AI BENCH LAB

Cost Estimator

Understand API costs and electricity costs before you run.

Version 1.0 | March 2026

aibenchlab.com

Guides · Professional AI Model Benchmarking for Windows

Contents

Understand API costs and electricity costs before you run. Generate professional cost analysis reports.

1. Overview

The Cost Estimator calculates projected costs for a benchmark before you run it. For cloud models, it estimates API token costs. For local models, it estimates electricity costs. Both estimates are available in the Benchmark Wizard (Step 7 in Advanced mode) and via the CLI and REST API.

2. Cloud Model Cost Estimation

How It Works

› **Token estimation**: The estimator calculates expected input and output tokens for the selected test suite. It uses historical data from previous runs (if available) or domain-specific defaults.

› **Pricing lookup**: Per-token pricing is resolved for the selected model and provider.

› **Cost calculation**: Input tokens and output tokens are priced separately, then summed.

Token Estimation Sources

Source	Confidence	When Used
Historical data	High	3+ previous runs with the same model family and test domain
Domain defaults	Medium	No historical data; uses conservative per-domain estimates
Fallback	Low	No data available; uses baseline estimates

A 15% output buffer is applied to all output token estimates to account for response length variability.

Pricing Resolution

Pricing is resolved in priority order:

› **Fresh cache** — cached pricing data that hasn't expired (default TTL: 24 hours)

› **Similar model match** — automatic matching for model variants (e.g., `gpt-4o` matches `gpt-4o-2024-08-06`)

› **Stale cache** — expired cache data used as fallback when the network is unavailable

› **Hardcoded fallback** — built-in pricing table for major providers

Reference Pricing (Built-In Fallback)

Provider	Model	Input Rate	Output Rate
OpenAI	gpt-4o	$2.50/M tokens	$10.00/M tokens
OpenAI	gpt-4o-mini	$0.15/M tokens	$0.60/M tokens
OpenAI	o1	$15.00/M tokens	$60.00/M tokens
OpenAI	o1-mini	$3.00/M tokens	$12.00/M tokens
Anthropic	claude-3-5-sonnet	$3.00/M tokens	$15.00/M tokens
Anthropic	claude-3-5-haiku	$0.80/M tokens	$4.00/M tokens
Google	gemini-2.0-flash	$0.10/M tokens	$0.40/M tokens
Google	gemini-1.5-pro	$1.25/M tokens	$5.00/M tokens
Groq	llama-3.3-70b	$0.59/M tokens	$0.79/M tokens
xAI	grok-2	$2.00/M tokens	$10.00/M tokens
Together AI	llama-3.1-405b	$3.50/M tokens	$3.50/M tokens

These are fallback values. The estimator prefers fresh pricing data when available.

Cost Breakdown

The estimate includes:

- **Input token count** and cost
- **Output token count** and cost (with buffer)
- **Total API cost**
- **Pricing source** (cache, fallback, or stale)
- **Confidence level** (High, Medium, or Low)

---

3. Local Model Cost Estimation

For models running on local hardware, the estimator calculates electricity costs.

Inputs

Parameter	Source	Default
GPU TDP (watts)	Auto-detected via `nvidia-smi`	Varies by GPU
GPU count	Auto-detected	1
System overhead (watts)	User setting	150W
Electricity rate ($/kWh)	User setting	$0.42/kWh
Tokens per second	Historical data or default	10 TPS

Calculation

```
GPU Power     = GPU TDP x 0.80 (load factor) x GPU Count
Total Power   = GPU Power + System Overhead
Duration (hr) = Total Tokens / Tokens Per Second / 3600
Energy (kWh)  = (Total Power / 1000) x Duration
Cost          = Energy x Electricity Rate
```

Example: RTX 4090 (350W TDP), 100K total tokens at 50 TPS, $0.16/kWh electricity:

- GPU Power: 350W x 0.80 = 280W
- Total Power: 280W + 150W = 430W
- Duration: 100,000 / 50 / 3600 = 0.556 hours
- Energy: 0.43 kW x 0.556 hr = 0.239 kWh
- **Cost: $0.04**

Configuring Electricity Settings

Go to **Preferences > Cost Settings** to set:

- **Electricity rate** ($/kWh) — check your utility bill or use your region's average
- **System overhead** (watts) — total non-GPU power draw (CPU, RAM, fans, etc.)

