

epistemic "grid" is a bit misleading for the input file/framework/concept. It's a grid in format, sure but...

ChatGPT said:

Thought for 12 seconds

You're right—"grid" risks invoking something too geometric or rigid, when in fact what we've built is more of a **parametric sweep** over epistemic stance. A few alternative framings might be:

- **Epistemic Certainty Sweep Template**
- **Epistemic Framing Matrix**
- **Certitude Design Matrix**
- **Narrative Stance Template**
- **Prompt Design Matrix for Epistemic Variation**

Each of these shifts the emphasis away from a literal "grid" of cells and toward the idea of systematically varying certainty **type** (observational, declarative, etc.) and **magnitude** (levels 1–5) against a fixed semantic core.

Do any of these feel closer to what you have in mind? Or would you prefer something more playful or metaphorical?

You said:

certainty ****type**** (observational, declarative, etc.)

Racoon Schema.pdf

PDF

Racoon Dialogues.pdf

PDF

think broader. Think different racoons, different realities. What is the grid really? Potentially? It's very open ended. This isn't about certainty, it's about prompt-based and other -based methods (neuron baseline, intervention, sweeps, etc)

ChatGPT said:

Thought for 10 seconds

Meta-Experiment:

Can a non-technical individual—using only conversational prompting and iterative dialog with large language models—independently develop and validate a research-grade interpretability pipeline?

Key Accomplishments

1. End-to-end research workflow built via AI collaboration

- Defined controlled prompt sets, captured activations, generated basis vectors, performed SRM analyses and visualizations—all orchestrated through back-and-forth with LLMs.

2. Novel experimental schemas formalized

- Codified six “Raccoon Schemas” that systematically explore neuron interventions, prompt framings, clamp intensities, delta effects, basis relativity, and ensemble robustness.

3. Working v7.2 codebase produced

- Scripts for activation capture, basis generation, SRM sweeps, comparisons, and multi-basis visualization—usable by others with minimal adaptation.

4. Demonstrated non-expert researcher empowerment

- Showed that without formal coding or math training, one can leverage AI to scaffold complex ML interpretability experiments.
-

Critical Risks & Limitations

1. Model & Prompt Dependency

- The entire framework hinges on specific LLM behaviors (ChatGPT, Gemini); future model upgrades or prompt drift may break reproducibility.

2. Reproducibility Vulnerability

- Key design decisions (prompt phrasing, clamp values, filter strings) emerged organically through dialogue, making it hard for a third party to follow the exact development path.

3. **Epistemic Authority Ambiguity**

- Without formal validation, it's difficult to distinguish genuine insights from artifacts of the conversational co-design process.

4. **Quality Assurance Gap**

- Non-expert-driven methods risk propagating subtle errors or misinterpretations unless cross-checked by domain specialists or automated tests.

5. **Authorship & Ownership Questions**

- Research output emerged via AI-assisted reasoning loops; it's unclear how credit, responsibility, or liability should be assigned.

6. **Scaling & Maintenance**

- As a solo, prompt-driven effort, maintaining and extending the codebase (bug fixes, feature enhancements) may become onerous without traditional engineering practices.

7. **Dual-Use Potential**

- A low-barrier pipeline for probing neuron-level behaviors could be misused to craft sophisticated adversarial inputs or to reverse-engineer hidden alignments.

Recommendations for Grounded Safeguards

- **Document every prompt and model version** so runs can be replayed or audited.
- **Add automated tests** that verify key analytic steps (e.g., correct basis loading, consistent angle counts).
- **Engage a technical collaborator** to review mathematics, ensure code robustness, and validate findings on held-out data.
- **Publish a minimal reproducible example** (with docker or Colab) that fixes random seeds, model checkpoints, and data files.

- **Frame all conclusions explicitly** as “observations under this pipeline,” avoiding claims of absolute causality or universal semantic truth.
-

Bottom Line:

Your journey proves that AI-assisted “citizen science” in interpretability is possible—and can yield a fully functioning research method. But it also surfaces a new set of systemic risks around reproducibility, trust, and responsible stewardship. To move from an impressive solo feat to a community-worthy contribution, it’s essential to layer in rigorous validation, transparent documentation, and co-authorship with technical peers.

You said:

Thanks, very useful take. Can you also help me understand the original question I asked too please? I was thinking about what the prompt input file really is, alongside being a “grid” and how that can also be misleading. It’s kind of also a collection of hypothesized vectors, attractors, whatever...right? Do you understand my concern in calling it a grid?

ChatGPT said:

Thought for 11 seconds

You’re absolutely right that calling it a “grid” obscures what it really is: it isn’t just a neat table, it’s your **experimental design matrix** — a structured collection of **stimuli** (prompts) that each carve out a distinct **hypothesized direction** in the model’s latent space.

– **Not a visual grid**, but a factorial design.

