

000
001  **HARPA: A TESTABILITY-DRIVEN, LITERATURE-
002 GROUNDED FRAMEWORK FOR RESEARCH IDEATION**
003
004
005

006 **Anonymous authors**
007 Paper under double-blind review
008
009

010 **ABSTRACT**
011

012 While there has been a surge of interest in automated scientific discovery (ASD),
013 especially with the emergence of LLMs, it remains challenging for tools to generate
014 hypotheses that are both testable and grounded in the scientific literature. Additionally,
015 existing ideation tools are not adaptive to prior experimental outcomes. We developed
016 HARPA¹ to address these challenges by incorporating the ideation workflow inspired
017 by human researchers. HARPA first identifies emerging research trends through
018 literature mining, then explores hypothesis design spaces, and finally converges on
019 precise, testable hypotheses by pinpointing research gaps and justifying design choices.
020 Our evaluations show that HARPA-generated hypothesis-driven research proposals
021 perform comparably to a strong baseline AI-researcher across most qualitative di-
022 mensions (e.g., specificity, novelty, overall quality), but achieve significant gains in
023 feasibility (+0.78, $p < 0.05$, bootstrap) and groundedness (+0.85, $p < 0.01$, bootstrap)
024 on a 10-point Likert scale. When tested with the ASD agent (CodeScientist), HARPA
025 produced more successful executions (20 vs. 11 out of 40) and fewer failures (16
026 vs. 21 out of 40), showing that expert feasibility judgments track with actual exe-
027 cution success. Furthermore, to simulate how researchers continuously refine their
028 understanding of what hypotheses are both testable and potentially interesting from
029 experience, HARPA learns a reward model that scores new hypotheses based on
030 prior experimental outcomes, achieving approx. a 28% absolute gain over HARPA’s
031 untrained baseline scorer. Together, these methods represent a step forward in the
032 field of AI-driven scientific discovery.

033 **1 INTRODUCTION**

034 Scientific discovery fundamentally depends on effective hypothesis generation—a creative, iterative,
035 and cognitively complex process. In the past year, advances in large language models (LLMs) have
036 revitalized the field of Automated Scientific Discovery (ASD) and AI-assisted ideation, by providing
037 the foundations for agents that can autonomously execute experiments (Lu et al., 2024; Gottweis et al.,
038 2025; Jansen et al., 2025; Li et al., 2024c). At the same time, these models have been applied to generate
039 novel research ideas (Radensky et al., 2024; Pu et al., 2024; Baek et al., 2024b; Wang et al., 2023; Li
040 et al., 2024b), supplying candidate ideas for the experimental agents to explore.

041 One of the central challenges of automated scientific discovery is that the hypotheses generated by large
042 language models rarely rise to the level of breakthrough discoveries (Gottweis et al., 2025). While
043 such hypotheses may be novel or creative, they are frequently infeasible as research proposals (Si et al.,
044 2025). Common issues include limited grounding in literature, omission of critical methodological
045 details, and reliance on resource-intensive experimental designs that exceed the capacity of ASD agents.
046 These challenges mirror findings from prior studies, where ideation systems often produce ideas that
047 are too abstract to be actionable, require substantial human intervention to refine into testable research
048 proposals (Li et al., 2024b; Vasu et al., 2025; Radensky et al., 2024; Wang et al., 2023; Pu et al., 2024),
049 or lack mechanisms to balance novelty with feasibility (Li et al., 2024c; Jansen et al., 2025; Gottweis
050 et al., 2025).

051 In this work, we present HARPA — **Hypothesis & Research Proposal Assistant** — a novel multi-stage
052 computational framework that generates literature-grounded research proposals with specific hypotheses
053 well-supported for ASD systems. HARPA is composed of a *proposal generator* and a *scorer*, as shown

054 ¹All code and data used in this paper will be made publicly available at GitHub Link: (removed for review).

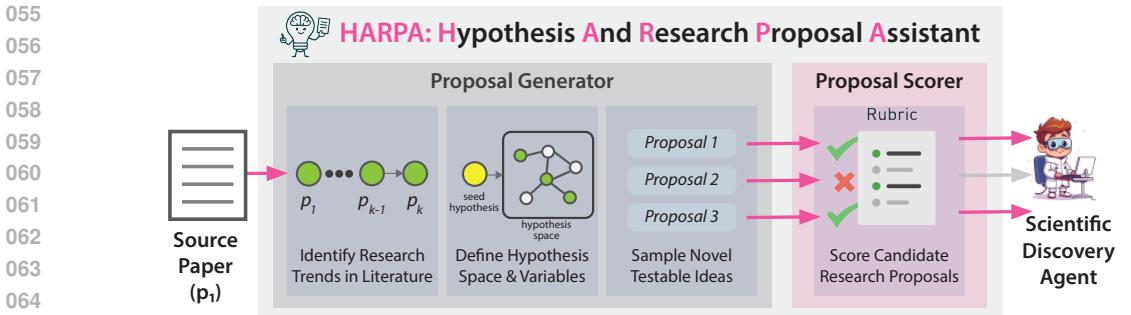


Figure 1: Overview of HARPA. Starting from a [source paper’s abstract](#), seed hypothesis derived from literature trends, HARPA constructs a *world model* of variables, values, and supporting evidence. The proposal generator consists of three stages (trend identification, hypothesis space exploration for divergence, proposal sampling for convergence) to produce candidate hypothesis-driven research proposals. A dedicated scorer employs reasoning-based reward model based on prior execution evidences to evaluate testability w.r.t target ASD agent.

in Figure 1. HARPA generates significantly more feasible research ideas by virtue of its generation approach being strongly grounded in the scientific literature: first identifying relevant research trends related to a user’s hypothesis of interest, then systematically exploring the constructed hypothesis space of variables and their values, and finally converging on novel (and significantly more feasible) hypotheses as well-reasoned combinations of variables and research trends that fill identifiable research gaps in the literature.

We evaluate HARPA using a combination of expert human studies as well as ASD agents. We empirically show that HARPA-generated proposals are more feasible and better grounded in the scientific literature than those produced by contemporary systems. Beyond these gains, we further demonstrate that HARPA’s reward-trained scorer, distilled in RM-R1 style (Chen et al., 2025c), can predict which research proposal is most likely to execute on the ASD agent. Unlike a black-box classifier, the scorer produces rubric-style reasoning traces, interpretable justifications generated from the proposal content and conditioned on the ASD agent’s capabilities, trained to reflect patterns distilled from prior execution outcomes. This enables HARPA to incorporate feedback from prior experimental evidences to selectively generate proposals tailored to the strength and constraints of specific ASD agent - much as a professor might guide a student toward research ideas aligned with the student’s prior knowledge and expertise.

Our results empirically demonstrate that HARPA nearly doubles the scientific output of automated discovery systems, measured as the number of successfully executed experiments, while also reducing costs by pruning infeasible proposals unlikely to succeed before they are attempted. Our contributions:

- **HARPA:** a novel literature-grounded framework for hypothesis generation for ASD systems, that combines identifying research trends, hypothesis space construction, and testability-aware convergence to generate proposals that are novel and executable.
- **Empirical demonstration:** studies with both human domain experts and automated scientific discovery systems showing that HARPA-generated proposals are rated higher in feasibility (+0.78, $p \leq 0.05$) and literature-grounding (+0.85, $p \leq 0.01$), and achieve higher execution success (~80% more, 20 vs. 11), compared to competing systems.
- **Learned feasibility:** We demonstrate that HARPA’s scorer — an interpretable reward model distilled from actual execution traces, can predict which research proposals are most likely to be executable by a given ASD agent, significantly saving time and cost by selectively pruning hypotheses that the system is unlikely to execute. HARPA’s scorer achieves a +0.28 absolute, 53% relative gain over the untrained baseline scorer.
- A publicly available implementation of this approach (HARPA), including the HARPA-Scorer model (to be released on Hugging Face), and first large-scale ASD execution traces and preference dataset to support reproducibility and future research.

Together, these contributions represent a step toward more capable hypothesis generation tools and help advance the rapidly growing field of AI-driven scientific discovery.

2 RELATED WORK

Human hypothesis generation. Cognitive science highlights that scientific hypothesis generation is a complex iterative process involving strategies such as analogical reasoning and model based thinking, where simplified representations guide inquiry (Dunbar, 2000; Nersessian, 2010; Klahr & Simon,

1999). The Scientific Discovery as Dual Search (SDDS) model (Klahr & Dunbar, 1988) identifies strategies such as searching memory for relevant hypotheses and generalizing from experimental results, underscores the need for the ASD systems that can reason over structured hypothesis spaces and adapt from experimental feedback. Prior work has also examined how researchers navigate the broader scientific landscape, where scientists often favor incremental, topic-adjacent experiments (Rzhetsky et al., 2015), with only a minority pursuing riskier but higher-impact directions (Foster et al., 2015).

116 **Automated ideation tools.**

117 Computational frameworks such
118 as Literature-Based Discovery
119 (LBD) (Swanson, 1986) illustrate
120 how disconnected literatures
121 can be bridged to reveal hid-
122 den hypotheses. More recent
123 systems (Radensky et al., 2024;
124 Wang et al., 2023) focus on
125 producing super-brief, novelty-
126 driven research ideas typically
127 assessed with human judgments
128 rather than execution. Systems
129 such as Chain of Ideas (Li
et al., 2024b) and HypER (Vasu

130 et al., 2025) identify literature trends but generate ideas that are too high-level to be actionable,
131 while Scideator (Radensky et al., 2024) generates diverse coarse-grained facets such as purpose,
132 mechanism, or contribution, offering novelty but lacking operational clarity and require human
133 refinement. IdeaSynth (Pu et al., 2024) transforms research ideas into proposals but demands substantial
134 human-in-the-loop involvement, limiting scalability. Recent systems such as Nova (Hu et al., 2024),
135 Dolphin (Yuan et al., 2025), and hypothesis proposers (Yang et al., 2024a; Qi et al., 2023) further
136 explore enhancing novelty and diversity in LLM-generated ideas, but they remain too high-level to
137 be actionable for proposal-level evaluation. Existing systems lack mechanisms to adapt their ideation
138 in response to experimental feedback (Table 1). In this paper, we compare HARPA with the AI
139 Researcher method (Si et al., 2024), which was custom-built for open-domain proposal generation and
140 has demonstrated state-of-the-art performance on this task.

141 **Bridging ideation and execution.** Large-scale evaluations (Si et al., 2024; 2025) show that while
142 AI-generated ideas may be perceived as more novel than expert-authored ones, they are often less
143 feasible experimentally. Other ideation frameworks, including MLR-Copilot (Li et al., 2024c) and Agent
144 Laboratory (Schmidgall et al., 2025), emphasize benchmark-guided or multi-agent settings but fall short
145 of systematic experimental comparisons. Execution focused systems like CODE SCIENTIST (Jansen et al.,
146 2025) and AI-Scientist (Lu et al., 2024) demonstrate end-to-end automated experimentation but assume
147 hypotheses are already well-structured and feasible. NovelSeek (Team et al., 2025) extends this direction
148 by performing multi-round optimization and debugging given an existing idea or codebase, rather than
149 generating open-domain full proposals. HARPA complements these systems by generating structured,
150 literature-grounded proposals. Unlike other ideators, HARPA integrates a reward model conditioned on
151 ASD capabilities, making research hypotheses generation novel, grounded, and experimentally feasible.
152 This makes it useful for human researchers positioning it as a building block toward the long-term vision
153 of “robot scientists” (King et al., 2009). Table 1 compares HARPA with representative systems in the
154 literature over different ideation attributes.

155 3 HARPA: HYPOTHESIS AND RESEARCH PROPOSAL ASSISTANT

156 HARPA’s design is inspired by studies of how humans generate hypotheses (Section 2). HARPA
157 consists of two core components: a *proposal generator* and a *scorer*. The proposal generator begins
158 with a user-given source paper and generates detailed, literature-grounded hypothesis-driven research
159 proposals by treating hypotheses as structured research artifacts, enriched with a rationale (literature-
160 based justification explaining how prior work motivates the preliminary hypothesis), related work, key
161 variables, and operationalization plans. The scorer complements this process by ranking and filtering
162 proposals with a learned reward model that predicts feasibility and testability without requiring full
163 execution. These components together allow HARPA to produce hypothesis-driven proposals that are
164 not only novel and grounded in prior work, but also prioritized for practical execution by ASD agents.

