Anthropic Subscriptions Offer 5x+ More Value Than OpenAI

Limit testing every AI subscription plan from Anthropic, OpenAI, Meta, SpaceXAI, MiniMax, Moonshot, Z.ai, Cursor, and Cognition

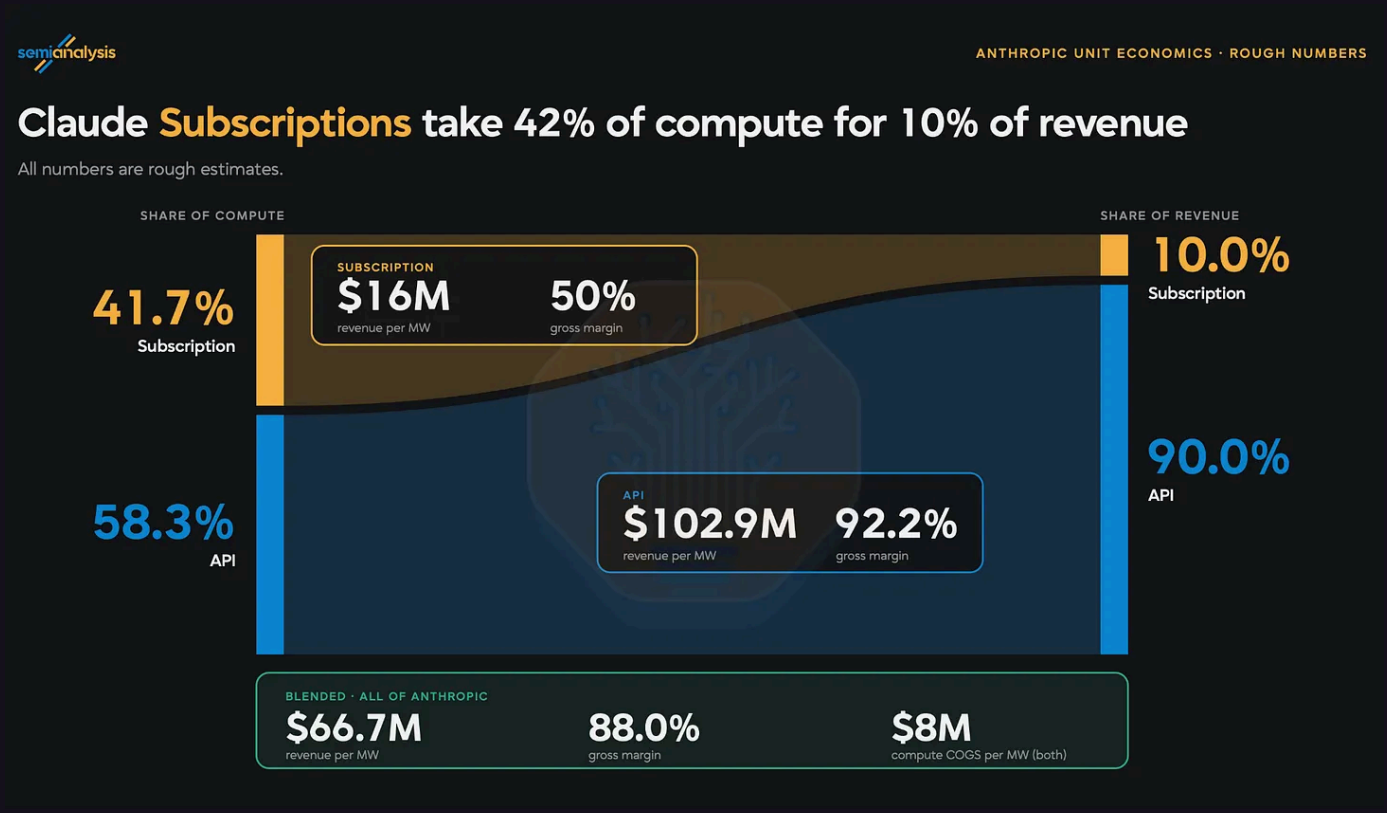
ANDREW MEGALAA, MAX KAN, AND DYLAN PATEL

