Summary:

I explored reasoning token support across APIs. Gemini doesn't provide reasoning tokens; DeepSeek via OpenRouter does (e.g., deepseek-coder-r1). Reasoning tokens aren’t visible due to OpenRouter normalizing outputs—no <think> tags or EOR markers. Gemini and Anthropic models support max reasoning tokens via reasoning.max\_tokens.

OpenRouter has request limits—20/minute for free models, daily limits depending on credits.

Dynamic controllable reasoning is hard via API. There's no way to append a "wait" token mid-prompt and resume with reasoning traces—it restarts or loses state. So I concluded we need a **local model** for real-time reasoning control.

API’s that give reasoning tokens:

Gemini doesn’t give

Only Deepseek I guess

Q: Will we give the reasoning tokens to verification model as well? Then we would need them, but if we don’t we can just count the reasoning tokens or the api can give it to us for any model I guess.

**Max Tokens for Reasoning**

**Supported models**

Currently supported by:

* Gemini thinking models
* Anthropic models (by using the **reasoning.max\_tokens** parameter)

Openrouter limits

1. Free usage limits: If you’re using a free model variant (with an ID ending in **:free**), you can make up to 20 requests per minute. The following per-day limits apply:

* If you have purchased less than 10 credits, you’re limited to 50 **:free** model requests per day.
* If you purchase at least 10 credits, your daily limit is increased to 1000 **:free** model requests per day.

Gemini also has a free api

And anythingllm

I also have codestral free 1 req per second

I am using openrouter, it has free tier deepseek r1 with reasoning traces.

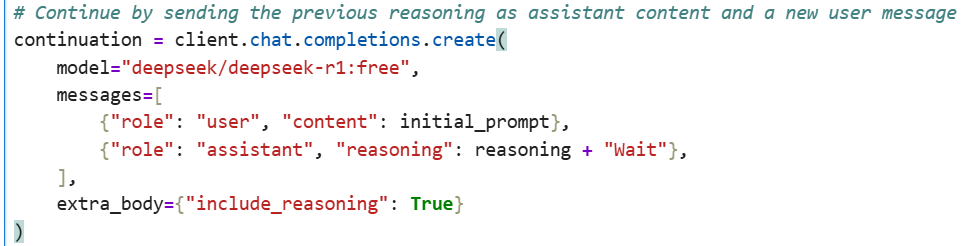
**🚫 Why you don’t see end of reasoning token:**

* OpenRouter’s API normalizes the output, so the <think> tags and internal markers are removed.
* The reasoning field only contains the **raw content** without explicit EOR markers.

It will be harder with a api to implement dynamical controllable reasoning.

Because I guess there is no way to append a wait token to a prompt via api and make it continue with that token.

You need to create a new prompt but then it doesn’t return a reasoning trace, it just answers on content.



This doesn’t work for some reason, it rethinks again as you have never given it. Do you know a way?

I gave up, I need a local model to implement dynamical adjusting reasoning.

Tasks:

1) Given a model that allows to dynamically change the amount of test-time compute (e.g. s1), measure how much compute is needed to solve each example in a given reasoning dataset.

2) For each example in the dataset, generate correct and wrong solutions. Then measure how much inference-time compute is needed to correctly identify whether a solution is correct or not.

3) Can we identify a pattern of how much compute is needed to verify a task of a certain solution (as measured by inference-time compute needed to solve it)?

Q) For task 1, Should I dynamically increase the reasoning length until I find a correct solution to find the minimum reasoning size, or just a solution is enough? If latter I can use and reasoning model, if not S1 is needed. But It’s a big model, the only way to reach it is via hugginface api, which does not return reasoning trace. Also its not possible to append wait token to a prompt. I think I should use bender right? Or I can just use the 1.5B L1 model which I think I can run locally. I can also use the false answer for the second task right?

Q) I am not really sure if there is a systematic way of finding wrong solutions. Increase temperature until false? Shorten the reasoning until false? Which one would you prefer?

Q) for task 3, in the graph, we have reasoning token size of the verification in y direction, and reasoning size of the answer in x right? Do we do this for the right answers or wrong answers? Or both?

Here is the jupiter notebook