165 ²Systems: GPT-5 (OpenAI, 2025), Scideator (Radensky et al., 2024), Moose-Chem (Yang et al., 2024b), CodeScientist Ideator (Jansen et al.,
166 2025), AI-Researcher (Si et al., 2024)

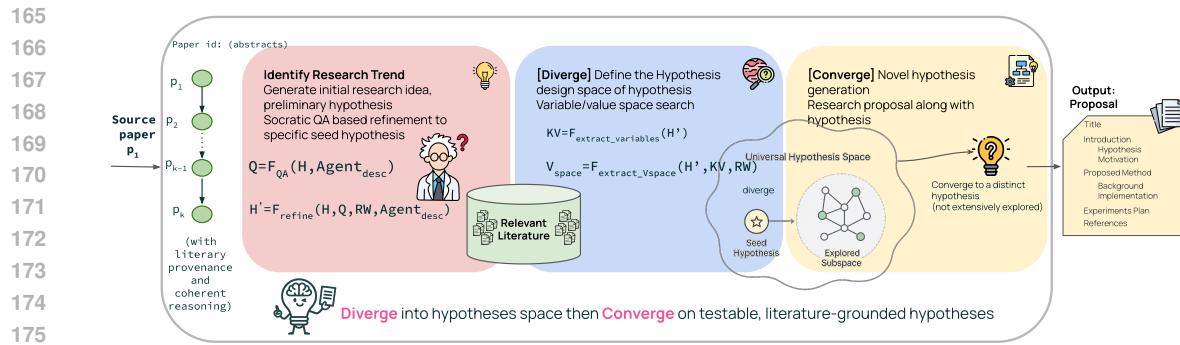


Figure 2: HARPA’s Proposal Generator: Divergence and convergence to literature grounded novel proposals

3.1 HARPA’s PROPOSAL GENERATOR

HARPA begins by constructing the scientific reasoning chain of papers given a source paper. The chain construction is based on (Vasu et al., 2025), where each paper is connected to the previous paper based on its scientific dependency and the citing relation. This reasoning chain enables HARPA to identify a preliminary research gap and the motivation to come up with a research problem and hypothesis (Appendix L 8). However, the seed hypothesis generated at this stage is not very specific and is not optimized for novelty or the feasibility of the idea. To systematically develop a literature-grounded research proposal that is also novel and feasible, we follow the following steps:

Refinement with Socratic Question Answering Recently, Socratic questioning has been applied to language models as a self-guiding mechanism (Chen et al., 2025b). We apply Socratic questioning to help the model think critically, uncover assumptions, and prompt a deeper understanding of the generic preliminary hypothesis. Given a set of relevant literature snippets (*RW*) extracted from related works associated with this preliminary hypothesis, and the description of the underlying ASD agent (*Agent_{desc}*) which executes this hypothesis (*H*), we generate a set of at least 20 questions, *Q*, that helps to navigate the specificity of this hypothesis (see Appendix L 10). This is denoted as $Q = \mathcal{F}_{QA}(H, Agent_{desc})$. Then, given this set of *Q* and the set of literature snippets *RW* relevant to *H* and *Agent_{desc}*, the language model can refine *H* by answering these *Q*. We denote this using $H' = \mathcal{F}_{refine}(H, Q, RW, Agent_{desc})$. The detailed instruction to refine and make the hypothesis more specific is given in Appendix L 11. Using *H* as the query, we systematically collect *RW* using the snippet search over S2ORC corpus (Singh et al., 2025). Specifically, *H* is progressively generalized *n* times (see Appendix L 9) and each version of *H* is used as a query to collect *RW*.

Defining the hypothesis design space using H' To understand the key concepts and variables around this hypothesis in hand, we first extract the set of key variables (*KV*) from it (see L12), denoted as $KV = \mathcal{F}_{extract_var}(H')$. However, the relevant literature might have already explored similar variables or different values of these variables (a brief illustration in Appendix L 7). We extract and define this key variable space as $V_{space} = \mathcal{F}_{extract_space}(H', KV, RW)$. In this process, we ensure that each of these key variables or values mentioned in the related literature is associated with H' . To do this, the extraction process also extracts metadata such as the source paper title and the specific details and description relevant to H' about this variable (see Appendix L13). We also allow the model to add as extra variables a small set of standard evaluation metrics (“accuracy”, “precision”) that were given as illustrative examples in the prompt. When these are added without direct literature evidence, they are explicitly marked as “LLM-recommended”.

Convergence to novel hypothesis The research trend, initial idea, and hypothesis design space can be seen as HARPA’s world model around the initial idea H' . It encodes the key components of the idea and if or how they are being addressed in the relevant literature. Given this hypothesis space, H' is now converged into a distinct hypothesis, H_{final} , which has not been extensively studied in the given space. We denote this process as $H_{final} = \mathcal{F}_{generate}(H', V_{space}, RW)$. Along with H_{final} , we also generate the detailed description of this hypothesis such as overview, detailed description of key variables, idea design including how the combination of the variables can be integrated or how the hypothesis can be implemented in a high level and some of the source papers (initial chain of papers and trend) from which

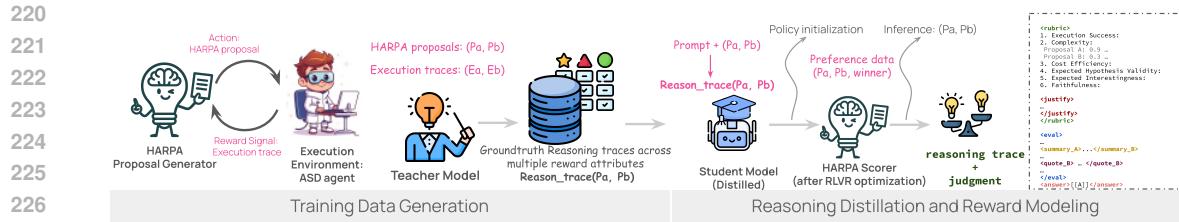


Figure 3: **HARPA Scorer:** 1. *Training Data Generation.* HARPA generates candidate proposals (P_a, P_b), which are executed in the ASD-agent environment to produce raw execution traces (E_a, E_b). A teacher LLM analyzes these traces and outputs a high-fidelity rubric-style reasoning trace with justification and answer ($Reason_trace(P_a, P_b)$). 2. *Reasoning Distillation and Reward Modeling.* The student model is distilled from these reasoning traces, initialized as a policy, and fine-tuned via RLVR using preference labels to produce a rubric-style reasoning trace and a preference label (e.g., ‘Proposal A wins’, an example trace in Appendix L 1).

this idea is evolved as related work (detailed instruction in L14, Appendix F). All LLM function calls in this pipeline were backed by GPT-4o model1.

HARPA also specifies the operationalization of this idea, so that the underlying ASD agent or human researcher can have more details about its implementation plan. For this, we utilize the functionality—idea to implementation plan—of CodeScientist (Jansen et al., 2025). All this information together forms the final $\mathcal{HARPA}_{proposal}$ (example in Appendix C).

3.2 HARPA SCORER: ESTIMATING TESTABILITY OF PROPOSALS

Generating and executing every candidate proposal, whether by human researchers or autonomous agents, is infeasible at scale. To address this challenge, we develop a *learned reward model* that predicts the likely success of a research proposal without requiring full execution. Existing approaches either rely on direct execution (costly) (Li et al., 2024c; Lu et al., 2024) or on heuristic judgments by LLMs on feasibility (Si et al., 2024; Chen et al., 2025a; Yang et al., 2024b; Baek et al., 2024a), which are often unreliable (Li et al., 2024a) and lack grounding in prior experimental evidence (Zhu et al., 2025). Our goal is to provide a scalable and interpretable mechanism to filter and rank research proposals, prioritizing those that are both novel and feasible for the given ASD agent. See Figure 3 for the overview of the HARPA scorer.

Training Data Generation. We collect preference data by executing HARPA-generated proposals using an off-the-shelf ASD agent, CODESCIENTIST, that runs containerized Python experiments. Each execution (E) produces raw traces of the experiment setup, intermediate errors, and automatic assessments, and a final report. We convert the structured experiment summaries (e.g., Appendix L 2) generated by CODESCIENTIST into categorical outcome labels using a meta-analysis scheme:

$$\text{Label}(E) = \begin{cases} \text{Success} & \text{if } \text{faithfulness_category} = \text{faithful}, \\ \text{Failure} & \text{if } \text{faithfulness_category} = \text{errors} \\ & \vee (\text{faithfulness_category} = \text{inconclusive} \wedge \\ & \text{hypothesis_category} = \text{inconclusive}), \\ \text{Uncertain} & \text{otherwise.} \end{cases}$$

where `faithfulness_category` indicates whether the experiment was executed faithfully without implementation errors, and `hypothesis_category` captures whether the observed outcomes ‘support,’ ‘reject,’ or remain ‘inconclusive’ w.r.t the original hypothesis. The *Uncertain* label captures executions that neither cleanly succeed nor fail, ensuring ambiguous traces do not distort the success or failure boundary. These labels are then used to construct pairwise preferences: for each pair (P_a, P_b) , a teacher LLM analyzed (see Appendix L 3) the corresponding traces and generated a rubric-style reasoning trace, along with a preference judgment based on the observed outcome. This yields high-quality training data for the distillation, consisting of pairwise comparisons with interpretable justifications that reflect the empirical feasibility.

Reasoning Distillation and Reward Modeling. We train the HARPA scorer in two stages following the RM-R1 framework (Chen et al., 2025c). First, we distill the teacher’s rubric-style reasoning traces into the student model. This facilitates the student with the ability to generate interpretable justifications

aligned with teacher rubrics. Next, we train the distilled model with preference-based optimization using the RLVR strategy (Chen et al., 2025c), aligning its scoring with empirically verifiable outcomes ('success,' 'failure') from CODESCIENTIST executions. The model outputs both (i) a comparative label (e.g., "Proposal A wins") and (ii) a rubric-style reasoning trace explaining the decision. This dual output allows the model to function not only as a black-box scorer but also as an explainer, providing transparent, human-readable justifications that can be used to refine research proposals. An example reasoning trace is provided in Appendix L 1, showing how the model assigns higher feasibility to one proposal using execution-derived factors, such as execution success, complexity (based on reflection), and cost efficiency parsed from the structured experiment summary of CODESCIENTIST.

Conditioning on ASD capabilities. To ensure judgements are adaptable to the targeted execution environment, the reward model is conditioned on an explicit ASD agent (see Appendix L 3), specifying constraints such as compute budget, permissible evaluation protocol, dataset access, and whether human involvement is allowed. During both training and inference, the agent profile is concatenated with the proposals and execution metadata. In our case, conditioning reflects the limits of CODESCIENTIST, but the same mechanism applies to other agents. For instance, proposals requiring human studies or private datasets are down-ranked for CODESCIENTIST but could go higher for a more capable agent. This makes HARPA's scorer adaptive, producing feasibility-aware rankings that generalize across different discovery settings.

4 EXPERIMENTS

We evaluate HARPA along two complementary axes: (1) a human-centric expert study to evaluate whether generated proposals are appealing to human researchers, and (2) an ASD-centric execution study, which measures the operational testability of proposals through the reward modeling.

4.1 BASELINES

We compare HARPA against different baselines depending on the evaluation axis. **Human-centric Evaluation:** We compare HARPA proposal generator against AI-Researcher (Si et al., 2024), a strong baseline for literature-grounded ideation. We standardized section headings to match proposal formats across systems. For references, we included the papers AI-Researcher internally retrieved, whereas HARPA had literature identified during its multi-stage pipeline. To ensure comparability, we generated topics from each source paper's abstract (since AI-Researcher expects a topic rather than a source paper). **Both systems are seeded with the same information (abstract of the source paper), and neither is given access to the full paper. Each system then uses its own retrieval pipeline, which is an integral part of its design and therefore kept unchanged for end-to-end comparison.** Apart from this topic generation step, all other settings followed the original AI-Researcher implementation. **Agent-centric Evaluation:** For the HARPA scorer, we compare the two variants: (i) an untrained LLM scorer applied directly to a pair of proposals, and (ii) the HARPA scorer, our distilled and RLVR-trained reward model. This setup allows us to isolate the benefit of training the scorer while keeping the proposal generator fixed. We use Qwen-7B-Instruct as the backbone, with the non-finetuned model as the LLM scorer baseline and the trained version as HARPA scorer.

4.2 HUMAN-CENTRIC EVALUATION SETUP

Participants: We recruited 12 experts who have experience in writing and reviewing scientific articles in their domain of interest via *Upwork.com*. See Appendix A.1 for detailed backgrounds and screening criteria. **Dataset:** Our evaluation corpus was constructed dynamically by the experts themselves. Each expert selected source papers (≥ 20 citations, published before 2025) in their domain of expertise. This design ensured informed and fair evaluation in a familiar context. For each source paper, we generated two proposals from HARPA and two from the baseline, and each expert evaluated proposals from at most two source papers of their choice. This process resulted in 40 proposals per system overall. Proposals were uniformly formatted with identical section headings — title, introduction, proposed method, experiments plan, and references, and covered diverse topics (e.g., NLP, RAG, RL, Optimization). (Corpus statistics in Appendix A.1.1)

Evaluation Rubric: We adapted our evaluation rubric from the idea review form of Si et al. (2024) for evaluating research proposals. Experts rated each proposal on a 10-point Likert scale for *Familiarity*, *Novelty*, *Feasibility*, *Expected effectiveness*, *Excitement*, *Overall*, and *Confidence*, providing brief

330 textual justifications (full rubric in Appendix B). In addition to the original rubric, we introduced four
 331 dimensions relevant to hypothesis-driven proposals and their operationalization: *Literature Grounding*,
 332 *Motivation from Literature*, *Coherence of Idea Composition*, and *Specificity of Proposed Method*. In
 333 total, the rubric covered 11 dimensions, with full wording provided in Appendix B. **Protocol:** Proposals
 334 were presented to each expert in randomized order, with system identities hidden. The same expert who
 335 provided the source paper independently assessed and rated all four proposals (including baseline and
 336 HARPA) to ensure fair comparison on the same topic. Some experts reviewed proposals for more than
 337 one source paper ³. Data collection was carried out using the Label Studio platform and experts were
 338 compensated at a rate of 35USD/hr.

339 4.3 AGENT-CENTRIC EVALUATION SETUP

340 Here we evaluate proposals by executing them with the CODESCIENTIST providing data.

341 **Data Curation.** We sampled 275 highly cited ACL papers as source papers and generated up to five
 342 HARPA proposals per paper (1, 222 total). Each proposal was executed *five* times each in CODESCIEN-
 343 TIST to avoid the stochasticity in LLM-based code generation. From each of the five runs, we selected
 344 the execution trace that most truly representing the research proposal and considered that for further
 345 analysis. Outcomes were labeled as SUCCESS (29.38%), FAILURE (51.55%), or UNCERTAIN (19.07%)
 346 according to the categorical outcome labels described before.

347 **Preference Construction and Training:** From these labeled executions, we constructed 3954 prefer-
 348 ence pairs on shared source paper topic (see Appendix L 5). Each pair with execution metadata was used
 349 to generate a rubric-style reasoning trace (including preference judgments) by an oracle model⁴, which
 350 achieved 87.48% accuracy. We filtered the pairs with correct judgements (3459) and their reasoning
 351 traces as ground truth for further experiments. **Distillation and RLVR:** We split the proposals into
 352 training (2595), validation (452), and test (412) subsets. Following the RM-R1 framework Chen et al.
 353 (2025c), we first distilled a student model to generate interpretable rubric-style reasoning aligned with
 354 teacher rubrics. We further applied RLVR training on preference pairs (using an additional 226,170
 355 success-failure pairs irrespective of shared topic). Finally, we evaluated the distilled reward model
 356 on a held-out set of success-failure pairs (186), using accuracy and qualitative analysis of reasoning
 357 traces. This two-stage process yields the *HARPA Scorer* that is both interpretable and adaptive to ASD
 358 execution (more implementation details in Appendix D.1).

