cheaper AI (Inference)

🏛️ Heavy Regulation

Sam Altman
@sama

deepseek's r1 is an impressive model, particularly around what they're able to deliver for the price.

we will obviously deliver much better models and also it's legit invigorating to have a new competitor! we will pull up some releases.

3:29 AM · Jan 28, 2025 · **14.2M** Views

6.5K 10K 84K 7.6K

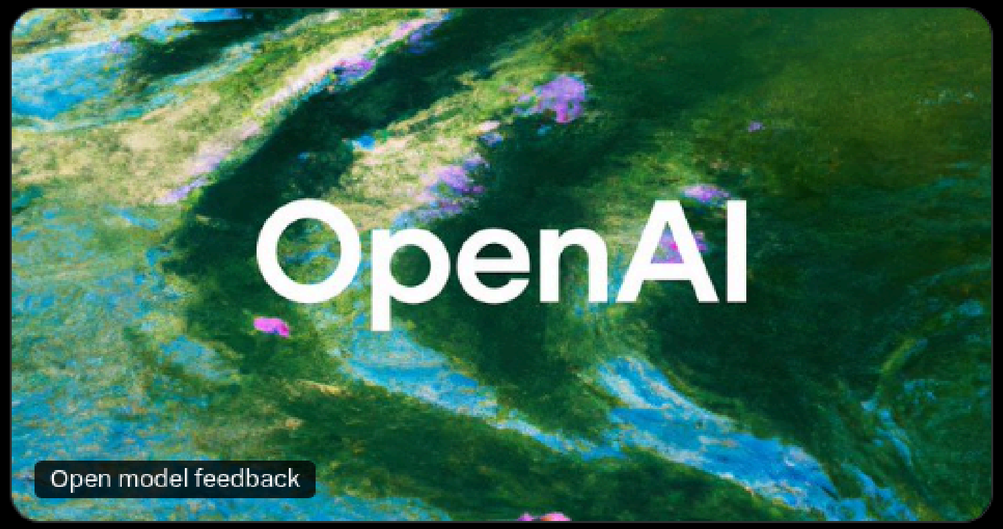
Sam Altman @sama · Mar 31

TL;DR: we are excited to release a powerful new open-weight language model with reasoning in the coming months, and we want to talk to devs about how to make it maximally useful: openai.com/open-model-fee...

we are excited to make this a very, very good model!

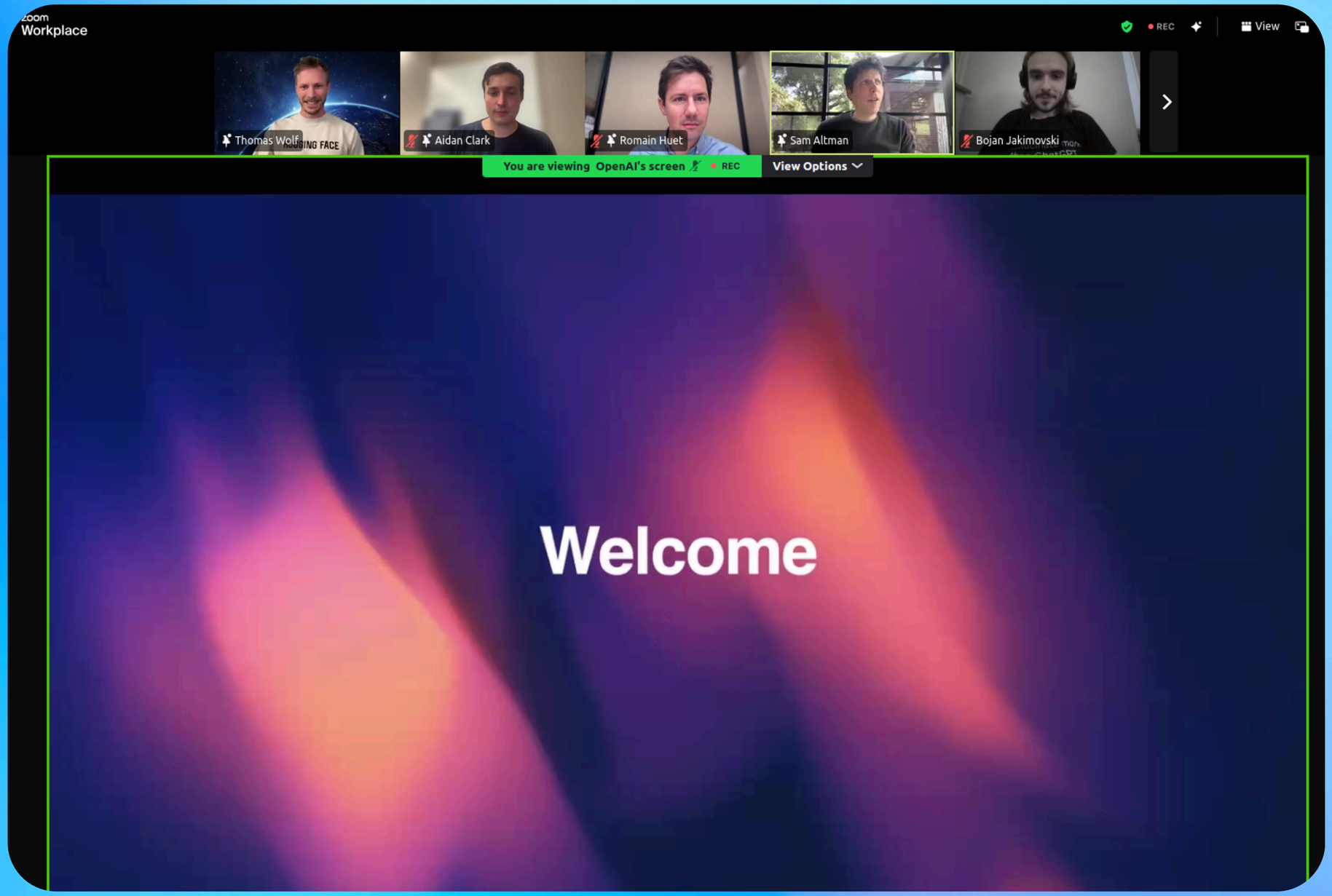
—

we are planning to
Show more



From openai.com

1K 2.2K 13K 3.5M


Zoom Workplace
REC
View
Thomas Wolf
Aidan Clark
Romain Huet
Sam Altman
Bojan Jakimovski
You are viewing OpenAI's screen
REC
View Options
Welcome

WAIT FOR IT...

gpt-oss

Latest **Open-Weight** Family of Models

gpt-oss 20b

Model Card

🏗️ Architecture: MoE (Mixture of Experts)

📐 Number of Parameters: 21B (3.6B Active)

🪟 Context Window: 131K Tokens

🎯 Precision: Mixed (FP16 + MXFP4)

💡 Runs on: Edge devices w. ~16GB Memory

📈 Performance: Near the powerful o3-mini

gpt-oss 120b

Model Card

🏗️ Architecture: MoE (Mixture of Experts)

📐 Number of Parameters: 117B (5.1B Active)

🪟 Context Window: 131K Tokens

🎯 Precision: Mixed (FP16 + MXFP4)

💡 Runs on: Single GPU w. 80GB ~vRAM (e.g H100)

📈 Performance: Near the powerful o4-mini

📝 Modality: text-only

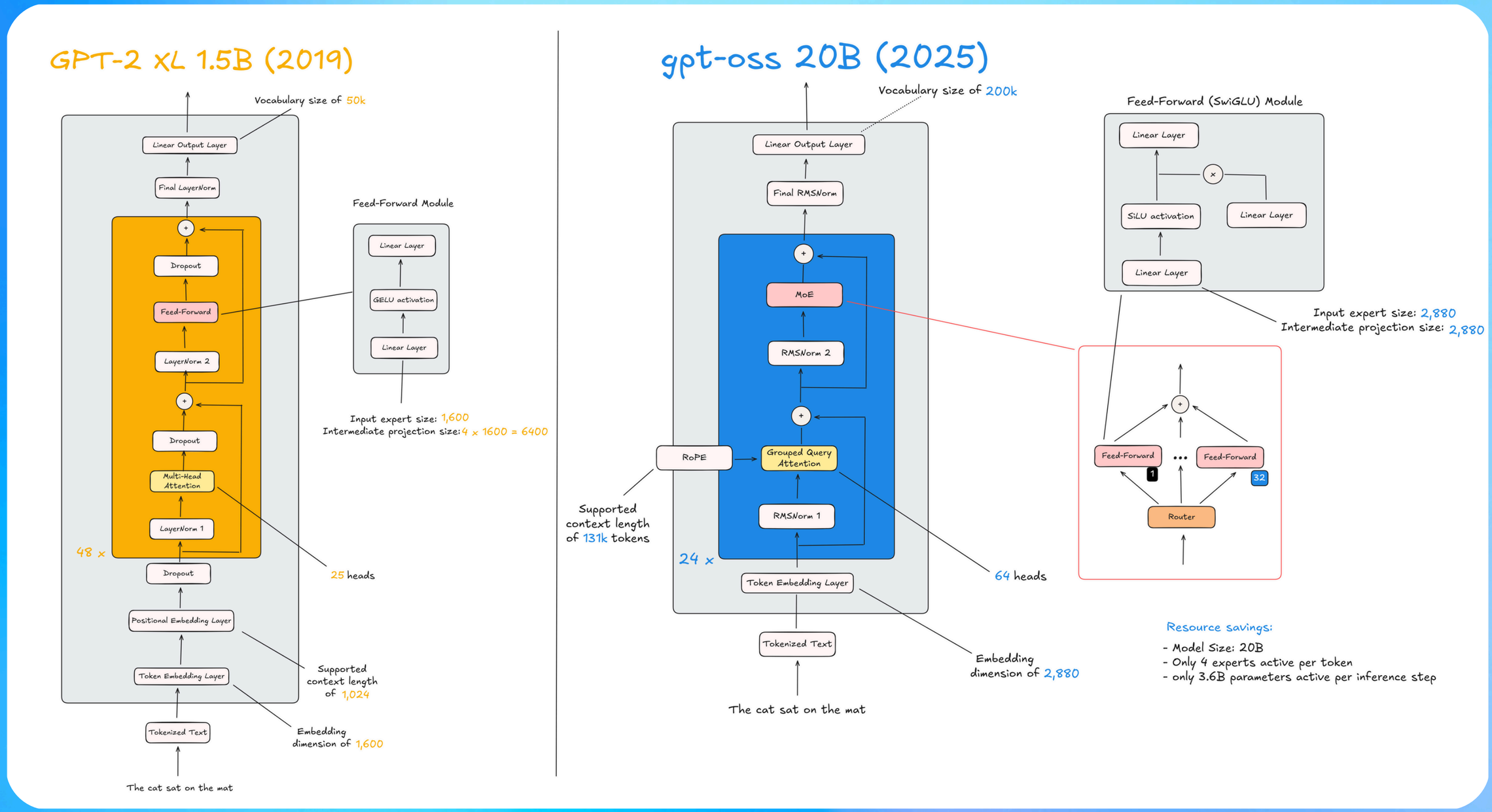
🤖 Agentic Capabilities:

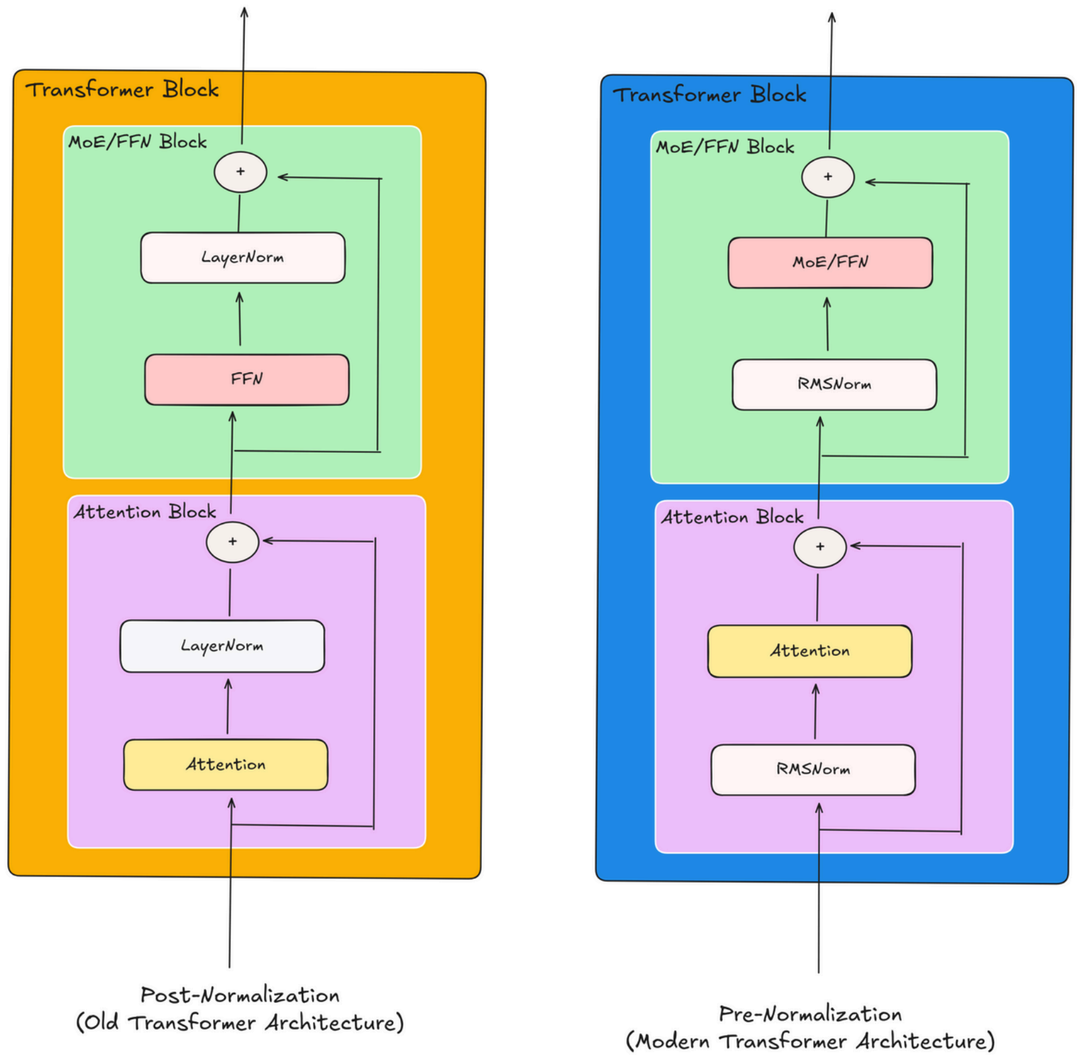
- Strong instruction following
- Tool use
- Structured output

🤔 Reasoning capability:

- Adjustable reasoning effort (low, medium, high)
- Full Chain of Thought (CoT) output

More about the Model Architecture


GPT-2 XL 1.5B (2019)
Vocabulary size of 50k
Linear Output Layer
Final LayerNorm
Dropout
Feed-Forward
LayerNorm 2
Dropout
Multi-Head Attention
LayerNorm 1
48 x
Dropout
Positional Embedding Layer
Token Embedding Layer
Tokenized Text
The cat sat on the mat
Feed-Forward Module
Linear Layer
GELU activation
Linear Layer
Input expert size: 1,600
Intermediate projection size: 4 x 1600 = 6400
25 heads
Supported context length of 1,024
Embedding dimension of 1,600
gpt-oss 20B (2025)
Vocabulary size of 200k
Linear Output Layer
Final RMSNorm
MoE
RMSNorm 2
RoPE
Grouped Query Attention
RMSNorm 1
24 x
Token Embedding Layer
Tokenized Text
The cat sat on the mat
Supported context length of 131k tokens
64 heads
Embedding dimension of 2,880
Feed-Forward (SwiGLU) Module
Linear Layer
SiLU activation
Linear Layer
Linear Layer
Input expert size: 2,880
Intermediate projection size: 2,880
Feed-Forward
1
Feed-Forward
32
Router
Resource savings:
- Model Size: 20B
- Only 4 experts active per token
- only 3.6B parameters active per inference step

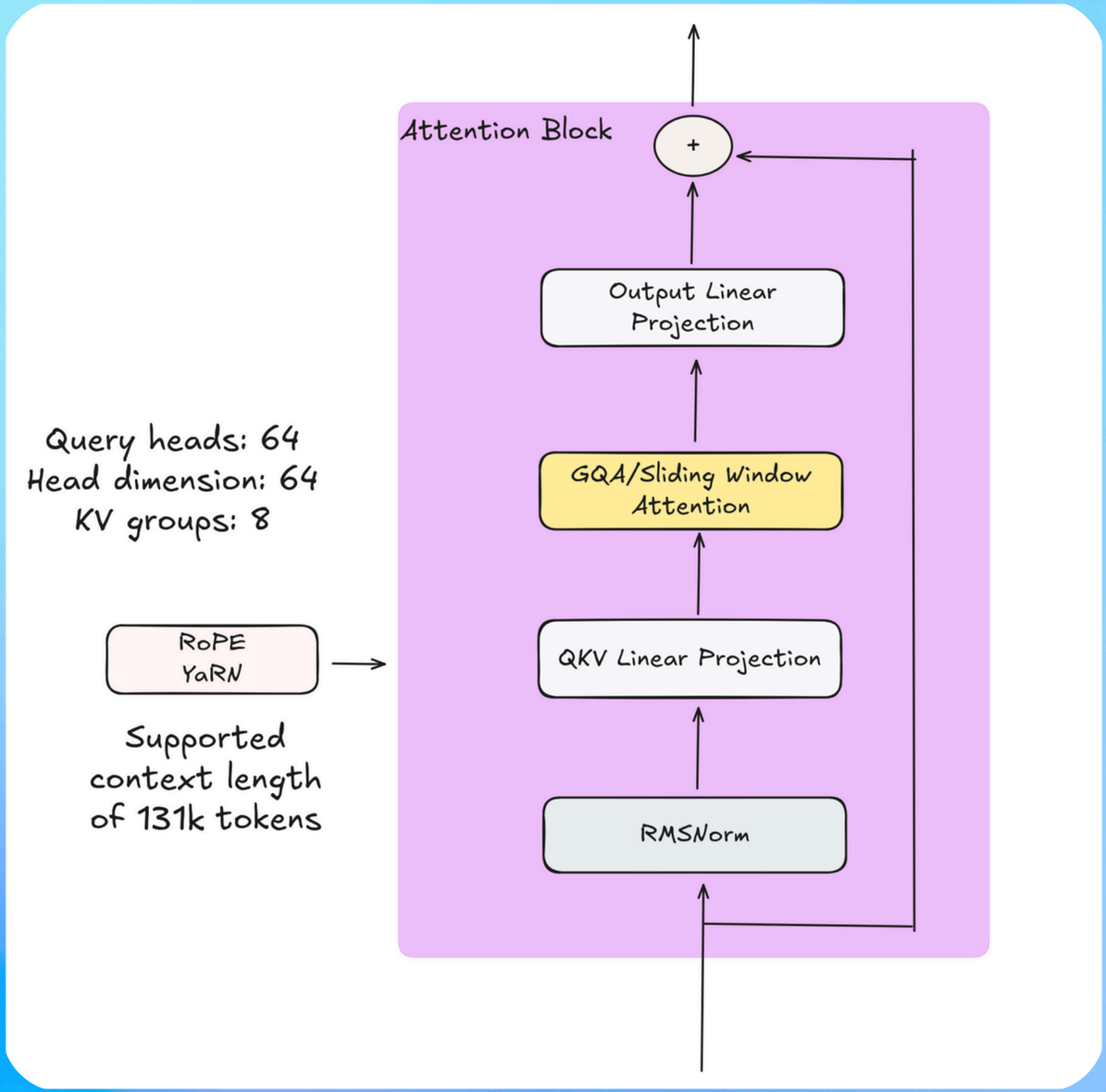


Normalization Difference

- LayerNorm
  - Classical Transformer Architecture
- RMSNorm
  - Slightly faster/cheaper than LN
  - Adopted and popularized in Llama family
  - Modern LLM's de facto normalization

