LLM Benchmark - Summer 2025

Rinat Abdullin

22 August, 2025

Table of Contents

Many have asked - when are the LLM benchmarks coming back online on a regular schedule? Here we are, with a lot of new material to catch up:

- Sharing the secrets - Schema-Guided Reasoning.
- OpenAI GPT-5 Releases are a Big Deal
- A structural problem with the GPT-5 release
- Grok-4 shares the top place
- Gemini 2.5 Pro
- Qwen-3 is still quite popular
- DeepSeek - incremental improvements
- Enterprise Reasoning Challenge (ERCr3)

So let's get started!

1 Schema-Guided Reasoning (SGR)

We finally have a term for the Custom Chain-of-Thought (or SO CoT) approach that we've been using heavily in various projects.

This approach was initially extracted from the successful cases in our AI portfolio and further refined by AI R&D work in the community (including successful submissions in Enterprise RAG Challenges).

In fact, all **evals from our Reasoning LLM Benchmark v2 (starting from January 2025) leverage specialised SGR schemas to drive the reasoning**.

You can read more about SGR here[1] or check out the publicly shared demo[2]. This demo demonstrates how to use SGR to build a business assistant capable of:

- planning and reasoning while using an inexpensive non-reasoning model
- calling tools to help manage customers in a fictional company that sells AGI courses (in the demo we simulate tools to create invoices, send emails and pull customer data)
- creating additional rules and memories for itself

All that is done in 160 lines of Python code without the use of any AI frameworks or tool calling. Just OpenAI SDK and Pydantic.

This topic of creating and orchestrating agents for business tasks is something that we are quite interested in and keen to push state-of-the-art in the field further. We'll have **one more interesting announcement** along these lines in this report later, but for now let's check out which models are the best for working with business tasks given Schema-Guided Reasoning.

1. https://abdullin.com/schema-guided-reasoning/
2. https://abdullin.com/schema-guided-reasoning/demo

2 OpenAI GPT-5 Releases are a Big Deal

Let's start with the obvious big wins. OpenAI has released a range of models recently:

- available via API: gpt-5, gpt-5-mini and gpt-5-nano (announcement[3])
- downloadable: gpt-oss-20b and gpt-oss-120b (announcement[4] / download[5])

gpt-5 from OpenAI is currently the TOP-1 model on our leaderboard!

#	Model	bi	compliance	code	reason	Score	Err	Local	Features
1	**openai/gpt-5-2025-08-07**	**54%**	**70%**	**100%**	**77%**	**79.4%**			**SO, Reason**
2	x-ai/grok-4	54%	70%	100%	77%	79.4%			SO, Reason
3	openai/o3-mini-2025-01-31	45%	70%	100%	74%	76.7%			SO, Reason
4	openai/o4-mini-2025-04-16	45%	70%	100%	74%	76.7%			SO, Reason
5	openai/gpt-5-mini-2025-08-07	54%	70%	93%	74%	76.7%			SO, Reason
6	openai/gpt-oss-120b	54%	67%	92%	72%	75.0%		✓	Open
7	x-ai/grok-3-mini	54%	62%	97%	71%	74.0%	3		
8	google/gemini-2.5-pro-preview-06-05	45%	70%	100%	71%	73.9%			Reason
9	google/gemini-2.5-flash-preview:thinking	45%	57%	100%	68%	71.2%	1		Reason
10	google/gemini-2.5-pro-preview-03-25	45%	70%	93%	68%	71.1%			Reason
11	qwen/qwen3-32b	54%	40%	96%	68%	71.1%	1	✓	Reason, Open

gpt-5 is the smartest model, also it is quite large, slow and expensive. It is also an overkill for day-to-day business automation tasks at scale. For that we have smaller models like GPT-5-mini. And this is where the interesting things start.

gpt-5-mini is currently on the 5th place of the leaderboard, when running under Schema-Guided Reasoning. It is a reasonably priced and capable model - very well balanced.

However, in addition to API models, OpenAI has also released two open-weights models that you can freely download and run on your hardware. For example:

- Download from HuggingFace[6]
- Run on your local hardware with ollama[7]
- Or experiment with via OpenRouter[8]

The most curious part is that **gpt-oss-120b model looks very similar to gpt-5-mini model on our benchmark. It is as if two models were almost the same**.