13 HOURS AGO • PAID

123 11 Share

Subscription plans are still the primary way consumers and small businesses pay for AI. These plans are highly subsidized—as we previously explained in June—but can still make economic sense as powerful customer acquisition and marketing tools. For example, the goodwill engendered by OpenAI's generous resets is partially responsible for the recent surge in Codex adoption and has forced Anthropic to repeatedly walk back planned subscription nerfs to avoid getting clobbered in the court of public opinion.

Furthermore, because subscription plans are so heavily subsidized, they can also have a large impact on margins and revenue per MW despite only being a small portion of total revenue. Consider the following rough numbers for Anthropic.

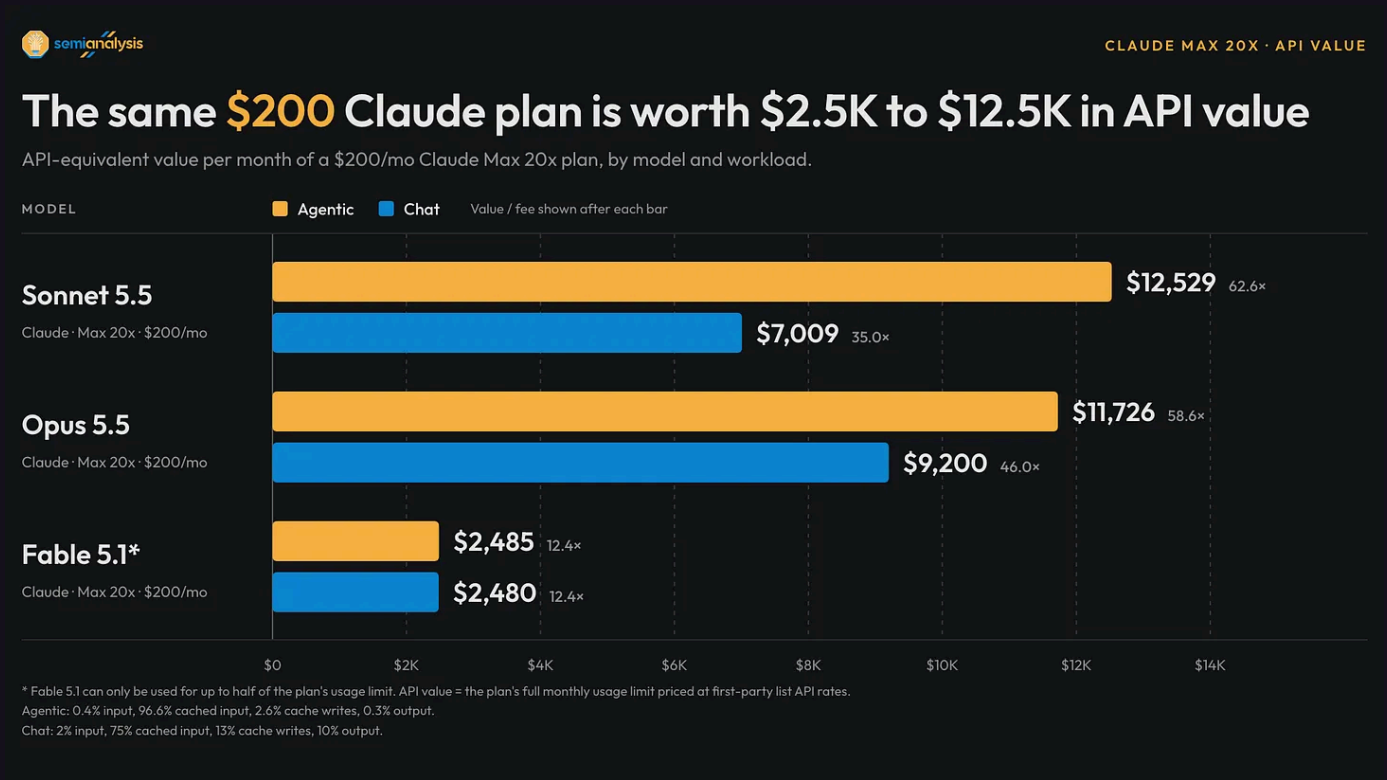


Despite being just 10% of overall revenue, subscriptions can take up over 40% of **inference compute and lower blended revenue per MW by ~$36M.** Subscriptions are even more important for OpenAI, as they make up a larger portion of their total revenue. For specific numbers, see our Tokenomics Model.

In other words, **if you want to accurately model AI lab financials, you need to understand their subscription limits.**

