

Ask drawer (over Results)

ask-surface.html · 1200×740

ChatMaestro

v2026.08

 PN priya.n

Results > Terse-prompt · Pilot-20 · run of Jul 16

Results — Composite & Cost

73.9
Composite mean

\$2.52
Run cost

Configuration	Composite	Groundedness	Cost
GPT-5	71.2		
Terse baseline	66.9		

Ask
This run · Pilot-20

Read-only · your permissions

PN How did GPT-5 do versus the terse baseline, and show the score trend?

GPT-5 averaged **71.2** composite across 20 enrollees — **+4.3** over the terse baseline (66.9). The gain is almost all in **groundedness** (+7.1); coherence is flat. Cost was **\$2.51**.

By configuration

Table Chart

Configuration	Composite	Groundedness	Cost
GPT-5	71.2	78.4	\$2.51
Terse baseline	66.9	71.3	\$1.98

Composite by self-improve pass

Table Chart

Read-only · checked

Hide query

Save prompt

```

select cfg, avg(score_composite) c,
       sum(token_cost) cost from run_result
where run_id = %(run)s group by cfg;

```

Ask about this run...

Ask

Open full Ask

Annotations

- Ask this question about the results.
- One engine, opened with context** The **same** Ask drawer serves every entry point (Results, Experiments, Cohorts, Models, Documents, global) — only the `scope_key` differs. Here `scope = results` with this run pre-bound. Context is a default, not a cage — “compare to other runs” still works.
- Bounded by your access** Answers are **read-only** and the generated SQL always runs under *your* permissions (RLS) — a shared saved question never widens what you can see.
- Narrative + result cards** An answer is a **narrative** plus zero or more result cards, each switchable between **table** and **chart**.
- Composite is one overall score that blends the separate quality scores into a single number.
- Groundedness scores how well an answer sticks to the source facts instead of making things up.
- Each pass is one round where the model reviews its own answer and tries to improve it.
- The query is inspectable** The generated SQL is shown below and can be hidden — every answer is traceable to the query that produced it.

-
- 9 **Save = a reusable button** Saving turns this question into a named button. In the save dialog you choose **per value** what becomes a fill-in parameter vs stays fixed; an admin can later approve it for everyone.
-
- 10 The database query that produced this answer — it only reads data and runs under **your** own permissions.
-
- 11 **How saved questions partition** The rail shows **this context + global**, filtered to what you may see — your drafts (private), shared, approved, and built-in, never a retired one — and always bounded by your RLS permissions.
-
- 12 **Parameterized question** A saved question with fill-in parameters opens a small dialog to collect them (category / dimension / scalar) before it runs.
-
- 13 Opens the full Ask across everything, not just this run.