Either way, this is the first time in forever to see a model from the TOP-5 that is shared publicly for free use.

---

3. https://openai.com/index/introducing-gpt-5/
4. https://openai.com/index/introducing-gpt-oss/
5. https://huggingface.co/blog/welcome-openai-gpt-oss
6. https://huggingface.co/blog/welcome-openai-gpt-oss
7. https://ollama.com/library/gpt-oss
8. https://openrouter.ai/models?q=gpt-oss

#	Model	bi	compliance	code	reason	Score	Err	Local	Features
1	**openai/gpt-5-2025-08-07**	**54%**	**70%**	**100%**	**77%**	**79.4%**			**SO, Reason**
2	x-ai/grok-4	54%	70%	100%	77%	79.4%			SO, Reason
3	openai/o3-mini-2025-01-31	45%	70%	100%	74%	76.7%			SO, Reason
4	openai/o4-mini-2025-04-16	45%	70%	100%	74%	76.7%			SO, Reason
5	**openai/gpt-5-mini-2025-08-07**	**54%**	**70%**	**93%**	**74%**	**76.7%**			**SO, Reason**
6	**openai/gpt-oss-120b**	**54%**	**67%**	**92%**	**72%**	**75.0%**		✓	**Open**
7	x-ai/grok-3-mini	54%	62%	97%	71%	74.0%	3		
8	google/gemini-2.5-pro-preview-06-05	45%	70%	100%	71%	73.9%			Reason
9	google/gemini-2.5-flash-preview:thinking	45%	57%	100%	68%	71.2%	1		Reason
10	google/gemini-2.5-pro-preview-03-25	45%	70%	93%	68%	71.1%			Reason
11	qwen/qwen3-32b	54%	40%	96%	68%	71.1%	1	✓	Reason, Open
12	anthropic/claude-3.7-sonnet:thinking	54%	32%	100%	67%	70.4%	1		Reason
13	openai/o1-2024-12-17	45%	70%	84%	67%	70.0%			SO, Reason
14	deepseek/deepseek-r1-0528	45%	62%	93%	66%	68.9%		✓	SO, Reason, Open
15	**openai/gpt-4.1-2025-04-14**	**45%**	**70%**	**77%**	**67%**	**67.2%**			**SO**
16	**openai/gpt-5-nano-2025-08-07**	**36%**	**67%**	**90%**	**63%**	**66.7%**			**SO, Reason**
17	deepseek/deepseek-r1	27%	64%	100%	63%	66.1%		✓	SO, Reason, Open
18	**openai/gpt-oss-20b**	**36%**	**70%**	**88%**	**63%**	**66.1%**		✓	**Open**

Likewise, gpt-5-nano model scored 16th place on our leaderboard. gpt-oss-20b model had very similar results, taking 18th place.

While the models are called gpt-oss, they are not exactly Open Source models, but rather Open Weights models. This means that one can download and use these models freely, but the original training data and pipelines are not shared.

gpt-oss models leverage Mixture-of-Experts (MoE)[9] architecture, where only a small part of the model is used to generate each new token. This makes these models really fast.

You can run gpt-oss-120B model on a single H100 GPU (it requires a modern GPU with 80 GB VRAM), while gpt-oss-20B requires a modern GPU with 16GB of VRAM like RTX 5090).

Now, the 'modern GPU' requirement normally means such models would not be natively supported by older GPUs like the 4090 or A100. However, there's another unusual catch.

Thanks to the MoE architecture of gpt-oss, you can also run these local models at lower speeds with surprisingly little VRAM as well.

9. https://en.wikipedia.org/wiki/Mixture_of_experts

For example, 120B can run fairly well (10-30 tokens per second) on older cards with just 5-8GB of VRAM. In that case we are keeping attention part of the model in GPU, while all the experts reside in usual RAM (you would need 64GB for that). You can read more about configuring llama.cpp to run these models on Reddit (discussion[10]).

TL;DR: there is a new `--cpu-moe` switch that supports MoE offloading to CPU (not yet supported in Ollama).

10. https://www.reddit.com/r/LocalLLM/comments/1mix4yp/getting_40_tokenssec_with_latest_openai_120b/

3 A structural problem with the GPT-5 release

There is one problem with the GPT-5 release. It uses a completely new response format for defining conversations, called: OpenAI Harmony[11]. This format doesn't play well with Structured Outputs[12] as of yet. This means that OpenAI API occasionally fails to fullfill its promise to always return JSON that complies with the provided schema.