So how do limits actually work? The way to think about subscriptions is that your monthly payment grants you some numbers of "credits". Each (model, token type) combo consumes a different amount of credits. **Because credit cost ratios can differ dramatically from API price ratios, the "value" of the same plan changes depending on what model and workload you're running.** Put differently, **it doesn't make sense to say a plan is worth $X in isolation—you need to consider the full (plan, model, workload) tuple.**

Here's how the API-equivalent value of the same $200/month Claude plan changes depending on the model and workload.



Source: SemiAnalysis Tokenomics Model

A single static snapshot isn't enough either. Labs publicly change their limits all the time with promos and new model releases. They can also silently change limits whenever they

This is exactly what SemiAnalysis has done with our new Subscriptions Dashboard available exclusively to Tokenomics Model subscribers. Besides every OpenAI and Anthropic subscription, we also track Meta, SpaceXAI, Cursor, Cognition, Z.ai, MiniMax, and Moonshot.



A screenshot of our dashboard showing a small subset of the available data.
Source: SemiAnalysis Tokenomics Model

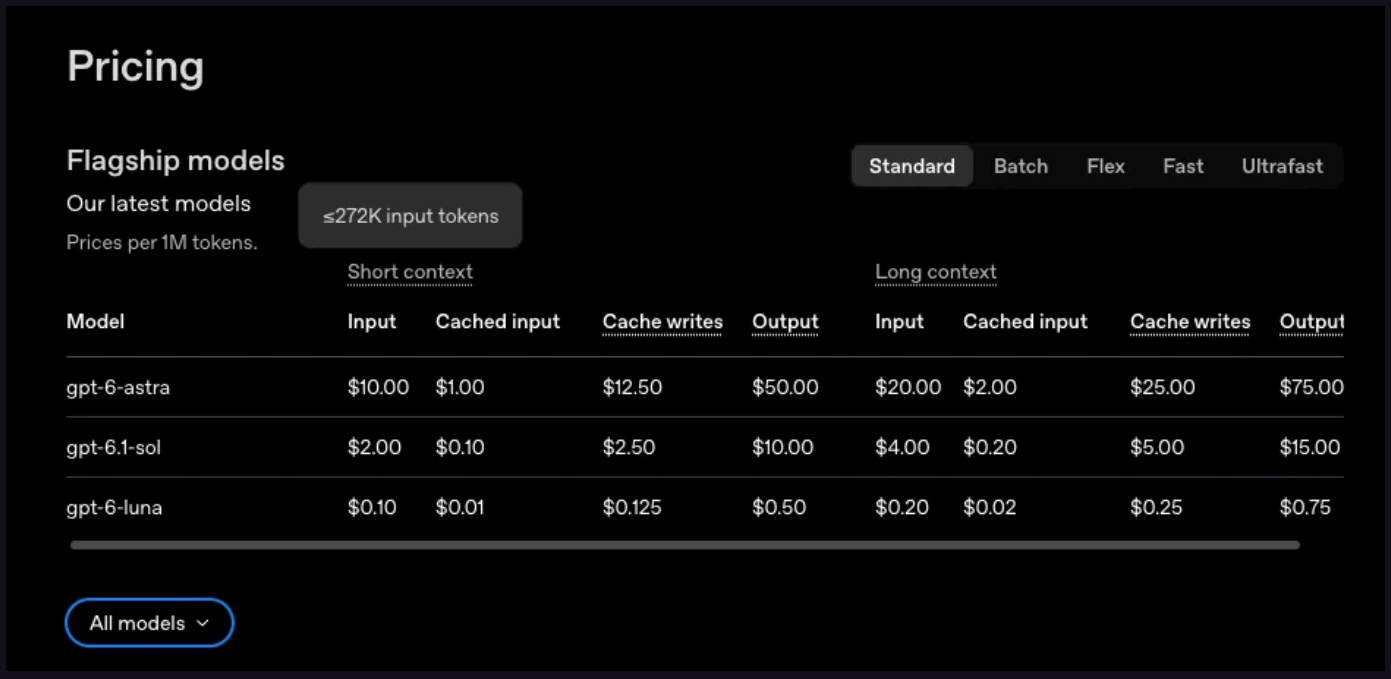
New providers, plans, and models will be added as soon as they're released. The rest of the article will give an overview of everyone's current limits. All token and dollar amounts below assume agentic usage unless otherwise specified.