359 A full end-to-end evaluation would be ideal, but it is too expensive and would require generating many
 360 additional proposals and obtaining impractical expert annotation on largely random samples. We evaluate
 361 the two components separately, since the scorer only becomes meaningful once the generator produces
 362 executable hypotheses: expert review on a small set of relevant samples and large-scale testability on
 363 diverse ACL papers. This setup provides a controlled evaluation of each component in isolation, and we
 364 leave a full end-to-end evaluation to future work.

366 5 MAIN RESULTS

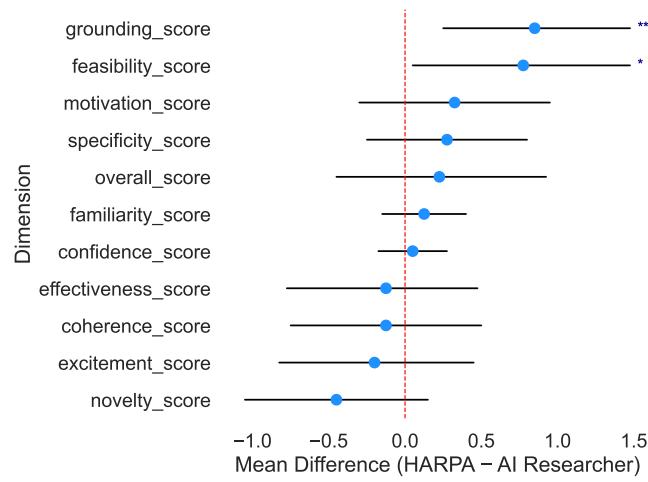
367 5.1 HUMAN-CENTRIC RESULTS

368 Figure 4 summarizes the expert evaluations of HARPA’s proposal generator against the baseline across
 369 11 dimensions. Nine dimensions define the research proposal quality (i.e., novelty, feasibility, expected
 370 effectiveness, excitement, grounding, specificity, coherence, motivation, and overall quality), while
 371 two meta-dimensions capture the user’s familiarity with the proposal topic and their confidence in the
 372 judgment (complete proposal evaluation form in Appendix B). HARPA shows statistically significant
 373 gains in feasibility (+0.78, $p < 0.05$, bootstrap) and grounding (+0.85, $p < 0.01$, bootstrap). For
 374 specificity, motivation, and overall scores, HARPA shows a positive trend, although it does not rise to
 375 the level of statistical significance. For other metrics, HARPA performs comparably to the baseline
 376 (Appendix Table 6), showing that improvements in feasibility and grounding without sacrificing clarity
 377 or novelty. Novelty scores for HARPA averaged 5.98 ± 1.33 compared to 6.43 ± 1.32 for the baseline,
 378 with both systems rarely falling below the midpoint of the 10-point scale. This indicates that HARPA
 379 produces ideas perceived as incrementally novel. This difference is not statistically significant, and both
 380 systems fall within the same *incremental-to-reasonably-novel* range defined in the evaluation rubric.
 381 These findings align with our design goal that grounding research proposals in literature and refining

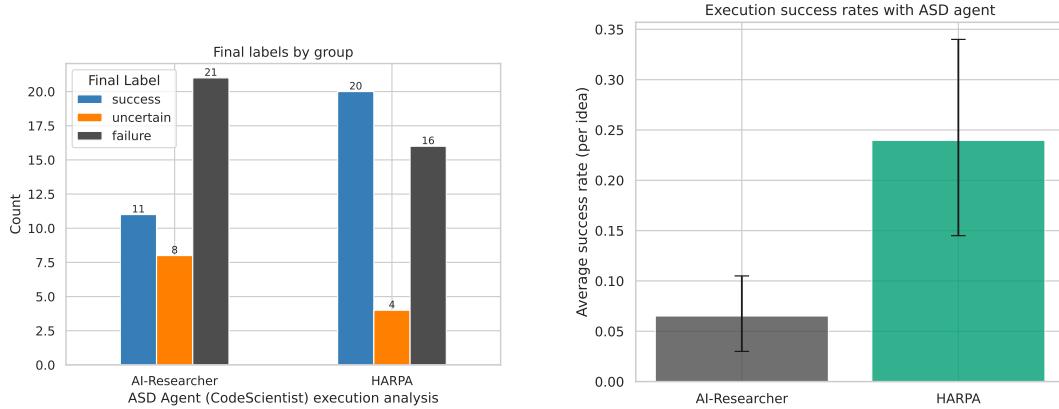
382 ³Since source papers were selected individually, proposals were unique to each expert and not cross-reviewed.

383 ⁴claude-sonnet-4 was used as an Oracle model.

385 hypotheses through a human-like workflow leads to more operational, testable research proposals.
 386 (Detailed rating distributions in Appendix A.1.)
 387



404
 405 Figure 4: Mean difference between HARPA’s proposal generator and AI-Researcher across nine evaluation
 406 dimensions. Also reporting the familiarity and confidence score differences. Points show average differences,
 407 horizontal bars indicate 95% bootstrap confidence intervals (10k resamples). Stars indicate significant difference
 408 computed using the nonparametric bootstrap test (* $p < 0.05$, ** $p < 0.01$)



423 (a) Execution outcomes (counts) from CODESCIENTIST
 424 runs, labeled as success, uncertain, or failure.

425 (b) Average per-idea success rates (95% bootstrap CIs)
 426 for HARPA vs. baseline AI-Researcher proposals.
 427

428 Figure 5: Execution results from CODESCIENTIST. Left: outcome distribution across groups. Right:
 429 paired comparison of mean success rates showing HARPA significantly outperforms the baseline AI-
 430 Researcher.

431 **Execution Success rates:** We evaluated whether HARPA proposals more often succeed when executed
 432 by a typical ASD agent (here, CODESCIENTIST). For each user-evaluated proposal, we executed five
 433 independent runs labeled outcomes using the meta-analysis labeling scheme (Section 3.2). Figure 5a
 434 shows the raw distribution of execution outcomes across groups. HARPA produced a higher number
 435 of successful executions (20 vs. 11 for the baseline) and fewer outright failures than the baseline AI-
 436 researcher system. We further aggregated results into per-idea success rate, defined as the proportion
 437 of faithful executions per idea. To ensure fair comparison, we paired HARPA and baseline proposals
 438 generated from the same source paper and computed within-source paper differences in success rates.
 439 Figure 5b summarizes per-idea success rates where HARPA achieved a higher mean success rate than
 the baseline (0.24 vs. 0.065), and the bootstrap test ($p < 0.001$) confirmed the difference was statistically
 significant. These results together demonstrate that HARPA proposals succeed more often in automated
 executions, consistent with expert ratings of higher feasibility.

440	441	442	443	System		Pairwise Consistency			Accuracy	
				Execution	Complexity	Cost	Hypothesis	Interesting-	Faithfulness	(win)
				Success		Efficiency	Validity	ness		
444	Baseline	–	–	–	–	–	–	–	–	0.52
445	HARPA	0.80	0.67	0.73	0.69	0.55	0.79	–	0.81	

447 Table 2: Pairwise consistency of **HARPA-Scorer** with oracle judgments across rubrics. Consistency is the fraction
 448 of proposal pairs where the scorer and oracle agree. Baseline lacks rubric-level judgments (–). Accuracy comparing
 449 baseline and **HARPA-Scorer** on the test data (N=186 success-failure pairs).

450	451	System	BLEU	BLEU BP	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-Lsum	Len-ratio
452	Baseline	0.08	0.79	0.43	0.13	0.18	0.17	0.81	
453	HARPA	0.22	1.00	0.55	0.22	0.26	0.26	1.12	

454 Table 3: Overlap-based evaluation of HARPA reasoning traces w.r.t reference traces.

455 **Qualitative Examples:** To illustrate these quantitative trends, an expert rated a HARPA proposal as
 456 highly feasible and grounded: “*Using the softmax trick allows backpropagation/gradient estimation,*
 457 *it is a well known trick and the implementation is not so complicated...*” (feasibility = 7, grounding =
 458 9). The expert highlighted that the ideas was concrete and testable, with direct support from the prior
 459 literature. By contrast, a baseline proposal as “*The proposed method looks feasible. The problem is that*
 460 *it lacks details. Everything related to the method is summarized in 2–3 lines in the ‘Proposed methods’*
 461 *without any mathematical language. ...*” was judged infeasible (feasibility = 4) and poorly grounded
 462 (grounding = 2). Although the expert noted that it was an exciting impact, they emphasized that the
 463 lack of detail and irrelevant literature made the proposal impossible to operationalize. Examples of full
 464 proposals and expert assessments are provided in Appendix C

465 **Novelty–Feasibility Trade-off.** HARPA is designed to filter out logically inconsistent or non-
 466 executable ideas, not to suppress high-risk innovation. While such feasibility-oriented filtering can
 467 modestly reduce groundbreaking ideas, the observed novelty difference is not statistically significant,
 468 and experts judged HARPA’s proposals to remain more actionable. This reflects novelty–feasibility
 469 trade-off (Guo et al., 2025), where highly novel ideas often become infeasible; within this trade-off,
 470 HARPA maintains comparable novelty while achieving substantially higher feasibility and grounding.
 471 HARPA therefore targets actionable novelty suitable for ASD agents rather than unconstrained creativity.

472 **Q Takeaway:** *In summary, HARPA bridges the gap between ideation and execution: it generates*
 473 *literature-grounded, feasible, and testable research proposals that succeed **nearly twice as often in ASD***
 474 *execution (20 vs. 11, $\approx 2 \times$), while also outperforming prior systems in expert evaluations.*

475 5.2 AGENT-CENTRIC RESULTS

476 We next evaluated HARPA-scorer against a baseline untrained LLM scorer, a Qwen-7B-instruct
 477 (section 4.1). HARPA-scorer improves accuracy with a +0.28 absolute gain (a 53% relative improvement),
 478 with more balanced performance across classes. **This improvement is largely driven by distilling**
 479 **structured reasoning traces, which provide feasibility cues that the untrained LLM does not capture.**

480 Importantly, scorer inference is far cheaper than executing a full CodeScientist run, so accurate feasibility
 481 prediction directly reduces compute by avoiding many failed executions. In our evaluation split, only 45
 482 of 120 proposals succeed end-to-end; with 81% accuracy, the scorer correctly flags most low-feasibility
 483 proposals before execution. We further tested 35 proposal pairs from the human-evaluation corpus,
 484 which includes a wider mix of computational topics; the scorer attained 74% accuracy. While this does
 485 not establish cross-domain generalization, it suggests the scorer can handle some proposals outside the
 486 ACL-derived distribution. The scorer is general in design and can be extended to new domains or ASD
 487 agents given corresponding execution traces.

488 In addition, HARPA-scorer produces rubric-aligned reasoning traces with explicit scoring on feasibility,
 489 cost efficiency, and complexity, like the teacher model. In contrast, the baseline model produced
 490 unstructured free text (e.g., in Appendix L 6) that lacks actionable justifications and 4.84% unknown

495 predictions. This alignment with oracle-style reasoning makes HARPA-scorer’s judgement easier to
 496 interpret and more reliable for refining the research proposals.
 497

498 Beyond accuracy, we further assessed *pairwise consistency*, whether the scorer agrees with the Oracle on
 499 which two proposals are preferred for each rubric dimension. HARPA-scorer achieves strong alignment
 500 on testability-oriented rubrics, with 0.80 consistency on *Execution Success* and 0.70 on *faithfulness*,
 501 and moderate alignment on *Complexity, Cost Efficiency, and Hypothesis Validity*, while alignment
 502 drops to 0.55 for the more subjective *Interestingness* dimension. These results indicate that the scorer
 503 is capturing reliably testability-related signals while remaining less consistent on subjective criteria.
 504 Finally, we compare HARPA-generated rationales with baseline ones. HARPA significantly ($p < 0.01$,
 505 paired t-test) outperforms the baseline across all overlap metrics with reference rationales. We see major
 506 improvements in BLEU scores (+0.14, a 166% increase) and strong gains in ROUGE-1 (+0.12, +27%),
 507 ROUGE-2 (+0.10, +77%), and ROUGE-L/Lsum (+0.09, +49%). The particularly strong improvements
 508 in higher-order n-grams—like ROUGE-2 and BLEU’s 3-4-gram scores—suggest that HARPA is not
 509 just matching individual words better, but is actually producing more coherent text with better content
 509 flow and sequencing.

510 **Q Takeaway:** *In summary, HARPA’s scorer delivers +0.28 absolute (~53% relative) higher accuracy*
 511 *than an untrained LLM scorer, while providing interpretable rubric-style judgments that enable reliable,*
 512 *execution-informed filtering of research proposals.*

513 6 CONCLUSION AND FUTURE WORK

514 We presented HARPA, a literature-grounded, testability-driven framework for the open-ended task of
 515 hypothesis generation. HARPA systematically extracts research trends, explores existing hypothesis
 516 spaces, and converges on testable hypothesis-driven proposals. We introduced an interpretable reward-
 517 trained scorer that adapts feasibility judgments to ASD agent capabilities, enabling HARPA to prioritize
 518 hypotheses that are executable. Our evaluations show significant improvements in feasibility and
 519 grounding, with HARPA’s proposals also succeeding more often in automated execution. As the scorer
 520 serves as a proxy for resource-intensive experimentation, HARPA enables execution-derived feedback
 521 into future proposal generation, selectively refining hypotheses in line with ASD agent capabilities.
 522 To our knowledge, HARPA is the first ideation framework to learn directly from execution outcomes,
 523 enabling feasibility-aware hypotheses generation, and points to further possible improvements using
 524 even richer training data and execution environments. Together, these contributions represent a step
 525 toward more capable hypothesis generation tools and help advance the rapidly growing field of AI-driven
 526 scientific discovery.