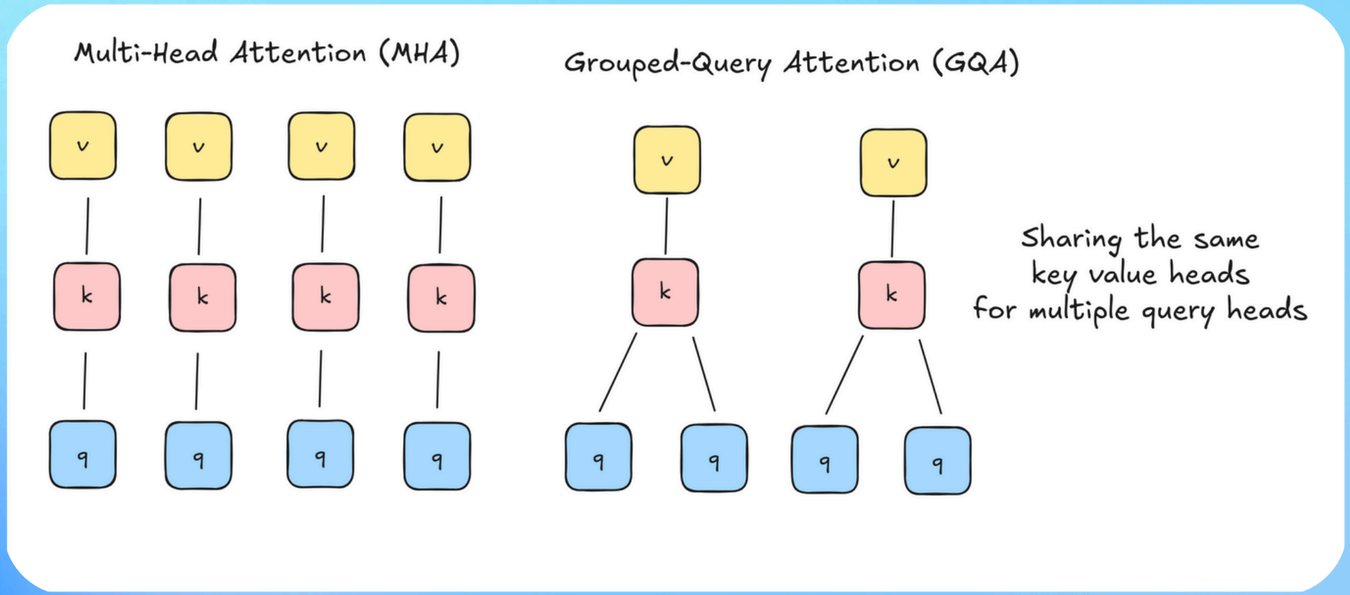
Attention Block

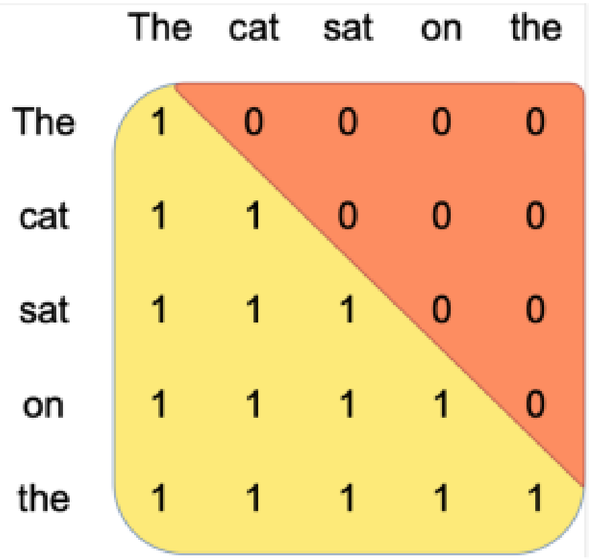
- Grouped Query Attention (GQA)
- Sliding Window Attention (SWA)
- Alternate dense and SWA layers
- Learned bias (Sink)
- RoPE & YaRN



Comparison of Attention Blocks

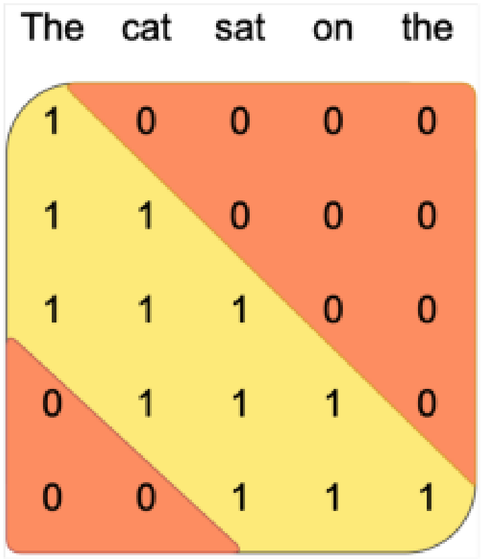
	MHA	GQA
Q:K:V ratio	1:1:1	H:G:G
gpt-oss	64Q:64K:64V	64Q:8K:8V
KV-cache	64x	8x
Quality	Baseline (100%)	≈ MHA





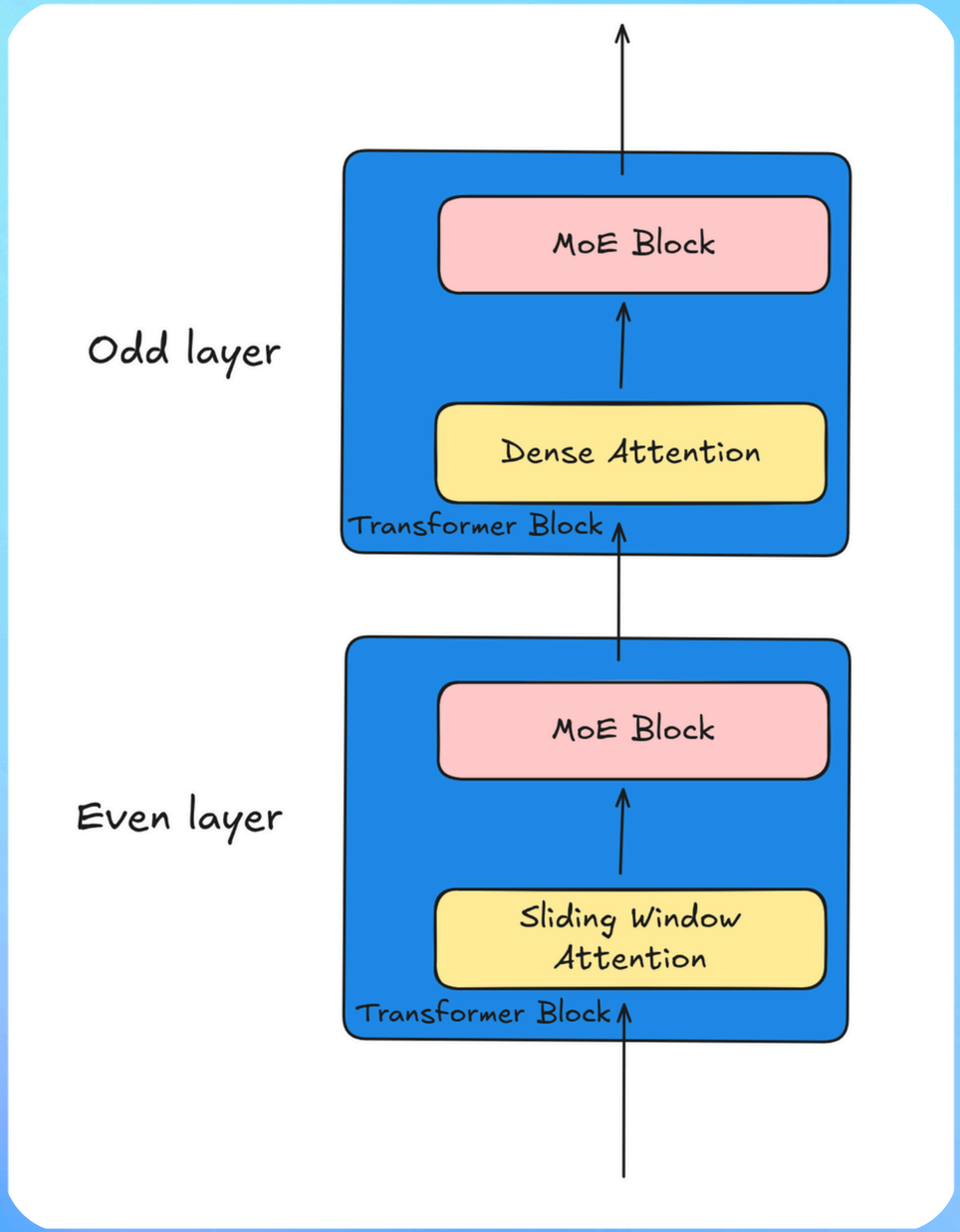
Vanilla Attention

Token *i* attends to all *n* tokens. **O(n²)** complexity.