We encountered failures with OpenAI SDK, when gpt-5, gpt-5-mini and gpt-5-nano returned responses that were incompatible with the provided schema. Here is the Gist that reliably reproduces the problem with all GPT-5 models: SGR triggers Harmony parsing bug with GPT-5 models[13]. We reported it to OpenAI directly, and also shared it with the OpenAI Community[14] with a repro, so you can check full details yourself.

Not only does **the problem go away when switching from gpt-5 models back to gpt-4o**, the responses also get much faster. There are two possible reasons for that:

- gpt-4o doesn't use more complicated Harmony response format
- gpt-4o is not a reasoning model. GPT-5 models like to think under the hood before responding, while gpt-4o just answers.

Issues with the new gpt-5 models don't stop at the API, they also affect gpt-oss-120B and gpt-oss-20B models. None of the public LLM providers that offer APIs with Structured Outputs yet provide access to these that works with StructuredOutputs. Even Ollama struggles with this new format (see ticket[15])

How did we get GPT-5 models to work reliably in our benchmark? We didn't. We simulated a working constrained decoding by discarding all responses with invalid schema until a valid one was produced.

We are pretty sure that the integration issues will be resolved soon enough. Then we would get a great local model that is smart, fast and can be led to reason within SGR, further boosting its capabilities.

11. https://github.com/openai/harmony
12. https://platform.openai.com/docs/guides/structured-outputs
13. https://gist.github.com/abdullin/332b03de6b86a134eedbc2e4b8379736
14. https://community.openai.com/t/harmony-based-gpt-5-models-return-malformed-structured-outputs-sdk-1-100-2/1353934
15. https://github.com/ollama/ollama/issues/11691

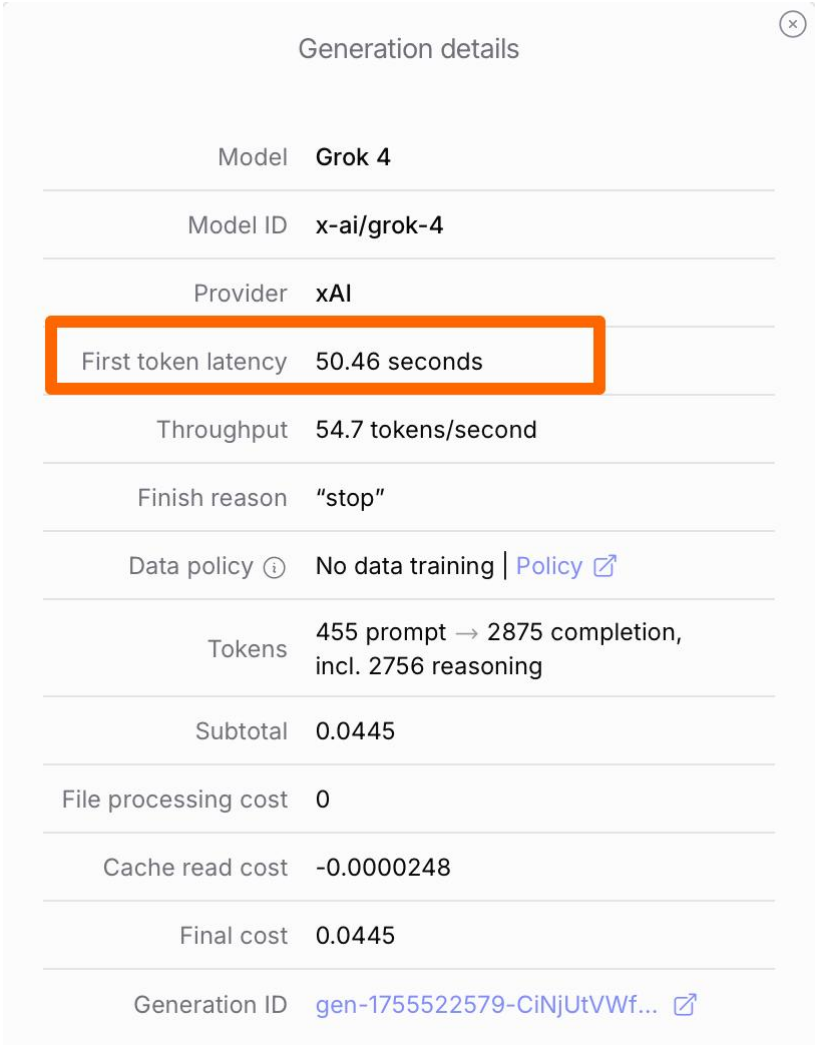
4 Grok-4 shares the top place with OpenAI GPT-5