527 ETHICS STATEMENT

528 We honor the Code of Ethics. No personally identifiable information is used in this work. The human
 529 evaluators were hired from Upwork using a detailed job post. We had Institutional Review Board (IRB)
 530 approval for obtaining written consent from our human evaluators. We shared an example task sheet
 531 with complete instructions during the recruitment. The evaluators were duly compensated based on
 532 minimum wage in the respective countries and always above their quotation.
 533

534 THE USE OF LARGE LANGUAGE MODELS (LLMS)

535 We used AI-based tools (Claude, ChatGPT, and Grammarly) for lightly polishing the grammar, clarity,
 536 and identifying errors, and generating code for plots; all ideas and content are the authors’ own.
 538

539 REFERENCES

540 Jinheon Baek, Sujay Kumar Jauhar, Silviu Cucerzan, and Sung Ju Hwang. Researchagent: Iterative
 541 research idea generation over scientific literature with large language models. *ArXiv*, abs/2404.07738,
 542 2024a. URL <https://api.semanticscholar.org/CorpusID:269042844>.

544 Jinheon Baek, Sujay Kumar Jauhar, Silviu Cucerzan, and Sung Ju Hwang. Researchagent: Itera-
 545 tive research idea generation over scientific literature with large language models. *arXiv preprint*
 546 *arXiv:2404.07738*, 2024b.

547 Hui Chen, Miao Xiong, Yujie Lu, Wei Han, Ailin Deng, Yufei He, Jiaying Wu, Yibo Li, Yue Liu, and
 548 Bryan Hooi. Mlr-bench: Evaluating ai agents on open-ended machine learning research. *ArXiv*,
 549 abs/2505.19955, 2025a. URL <https://api.semanticscholar.org/CorpusID:278911978>.

550 Lin Chen, Changshui Zhang, Feng Zhou, Qi Yang, Changming Xiao, Wanpeng Hu, and Haodi Liu.
 551 Socratic questioning: Learn to self-guide multimodal reasoning in the wild, 2025b. URL <https://arxiv.org/abs/2501.02964>.

553 Xiusi Chen, Gaotang Li, Ziqi Wang, Bowen Jin, Cheng Qian, Yu Wang, Hongru Wang, Yu Zhang,
 554 Denghui Zhang, Tong Zhang, et al. Rm-r1: Reward modeling as reasoning. *arXiv preprint*
 555 *arXiv:2505.02387*, 2025c.

557 Kevin Dunbar. How scientists think in the real world: Implications for science education. *Journal of*
 558 *applied developmental psychology*, 21(1):49–58, 2000.

559

560 Jacob G Foster, Andrey Rzhetsky, and James A Evans. Tradition and innovation in scientists' research
 561 strategies. *American sociological review*, 80(5):875–908, 2015.

562 Juraj Gottweis, Wei-Hung Weng, Alexander Daryin, Tao Tu, Anil Palepu, Petar Sirkovic, Artiom
 563 Myaskovsky, Felix Weissenberger, Keran Rong, Ryutaro Tanno, et al. Towards an ai co-scientist.
 564 *arXiv preprint arXiv:2502.18864*, 2025.

565

566 Sikun Guo, Amir Hassan Shariatmadari, Guangzhi Xiong, Albert Huang, Myles Kim, Corey M Williams,
 567 Stefan Bekiranov, and Aidong Zhang. Ideabench: Benchmarking large language models for research
 568 idea generation. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and*
 569 *Data Mining* V. 2, pp. 5888–5899, 2025.

570

571 Xiang Hu, Hongyu Fu, Jinge Wang, Yifeng Wang, Zhikun Li, Renjun Xu, Yu Lu, Yaochu Jin, Lili
 572 Pan, and Zhenzhong Lan. Nova: An iterative planning and search approach to enhance novelty and
 573 diversity of llm generated ideas. *arXiv preprint arXiv:2410.14255*, 2024.

574

575 Peter Jansen, Oyvind Tafjord, Marissa Radensky, Pao Siangliulue, Tom Hope, Bhavana Dalvi Mishra,
 576 Bodhisattwa Prasad Majumder, Daniel S Weld, and Peter Clark. Codescientist: End-to-end semi-
 577 automated scientific discovery with code-based experimentation. *arXiv preprint arXiv:2503.22708*,
 578 2025.

579

580 Ross D. King, Jem Rowland, Stephen G. Oliver, Michael Young, Wayne Aubrey, Emma Byrne, Maria
 581 Liakata, Magdalena Markham, Pinar Pir, Larisa N. Soldatova, Andrew Sparkes, Kenneth E. Whelan,
 582 and Amanda Clare. The automation of science. *Science*, 324(5923):85–89, 2009. doi: 10.1126/
 583 science.1165620. URL <https://www.science.org/doi/abs/10.1126/science.1165620>.

584

585 David Klahr and Kevin Dunbar. Dual space search during scientific reasoning. *Cognitive science*, 12(1):
 586 1–48, 1988.

587

588 David Klahr and Herbert A Simon. Studies of scientific discovery: Complementary approaches and
 589 convergent findings. *Psychological Bulletin*, 125(5):524, 1999.

590

591 Haitao Li, Qian Dong, Junjie Chen, Huixue Su, Yujia Zhou, Qingyao Ai, Ziyi Ye, and Yiqun Liu. Llms-as-
 592 judges: a comprehensive survey on llm-based evaluation methods. *arXiv preprint arXiv:2412.05579*,
 593 2024a.

594

595 Long Li, Weiwen Xu, Jiayan Guo, Ruochen Zhao, Xingxuan Li, Yuqian Yuan, Boqiang Zhang, Yuming
 596 Jiang, Yifei Xin, Ronghao Dang, et al. Chain of ideas: Revolutionizing research via novel idea
 597 development with llm agents. *arXiv preprint arXiv:2410.13185*, 2024b.

598

599 Ruochen Li, Teerth Patel, Qingyun Wang, and Xinya Du. Mlr-copilot: Autonomous machine learning
 600 research based on large language models agents, 2024c. URL <https://arxiv.org/abs/2408.14033>.

601

602 Chris Lu, Cong Lu, Robert Tjarko Lange, Jakob Foerster, Jeff Clune, and David Ha. The ai scientist:
 603 Towards fully automated open-ended scientific discovery. *arXiv preprint arXiv:2408.06292*, 2024.

604

605 Nancy J Nersessian. *Creating scientific concepts*. MIT press, 2010.

606

607 OpenAI. Gpt-5 system card. <https://cdn.openai.com/gpt-5-system-card.pdf>, 2025. Accessed:
 608 2025-09-21.

605 Kevin Pu, KJ Feng, Tovi Grossman, Tom Hope, Bhavana Dalvi Mishra, Matt Latzke, Jonathan
 606 Bragg, Joseph Chee Chang, and Pao Siangliulue. Ideasynth: Iterative research idea development
 607 through evolving and composing idea facets with literature-grounded feedback. *arXiv preprint*
 608 *arXiv:2410.04025*, 2024.

609 Biqing Qi, Kaiyan Zhang, Haoxiang Li, Kai Tian, Sihang Zeng, Zhang-Ren Chen, and Bowen Zhou.
 610 Large language models are zero shot hypothesis proposers. *arXiv preprint arXiv:2311.05965*, 2023.

612 Marissa Radensky, Simra Shahid, Raymond Fok, Pao Siangliulue, Tom Hope, and Daniel S Weld.
 613 Scideator: Human-llm scientific idea generation grounded in research-paper facet recombination.
 614 *arXiv preprint arXiv:2409.14634*, 2024.

615 Andrey Rzhetsky, Jacob G Foster, Ian T Foster, and James A Evans. Choosing experiments to accelerate
 616 collective discovery. *Proceedings of the National Academy of Sciences*, 112(47):14569–14574, 2015.

618 Samuel Schmidgall, Yusheng Su, Ze Wang, Ximeng Sun, Jialian Wu, Xiaodong Yu, Jiang Liu, Michael
 619 Moor, Zicheng Liu, and Emad Baroum. Agent laboratory: Using llm agents as research assistants,
 620 2025. URL <https://arxiv.org/abs/2501.04227>.

622 Chenglei Si, Diyi Yang, and Tatsunori Hashimoto. Can llms generate novel research ideas? a large-scale
 623 human study with 100+ nlp researchers. *arXiv preprint arXiv:2409.04109*, 2024.

624 Chenglei Si, Tatsunori Hashimoto, and Diyi Yang. The ideation-execution gap: Execution outcomes of
 625 llm-generated versus human research ideas. *arXiv preprint arXiv:2506.20803*, 2025.

627 Amanpreet Singh, Joseph Chee Chang, Dany Haddad, Aakanksha Naik, Jena D. Hwang, Rodney
 628 Kinney, Daniel S Weld, Doug Downey, and Sergey Feldman. Ai2 scholar QA: Organized literature
 629 synthesis with attribution. In Pushkar Mishra, Smaranda Muresan, and Tao Yu (eds.), *Proceedings*
 630 *of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 3: System*
 631 *Demonstrations)*, pp. 513–523, Vienna, Austria, July 2025. Association for Computational Linguistics.
 632 ISBN 979-8-89176-253-4. URL <https://aclanthology.org/2025.acl-demo.49/>.

633 Don R Swanson. Fish oil, raynaud’s syndrome, and undiscovered public knowledge. *Perspectives in*
 634 *biology and medicine*, 30(1):7–18, 1986.

635 NovelSeek Team, Bo Zhang, Shiyang Feng, Xiangchao Yan, Jiakang Yuan, Zhiyin Yu, Xiaohan He,
 636 Songtao Huang, Shaowei Hou, Zheng Nie, et al. Novelseek: When agent becomes the scientist–
 637 building closed-loop system from hypothesis to verification. *arXiv preprint arXiv:2505.16938*,
 638 2025.

640 Rosni Vasu, Chandrayee Basu, Bhavana Dalvi Mishra, Cristina Sarasua, Peter Clark, and Abraham
 641 Bernstein. Hyper: Literature-grounded hypothesis generation and distillation with provenance. *arXiv*
 642 *preprint arXiv:2506.12937*, 2025.

643 Qingyun Wang, Doug Downey, Heng Ji, and Tom Hope. Scimon: Scientific inspiration machines
 644 optimized for novelty. *arXiv preprint arXiv:2305.14259*, 2023.

646 Zonglin Yang, Xinya Du, Junxian Li, Jie Zheng, Soujanya Poria, and Erik Cambria. Large language
 647 models for automated open-domain scientific hypotheses discovery. In *Findings of the Association*
 648 *for Computational Linguistics: ACL 2024*, pp. 13545–13565, 2024a.

650 Zonglin Yang, Wanhai Liu, Ben Gao, Tong Xie, Yuqiang Li, Wanli Ouyang, Soujanya Poria, Erik
 651 Cambria, and Dongzhan Zhou. Moose-chem: Large language models for rediscovering unseen
 652 chemistry scientific hypotheses. *arXiv preprint arXiv:2410.07076*, 2024b.

653 Jiakang Yuan, Xiangchao Yan, Bo Zhang, Tao Chen, Botian Shi, Wanli Ouyang, Yu Qiao, Lei Bai,
 654 and Bowen Zhou. Dolphin: moving towards closed-loop auto-research through thinking, practice,
 655 and feedback. In *Proceedings of the 63rd Annual Meeting of the Association for Computational*
 656 *Linguistics (Volume 1: Long Papers)*, pp. 21768–21789, 2025.

658 Minjun Zhu, Qijie Xie, Yixuan Weng, Jian Wu, Zhen Lin, Linyi Yang, and Yue Zhang. Ai scientists
 659 fail without strong implementation capability. *arXiv preprint arXiv:2506.01372*, 2025.

660 Table 4: Distribution of source papers across shared topics.
661

662 shared topic	663 Count
664 Graph Neural Networks and Graph Learning	665 3
666 Generative Models for Images	667 2
668 Recommender Systems with LLMs	669 2
670 Medical and Healthcare Applications	671 2
672 Differentiable Optimization	673 2
674 Bias and Fairness in NLP	675 2
676 Hallucination and Factuality in LLMs	677 2
678 Reinforcement Learning for Scheduling	679 2
680 Retrieval-Augmented Generation and Information Refinement	681 1
682 Continual Learning and Knowledge Distillation	683 1
684 Fake News Detection	685 1

675
676

A APPENDIX

677678

A.1 HUMAN EVALUATION DETAILS

679

680 We recruited 12 experts with demonstrated research experience, spanning diverse academic and research
681 backgrounds. The expert pool included 6 PhD students in Computer Science or related fields, 3 faculty
682 members/academic researchers, and 2 postdoctoral researchers. Their expertise covered a broad range
683 of topics in ML and NLP: bias and fairness in language models (4), multilingual and low-resource
684 NLP (3), factuality and hallucination detection (3), code generation and programming with LLMs (2),
685 uncertainty estimation and interpretability (2), and mathematical reasoning/structured predictions (2).
686 Additional specialized domains included recommender systems and IR, mathematical modeling, deep
687 reinforcement learning, and AI safety/robustness. Note that the counts are not mutually exclusive, as
688 evaluators could select multiple primary research areas.