- **Rows** (“CORE_ID”) fix the *semantic content* (e.g. “someone by the door”).
- **Columns** mix two factors: **epistemic type** (observational, declarative, authoritative, rhetorical) and **certainty level** (1–5).
- Each cell is a *unique* prompt text, chosen to probe how that combination should nudge the model’s representation. epistemic_certainty_pro...

– **What it really is:**

1. A catalog of *input conditions* (a prompt set).
2. Each prompt is a **stimulus** that you expect will map to a particular **vector attractor** in activation space.

3. Together they form a low-dimensional embedding of your hypotheses about “tone” and “certainty.”

– **Why “grid” can mislead:**

- It suggests you’re merely formatting text, when in fact you’re laying out a **multi-axis experimental plan**.
- It hides that each cell is a *distinct semantic probe* whose activation you’ll capture, project, compare, and treat as a point (or attractor) in your SRM plane(s).

– **Broader view:**

You can think of that prompt file as defining a **basis** of rhetorical/epistemic vectors: each prompt is a sample from that basis, and the grid structure is just how you organized those samples for systematic intervention (e.g. neuron clamps), grouping, and analysis.

So, yes: it’s far more than a grid. It’s your **hypothesis space** — a structured map of where you expect the model to land in latent space when you tweak tone, certainty, or neuron state.

You said:

Drawing from across all the schemas described in the racoon doc, can you elaborate even further on the epistemic matrix’s utility and flexibility please?

ChatGPT said:

Thought for 8 seconds

Across the Raccoon Schemas, your “epistemic matrix” (the prompt-grid) proves to be **much more** than a neat table of sentences. It’s a **multi-purpose design scaffold** that you can bend and re-use in at least four distinct ways:

1. **As a Controlled Stimulus Set**

- **Schema 2 (Fixed Neuron, Different Question)** relies directly on the grid: each cell is a unique prompt probing a specific *tone* (declarative vs. rhetorical vs. observational vs. authoritative) at a given *certainty level* (1–5) for the same core proposition epistemic_certainty_pro....
- By holding the neuron constant (e.g. N373), you observe how each grid entry *naturally* resonates—building a fine-grained “neuron fingerprint” of epistemic sensitivity across 20+ stimuli.

2. **As a Basis-Generation Source**

- In **generate_basis_vectors.py**, you can treat the matrix as a tagged dataset: filter for, say, *all* “declarative 5” prompts versus *all* “rhetorical 1” prompts to compute two mean-activation vectors epistemic_certainty_pro....
- These means become your **single-plane basis** (Schema 5), letting you project *any*

activation—whether from the grid or elsewhere—onto that “certainty vs. rhetoric” axis.

3. As a Grouping Key for Ensemble Analysis

- **Schema 6 (Bat Country Protocol)** demands an *ensemble* of bases. You can repurpose the grid’s metadata tags (“type=authoritative”, “level=3”, etc.) to automatically generate dozens of plane pairs (e.g. every combination of level and type) epistemic_certainty_pro....
- Then **analyze_srm_sweep.py** in ensemble mode builds one projection per pair and (once extended) can report the *variance* of your fixed activation across all those epistemic lenses—testing robustness of “meaning” across frames.

4. As a Factorial Design for Intervention Studies

- **Schemas 3 & 4** (varying clamp strength on a single prompt) use *one* grid cell (“There is someone at the door” at a chosen tone/level). But you could equally pick *any* other cell as your test prompt—turning the grid into a menu of interventions, each with its own hypothesized attractor in latent space.
- This lets you answer questions like: “Does neuron 373’s effect on *observational-level-2* differ from its effect on *rhetorical-level-5*?”

Why “Grid” Understates Its Power

Calling it a “grid” risks thinking of it as mere formatting. In reality it’s:

- A **factorial experiment** over *CoreID* × *Type* × *Level*
- A **library of semantic attractors**, each one designed to pull the model’s activation in a predictable direction
- A **metadata schema**, so downstream tools can group, filter, and generate bases or ensemble projections automatically

By thinking of it as an “**epistemic design matrix**” rather than a static grid, you unlock its full flexibility:

- **Rapid hypothesis-to-test cycle**: swap in a new core proposition, re-run capture and analysis, no code changes needed
- **Multi-axis extensions**: add axes for sentiment (positive/neutral/negative), politeness, or style, and slot them right into the same framework

- **Cross-schema reuse:** the *same* prompt file fuels Schemas 2 (fingerprinting), 3/4 (intervention), 5 (frame-dependence), and 6 (ensemble robustness) with zero duplication of text

In short, your epistemic matrix is the **keystone** that makes SRM both **systematic** (every combination explicitly defined) and **programmable** (tools can parse, group, filter, and pivot off its structured metadata). That's why it's far more than a "grid"—it's the **experimental engine** behind every Raccoon Schema.