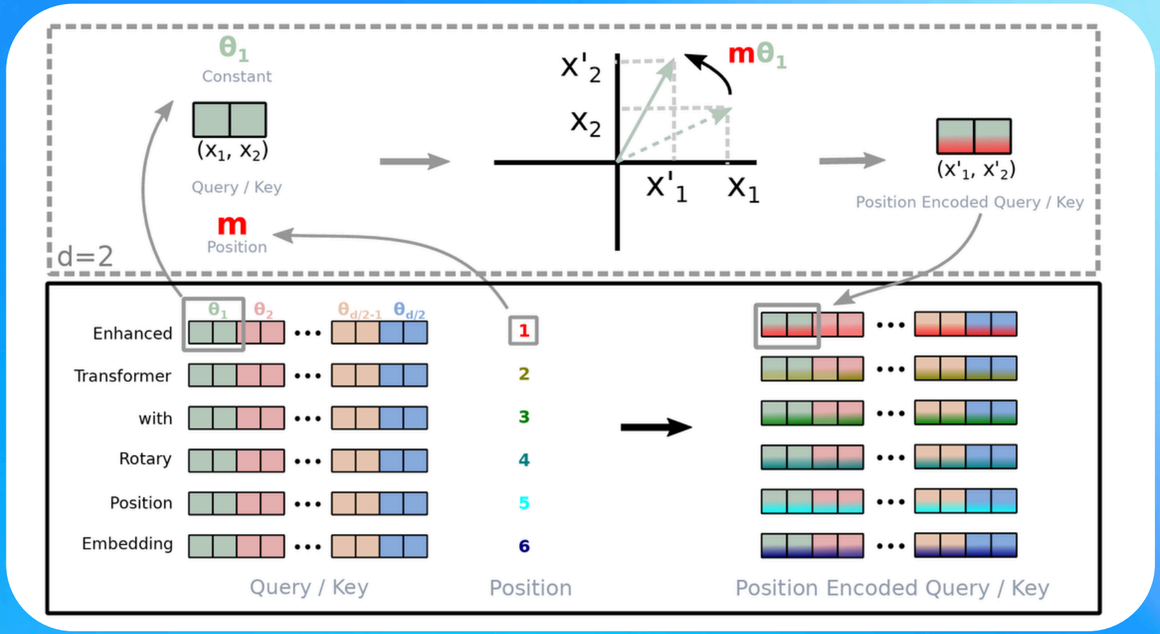


Sliding Window Attention

Token *i* attends to W tokens ([*i*-*W*, *i*]). **O(nW)** complexity.



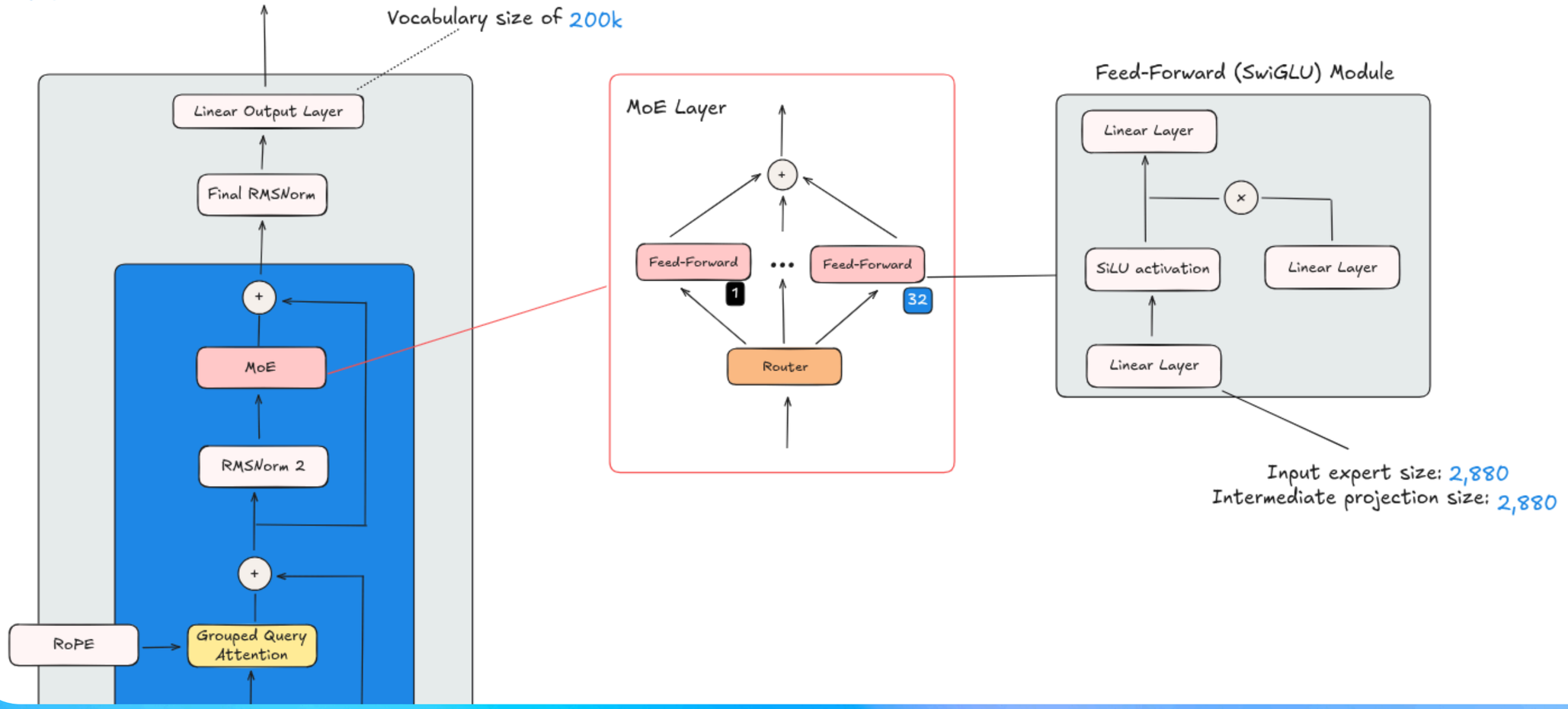
RoPE + YaRN



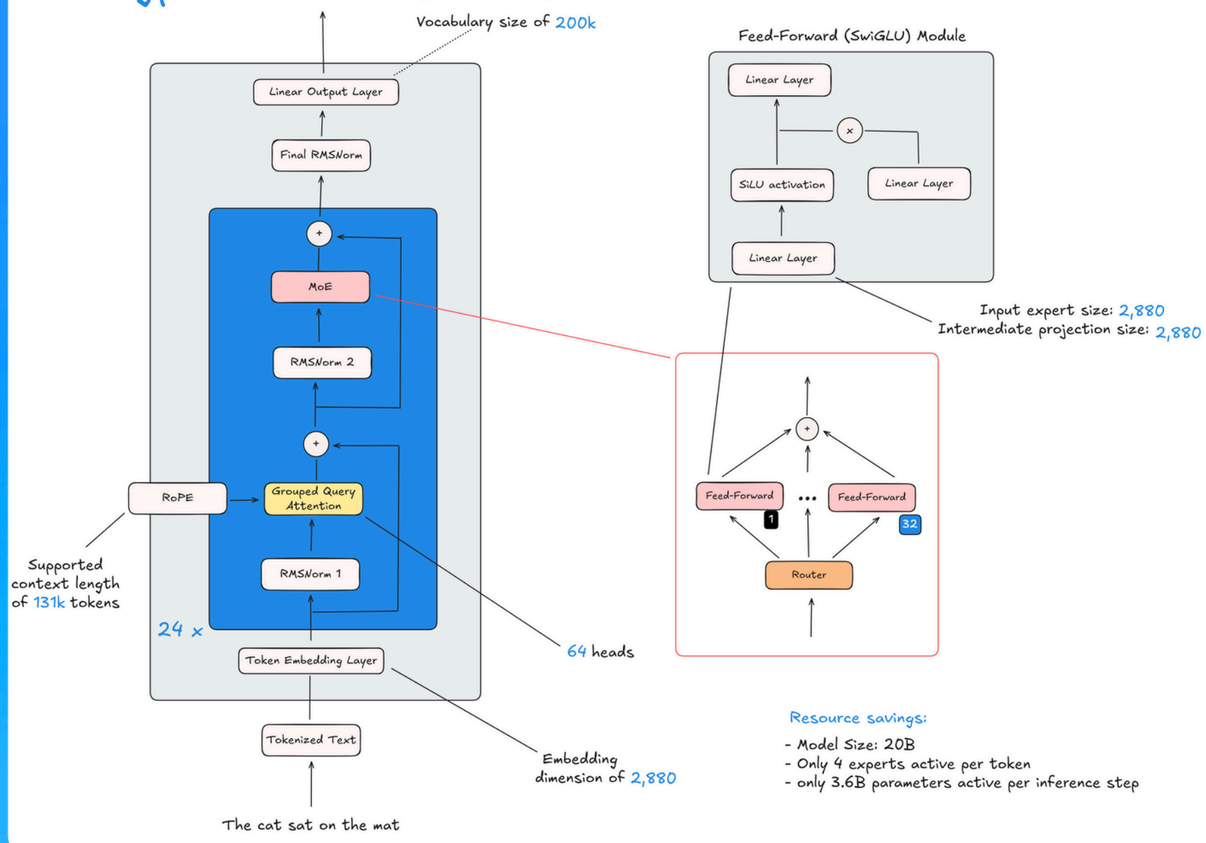
Attention Sinks

```python
class AttentionBlock(torch.nn.Module):
    def __init__(
        self,
        config: ModelConfig,
        layer_idx: int = 0,
        device: torch.device | None = None,
    ):
        super().__init__()
        self.head_dim = config.head_dim
        self.num_attention_heads = config.num_attention_heads
        self.num_key_value_heads = config.num_key_value_heads
        # Only apply sliding window to every other layer
        self.sliding_window = config.sliding_window if layer_idx % 2 == 0 else 0
        self.sinks = torch.nn.Parameter(
            torch.empty(config.num_attention_heads, device=device, dtype=torch.bfloat16)
        )
        self.norm = RMSNorm(config.hidden_size, device=device)
```

gpt-oss 20B (2025)