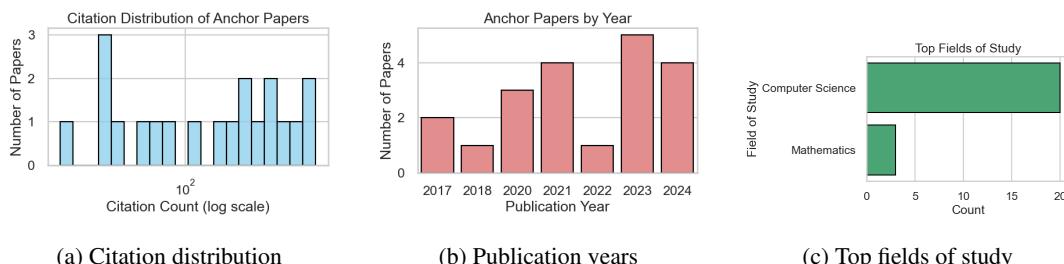
689 In terms of research experience, half of the participants (6/12) reported 3-5 years of active work in their
690 field, three reported 6-10 years, and three reported 10+ years. As part of the screening, each expert
691 shared their Google Scholar profile (or equivalent evidence of publications). The citation count of
692 experts' scholarly work ranged from 7 to 1256 (median =147, mean=297.3). This distribution shows that
693 our evaluation pool included both early-career researchers and more senior researchers with substantial
694 publication records.

695

A.1.1 SOURCE PAPER DOMAINS AND TOPIC DISTRIBUTION

696

697 Table 4 summarizes the distribution of source papers across shared topics, obtained by classifying
698 abstracts into broad topics using the same approach from Listing 5. The topics span from graph neural
699 networks to health-care applications, optimization to fairness. Figure 6 and Table 5 summarize the
700 source papers selected by experts. These papers span recent years (2018-2023), show moderate citation
701 counts, and cover diverse venues.



Venue	Count
arXiv.org	3
Neural Information Processing Systems	2
Annual Meeting of the Association for Computational Linguistics	2
Computer Vision and Pattern Recognition	2
North American Chapter of the Association for Computational Linguistics	2
International Conference on Computer Graphics and Interactive Techniques	1
IEEE Access	1
Knowledge Discovery and Data Mining	1
ACM Transactions on Intelligent Systems and Technology	1
ACM Conference on Health, Inference, and Learning	1

Table 5: Venues of source papers selected by experts.

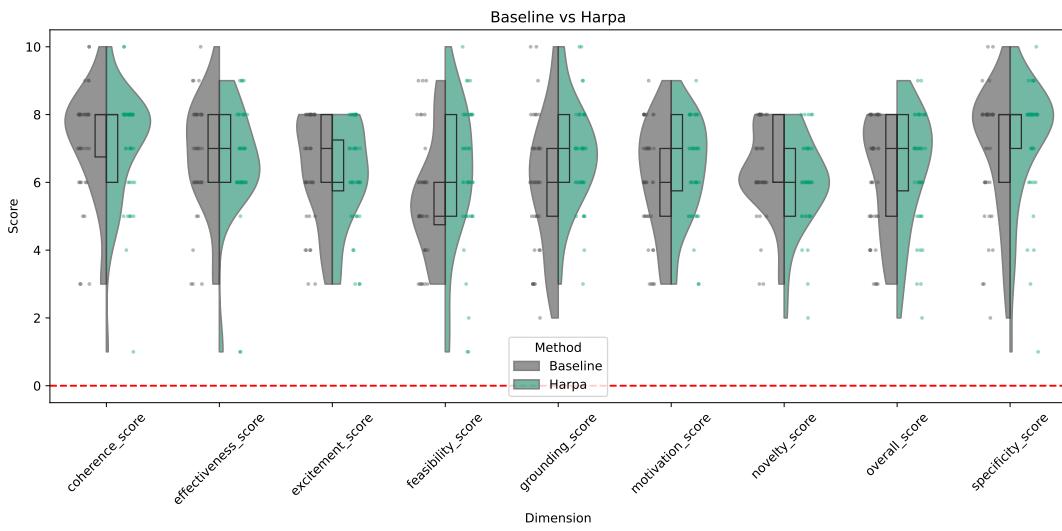


Figure 7: Distribution of expert ratings across nine dimensions for HARPA vs baseline. Shown for completeness (complementary to Fig. 4).

B PROPOSAL REVIEW FORM

We use the following proposal assessment form to elicit reviews from all the experts. Our assessment questions largely follow the expert evaluation protocol introduced by Si et al. (2024) for proposal assessment, but we extended it with several additional questions tailored to hypothesis-driven research proposals. In particular, we added dimensions for *motivation*, *specificity*, *coherence*, and *literature grounding*, as these aspects are critical for ensuring that proposals are both operational and directly testable. To ensure consistency, each question was accompanied by a detailed description of the scale points as well as hints on what evidence to consider (e.g., explicit references, prior knowledge).

The full questionnaire (including all Likert-scale anchors and instructions shown to experts) is reproduced below.

1. Familiarity: Before reviewing the idea, please indicate how familiar you are with the given topic on a scale of 1 - 5 (this is just for us to understand potential confounders).

1. You have never read about this topic before
2. You have read at least one paper on this topic
3. You have read multiple papers on this topic but have not published any paper on it
4. You have co-authored at least one paper on this topic
5. You have co-authored multiple papers on this topic or have published at least one first-author paper on this topic

770 **2. Novelty Score:** Whether the idea is creative and different from existing works on the topic, and brings
 771 fresh insights. You are encouraged to search for related works online. You should consider all papers
 772 that appeared online prior to July 2024 as existing work when judging the novelty.
 773

- 774 1. Not novel at all - there are many existing ideas that are the same
- 775 2.
- 776 3. Mostly not novel - you can find very similar ideas
- 777 4.
- 778 5. Somewhat novel - there are differences from existing ideas but not enough to turn into a new
 779 paper
- 780 6. Reasonably novel - there are some notable differences from existing ideas and probably enough
 781 to turn into a new paper
- 782 7.
- 783 8. Clearly novel - major differences from all existing ideas
- 784 9.
- 785 10. Very novel - very different from all existing ideas in a very interesting and clever way

786 **Novelty Rationale:** Short justification for your score. If you give a low score, you should specify similar
 787 related works. (Your rationale should be at least 2-3 sentences.) *Hint: If the idea is not novel, point out
 788 what is already similar in prior work (e.g., method, task, or setting), and briefly mention any minor
 789 differences if they exist. If the idea is novel, explain what is new—such as a novel method, a new task, or
 790 applying an existing idea to a new domain.*

791 **3. Feasibility Score:** How feasible it is to implement and execute this idea as a research project?
 792 Specifically, how feasible the idea is for a typical CS PhD student to execute within 1-2 months of time.
 793 You can assume that we have abundant OpenAI / Anthropic API access, but limited GPU compute.

- 794 1. Impossible: the idea doesn't make sense or the proposed experiments are flawed and cannot be
 795 implemented
- 796 2.
- 797 3. Very challenging: there are flaws in the proposed method or experiments, or the experiments
 798 require compute/human resources beyond any academic lab
- 799 4.
- 800 5. Moderately feasible: It can probably be executed within the given time frame but would require
 801 careful planning, efficient use of APIs or some advanced computational strategies to overcome
 802 the limited GPU resources, and would require some modifications to the original proposal to
 803 make it work
- 804 6. Feasible: Can be executed within the given constraints with some reasonable planning
- 805 7.
- 806 8. Highly Feasible: Straightforward to implement the idea and run all the experiments
- 807 9.
- 808 10. Easy: The whole proposed project can be quickly executed within a few days without requiring
 809 advanced technical skills

810 **Feasibility Rationale:** Short justification for your score. If you give a low score, you should specify
 811 what parts are difficult to execute and why. (Your rationale should be at least 2-3 sentences.)

812 **4. Expected Effectiveness Score:** How likely the proposed idea is going to work well (e.g., better than
 813 existing baselines).

- 814 1. Extremely Unlikely: The idea has major flaws and definitely won't work well
- 815 2.
- 816 3. Low Effectiveness: The idea might work in some special scenarios but you don't expect it to
 817 work in general

825 4.

826 5. Somewhat ineffective: There might be some chance that the proposed idea can work better

827 than existing baselines but the improvement will be marginal or inconsistent

828 6. Somewhat effective: There is a decent chance that the proposed idea can beat existing baselines

829 by moderate margins on a few benchmarks

830 7.

831 8. Probably Effective: The idea should offer some significant improvement over current methods

832 on the relevant benchmarks

833 9.

834 10. Definitely Effective: You are very confident that the proposed idea will outperform existing

835 methods by significant margins on many benchmarks

836

837 Expected Effectiveness Rationale: Short justification for your score. (Your rationale should be at least 2-3 sentences.) *Hint: You must consider how the novelty of the idea relates to its excitement or impact — if the idea is not novel (e.g., already done before), it should generally not be rated as very exciting.*

838 5. Excitement Score: How exciting and impactful this idea would be if executed as a full project.

839 Would the idea change the field and be very influential.

840 1. Poor: You cannot identify the contributions of this idea, or it's not interesting at all and you

841 would fight to have it rejected at any major AI conference

842 2.

843 3. Mediocre: this idea makes marginal contributions and is very incremental

844 4.

845 5. Leaning negative: it has interesting bits but overall not exciting enough

846 6. Learning positive: exciting enough to be accepted at a major AI conference, but still has some

847 weaknesses or somewhat incremental

848 7.

849 8. Exciting: would deepen the community's understanding or make major progress in this research

850 direction

851 9.

852 10. Transformative: would change the research field profoundly and worth a best paper award at

853 major AI conferences

854

855 Excitement Rationale: Short justification for your score. (Your rationale should be at least 2-3

856 sentences.) *Hint: You must consider how the novelty of the idea relates to its excitement or impact — if*

857 the idea is not novel (e.g., already done before), it should generally not be rated as very exciting.

858 6. Literature Grounding: To what extent are the key components (e.g., model choice, tasks, evaluation

859 strategies) grounded in existing scientific literature? You may also consider whether ideas reflect

860 well-established domain knowledge or listed references.

861 1. Not at all grounded: Mostly speculative or hallucinated; no support from literature or well-

862 established concepts

863 2.

864 3. Weak grounding: A few connections to existing work, but most claims lack clear support from

865 the listed references or alignment with well-known concepts

866 4.

867 5. Partially grounded: About half the components are linked to literature or reflect widely accepted

868 ideas in the field

869 6. Strong grounding: Most core elements are supported by the listed references or well-established

870 concepts, with only minor gaps

871 7.

880 8. Very strong grounding: The vast majority of components are supported by listed references or
 881 widely accepted domain knowledge, though one or two key claims still lack clear support
 882 9.
 883 10. Fully grounded: Every major concept and step is well-supported by listed references or clearly
 884 based on well-established domain knowledge; no unsupported claims remain.

886 **Explanation:** You should also provide a rationale for your score. (Your rationale should be at least
 887 2-3 sentences.) *Hint: If a claim is grounded in well-known concepts but not supported by the listed
 888 references, explain why it is reasonable based on your domain knowledge. Indicate whether your
 889 assessment relies on (a) the proposal's reference list, (b) external sources you know, or (c) generally
 890 accepted field knowledge.*

891 **7. Motivation from Literature:** Is the problem statement/overall idea clearly defined and motivated by
 892 a specific, well-scoped research gap, or limitation identified in the widely recognized field knowledge?

894 1. No clear motivation: idea feels arbitrary or disconnected
 895 2.
 896 3. Weakly motivated: mentions general themes but lacks a compelling rationale
 897 4.
 898 5. Somewhat motivated: a recognizable problem is present, but vague
 899 6. Well motivated: builds on a clear and relevant research direction
 900 7.
 901 8. Strongly motivated: clearly addresses a known issue or opportunity from existing work or
 902 widely acknowledged field challenges
 903 9.
 904 10. Exceptionally motivated: makes a compelling case for a timely and important problem grounded
 905 in the reference list or broadly recognized research needs

906 **Explanation:** You should also provide a rationale for your score. (Your rationale should be at least 2-3
 907 sentences.) Also specify which part of the idea was most clearly linked to a literature-based motivation.

912 **8. Coherence of Idea Composition:** Are the combined components (problem, methods, tasks, and
 913 metrics) logically integrated and literature-informed?

914 1. Incoherent: parts don't fit together; lacks logical or conceptual connection
 915 2.
 916 3. Loosely connected: some rationale exists, but combination feels forced
 917 4.
 918 5. Reasonable fit: elements are compatible, though not deeply integrated
 919 6. Moderate coherence: combination makes general sense with limited justification
 920 7.
 921 8. Coherent and justified: combination makes sense and is literature-informed
 922 9.
 923 10. Highly coherent: seamless integration of ideas with strong literature basis

928 **Explanation:** You should also provide a rationale for your score. (Your rationale should be at least 2-3
 929 sentences.) *Hint: If the fit between components is strong, note which elements are well connected and
 930 clearly defined for implementation (e.g., problem-task pairing, method-metric match). If weak, specify
 931 which parts feel vague, disconnected, or hard to execute.*

932 **9. Specificity of Proposed Method:** How clearly does the proposed method present a testable research
 933 goal or hypothesis? To what extent is it sufficiently detailed to be operationalized in a way that aligns
 934 with prior literature or accepted practices?

935 1. Extremely unclear: the method is explained in an extremely vague or ambiguous manner,
 936 making it impossible to understand or replicate the approach without additional information or
 937 clarification.

938 2.

939 3. Unclear: the method is described with some detail, but significant gaps in explanation or logic
 940 leave the reader with considerable confusion and uncertainty about how to apply or replicate
 941 the approach.

942 4.

943 5. Somewhat clear: method is described with sufficient detail to understand the basic approach,
 944 but important elements remain vague or underdeveloped

945 6. Moderately clear: method is described with sufficient detail to understand the basic approach,
 946 but lacks the precision or specificity needed to fully replicate or grasp the nuances of the
 947 methodology without further guidance.

948 7.

949 8. Clear and testable: method is clearly and precisely described, with most details provided
 950 to allow for replication and comprehension, though minor areas may benefit from further
 951 clarification or elaboration.

952 9.

953 10. Highly clear and specific: method is articulated in an exceptionally clear, precise, and detailed
 954 manner, enabling straightforward replication and thorough understanding of the approach with
 955 no ambiguities

956

957

Explanation: You should also provide a rationale for your score. (Your rationale should be at least 2-3 sentences.)