Methodology

Tokens are generally priced per MTok (million tokens) across the following types:

- Input: Fresh tokens added to the LLM's context window that aren't cached
- Cache write: Input tokens that are cached for multi-turn conversations. Generally slightly more expensive than regular input tokens.
- Cache read: Tokens from previous turns that are already cached in the conversation. Generally extremely cheap compared to uncached tokens.

models typically compact before that expensive context window is reached, so we also keep our measurements below it.



Pricing

Flagship models

Our latest models

Prices per 1M tokens.

Standard | Batch | Flex | Fast | Ultrafast

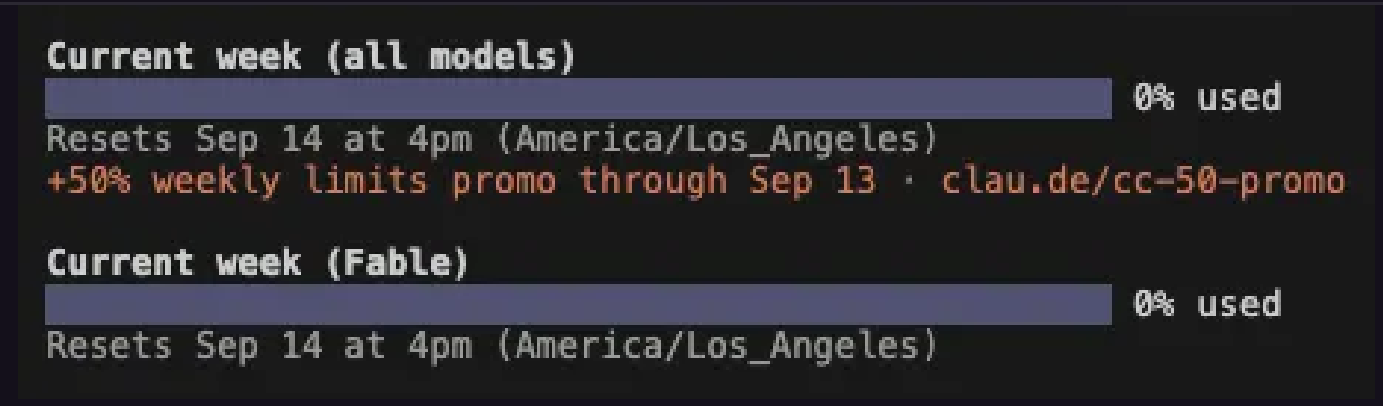
≤272K input tokens

Model	Short context Input	Short context Cached input	Short context Cache writes	Short context Output	Long context Input	Long context Cached input	Long context Cache writes	Long context Output
gpt-6-astra	$10.00	$1.00	$12.50	$50.00	$20.00	$2.00	$25.00	$75.00
gpt-6.1-sol	$2.00	$0.10	$2.50	$10.00	$4.00	$0.20	$5.00	$15.00
gpt-6-luna	$0.10	$0.01	$0.125	$0.50	$0.20	$0.02	$0.25	$0.75

