

OpenClaw Token Calibration

The Real-World 5-Step Method

ScaleUP Media | SPRINT Program

What You Are Going to Do

This is how you train your OpenClaw agent to accurately predict the cost of a task before it runs. The method is simple: ask it to guess, check the real number against your Anthropic dashboard, show it the difference. Repeat that a few times until the guesses are tight.

No complex setup. No log files. No automation required to get started. Just a conversation loop between you, your agent, and your Anthropic usage dashboard.

This works because your agent learns from correction. Every time you show it the real cost versus what it guessed, it adjusts its internal model. By round 3 or 4, it will be estimating within 10-15% on similar tasks.

1 Tell Your Agent to Estimate Before It Starts

Before you give your agent a task, add this instruction to your system prompt or paste it at the start of your session. This flips your agent into estimation mode. It will now guess the token cost of every task before it touches it.

PASTE THIS INTO YOUR SYSTEM PROMPT OR SESSION START

Before you begin any task I give you, first output a cost estimate. Do not start the task until you have written the estimate.

Format it exactly like this:

```
---  
PRE-TASK ESTIMATE  
Task: [brief description of what you are about to do]  
Estimated input tokens: [your guess]  
Estimated output tokens: [your guess]  
Estimated total tokens: [your guess]  
Estimated cost: $[your guess]  
Confidence: [Low / Medium / High]  
---
```

After the estimate, proceed with the task normally.
When the task is complete, output your best guess at actual tokens used in the same format, labeled POST-TASK ESTIMATE.

You only need to paste this once per session, or add it to your persistent system prompt in OpenClaw. After that, every task gets an estimate automatically.

2 Run the Task — Agent Reports What It Thinks It Used

Now run your task like normal. Your agent will do three things in sequence:

1. **Output the PRE-TASK ESTIMATE** before it starts.
2. **Complete the task.**
3. **Output a POST-TASK ESTIMATE** — its best guess at actual tokens consumed.

The post-task estimate is the agent's internal count. It is not the real number — it is an approximation based on what the model thinks it generated. That is fine. That is the point. You are going to compare it to the real number in the next step.

Do not correct the agent mid-task. Let it run the full cycle and give you both estimates. The gap between pre and post is your first data point.

3 Pull the Real Cost from Your Anthropic Dashboard

Go to your Anthropic API usage dashboard and find the actual token usage for the task you just ran. You can do this two ways:

Option A — Screenshot

Take a screenshot of the usage row that matches your task timestamp. Make sure it shows input tokens, output tokens, and total cost. You will paste or share this directly back to your agent in Step 4.

Option B — Export the CSV

Export your usage log as a CSV and find the row that corresponds to your task. Copy the three numbers you need: input tokens, output tokens, and cost.

Match by timestamp. Run your task, note the time, then filter your dashboard to that window. Each agent call shows as a separate API call — grab the one that lines up with when your task ran.

4

Post the Real Numbers Back to Your Agent

This is the calibration step. Paste your actual numbers back into the conversation using the prompt below. Your agent will compare its estimates to reality, identify where it was off, and explain how it will adjust going forward.

PASTE THIS AFTER YOU HAVE THE REAL NUMBERS FROM YOUR DASHBOARD

Here are the actual token numbers from my Anthropic dashboard for the task you just completed:

Actual input tokens: [paste number]
Actual output tokens: [paste number]
Actual total tokens: [paste number]
Actual cost: \$[paste number]

Compare these to your PRE-TASK ESTIMATE and POST-TASK ESTIMATE.

Tell me:

1. How far off was your pre-task estimate? (% variance)
2. How far off was your post-task estimate? (% variance)
3. What caused the gap? (Did you undercount input context?
Overestimate output length? Miss something else?)
4. How will you adjust your estimates for similar tasks
going forward?

Update your internal calibration and confirm you have done so.

Your agent will walk through the math, identify the error pattern, and update its estimation model. This is the learning moment. The more specific the real numbers, the faster it calibrates.

If you have a screenshot from your dashboard, paste the image directly into the conversation. Your agent can read the numbers from it — no need to type them out.

5

Repeat Until Estimates Match — Usually 3 to 4 Rounds

Run another task. Get the estimate. Check the dashboard. Post the actuals. Do this loop 3 to 4 times. By the end, your agent's estimates will be consistently within 10-15% of real usage on similar tasks.

Round	Pre-Task Estimate	What You Do	What Happens
Round 1	Way off — agent guessing cold	Post actuals	Agent identifies the error pattern
Round 2	Closer — off by 20-30%	Post actuals	Agent tightens its model
Round 3	Getting tight — within 15%	Post actuals	Agent fine-tunes edge cases
Round 4+	Consistently within 10%	Spot check only	Calibration complete

If an estimate is still off after Round 3, ask your agent directly: 'What signals are you using to estimate this task type?' Then correct its assumptions explicitly and re-run.

When the Task Type Changes

Your agent calibrates per task type. A well-calibrated estimate for code generation will not automatically carry over to data extraction or email writing. When you switch to a meaningfully different task type, run the calibration loop again. Usually 2 rounds is enough once the agent already understands the concept.

The Full Loop at a Glance

1 Add the estimation instruction to your system prompt or session start.

2 Run your task. Agent outputs PRE and POST estimates automatically.

3 Go to your Anthropic dashboard. Pull the real token numbers.

4 Paste the real numbers back. Agent explains the gap and adjusts.

5 Repeat 3-4 times. Estimates get tight. Calibration done.

Once calibrated, your agent will pre-flight estimate every task automatically. Use those estimates to route cheap tasks to Haiku, medium tasks to Sonnet, and only send the heavy stuff to Opus. That is where the 60-80% cost reduction comes from.

ScaleUP Media | SPRINT Program

Bring questions to the live session or drop them in the community.