10. Overall Score: Overall score: Apart from the above, you should also give an overall score for the idea on a scale of 1 - 10 as defined below (Major AI conferences in the descriptions below refer to top-tier NLP/AI conferences such as *ACL, COLM, NeurIPS, ICLR, and ICML.):

964 1. Critically flawed, trivial, or wrong, would be a waste of students' time to work on it

965 2. Strong rejection for major AI conferences

966 3. Clear rejection for major AI conferences

967 4. Ok but not good enough, rejection for major AI conferences

968 5. Decent idea but has some weaknesses or not exciting enough, marginally below the acceptance
 969 threshold of major AI conferences

970 6. Marginally above the acceptance threshold of major AI conferences

971 7. Good idea, would be accepted by major AI conferences

972 8. Top 50% of all published ideas on this topic at major AI conferences, clear accept

973 9. Top 15% of all published ideas on this topic at major AI conferences, strong accept

974 10. Top 5% of all published ideas on this topic at major AI conferences, will be a seminal paper

975

976

Overall Rationale: You should also provide a rationale for your overall score. (Your rationale should be at least 2-3 sentences.) *Hint: This is just an idea. Please evaluate its potential — assuming it is properly fleshed out, implemented, and empirically validated, would it be acceptable at a future major AI conference? If the idea is too vague to envision as a strong paper, it should be rated lower.*

11. Confidence: Additionally, we ask for your confidence in your review on a scale of 1 to 5 defined as following:

984 1. Your evaluation is an educated guess

985 2. You are willing to defend the evaluation, but it is quite likely that you did not understand central
 986 parts of the paper

987 3. You are fairly confident that the evaluation is correct

988 4. You are confident but not absolutely certain that the evaluation is correct

989

990 5. You are absolutely certain that the evaluation is correct and very familiar with the relevant
991 literature
992

993 **Time:** How many minutes did you spend on this task?
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044

1045 1046 1047 1048 1049 1050 1051 1052 1053 1054 1055	Baseline				HARPA			
	Dimension	Mean	Std	Min–Max	Median	Mean	Std	Min–Max
Coherence	7.20	1.65	3–10	8.0	7.08	1.65	1–10	8.0
Confidence	4.33	0.66	3–5	4.0	4.38	0.67	3–5	4.0
Effectiveness	6.78	1.66	3–10	7.0	6.65	1.69	1–9	7.0
Excitement	6.45	1.54	3–8	7.0	6.25	1.48	3–8	6.0
Familiarity	3.93	0.94	2–5	4.0	4.05	0.93	2–5	4.0
Feasibility	5.50	1.72	3–9	5.0	6.28	2.08	1–10	6.0
Grounding	5.98	1.94	2–10	6.0	6.83	1.47	3–10	7.0
Motivation	6.13	1.64	3–9	6.0	6.45	1.43	3–9	7.0
Novelty	6.43	1.32	3–8	6.0	5.98	1.33	2–8	6.0
Overall	6.20	1.71	3–8	7.0	6.43	1.69	2–9	7.0
Specificity	7.00	1.88	2–10	8.0	7.28	1.78	1–10	8.0

Table 6: Expert ratings across 11 dimensions. Values report mean, std, min–max, and median (10-point Likert scale, higher is better). $n = 40$ proposals per system.

B.1 STATISTICAL TESTS FOR HUMAN EVALUATION

For each dimension, we computed paired differences between HARPA and the baseline on expert ratings. Statistical significance was assessed using bootstrap resampling (10,000 iterations) and Wilcoxon signed-rank tests. We report bootstrap as our primary test, since it makes no distributional assumptions and is appropriate for small sample sizes and ordinal scores. Table 7 shows the mean differences and p -values.

Dimension	MeanDiff	Bootstrap_p	Boot*	Wilcoxon_p	Wilcoxon*
coherence_score	-0.125	0.666		0.806	
effectiveness_score	-0.125	0.663		0.716	
excitement_score	-0.200	0.753		0.435	
familiarity_score	0.125	0.210		0.394	
confidence_score	0.050	0.360		0.660	
feasibility_score	0.775	0.017	*	0.016	*
grounding_score	0.850	0.002	**	0.017	*
motivation_score	0.325	0.163		0.286	
novelty_score	-0.450	0.937		0.107	
overall_score	0.225	0.275		0.598	
specificity_score	0.275	0.168		0.430	

Table 7: Mean differences (HARPA – baseline) with significance tests. Stars indicate significance (* $p < 0.05$, ** $p < 0.01$). Bootstrap resampling is our primary test.

C FULL EXAMPLES OF EXPERT REVIEW AND PROPOSALS

Table 8 shows two representative pairs of hypotheses (HARPA vs. baseline), along with expert assessment across all evaluation dimensions. Each row corresponds to one proposal. Complete dataset generated and assessed for human evaluation are available in the supplementary files.

1100 We include an example of full proposal evaluated by experts and generated by HARPA and by the
1101 baseline ideator.
1102

1103 **HARPA PROPOSAL**
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154

1155

1156

1157

1158

1159

1160

Paper ID

1163

1164 3bfb5f836d944414c171f8f843eaf90cf5604243

1165

1166

1167

1168

1169

Title

1171

1172

1173 Combining stochastic softmax tricks with control variates for improved spanning tree
1174 optimization.

1175

1176

1177

1178

1179

Introduction

1181

1182

Problem Statement

1184

1185 Integrating stochastic softmax tricks with control variates will significantly improve
1186 convergence speed and stability in spanning tree optimization problems compared to
1188 using stochastic softmax tricks alone.

1189

1190

Motivation

1192

1193

1194 Existing methods for variance reduction in discrete optimization problems often focus
1195 on individual techniques like Rao-Blackwellization or stochastic softmax tricks in
1196 isolation. However, these approaches do not fully exploit the potential synergies
1197 between different variance reduction techniques, particularly in complex combinatorial
1198 spaces like spanning trees and arborescences. No prior work has explored the
1199 integration of stochastic softmax tricks with control variates specifically for spanning
1200 tree problems, which could offer significant improvements in convergence speed and
1203 stability by leveraging structured relaxations and variance reduction simultaneously.

1205

1206

1207

1208

1209

Proposed Method

1210
1211
1212 The research aims to explore the integration of stochastic softmax tricks with control
1213 variates to enhance variance reduction in spanning tree optimization problems.
1214
1215 Stochastic softmax tricks provide structured relaxations that allow for gradient
1216 estimation in combinatorial spaces, while control variates reduce the variance of these
1217 gradient estimators by incorporating additional information. By combining these two
1218 techniques, the hypothesis posits that the model will achieve faster convergence and
1219 more stable performance. This approach addresses the gap in existing research where
1220 these techniques are typically applied in isolation. The expected outcome is a more
1221 efficient optimization process, particularly in graph-based problems like spanning trees,
1222 where maintaining the graph structure is crucial for accurate gradient computation. This
1223 combination is expected to reduce the variance of gradient estimates more effectively
1224 than either technique alone, leading to improved model performance metrics such as
1225 convergence speed and stability.
1226
1227

Background

1231
1232
1233
1234
1235 [Stochastic Softmax Tricks](#): Stochastic softmax tricks are used to create structured
1236 relaxations for combinatorial optimization problems, such as spanning trees. This
1237 involves using the Gumbel-Max trick to reparameterize distributions over one-hot binary
1238 vectors, allowing for gradient estimation in discrete distributions. The structured
1239 relaxation maintains the graph structure, enabling efficient gradient computation. This
1240 technique was selected for its ability to handle complex combinatorial spaces and its
1241 compatibility with gradient-based optimization methods.
1242
1243

1244 [Control Variates](#): Control variates are used to reduce the variance of gradient
1245 estimators by incorporating additional information into the estimation process. This
1246 involves constructing a control variate based on an analytical linear approximation to
1247 the gradient estimator, which is then combined with a naïve gradient estimate. This
1248 method remains unbiased while achieving lower variance, particularly effective in
1249 Gaussian approximating families. The control variate is expected to enhance the
1250 efficiency of the stochastic softmax tricks by further reducing the variance of the
1251 gradient estimates.
1252
1253

Implementation

1254
1255
1256 The proposed method involves integrating stochastic softmax tricks with control variates
1257 to optimize spanning tree problems. First, the stochastic softmax tricks are applied to
1258
1259

1265 create a structured relaxation of the spanning tree problem, allowing for gradient
1266 estimation in a differentiable manner. This is achieved by representing the problem as a
1267 linear program and applying a softmax function to approximate the selection of edges.
1268 Next, control variates are introduced to further reduce the variance of the gradient
1269 estimators. This involves constructing a control variate based on an analytical linear
1270 approximation to the gradient estimator, which is then combined with the gradient
1271 estimates obtained from the stochastic softmax tricks. The integration occurs at the
1272 gradient computation stage, where the control variate is used to adjust the gradient
1273 estimates, leading to lower variance and improved convergence. The data flows from
1274 the structured relaxation to the control variate adjustment, with the final output being a
1275 more stable and efficient gradient estimate. This method is implemented using libraries
1276 that support automatic differentiation, such as TensorFlow or PyTorch, and is evaluated
1277 against baseline methods like the vanilla Gumbel-Softmax estimator.
1278
1279
1280
1281
1282
1283
1284

1285 1286 **Experiments Plan**

1287 1288 **Operationalization Information**

1289
1290 Please implement an experiment to test
1291 the hypothesis that integrating stochastic
1292 softmax tricks with control variates will
1293 significantly improve convergence speed
1294 and stability in spanning tree optimization
1295 problems compared to using stochastic
1296 softmax tricks alone.
1297
1298

1299 1300 **Experiment Overview**

1301 This experiment will compare three
1302 methods for spanning tree optimization:
1303 1. **Baseline 1:** Vanilla Gumbel-Softmax
1304 estimator
1305 2. **Baseline 2:** Stochastic softmax tricks
1306 without control variates
1307 3. **Experimental:** Stochastic softmax
1308 tricks integrated with control variates
1309
1310

1311 1312 **2. Stochastic Softmax Tricks without 1313 Control Variates (Baseline 2)**

1314 Implement structured relaxations for
1315 spanning trees:

- Represent the spanning tree polytope
1316 using the cycle constraints
- Apply stochastic softmax tricks to
1317 maintain the graph structure
- Use automatic differentiation to
1318 compute gradients
- Implement a projection step to ensure
1319 the solution is a valid spanning tree

1320 1321 **3. Integrated Approach with Control 1322 Variates (Experimental)**

1320 The experiment should measure
 1321 convergence speed (iterations to reach a
 1322 predefined accuracy threshold) and
 1323 stability (variance of predictions across
 1324 different runs) for each method.
 1325

1327 **Implementation Details**

1330 **Pilot Mode Settings**

1332
 1333 Implement a global variable PILOT_MODE
 1334 with three possible settings: MINI_PILOT,
 1335 PILOT, or FULL_EXPERIMENT.
 1336
 1337 - **MINI_PILOT**: Use 5 small random
 1338 graphs (10-15 nodes) and run 10
 1339 optimization iterations with 3 independent
 1340 runs per method
 1341
 1342 - **PILOT**: Use 20 medium-sized random
 1343 graphs (20-50 nodes) and run 50
 1344 optimization iterations with 10
 1345 independent runs per method
 1346
 1347 - **FULL_EXPERIMENT**: Use 100 graphs
 1348 of varying sizes (up to 100 nodes) and
 1349 run 200 optimization iterations with 30
 1350 independent runs per method
 1351
 1352

1353 The experiment should first run in
 1354 MINI_PILOT mode, then PILOT mode if
 1355 successful, but stop before
 1356
 1357 **FULL_EXPERIMENT** (which will be
 1358 manually triggered after human
 1359 verification).
 1360

1365 **Graph Dataset Generation**

1366
 1367 Use NetworkX to generate the following
 1368 types of random graphs for the
 1369 experiment:
 1370

1371 1. Erdős–Rényi random graphs
 1372

1373
 1374 Extend the stochastic softmax tricks
 1375 implementation with control variates:
 1376 - Construct a control variate based on an
 1377 analytical linear approximation to the
 1378 gradient estimator
 1379 - Combine the control variate with the
 1380 naïve gradient estimate from the
 1381 stochastic softmax tricks
 1382 - Implement the optimal scaling
 1383 parameter for the control variate
 1384 - Apply the adjusted gradient in the
 1385 optimization process

Optimization Task

Implement a minimum spanning tree
 optimization task where the objective is
 to find the spanning tree with minimum
 total edge weight. Additionally, implement
 a maximum spanning tree task as a
 secondary objective.

Evaluation Metrics

1. **Convergence Speed**: Measure the number of iterations required to reach 95% of the optimal solution
2. **Stability**: Calculate the variance of the solutions across multiple independent runs
3. **Solution Quality**: Compare the final solution to the true optimal spanning tree (computed using standard MST algorithms)
4. **Gradient Variance**: Measure the variance of the gradient estimates during optimization

Experiment Procedure

```

2. Barabási–Albert preferential
1375 attachment graphs
1376
1377 3. Watts–Strogatz small-world graphs
1378
1379 For each graph, assign random edge
1380 weights from a uniform distribution [0.1,
1381 10.0].
1382
1383
1384
1385 Method Implementations
1386
1387 1. Vanilla Gumbel-Softmax Estimator
1388 (Baseline 1)
1389
1390
1391 Implement the standard Gumbel-Softmax
1392 trick for spanning tree optimization:
1393 - Represent the graph as an edge
1394 selection problem
1395 - Apply the Gumbel-Max trick to sample
1396 spanning trees
1397 - Use the softmax temperature parameter
1398 to control the discreteness of the
1399 distribution
1400 - Implement straight-through estimation
1401 for the backward pass
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429

```

Summary

1. For each graph in the dataset:
 - a. Run each method (Baseline 1, Baseline 2, Experimental) multiple times with different random seeds
 - b. Record the optimization trajectory (objective value vs. iteration)
 - c. Measure the gradient variance at each iteration
 - d. Calculate the final solution quality
1. Aggregate results across all graphs and runs:
 - a. Calculate average convergence speed for each method
 - b. Calculate average stability for each method
 - c. Perform statistical significance tests (bootstrap resampling) to compare methods

Visualization and Reporting

1. Generate convergence plots showing objective value vs. iteration for each method
2. Create box plots showing the distribution of convergence speeds and stability metrics
3. Generate tables with summary statistics for each method
4. Visualize example spanning trees produced by each method on selected graphs

Implementation Notes

- Use PyTorch for automatic differentiation and gradient computation
- Use NetworkX for graph manipulation and visualization
- Implement proper seeding for reproducibility
- Use a learning rate scheduler to improve convergence
- Save intermediate results to allow for experiment resumption

Expected Output

The experiment should produce:

1. A comprehensive report comparing the three methods
2. Convergence plots for each method
3. Statistical analysis of the differences between methods
4. Visualizations of example spanning trees
5. Raw data for further analysis

Please implement this experiment and run it first in MINI_PILOT mode, then in PILOT mode if successful. Do not proceed to FULL_EXPERIMENT mode without human verification.

