

AI4MetaScience Hackathon

Three candidate tasks

Task 1 — “Is this question worth asking?” Ideation stress-test tool

BACKGROUND / CONTEXT

A huge amount of wasted research effort traces back to the very first step: the question itself. Researchers often commit months of data collection, coding, or lab time to a question that turns out to be vague (“does social media affect wellbeing?”), unanswerable with available data, or already answered by prior work they hadn't fully checked. Advisors and reviewers catch these problems eventually, but usually late — after a draft exists, when it's expensive to go back and reframe. There's no consistent, early gut-check on whether a question is precise enough, testable, and worth the resources it will consume.

TASK DESCRIPTION

Build a tool where a researcher pastes in one research question. The tool should:

- Extract and restate the question's key terms, flagging any that are ambiguous or used inconsistently in the literature
- Surface the implicit assumptions baked into the question
- Identify what evidence or data would actually be needed to answer it, and flag if that evidence is unlikely to be obtainable
- Output a recommendation: proceed / narrow / revise / stop, with reasoning

Success criterion: given a deliberately vague or overly broad question, the tool should produce a specific, actionable critique — e.g., “your term 'engagement' is used in three different senses in the literature” — rather than generic advice like “consider clarifying your terms.”

Task 2 — Provenance-preserving lab notebook for AI-assisted execution

BACKGROUND / CONTEXT

Good research has always depended on a trail: why a method was chosen, what was tried and abandoned, which version of the code produced which result. Lab notebooks and version control exist precisely because memory is unreliable and because reviewers, collaborators, and future-you need to reconstruct decisions after the fact. As more of the actual execution — coding, running experiments, analysis — gets delegated to AI tools that iterate quickly, this trail is exactly what tends to disappear. It's easy to re-run, re-prompt, and move on without anyone recording why the approach changed or what the failed attempts revealed, which quietly erodes the accountability and reproducibility that research depends on.

TASK DESCRIPTION

Build a lightweight logging/wrapper tool that sits around an AI-assisted research task (e.g., an agent doing data cleaning, running experiments, or iterating on a model). For a single defined task, it should automatically capture:

- Each plan or hypothesis proposed and the reasoning behind it
- What was actually run (code, parameters, data version)
- What failed or contradicted expectations, and how the plan changed in response
- A timeline a human can walk back through to see the actual decision path, not just the final output

Success criterion: a reviewer who wasn't present can look at the log after the fact and answer "why did they change approach at step 4?" without having to ask the researcher.

Task 3 — Claim-to-evidence linker for one paper

BACKGROUND / CONTEXT

Peer review is supposed to be the mechanism that separates well-supported claims from overreaching ones, but in practice reviewers rarely have time to trace every claim in a paper back to its specific supporting evidence — a figure, a table, a results paragraph, a line of code. Readers end up trusting a paper largely based on how polished and confident it reads, which has always been an imperfect proxy and becomes a much weaker one when polished writing is cheap to produce. What's missing is a routine way to check specific claims against the specific evidence meant to support them, rather than reducing an entire paper's trustworthiness to one impression or score.

TASK DESCRIPTION

Build a tool that takes one paper (or preprint) as input and, for each major claim in the text, attempts to locate and link the exact evidence for it — a specific figure, table, results paragraph, or code path — and flags:

- Claims that have no traceable supporting evidence in the document
- Claims where the evidence is weaker than the stated confidence (e.g., "we show X" backed only by a single small-n experiment)
- What was checked vs. what remains unverified

Output should be a claim map (claim → evidence → confidence/caveat), not a single score, so the reader can inspect each link individually.