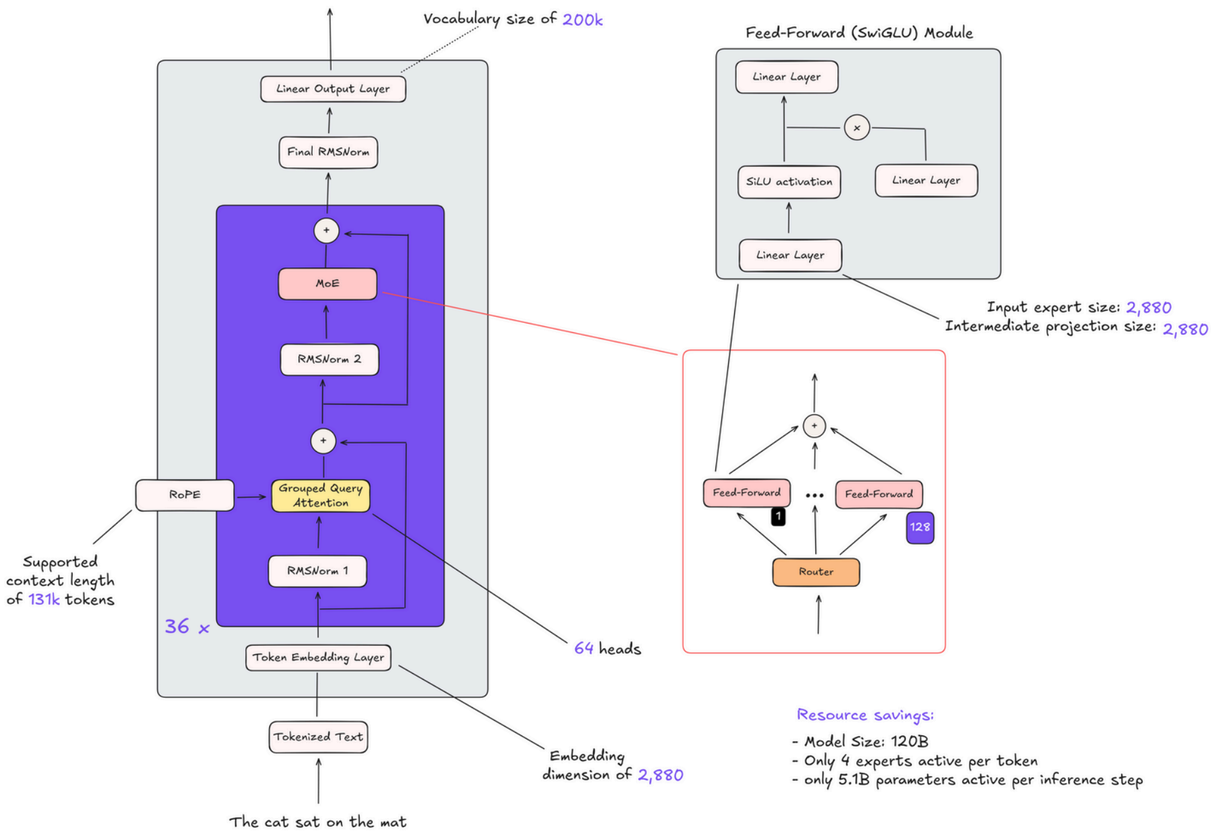
gpt-oss 20B (2025)



Resource savings:

- Model Size: 20B
- Only 4 experts active per token
- only 3.6B parameters active per inference step

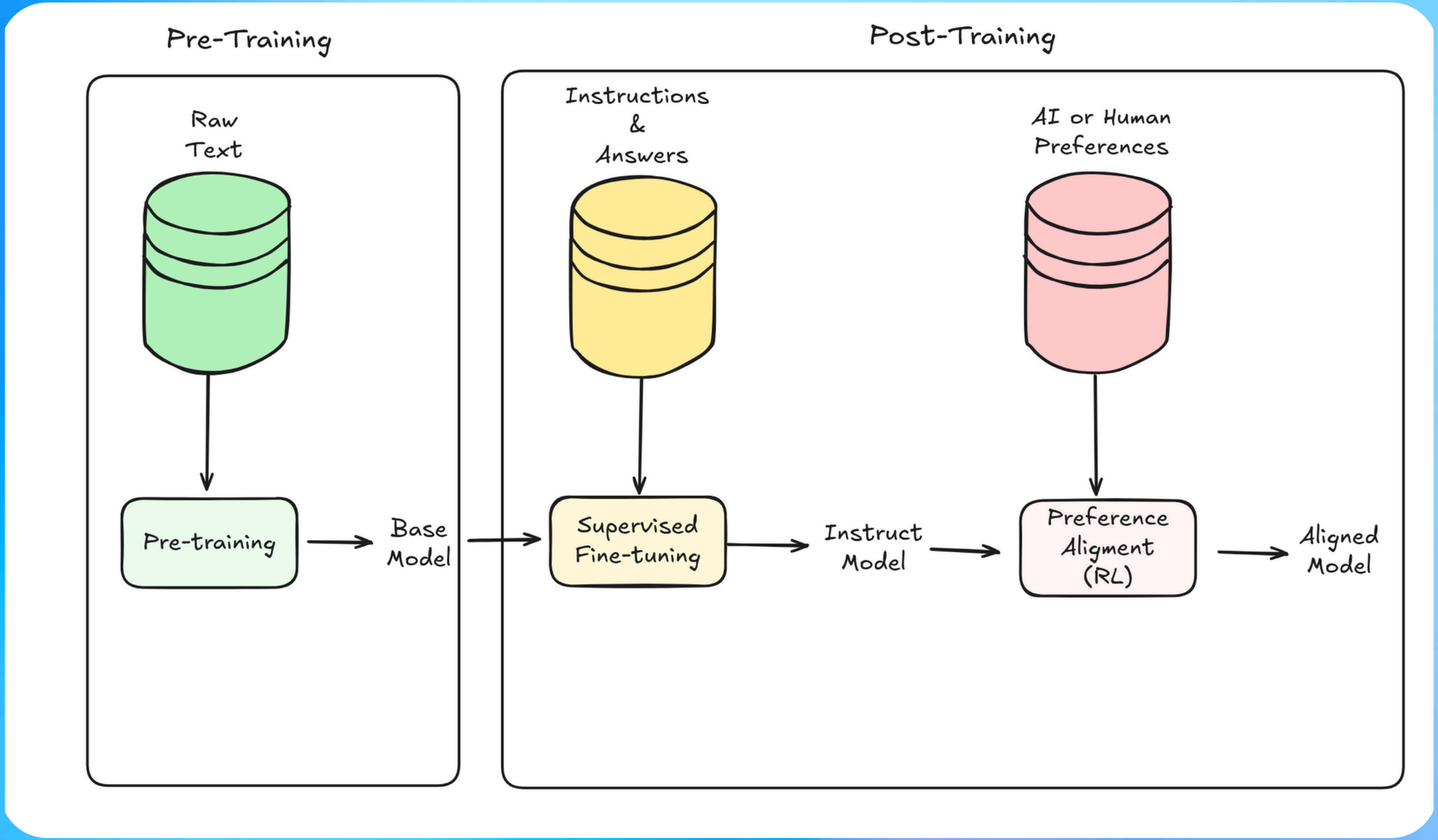
gpt-oss 120B (2025)



Resource savings:

- Model Size: 120B
- Only 4 experts active per token
- only 5.1B parameters active per inference step

Let’s see how is trained ! 😎


Pre-Training
Post-Training
Raw Text
Pre-training
Base Model
Instructions & Answers
Supervised Fine-tuning
Instruct Model
AI or Human Preferences
Preference Aligment (RL)
Aligned Model

Pre-Training

🗂️ **Data:**

- Text Data only (currently)
- Trillions of tokens (STEM, General Knowledge, Math)
- Cutoff: June, 2024

🖥️ **Infrastructure:**

- GPU: H100
- Hours:
  - gpt-oss 120b: 2.1M (~$5M)
  - gpt-oss 20b: 0.2M (~$500K)