End Note:

1476 The source paper is [Paper 0: Learning with Differentiable Perturbed Optimizers \(109](#)
 1477 citations, 2020). This idea draws upon a trajectory of prior work, as seen in the following
 1478 sequence: [Paper 1](#) --> [Paper 2](#) --> [Paper 3](#) --> [Paper 4](#) --> [Paper 5](#). The analysis
 1480 reveals a consistent theme of addressing the high variance in gradient estimation for
 1481 discrete latent variables, a challenge initially highlighted in the source paper. The
 1482

1483

1484

1485 progression of research has introduced various techniques like stochastic softmax
1486 tricks, Rao-Blackwellization, and coupled gradient estimators to tackle this issue.
1487 However, these approaches often focus on specific applications or settings, such as
1488 combinatorial spaces or categorical variables. A novel research idea could involve
1489 developing a generalized framework that unifies these variance reduction techniques,
1490 making them adaptable to a broader range of discrete optimization problems. This
1491 would advance the field by providing a more versatile tool for training models with
1492 discrete components, addressing the limitations of existing methods that are often
1493 application-specific.
1494

1495 The initial trend observed from the progression of related work highlights a consistent
1496 research focus. However, the final hypothesis proposed here is not merely a
1497 continuation of that trend — it is the result of a deeper analysis of the hypothesis space.
1498 By identifying underlying gaps and reasoning through the connections between works,
1499 the idea builds on, but meaningfully diverges from, prior directions to address a more
1500 specific challenge.
1501

1502 References

- 1513 1. [Learning with Differentiable Perturbed Optimizers](#) (2020)
- 1514 2. [Gradient Estimation with Stochastic Softmax Tricks](#) (2020)
- 1515 3. [Rao-Blackwellizing the Straight-Through Gumbel-Softmax Gradient Estimator](#)
1516 (2020)
- 1517 4. [Coupled Gradient Estimators for Discrete Latent Variables](#) (2021)
- 1518 5. [Training Discrete Deep Generative Models via Gapped Straight-Through](#)
1519 [Estimator](#) (2022)
- 1520 6. [Revisiting the Gumbel-Softmax in MADDPG](#) (2023)
- 1521 7. [Rao-Blackwellized Stochastic Gradients for Discrete Distributions](#) (2018)
- 1522 8. [Tackling Data Heterogeneity: A New Unified Framework for Decentralized SGD](#)
1523 [with Sample-induced Topology](#) (2022)
- 1524 9. [A generalized approximate control variate framework for multifidelity uncertainty](#)
1525 [quantification](#) (2018)
- 1526 10. [Adaptive Variance Reduction for Stochastic Optimization under Weaker](#)
1527 [Assumptions](#) (2024)
- 1528 11. [On Variance Reduction in Stochastic Gradient Descent and its Asynchronous](#)
1529 [Variants](#) (2015)



1540 12. [Partial Variance Reduction improves Non-Convex Federated learning on](#)
1541 [heterogeneous data](#) (2022)

1542 13. [Improved Analysis and Rates for Variance Reduction under Without-replacement](#)
1543 [Sampling Orders](#) (2021)

1544 14. [Stochastic Variance Reduction Methods for Policy Evaluation](#) (2017)

1545 15. [Reducing Reparameterization Gradient Variance](#) (2017)

1546

1547

1548

1549

1550

1551

1552

1553

1554

1555

1556

1557

1558

1559

1560

1561

1562

1563

1564

1565

1566

1567

1568

1569

1570

1571

1572

1573

1574

1575

1576

1577

1578

1579

1580

1581

1582

1583

1584

1585

1586

1587

1588

1589

1590

1591

1592

1593

1594

1595 **BASELINE PROPOSAL**

1596

1597

1598

1599

1600

1601

1602

1603

1604

1605

1606

1607

1608

1609

1610

1611

1612

1613

1614

1615

1616

1617

1618

1619

1620

1621

1622

1623

1624

1625

1626

1627

1628

1629

1630

1631

1632

1633

1634

1635

1636

1637

1638

1639

1640

1641

1642

1643

1644

1645

1646

1647

1648

1649

1650

1651

1652

1653

1654

1655

1656

1657 **Paper ID**

1658

1659

1660 3bfb5f836d944414c171f8f843eaf90cf5604243

1661

1662

1663

1664

1665 **Title**

1666

1667

1668 Adaptive Stochastic Gradient Clipping: Enhancing Stability and Convergence in Deep
1669 Learning Pipelines

1671

1672

1673

1674

1675 **Introduction**

1676

1679 **Problem Statement**

1680

1681 Gradient-based optimization in deep learning often suffers from instability and slow
1682 convergence, especially in complex decision-making pipelines where gradients can
1683 become extremely large or vanishingly small. This issue can lead to poor model
1684 performance, slow training, and difficulties in fine-tuning models for specific tasks.
1685

1687

1688

1689 **Motivation**

1690

1691 Existing methods like fixed gradient clipping, adaptive learning rates, and normalization
1692 techniques often struggle to balance stability and convergence speed across different
1693 layers and tasks within a pipeline. Inspired by the success of noise injection in
1695 improving generalization and the adaptive nature of biological neural systems, we
1696 propose a method that dynamically adjusts gradient updates based on local statistics
1698 and stochastic perturbations. This approach allows for aggressive updates in stable
1700 regions while dampening oscillations in sensitive areas, all while introducing beneficial
1701 noise for improved exploration and generalization.
1703

1704

1705

1706

Proposed Method

1707

1708

1709 We introduce Adaptive Stochastic Gradient Clipping (ASGC), which combines layer-
1710 wise gradient statistics with controlled stochastic perturbations. For each layer, we
1711 maintain running estimates of gradient mean and variance. During each update, we
1712 compute a clipping threshold as a function of these statistics. Before applying the
1713 threshold, we add Gaussian noise scaled by the layer's gradient variance. The clipping
1714 function is smoothed using a differentiable approximation, allowing end-to-end training.
1715 The noise scale and clipping function parameters are meta-learned across a diverse set
1716 of tasks.
1717

1718

1719

1720

1721

1722

1723

1724

1725

Experiments Plan

1726

1727

1728

1729

Step-by-Step Experiment Plan

1730

1731

Step 1: Implement ASGC

1732

1733

1734 Implement the ASGC algorithm as a PyTorch optimizer. This involves creating a custom
1735 optimizer class that inherits from `torch.optim.Optimizer` and overrides the `step()` method.
1736 The key components are: (1) Maintaining running estimates of gradient mean and
1737 variance for each layer. (2) Computing the adaptive clipping threshold. (3) Adding
1738 scaled Gaussian noise to the gradients. (4) Applying the smoothed clipping function. (5)
1739 Updating the parameters using the clipped and noisy gradients.
1740

1741

1743

Step 2: Prepare Datasets

1744

1745

1746 Prepare the following datasets for evaluation: (1) ImageNet for image classification. (2)
1747 WMT14 English-German for machine translation. (3) Atari suite (specifically Breakout,
1748 Pong, and Space Invaders) for reinforcement learning.
1749

1750

1751

Step 3: Setup Baseline Models

1752

1753 Implement baseline models for each task: (1) ResNet-50 for ImageNet. (2) Transformer
1754 for WMT14. (3) DQN for Atari games. Train these models using standard optimizers:
1755 Adam, SGD with momentum, Adagrad, and RMSprop.
1756

1757

1758

1759

Step 4: Train Models with ASGC

1760
1761 Train the same model architectures using ASGC. Use a grid search to find optimal
1762 hyperparameters for ASGC, including the initial noise scale and clipping function
1763 parameters.
1764
1765

Step 5: Evaluate Performance

1766
1767 Compare ASGC against baselines on the following metrics: (1) Final test
1768 accuracy/BLEU score/game score. (2) Training time to reach a specific performance
1769 threshold. (3) Stability of training (measured by the variance of validation performance
1770 across epochs). (4) Generalization (measured by the gap between training and test
1771 performance).
1772
1773

Step 6: Analyze Robustness

1774
1775 Evaluate the robustness of ASGC to hyperparameter choices by training models with
1776 randomly sampled hyperparameters and comparing the distribution of final
1777 performances against baselines.
1778
1779

Step 7: Visualize Gradient Statistics

1780
1781 Plot the distribution of gradient magnitudes before and after clipping for different layers
1782 and at different stages of training. Compare these distributions between ASGC and
1783 baseline optimizers.
1784
1785

Step 8: Analyze Meta-Learned Parameters

1786
1787 Examine the learned noise scales and clipping function parameters across different
1788 tasks and model architectures. Visualize how these parameters evolve during training.
1789
1790

Step 9: Ablation Studies

1791
1792 Conduct ablation studies to isolate the effects of adaptive clipping and stochastic
1793 perturbations. Train models with only adaptive clipping (no noise) and only stochastic
1794 perturbations (fixed clipping threshold).
1795
1796

Step 10: Write Up Results

1797
1798 Compile all results, visualizations, and analyses into a comprehensive report or paper
1799 draft.
1800
1801

1802

1803

1804

1805

1806

1807

1808

1809

1810

1811

1812

1813

1814

Test Case Examples

1815

1816

Baseline Prompt Input

1818

1819 Train a ResNet-50 model on ImageNet using Adam optimizer with default
1820 hyperparameters.
1821

1822

Baseline Prompt Expected Output

1824

1825 Final Top-1 Accuracy: 76.1%, Training Time: 90 hours, Stability (std dev of validation
1826 accuracy over last 10 epochs): 0.5%
1827

1828

Proposed Prompt Input

1830

1831 Train a ResNet-50 model on ImageNet using ASGC optimizer with meta-learned
1832 hyperparameters.
1833

1834

Proposed Prompt Expected Output

1836

1837 Final Top-1 Accuracy: 77.3%, Training Time: 85 hours, Stability (std dev of validation
1838 accuracy over last 10 epochs): 0.3%
1840

Explanation

1842

1843 ASGC achieves higher accuracy in less training time, with improved stability during the
1844 final stages of training. This demonstrates the benefits of adaptive clipping and
1845 stochastic perturbations in balancing aggressive updates and stability.
1847

1848

Fallback Plan

1851

1852 If ASGC does not outperform baselines as expected, we can pivot the project to an in-
1853 depth analysis of why adaptive stochastic methods struggle in certain scenarios. We
1854 would conduct a series of experiments to isolate the effects of gradient clipping, noise
1855 injection, and adaptive thresholds on different types of neural architectures and tasks.
1857 This could involve visualizing gradient flow through networks, analyzing the spectrum of
1859 the Hessian at different stages of training, and studying how different optimization
1860 techniques affect the loss landscape. We could also explore combining ASGC with
1861 other advanced optimization techniques like layer-wise adaptive rates or Hessian-based
1863 preconditioning. The goal would be to provide insights into the interplay between
1865 network architecture, task complexity, and optimization dynamics, potentially informing
1866 the development of next-generation optimization algorithms.
1868

1869

1870

1871

References

1872

1873

1874

1. [The Limited Multi-Label Projection Layer](#) (2019)
2. [Optimizing Rank-Based Metrics With Blackbox Differentiation](#) (2019)
3. [Categorical Reparameterization with Gumbel-Softmax](#) (2016)
4. [Differentiable Top-k Operator with Optimal Transport](#) (2020)
5. [Tackling Prevalent Conditions in Unsupervised Combinatorial Optimization: Cardinality, Minimum, Covering, and More](#) (2024)
6. [Differentiable Combinatorial Scheduling at Scale](#) (2024)
7. [Fast Differentiable Sorting and Ranking](#) (2020)
8. [Differentiation of Blackbox Combinatorial Solvers](#) (2019)
9. [Deep Network Flow for Multi-object Tracking](#) (2017)
10. [Learning Latent Trees with Stochastic Perturbations and Differentiable Dynamic Programming](#) (2019)