All models

Source: OpenAI

Subscription plans don't expose this fine grained pricing. Instead, they provide a simple 0 to 100% usage meter across 5-hour and 7-day windows, and sometimes a separate meter for a model like Fable. Subscription tiers are then differentiated by usage multipliers relative only to the provider's other plans. OpenAI for example used to advertise "Expanded Codex usage" in their $20 Plus plan, "5x more usage than Plus" in the $100 Pro plan, and "20x more usage" in the $200 Pro plan until they cut their $200 plan usage in half and removed all relative usage from their pricing page.



Screenshot of Claude Code meters. Source: Claude Code CLI

Setup

We compute the subscription-rate of each (plan, model, token type) triple by running experiments that isolate one token type at a time and watching how far the model provider's meter moves. An experiment is some number of repeated calls using a specific prompt that maximizes one token type while minimizing all the others.

Source: SemiAnalysis Tokenomics Model

Input, cache writes, and cache reads share the same prompt template. We use a portion of *War and Peace* since some models will refuse to respond if given large blocks of gibberish. For the input token experiments, we use a random tag on every call to ensure nothing gets cached. The cache write experiments run the same way but with the prompt marked for

For the output token experiments, we used a technical essay to force long outputs because models will refuse mechanical prompts like “repeat SemiAnalysis 100,000 times”.

Computing the Results

For every call we record two things: how many tokens of each type the provider billed, and what the usage meter read. Turning those into a price takes three steps.

1. Measure the rate of each token type

Providers generally report usage on a meter that moves in fixed amounts. This could be a whole percentage point, a credit, a cent etc... A single request often does not move the meter at all, so the cost of one request cannot be measured directly.

Instead, we measure in steps. As requests run, we keep a running total of the tokens used. Each time the meter rises, we store that total. The tokens used between two meter moves make up one step, which is the cost of moving the meter by one unit.

We drop two partial steps. When a run starts, the meter is already part of the way to its next move, so the tokens before the first move are less than a full step and would make the rate look higher than it is. The tokens after the last move never finish a step, so we drop those as well.

To get a rate, we add up how far the meter moved over the complete steps and divide by the tokens used in them. Providers charge differently for input, output, and cached tokens, so we calculate a separate rate for each.

Source: SemiAnalysis Tokenomics Model

Caveats:

- Because we see the meter only once per request, a single step can be off by up to one request. These errors don't add up with more consecutive steps, so we track a range instead of a single number, and the range gets narrower the more steps we count. We keep adding steps until the range is within ±5%, then report the rate across all of them.
- No request contains only one type of token. Some plans, for example, require a set of instructions on every request. We subtract that extra amount using the prices we measured for the other types.
- Reading from the cache costs almost nothing on some plans and may not move the meter at all. If the meter does not move after 500M cache read tokens, we assume that cache reads are free.

2. Convert rates into tokens per window and per month

We read every meter a plan shows after every request. This could be a 5-hour limit, a weekly limit, and for models like Fable, its own weekly limit. So each token type gets a price on each meter. This is how much of that limit a million tokens uses. From each price we work out how many tokens fit e.g. if a million tokens use 5% of a limit, the whole limit holds 20 million.

Source: SemiAnalysis Tokenomics Model

The 5-hour limit resets many times within a week, so a month is capped by the weekly limits.

3. Pricing

Lastly, we compute the API-equivalent value by converting the limit % consumed by a million tokens of each type into a dollar amount. For example, if you consume 1% of a $200 plan's monthly limit, that's $2.

Then, we assume a workload shape, and compute how many total tokens you can get from each plan given this ratio. For the agentic workload, we use our own usage ratios in September as reported in our Tokenomics Model.

Finally, we multiply by the blended price per MTok at API prices to get the API-equivalent value.