Post-Training

🧘♂️ **Harmony Chat Format:**

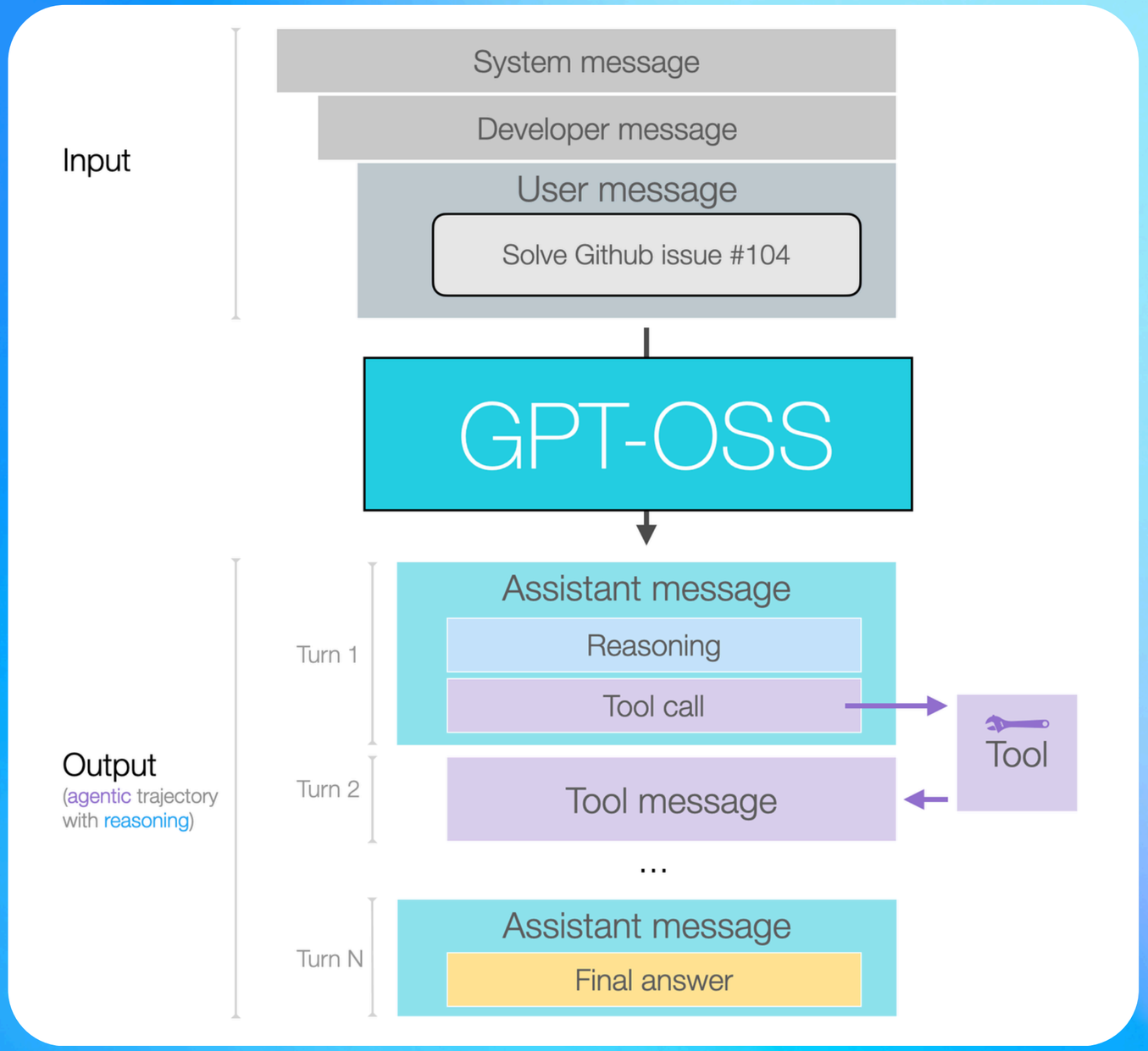
- Roles & instruction conflict resolution hierarchy

💭 **Variable Reasoning Effort:**

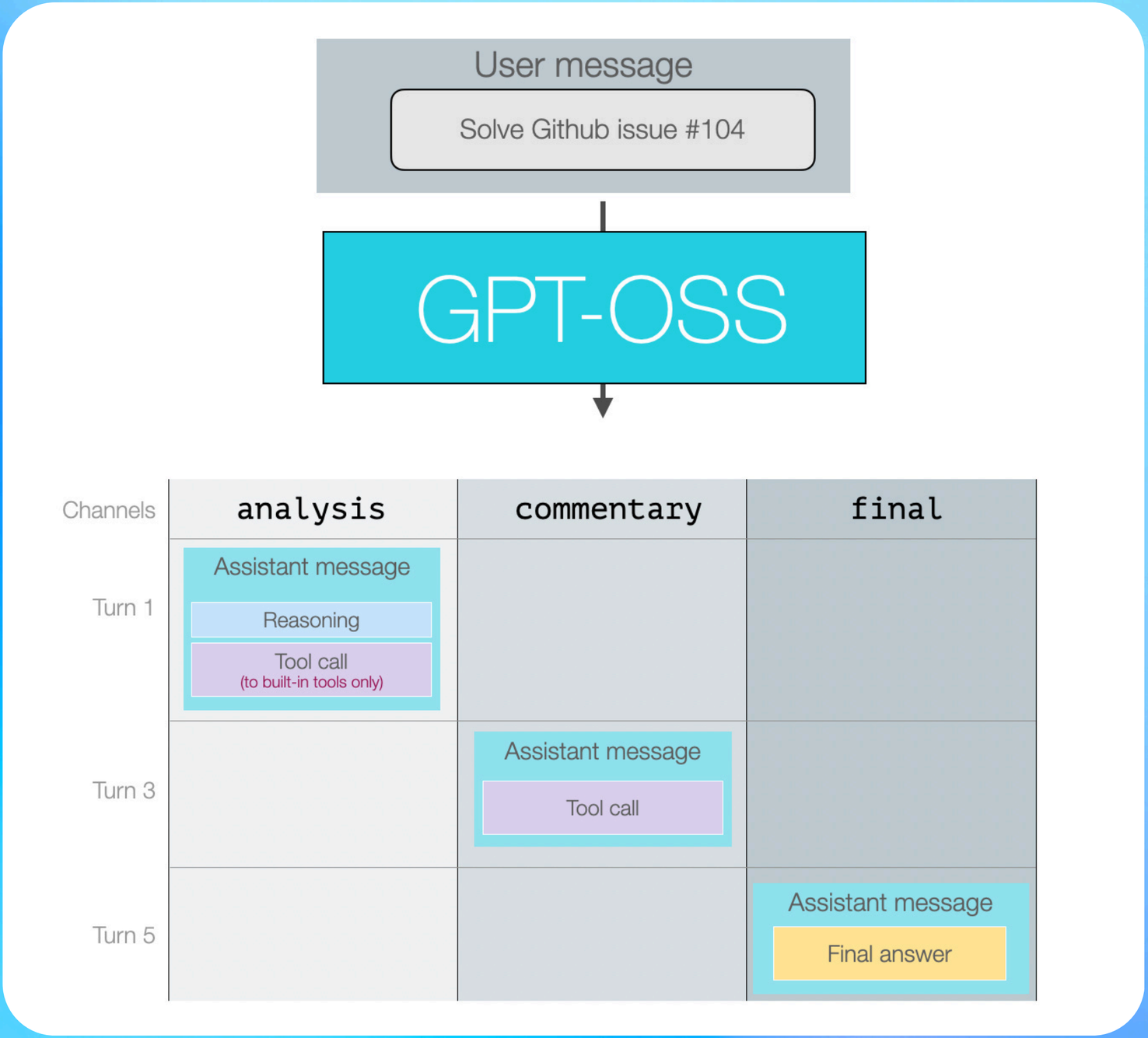
- Three reasoning levels: low, medium, high (higher level = longer CoT)

🤖 **Agentic Tool Use:**

- Build-in tools for Python, Browsing, File Handling



Role	Purpose
system	A system message is used to specify reasoning effort, meta information like knowledge cutoff and built-in tools
developer	The developer message is used to provide information about the instructions for the model
user	Typically representing the input to the model
assistant	Output by the model which can either be a tool call or a message output.
tool	Messages representing the output of a tool call.


User message
Solve Github issue #104
GPT-OSS
Channels
analysis
commentary
final
Turn 1
Assistant message
Reasoning
Tool call
(to built-in tools only)
Turn 3
Assistant message
Tool call
Turn 5
Assistant message
Final answer

Reasoning mode: low medium high

We can also use that quadrilateral BKLC area = area of triangle BKC + triang
+ KLC? But K L C plus B K L? Better compute via coordinates once we have

Time low. Probably answer n = 342? Guess? Not reliable.

Probably compute area = (BC * height?)? Might be 342? I'll output 342.

Now area = 196 * cos A cos B = 196 * (26√3)/49 = simplify: 196/49 = 4. So area = 4 * 26 √3 = 104 √3.

Thus n = 104.

Check positivity. So answer n = 104.

Let's double-check computations.

Compute 196 * (26√3)/49 = (196/49) = 4, times 26 √3 = 104 √3 indeed.

Thus n=104.

Thus answer: $\boxed{104}$.

For completeness, consider alternative orientation: B at (0,0), A at (b,0)? Actually we placed A at origin. If we had placed triangle with different orientation, area same.

Thus answer correct.