---

4. Operational Projections

The estimator projects costs beyond a single benchmark run, estimating what it would cost to operate the model in production.

Cloud Scenarios

Scenario	Requests/Day	Typical Use Case
Light	100	Individual developer, internal tool
Moderate	1,000	Small application, team use
Production	10,000	Customer-facing application
Heavy	100,000	Platform-scale deployment

Projections are shown as daily, monthly, and annual cost estimates based on the per-request cost from the benchmark estimate.

Local Scenarios

Scenario	Hours/Day	Typical Use Case
Part-Time	4	Development machine
Business Hours	8	Standard workday
Extended	12	Extended operations
Always-On	24	Production deployment

---

5. Confidence Levels

Every estimate includes a confidence indicator:

Level	Meaning
High	Historical token data available + fresh pricing cache
Medium	Using domain defaults or stale pricing data
Low	No data available; using hardcoded fallbacks only

The final confidence is the minimum of the token estimation confidence and the pricing confidence.

---

6. Cost Warning Threshold

When the estimated cost for a cloud benchmark exceeds a configurable threshold, a confirmation dialog appears before execution begins.

- **Default threshold**: $5.00

- **Configurable in**: Preferences > Cost Settings
- **Range**: $0 – $1,000

The dialog shows the estimated cost, test count, and model count. You can confirm, cancel, or update the threshold directly from the dialog.

---

7. Cost Reports

Generate a standalone cost analysis report for any benchmark session.

Report Contents

- **Summary**: Total cost, model, suite, test count, confidence level
- **Token breakdown**: Input/output token counts with pricing source
- **Estimate vs. actuals** (if the benchmark has run): Variance percentage between projected and actual costs
- **Hardware summary**: GPU model, power draw, system configuration
- **Operational projections**: Daily/monthly/annual cost scenarios
- **Assumptions**: All inputs and defaults used in the calculation

Report Options

Option	Default	Description
Cover page	On	Branded cover with session metadata
Configuration details	On	Full hardware and provider configuration
Projections	On	Operational cost scenarios
Assumptions	On	All estimation inputs and defaults
White label	Off	Remove AiBenchLab branding

Generating a Report

From the GUI: After a benchmark completes, click **Cost Report** in the results view.

From the CLI:

BASH

```
aibenchlab-cli run --model openai:gpt-4o --cost_report ./cost-report.json
```

From the REST API:

BASH

```
curl -X POST http://localhost:8787/estimate-cost \
  -H "x-aibl-token: YOUR_TOKEN" \
  -H "Content-Type: application/json" \
  -d '{"model_name": "gpt-4o", "provider_name": "openai", "provider_type": "cloud"}'
```

---

8. Tips

• **Run a small benchmark first.** Use the Quick Compare suite (free tier) with a cloud model to calibrate token estimates. Future estimates for the same model family will use this data and show "High" confidence.

• **Compare cloud vs. local costs.** The estimator handles both — run the same suite against a cloud model and a local model to see the cost trade-off.

• **Adjust electricity rate.** The default ($0.42/kWh) is conservative. Check your actual utility rate for more accurate local cost estimates.

• **Watch the confidence level.** A "Low" confidence estimate may be off by 2–3x. Run a calibration benchmark to build historical data.

Resources

• Documentation: **aibenchlab.com/docs**
• Support: **aibenchlab.com/contact**
• YouTube walkthroughs: **youtube.com/@aibenchlab**
• Email: **support@aibenchlab.com**

AiBenchLab is built by a solo founder with 40+ years of software engineering experience. Every feature is designed with the conviction that you deserve honest, transparent, and reproducible AI model evaluations. No hype. Just truth.