Source: SemiAnalysis Tokenomics Model

Catching a Provider A/B Test

While refining our methodology, we ran into a very confusing situation where just 1 of 3 of the same subscription we tested for a particular provider had ~20% lower limits than the other 2. This unlucky account happened to be significantly older as well, and we were worried that the provider had some cursed setup where they changed subscription limits depending on the age of the account.

Fortunately, after reaching out to the provider, we were able to confirm this was not the case and that our unlucky account was part of an "extremely tiny" A/B test they were running on limits. Additionally, they wanted to emphasize that they "didn't just decrease limits wholesale" and were instead testing "how to better balance when people hit limits".

This conclusion is interesting for two reasons:

1. It proves providers can silently change subscription limits at any time
2. Our method is sensitive enough to detect these subtle changes!

OpenAI vs Anthropic

The biggest question in AI subscription land is how Anthropic's limits compare to OpenAI's. Note that OpenAI just made some massive changes to their subscriptions last week, halving the API-equivalent value for the $200 plan and introducing a new $500 tier.

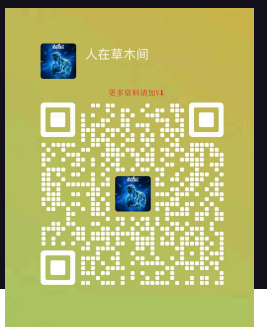
Source: SemiAnalysis Tokenomics Model

Limits are quite similar across the board for GPT-6 Astra vs Fable 5.1, though remember that Fable can only be used for 50% of your limit. In other words, your $200 ANT plan would still have 50% left after consuming $2,485 worth of Fable 5.1, whereas the equivalent OAI plan would be fully exhausted after $2,897 of Astra.

This extra 50% starts becoming very relevant when we compare Opus 5.5 vs GPT 6.1 Sol.

Source: SemiAnalysis Tokenomics Model

At this mid-tier of models—which both companies market as the intended daily driver for most users—Anthropic is an overwhelmingly better deal, offering ~5x the API-equivalent value across the board. You could argue that this is unfair for OAI because 6.1 Sol is much cheaper per token than Opus 5.5, but the gap is still massive even if you switch to comparing the number of tokens.



Overall, we believe API-equivalent value is the single best number for capturing the "value of a subscription plan, but there are times where raw token volumes can be more appropriate. For example, if a model is a particularly bad (or good) deal at API prices, then the API-equivalent value will of course be misleading.

Token efficiency is also an extremely relevant factor, but the industry unfortunately lacks reliable data here. Many people like to cite this chart from Artificial Analysis, but we do not believe the benchmark tasks in the AA Intelligence Index are at all representative of real work people do with LLMs.

OpenAI's new $500 plan and massive limit reduction

Our tracking confirmed the 50% value reduction for OpenAI's $200 plan announced by Tibo. $200 plans purchased before the cut will keep the old higher limits until October 29th, but newly purchased plans immediately start with the lower limits.

Source: SemiAnalysis Tokenomics Model

As you can see, OpenAI specifically halved the number of tokens you get for each model tier. This caused the API-equivalent value for Sol class models to drop by over 50% because OAI also cut cached input token pricing for 6.1 Sol.

50% price cut.

Source: SemiAnalysis Tokenomics Model

Of course, 300 TPS Ultrafast was the real headline feature for the $500 plan and we are currently testing its limits. Results will be published to Tokenomics Model subscribers as soon as they're available. We are also actively testing the limits when using your ChatGPT subscription on third party apps like Devin.

The $200/month plan also used to be significantly more subsidized than OpenAI's other subscriptions. If we normalize each (plan, model) combo to the API-equivalent value and tokens per dollar, we can see that Pro 100 offered ~2x more per-dollar value than Plus on Astra and Pro 200 was another ~2x on top of Pro 100 across all models.

Source: SemiAnalysis Tokenomics Model

With last week's cut, Pro 100, 200, and 500 all offer the same tokens per dollar for every model. Plus is also comparable for Sol but offers a relatively worse deal on Astra.