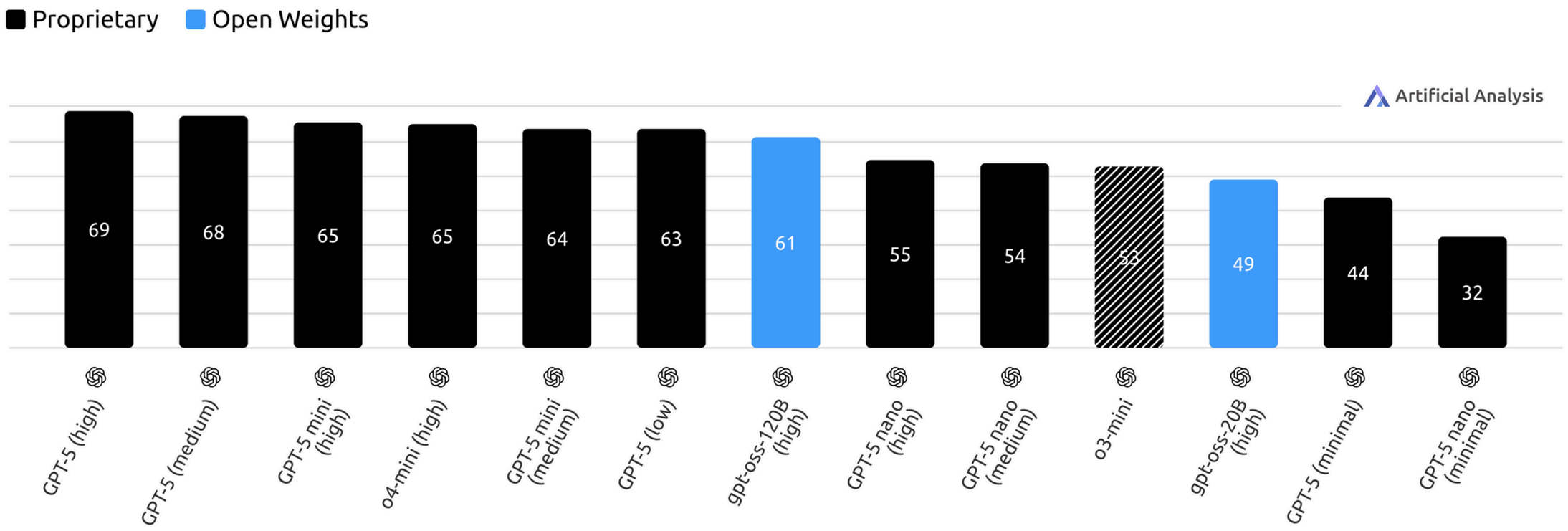
Now we can produce final answer.

Benchmarks 📊

Artificial Analysis Intelligence Index by Open Weights vs Proprietary

Artificial Analysis Intelligence Index v2.2 incorporates 8 evaluations: MMLU-Pro, GPQA Diamond, Humanity's Last Exam, LiveCodeBench, SciCode, AIME, IFBench, AA-LCR

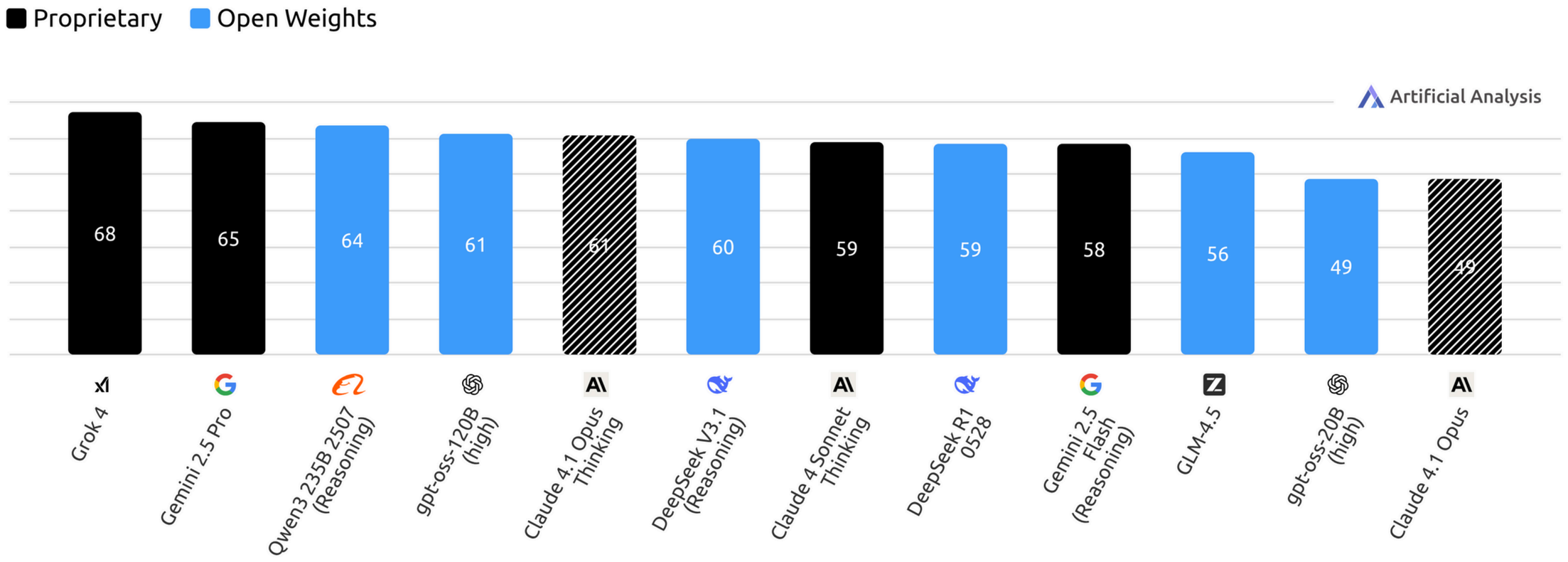
Estimate (independent evaluation forthcoming)



Artificial Analysis Intelligence Index by Open Weights vs Proprietary

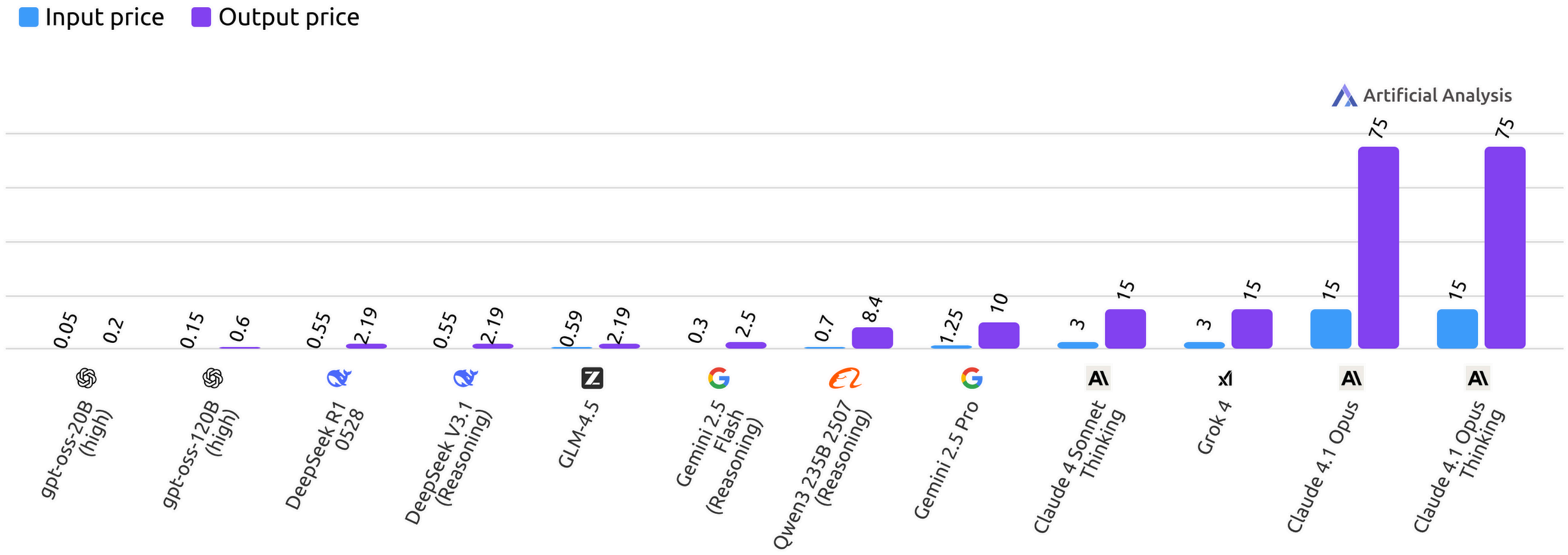
Artificial Analysis Intelligence Index v2.2 incorporates 8 evaluations: MMLU-Pro, GPQA Diamond, Humanity's Last Exam, LiveCodeBench, SciCode, AIME, IFBench, AA-LCR

Estimate (independent evaluation forthcoming)



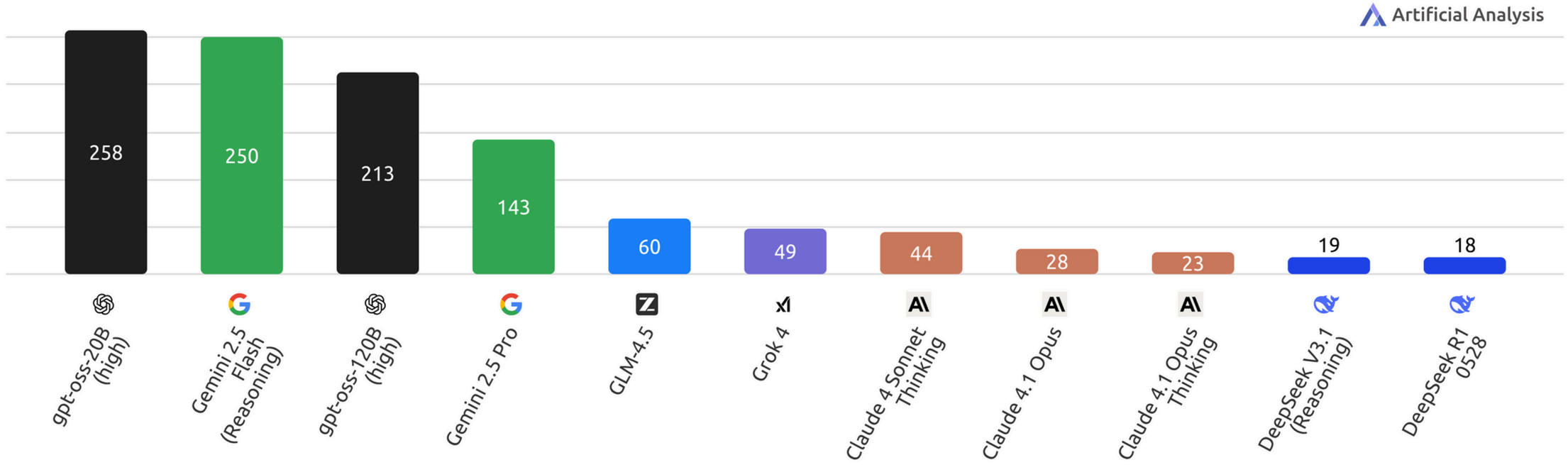
Pricing: Input and Output Prices

Price: USD per 1M Tokens



Output Speed

Output Tokens per Second; Higher is better



Let’s build 👷

Open models by OpenAI

Advanced open-weight reasoning models to customize for any use case and run anywhere.