Grok models historically scored low on our benchmarks. However Grok-4[16] suddenly jumped to the top of the leaderboard, getting scores similar to GPT-5 (medium reasoning effort).

#	Model	bi	compliance	code	reason	Score	Err	Local	Features
1	openai/gpt-5-2025-08-07	54%	70%	100%	77%	79.4%			SO, Reason
2	**x-ai/grok-4**	**54%**	**70%**	**100%**	**77%**	**79.4%**			**SO, Reason**
3	openai/o3-mini-2025-01-31	45%	70%	100%	74%	76.7%			SO, Reason
4	openai/o4-mini-2025-04-16	45%	70%	100%	74%	76.7%			SO, Reason
5	openai/gpt-5-mini-2025-08-07	54%	70%	93%	74%	76.7%			SO, Reason
6	openai/gpt-oss-120b	54%	67%	92%	72%	75.0%		✓	Open
7	**x-ai/grok-3-mini**	**54%**	**62%**	**97%**	**71%**	**74.0%**	**3**		
8	google/gemini-2.5-pro-preview-06-05	45%	70%	100%	71%	73.9%			Reason
9	google/gemini-2.5-flash-preview:thinking	45%	57%	100%	68%	71.2%	1		Reason
10	google/gemini-2.5-pro-preview-03-25	45%	70%	93%	68%	71.1%			Reason
11	qwen/qwen3-32b	54%	40%	96%	68%	71.1%	1	✓	Reason, Open
12	anthropic/claude-3.7-sonnet:thinking	54%	32%	100%	67%	70.4%	1		Reason
13	openai/o1-2024-12-17	45%	70%	84%	67%	70.0%			SO, Reason

The primary gotcha with **Grok-4 is that it can get quite expensive and slow**. Here is an example of one request from our benchmark, where it took 50 seconds just to start getting response.

16. https://x.ai/news/grok-4


Generation details
Model
Grok 4
Model ID
x-ai/grok-4
Provider
xAI
First token latency
50.46 seconds
Throughput
54.7 tokens/second
Finish reason
"stop"
Data policy
No data training | Policy
Tokens
455 prompt → 2875 completion, incl. 2756 reasoning
Subtotal
0.0445
File processing cost
0
Cache read cost
-0.0000248
Final cost
0.0445
Generation ID
gen-1755522579-CiNjUtVWf...

5 Gemini 2.5 Pro

Gemini 2.5 Pro is currently one of the best general-purpose models to use in business automation tasks. It features a large context (that it can actually work with), can handle multiple modalities and is quite cheap.

The only problem is that **Google LLMs still don't have a proper Structured Output** (similar to the capabilities of Mistral, OpenAI, Fireworks, Cerberas, Grok, or any local deployment). They feature only a limited subset that can be a pain to work with.

Anthropic models have been mediocre at best in the past months. The highest they reached was TOP-12 with a rather expensive claude-3.7-sonnet in thinking mode. Anthropic, however, doesn't support Structured Outputs via constrained decoding, which makes integration with their LLMs rather unreliable.

6 Qwen-3 models are still popular

New Qwen-3 models gained popularity immediately after the initial release at the end of April and are still praised for their quality. In fact, qwen-3-32B holds 11th place on our leaderboard, right above the Claude-3.7-sonnet:thinking