Anthropic, on the other hand, already offered the same per-dollar value for all of their subscription tiers and currently crushes OpenAI.

Source: SemiAnalysis Tokenomics Model

subscription offers far better value than anything from OpenAI.

The only counter argument for OpenAI is that none of their Pro plans have a 5 hour limit, which makes it easier to consume a higher % of your total monthly limit. However, we don't think this offsets the ~4x higher API-equivalent value offered by Opus 5.5 on Claude plans.

Do labs adjust subscription limits based on API pricing changes?

We've seen a number of significant price cuts recently to Anthropic and OpenAI models.

- Fable 5.1 cut cache reads by 75% vs Fable 5
- Opus 5.5 cut input and output token pricing by 20% vs Opus 5 and cache reads by 60%.
- GPT 6.1 Sol cut cache reads by 50% vs GPT 6 sol. This was after making GPT 6 Sol 60-67% cheaper vs 5.6 Sol.

One important question is whether the labs adjust token limits after these price cuts, such that the API-equivalent value of your subscription stays constant.

We already showed earlier that OpenAI didn't change Sol token limits on the $200/month plan when they released 6.1 Sol. This caused the API-equivalent value to drop ~30%.

Anthropic also didn't increase Fable token limits at all with the release of 5.1, but they did up Opus token limits with the 5.5 price cut.

Source: SemiAnalysis Tokenomics Model

However, the ~20% token increase for Max plans and ~50% increase for the Pro plan weren't enough to fully offset the API price cut.

Two different approaches to increasing subscription margins

As we explained at the start of this article, subscription gross margins are way lower than API and meaningfully reduce revenue per MW for both OpenAI and Anthropic. Both companies are well aware of this fact but have chosen two different strategies for reducing subscriber subsidies.

For a given plan, Anthropic decreases the API-equivalent value for more premium models. The drop off is small between Sonnet 5.5 and Opus 5.5, but becomes very meaningful with Fable 5.1.

Source: SemiAnalysis Tokenomics Model

Reducing the API-equivalent value as you release more powerful models at higher list prices is the more subtle way to stop subsidizing subscriptions. Assuming 100% utilization and 92% API gross margins, maxing out Opus 5.5 vs Fable 5.1 usage corresponds to -369% and 1% gross margins respectively. If we were to use a more realistic average utilization of 20%, then gross margins rise to 6% and 80% respectively.

In other words, **Anthropic would likely already have software like margins on their subscription plans if everyone only used Fable**. Opus class models will continue to become cheaper to serve over time as Anthropic figures out how to make the model smaller, and new model tiers above Fable will have even lower subscription limits (assuming they're included at all).

OpenAI, on the other hand, picked the nuclear option of just immediately cutting to Fable-level limits across the board.

Source: SemiAnalysis Tokenomics Model

The downside of this approach is supposed to be public backlash, but OpenAI has somehow managed to more or less avoid this entirely. It's likely some combination of all the good DevDay PR drowning out the negative news from the day prior and grandfathering existing plans for an additional month.

Chinese subscription plans are still a good deal

Despite being compute poor, the Chinese labs still offer subsidized subscription plans, but the level of subsidization varies dramatically between models.

Source: SemiAnalysis Tokenomics Model

Here's how they compare to OpenAI and Anthropic after normalizing per dollar.

Source: SemiAnalysis Tokenomics Model

Value per dollar increases as you buy higher tier subscriptions from the Chinese labs. On average, the API-equivalent value per dollar is a little less than the ~12x you get from

Third party plans are worse than first party

The final comparison we'll consider is OpenAI and Anthropic models on their own first party plans vs third party wrappers like Cursor and Cognition (Devin).

This post is for paid subscribers

Subscribe

Already a paid subscriber? Sign in

Previous

© 2026 Dylan Patel · Privacy · Terms · Collection notice

Start your Substack

Get the app

Substack is the home for great culture