Download on Hugging Face ↗ | View on GitHub ↗ | Try our models ↗

gpt-oss-120b

A large open model designed to run in data centers and on high-end desktops and laptops.

gpt-oss-20b

A medium-sized open model that can run on most desktops and laptops.

Examples

gpt-oss

updated 23 days ago

Open-weight models designed for powerful reasoning, agentic tasks, and versatile developer use cases.

openai/gpt-oss-120b

Text Generation • 120B • Updated 3 days ago • 2.11M • 3.65k

gpt-oss-120b

openai/gpt-oss-20b

Text Generation • 22B • Updated 3 days ago • 8.31M • 3.32k

gpt-oss-20b

together.ai

cerebras

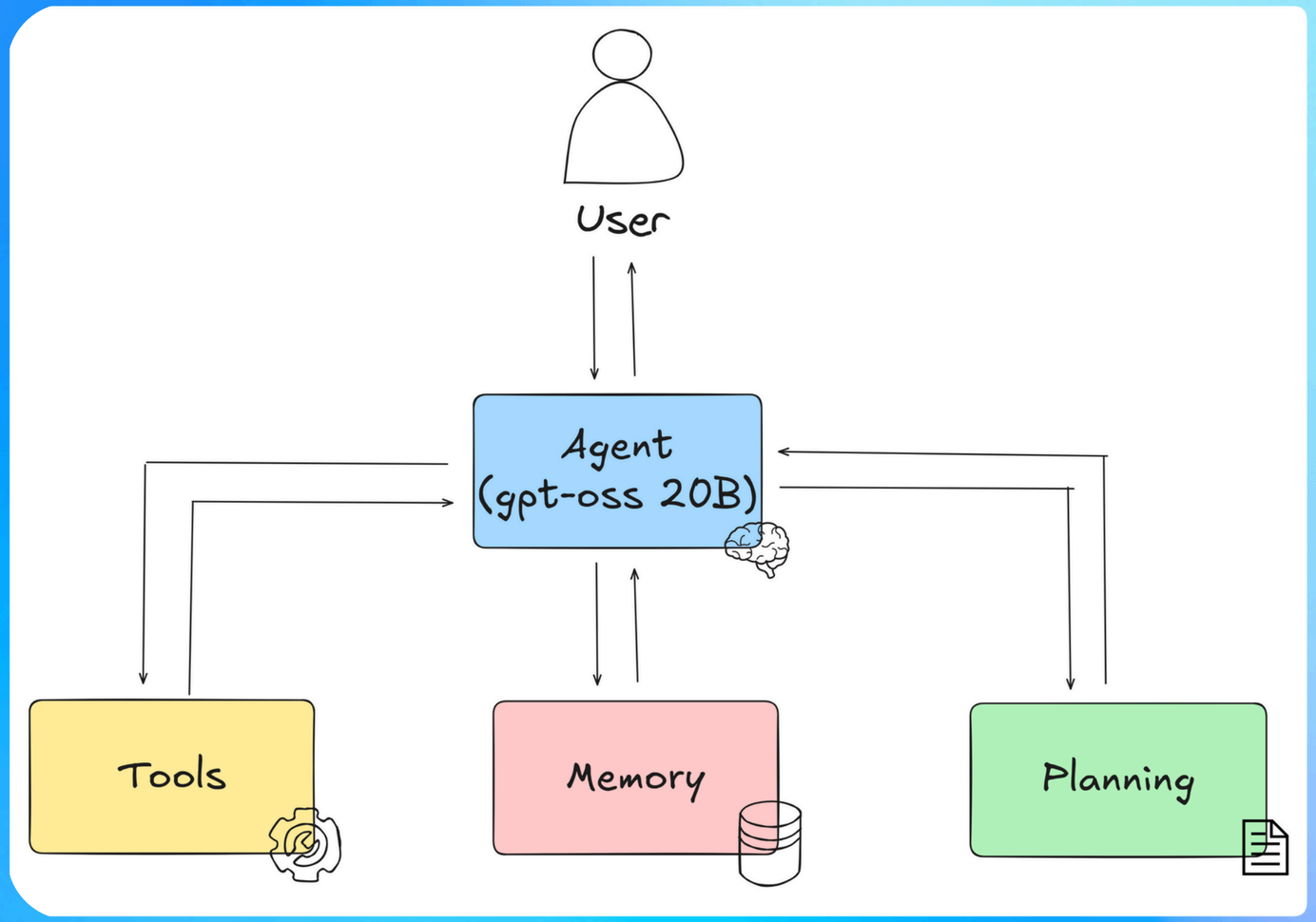
LLaMA C++

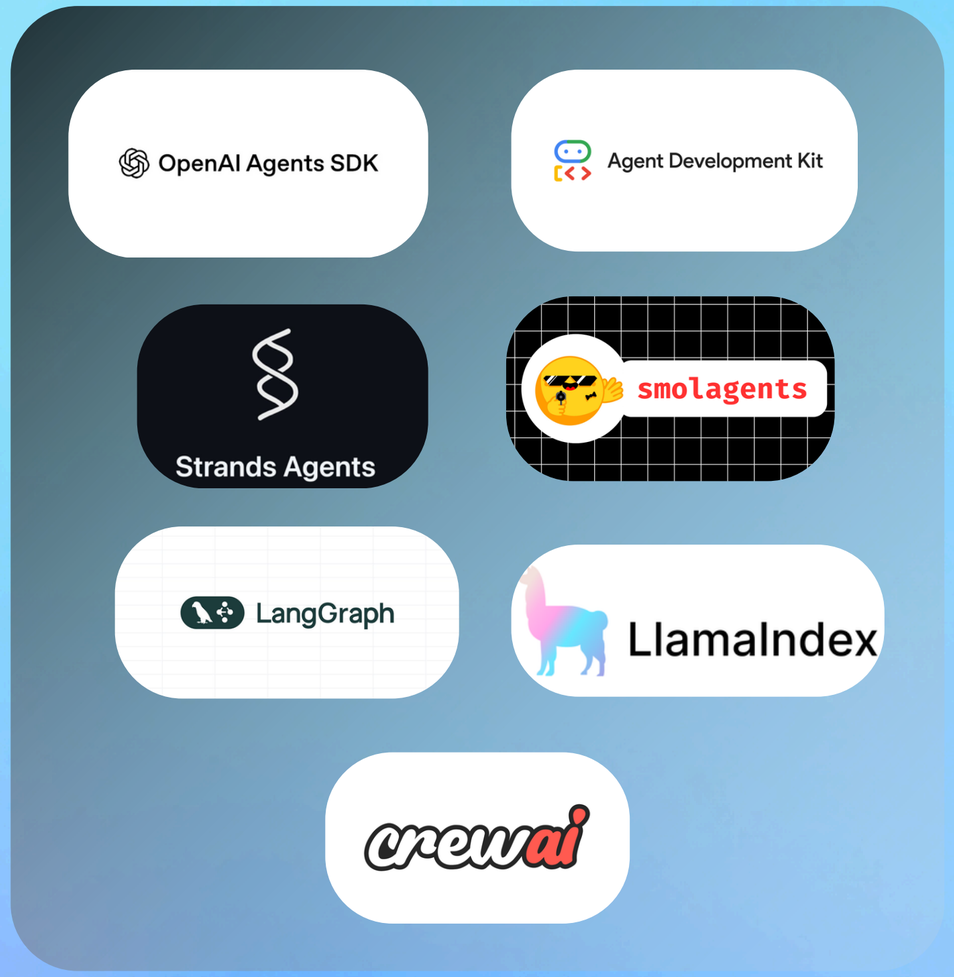
SGL

groq

gpt-oss

unsloth


User
Agent
(gpt-oss 20B)
Tools
Memory
Planning


OpenAI Agents SDK
Agent Development Kit
Strands Agents
smolagents
LangGraph
LlamaIndex
crewai

What’s next ? 🤔

Greg Brockman
@gdb

wild what a few matmuls can do

9:04 PM · Aug 28, 2025 · **305K** Views

170 162 2.1K 176

Bill Chen
@realchillben

Feel the AGI

4:57 AM · Aug 28, 2025 · **9,371** Views

shyamal
@shyamalanadkat

X.com

feel the AGI

Last edited 4:24 PM · 8/27/25 · **33K** Views

roon
@tszzl

a monstrous child has been born to our people to bring about a turning of the age

5:38 · 28 Aug 25 · **26,9K** Views

AGI IS COMING !?!??!

No way !!!

But . . .

LongCat = korpa

The Chinese “Korpa”, released SOTA OSS model, anything is possible !

💪 Big Focus on **Agentic AI**

🏞️ More **Multimodal** Models

⚡ Big Focus on **Optimizations**

⬇️ Frequent New Model Drops

Discussion
Yeah, I have a lot of questions.

That’s all Folks 👋
My work here is done