#	Model	bi	compliance	code	reason	Score	Err	Local	Features
1	openai/gpt-5-2025-08-07	54%	70%	100%	77%	79.4%			SO, Reason
2	x-ai/grok-4	54%	70%	100%	77%	79.4%			SO, Reason
3	openai/o3-mini-2025-01-31	45%	70%	100%	74%	76.7%			SO, Reason
4	openai/o4-mini-2025-04-16	45%	70%	100%	74%	76.7%			SO, Reason
5	openai/gpt-5-mini-2025-08-07	54%	70%	93%	74%	76.7%			SO, Reason
6	openai/gpt-oss-120b	54%	67%	92%	72%	75.0%		✓	Open
7	x-ai/grok-3-mini	54%	62%	97%	71%	74.0%	3		
8	google/gemini-2.5-pro-preview-06-05	45%	70%	100%	71%	73.9%			Reason
9	google/gemini-2.5-flash-preview:thinking	45%	57%	100%	68%	71.2%	1		Reason
10	google/gemini-2.5-pro-preview-03-25	45%	70%	93%	68%	71.1%			Reason
11	**qwen/qwen3-32b**	**54%**	**40%**	**96%**	**68%**	**71.1%**	**1**	**✓**	**Reason, Open**
12	anthropic/claude-3.7-sonnet:thinking	54%	32%	100%	67%	70.4%	1		Reason
13	openai/o1-2024-12-17	45%	70%	84%	67%	70.0%			SO, Reason
14	deepseek/deepseek-r1-0528	45%	62%	93%	66%	68.9%		✓	SO, Reason, Open
15	openai/gpt-4.1-2025-04-14	45%	70%	77%	67%	67.2%			SO
16	openai/gpt-5-nano-2025-08-07	36%	67%	90%	63%	66.7%			SO, Reason

In fact, when one of the community members decided to check out what is the smallest possible LLM that can be used to drive AI agents in business scenarios, they managed to port SGR Demo to run on top of Qwen-3-4B via a local llama.cpp deployment.

> The concrete version being used in this sample is quantised version of Qwen3-4B-Instruct-2507[17]: `Qwen3-4B-Instruct-2507-Q8_0`

This is worthy of two comments:

- Teams currently choose Qwen-3 models when they need the smallest capable model.
- Obviously, in real-world business cases running something larger would give some safety margin (e.g. gpt-oss-20B or qwen-3-32B). However, it is just crazy that such a small model can make some sense in fairly complex scenarios.

17. https://huggingface.co/Qwen/Qwen3-4B-Instruct-2507

```
- Got elon@x.com data
- Issued invoice with 2x SKUs
- Sent email to finance@x.com
```

```
╭─ Launch agent with task ──────────────────────────────────────────────────────────────────────────────╮
│ Void last elon@x.com invoice and make new one: provide the discount - 3x of what sama@openai.com got for the same │
│ products and email it to elon                                                                         │
╰───────────────────────────────────────────────────────────────────────────────────────────────────────╯
Planning step_1... Get customer data for sama@openai.com to determine their discount
  tool='get_customer_data' email='sama@openai.com'
Planning step_2... Get customer data for elon@x.com to retrieve last invoice ID
  tool='get_customer_data' email='elon@x.com'
Planning step_3... Void last invoice for elon@x.com (ID: INV-2)
  tool='void_invoice' invoice_id='INV-2' reason='Reissuing invoice with updated discount.'
Planning step_4... Issue new invoice to elon@x.com with 15% discount
  tool='issue_invoice' email='elon@x.com' skus=['SKU-205', 'SKU-205', 'SKU-210', 'SKU-210', 'SKU-220', 'SKU-220']
discount_percent=15
Planning step_5... Send email to finance@x.com with new invoice
  tool='send_email' subject='Your Invoice - INV-3' message='Dear Elon,\n\nHere is your invoice for the products you
ordered:\n\n- AGI 101 Course Personal (SKU-205) x2\n- AGI 101 Course Team (5 seats) (SKU-210) x2\n- Building AGI - online
exercises (SKU-220) x2\n\nTotal amount: $3726.00\nDiscount applied: 15% ($558.90)\n\nThank you!'
files=['/invoices/INV-3.pdf'] recipient_email='finance@x.com'
Planning step_6... agent completed.
──────────────────────────────────────── Summary ────────────────────────────────────────
- Got sama@openai.com's discount rule (5%)
- Calculated 3×5% = 15% for Elon
- Voided INV-2
- Issued new invoice INV-3 with 15% discount
- Sent email to finance@x.com
```

Here is the source code that shows how to upgrade a classical SGR demo to work with Qwen-3-4B[18]. It includes three major changes:

- Removes OpenAI SDK and composes raw requests to the API exposed by llama.cpp
- Extends the prompt by spelling out business rules in more detail
- Adds one more reasoning field in the beginning of the SGR Cascade in `NextStep`

These changes are enough to make even Qwen-3-4B start making sense in a task that requires a multi-step reasoning agentic behaviour.

18. https://gist.github.com/abdullin/bacab6ee38ae23ab78024a85f316c8eb

7 DeepSeek - incremental improvements

DeepSeek models were a big hit when they first came out. However multiple better models came out since then, pushing them down the benchmark ladder.

At the time of writing, the initial deepseek-r1 version has fallen to 17th place, getting the same score in SGR-driven mode as gpt-oss-20B model. deepseek-r1-0528 was just an incremental improvement, getting it to a current 14th place.

However, not that many people would be interested in running 671B model, when there is a much better 120B model available. Qwen3-32B is smaller and also better.

Recently released Deepseek Chat v3.1 didn't perform much better on our SGR benchmark either:

#	Model	bi	compliance	code	reason	Score	Err	Local	Features
14	deepseek/deepseek-r1-0528	45%	62%	93%	66%	68.9%		✓	SO, Reason, Open
17	deepseek/deepseek-r1	27%	64%	100%	63%	66.1%		✓	SO, Reason, Open
26	deepseek/deepseek-r1-distill-llama-70b	36%	32%	96%	56%	60.0%	4	✓	Open
27	deepseek/deepseek-chat-v3-0324	45%	60%	70%	55%	59.6%		✓	Reason, Open
30	**deepseek/deepseek-chat-v3.1**	**36%**	**62%**	**68%**	**57%**	**58.2%**		**✓**	**SO, Open**
31	deepseek/deepseek-r1-0528-qwen3-8b	27%	62%	82%	52%	56.7%	2	✓	Reason, Open
36	deepseek-v3	36%	47%	58%	49%	50.6%	1	✓	SO, Open
66	deepseek/deepseek-r1-distill-qwen-32b	9%	22%	29%	17%	21.2%	2	✓	SO, Open

8 Enterprise Reasoning Challenge (ERCr3)

As you can see, there is a wide variety of capable models showing up. As soon as community figures out how to reliably use Structured Outputs with gpt-oss models and their Harmony response format, we will get into a very interesting situation:

- there are LLMs that handle business tasks with SGR really well (within the TOP-20)
- you can freely download and use them
- even on a rather non-demanding hardware

This is a big deal, but what about **pushing state of the art in enterprise automation even further together** and **finding patterns to do even more with less**?

If you followed our previous work, you know that we do that by running massive crowdsourced experiments together with the community of enthusiasts and independent teams (e.g. see Enterprise RAG Challenge round 2[19]).

We are planning to run the 3rd round of our Enterprise Challenges this Fall/Winter. This time we will **focus on business automation with agents** via APIs.

The objective of the challenge for teams will be: to write an agent that will get human requests like "*redo last elon@x.com[20] invoice: use 3x discount of sama@openai.com[21]*". It will then need to use available (simulated) APIs to find its way and carry out the operation properly.

We will provide participants with the simulated APIs that their agents can call in order to accomplish the tasks. However, it will be the job of an agent to figure out which APIs to call and in which order to accomplish the task.

Most of the times, the full solution will not be known in advance before a request to some API provides missing piece of the puzzle. So it will be the job of an agent to reason through and use proper tools to get the job done.

> Implementation-wise, the solution doesn't have to be "an agent". It could be a multi-agent system, orchestrator, a single prompt with MCP plugins, - whatever solves the problem. And we will compare the performance off these radically different approaches within the same setup.

Similar to the spirit of the previous ERC competitions, we will open source and share[22] as much as possible, including:

- source code of the simulation runtime
- source code of the task generator
- all submissions
- analysis results and reports

19. https://www.timetoact-group.at/landingpages/enterprise-rag-challenge
20. mailto:elon@x.com
21. mailto:sama@openai.com
22. https://github.com/trustbit/enterprise-rag-challenge/tree/main

Just like before, we will also hold a public test run before the main event, in order to give everybody a chance to practice and test their agents.

Similar to ERCr1 and ERCr2 we are planning to have multiple leaderboards, including the leaderboard for the local models. This time local models have a fair competing chance against even the best models.

Stay tuned for the updates!