1890

1891

1892

1893

1894

1895

1896

1897

1898

1899

1900

1901

1902

1903

1904

1905

1906

1907

1908

1909

1910

1911

1912

1913

1914

1915

1916

1917

1918

1919

1920

1921

1922

1923

1924

System		Hypothesis/Problem Statement, Ratings, and Justification
1925	HARPA	<p><i>“Integrating stochastic softmax tricks with control variates will significantly improve convergence speed and stability in spanning tree optimization problems compared to using stochastic softmax tricks alone.”</i></p> <p>Novelty = 6 (“The following paper is a neurips 2020 paper that has spanning tree optimization as an application: “Gradient Estimation with Stochastic Softmax Tricks ””. It mainly uses softmax trick for some discrete problems such as spanning tree optimization. The main novelty comes to add the control variates into the loop.”)</p> <p>Feasibility = 7 (“Using the softmax trick allows backpropagation/gradient estimation, it is a well known trick and the implementation is not so complicated, although the control variates is not so clear how would be implemented (with examples) ”)</p> <p>Grounding = 9 (“It is very grounded on the listed references, mostly similar to “Gradient Estimation with Stochastic Softmax Tricks (2020)”. The control variates ideas, although slightly mentioned in the same paper, are more frequent described as “A generalized approximate control variate framework for multifidelity uncertainty quantification (2018) ””)</p> <p>Specificity = 6 (“I feel the control variates part is not so clear how it would be implemented. What is the additional variable that is correlated to the target? what would be the target in that case? I can see the motivation for that, but it is not so clear how it would be implemented. Examples would be appreciated. ”)</p> <p>Coherence = 8 (“It is very clear that softmax trick is for gradient estimation and control variates is where the novelty is, to improve efficiency. So I see it is coherent. ”)</p> <p>Motivation = 6 (“The motivation only comes from adding control variates to speedup convergence to the already existing methods using “stochastic softmax tricks”. I can’t see anything else regarding motivation. ”)</p> <p>Excitement = 5 (“I would say it is not exciting due to the lack of novelty (compared to the given neurips paper in 2020). The experiments are also vanilla (mostly ablation studies). The experiment overview is basically removing the control variates and the softmax to compare with the method with both. ”)</p> <p>Effectiveness = 6 (“It is very clear that softmax trick is for gradient estimation and control variates is where the novelty is, to improve efficiency. So I see it is coherent. ”)</p> <p>Overall = 6 (“The motivation of the proposed method is clear, increase efficiency when bringing the control variates into the loop. However, details on how to incorporate the control variate ideas, which is the core of the novelty, are not so clear. ”)</p> <p>Confidence = 4</p>
1945	Baseline	<p><i>“Gradient-based optimization in deep learning often suffers from instability and slow convergence, especially in complex decision-making pipelines where gradients can become extremely large or vanishingly small. This issue can lead to poor model performance, slow training, and difficulties in fine-tuning models for specific tasks.”</i></p> <p>Novelty = 5 (“From one perspective, I don’t score high the novelty regarding this proposal since this would depend on some details that are not explicitly mentioned. Example: “During each update, we compute a clipping threshold as a function of these statistics”. How exactly those statistics will be used would impact on the novelty. For example, Adam optimizer also use statistics for updating gradients. However, the overall method has its novelty characteristics by combining the statistics with stochastic perturbations.”)</p> <p>Feasibility = 4 (“The proposed method looks feasible. The problem is that it lacks details. Everything related to the method is summarized in 2 or 3 lines in the “Proposed methods” without any mathematical language. How would we smooth the clipping function, for example?”)</p> <p>Grounding = 2 (“The proposal reference list is not linked to the proposed methods. Soe papers I am really aware of have nothing to do with the method proposed. For example, “Differentiation of Blackbox Combinatorial Solvers (2019)” is specifically about imitation learning of combinatorial labels, there is no novelty towards unconstrained optimizers.”)</p> <p>Specificity = 2 (“As I mentioned before, the details are the problem in this proposal. There is no details of the proposed methods, and therefore the specificity is extremely unclear.”)</p> <p>Coherence = 6 (“The proposed method is very weak and lack of important details. The experimental plan looks all correct, but they are not so important with respect to the method itself. e, it is obvious from the Proposed metod that it is a new optimizer, and then the step 1 of experimental plan it just repeat the steps without detailing it.”)</p> <p>Motivation = 6 (“The section of motivation makes sense, although not grounded in the list of papers.”)</p> <p>Excitement = 7 (“It is definitely exciting in a sense that the impact of this type of research is extremely high, since it can substitute, for example, specific pytorch optimizers that are widely used (for example, Adam) and sometimes suffer from convergences difficulties depending on the architecture used (for example, RNNs).”)</p> <p>Effectiveness = 3 (“The main problem of this research proposal is the lack of details in the proposed method subsection. I still don’t see how the parts that matter would be really implemented, such as the clipping part. And what is end-to-end? What is optimized for the adaptation?”)</p> <p>Overall = 3 (“The main problem of this proposal is the lack of details. The method itself, in a high-level idea, makes sense. But the “how” is extremely unclear. There is no depth of the methodology. The ideas ends up in the high-level buzzwords.”)</p> <p>Confidence = 4</p>

Table 8: Representative HARPA vs. baseline hypotheses with expert assessment.

D HARPA-SCORER

D.1 HARPA-SCORER: ADDITIONAL EXPERIMENT DETAILS

Implementation details. We follow almost the same hyperparameters as the RM-R1 pipeline (Chen et al., 2025c), consisting of two stages: (i) reasoning distillation from oracle rubric-style traces, and (ii) RLVR fine-tuning on execution-derived preference pairs. The backbone is Qwen2.5-7B-Instruct, trained with openrlhf and DeepSpeed using full fine-tuning.

Distillation Stage. We fine-tuned on 3,459 rubric-aligned preference pairs (Section 4.3), split into 2,595 train, 452 validation, and 412 test. Training used a global batch size of 4 (micro-batch size 1), maximum sequence length 12,288, and Adam optimizer with offloading at a learning rate of 5×10^{-6} .

1980 We trained for 1 epoch in bfloat16 precision with ZeRO stage-2 optimization, gradient checkpointing,
 1981 FlashAttention, and sample packing. Training was performed on 4× NVIDIA A100-SXM4-80GB.
 1982

1983 **RLVR Stage.** We further optimized the distilled model with RLVR on execution-derived success/failure
 1984 pairs, using the ver1 PPO trainer. Training was performed on 8×NVIDIA A100-SXM4-80GB GPUs.
 1985 We set the rollout batch size to 64, PPO mini-batch size to 16, and micro-batch size per GPU to 4. The
 1986 learning rate was 1×10^{-6} with constant warmup. Maximum input and output lengths were both 8,192
 1987 tokens. KL regularization was applied with a coefficient of 10^{-3} and a clip ratio of 0.2, with entropy
 1988 coefficient set to 0. GPU memory utilization was capped at 0.5 to prevent OOM issues. Sampling used
 1989 temperature 1.0 and top-p 1.0. Training ran for a single epoch, with checkpoints saved every 1,000 steps.
 1990

1991 **Compute Cost.** The distillation stage required 3h08m of training on 4×A100-80GB GPUs, and the
 1992 RLVR stage required 8d10h on 8×A100-80GB GPUs. Constructing the scorer’s training dataset is the
 1993 dominant cost, since each supervision signal requires a full ASD-agent execution (code synthesis and
 1994 experiment run), as described in Section 4.3.
 1995

1996
 1997

1998 Example reasoning trace

2000 **<rubric>**

- 2001 1. Execution Success:
 - Proposal A: 0.9 - Well-structured multilingual toxicity detection experiment
 - ... with clear baseline/experimental comparison, proper statistical analysis, and
 - ... realistic implementation scope
 - Proposal B: 0.7 - Complex RAG system comparison with multiple components that
 - ... may face integration challenges and dataset loading issues
- 2002 2. Complexity:
 - Proposal A: 0.3 - Moderate complexity with toxicity classification, adversarial
 - ... generation, and iterative improvement but manageable scope
 - Proposal B: 0.7 - High complexity requiring multiple RAG architectures,
 - ... real-time retrieval, adaptive mechanisms, and complex integration
- 2003 3. Cost Efficiency:
 - Proposal A: 0.8 - Efficient use of pre-trained models for toxicity detection
 - ... with limited language scope in pilot mode
 - Proposal B: 0.6 - Multiple model comparisons and complex retrieval operations
 - ... likely to consume significant computational resources
- 2004 4. Expected Hypothesis Validity:
 - Proposal A: 0.6 - Clear metrics (compliance ratio, adversarial accuracy drop)
 - ... but dynamic adversarial integration may not necessarily improve performance
 - Proposal B: 0.7 - Well-defined metrics (BLEU/ROUGE scores) with logical
 - ... expectation that adaptive architectures should improve performance
- 2005 5. Expected Interestingness:
 - Proposal A: 0.8 - Novel integration of dynamic data collection with adversarial
 - ... evaluation in multilingual safety context
 - Proposal B: 0.7 - Interesting combination of real-time and adaptive retrieval
 - ... but more incremental advancement
- 2006 6. Faithfulness:
 - Proposal A: 0.8 - Clear operationalization with realistic simulation of
 - ... human-in-the-loop processes
 - Proposal B: 0.6 - Complex integration requirements may lead to simplified
 - ... implementations that deviate from intended design

2007 **<justify>**

2035
 2036 Proposal A scores higher on execution success due to its more straightforward
 2037 ... implementation using established toxicity detection methods, while Proposal B's
 2038 ... multiple complex components increase failure risk. Proposal A is less complex,
 2039 ... focusing on a single domain (toxicity detection) with clear iterative
 2040 ... improvement, whereas Proposal B requires implementing multiple RAG architectures
 2041 ... with real-time and adaptive components. Cost efficiency favors Proposal A due to
 2042 ... its focused scope and efficient use of pre-trained models. Both proposals have
 2043 ... reasonable hypothesis validity, but Proposal B's expectation of improvement from
 2044 ... adaptive architectures is slightly more logical. Proposal A offers higher
 2045 ... interestingness due to its novel approach to multilingual safety alignment.
 2046 ... Faithfulness is higher for Proposal A because its human-in-the-loop simulation
 2047 ... is more realistic than Proposal B's complex integration requirements.
 2048
 2049 </justify>
 2050 </rubric>
 2051
 2052 <eval>
 2053 Both proposals address benchmark creation and evaluation but take different
 2054 ... approaches.
 2055
 2056 <summary_A>Proposal A focuses on multilingual toxicity detection, comparing a
 2057 ... baseline system using static data against an experimental system that integrates
 2058 ... dynamic data collection with adversarial evaluation</summary_A>. <quote_A>"The
 2059 ... experiment should compare a baseline system against an experimental system, with
 2060 ... appropriate metrics to measure performance differences"</quote_A> and uses
 2061 ... <quote_A>"compliance ratios in multilingual toxicity detection tasks and
 2062 ... adversarial accuracy drop"</quote_A> as key metrics.
 2063
 2064 <summary_B>Proposal B targets retrieval-augmented generation systems, comparing
 2065 ... three configurations: baseline RAG, real-time RAG, and dynamic RAG that combines
 2066 ... both real-time retrieval and adaptive architectures</summary_B>. <quote_B>"This
 2067 ... experiment will compare three RAG system configurations: 1. Baseline RAG: A
 2068 ... standard RAG system with static retrieval strategies 2. Real-time RAG: A RAG
 2069 ... system with real-time information retrieval 3. Dynamic RAG (Experimental): A RAG
 2070 ... system that integrates both real-time information retrieval and adaptive
 2071 ... retrieval architectures"</quote_B>.
 2072
 2073 In terms of implementation complexity, Proposal A presents a more manageable scope.
 2074 ... <quote_A>"Use 100 examples per language from 5 languages (English, Spanish,
 2075 ... French, German, Chinese) for training and 50 examples per language for testing.
 2076 ... Run 5 iterations of the dynamic data collection process"</quote_A> in pilot
 2077 ... mode, with clear simulation strategies: <quote_A>"Since actual human annotators
 2078 ... aren't available, simulate them by: 1. Creating a function that identifies model
 2079 ... errors and generates challenging examples 2. Implementing different 'annotator
 2080 ... personas' with varying expertise"</quote_A>.
 2081
 2082 Proposal B involves significantly more complex integration requirements.
 2083 ... <quote_B>"Implement a module that: - Dynamically accesses and integrates
 2084 ... external knowledge bases during generation - Updates retrieval strategies based
 2085 ... on the current generation state - Monitors the generation process to identify
 2086 ... when new information is needed"</quote_B> and <quote_B>"Implement an
 2087 ... architecture that: - Dynamically adjusts retrieval strategies based on task
 2088 ... complexity and information needs - Uses a feedback mechanism to evaluate the
 2089 ... quality of retrieved documents - Maintains multiple retrieval
 2089 ... strategies"</quote_B>.
 2089

```

2090
2091 The evaluation metrics are well-defined in both cases, but Proposal A's metrics are
2092 ... more directly measurable: <quote_A>"Compliance Ratio: Calculate the proportion
2093 ... of model outputs that adhere to safety guidelines" and "Adversarial Accuracy
2094 ... Drop: Measure the decrease in model performance when exposed to adversarial
2095 ... inputs"</quote_A>. Proposal B uses standard NLG metrics: <quote_B>"BLEU score
2096 ... between generated text and reference text" and "ROUGE scores (ROUGE-1, ROUGE-2,
2097 ... ROUGE-L)"</quote_B>.
2098 </eval>
2099 <answer>[[A]]</answer>
2100

```

Listing 1: Example of rubric-style reasoning trace given the proposal pairs and their execution metadata from an oracle model

Example extracted execution trace

```

2103 {
2104   "summary_results": {
2105     "summary": "This experiment tested whether dynamic sparsification tailored to
2106     ... syntactic complexity of queries and semantic density of documents would improve
2107     ... retrieval speed and accuracy compared to static sparsification methods on the
2108     ... MSMARCO dataset. The experiment used 200 queries with 730 documents in pilot
2109     ... mode. The baseline method employed static TF-IDF with BM25 scoring using a fixed
2110     ... 1000 features, while the experimental method dynamically adjusted the number of
2111     ... features (500-1500) based on query complexity scores and applied density-aware
2112     ... similarity weighting. However, the results showed identical performance between
2113     ... both methods: both achieved exactly the same MRR (0.5384), Precision@1 (0.255),
2114     ... and Precision@5 (0.071). The only difference was processing time, with the
2115     ... dynamic method being slower (4.01ms vs 2.41ms). Statistical significance testing
2116     ... using bootstrap resampling with 10,000 iterations yielded p-values of 1.0 for
2117     ... all metrics, indicating no significant differences. The experiment was
2118     ... well-implemented and faithful to the original design, successfully calculating
2119     ... query syntactic complexity using NLTK-based metrics and document semantic
2120     ... density using LLM embeddings for the first 100 documents. The lack of
2121     ... improvement suggests that either the dynamic sparsification approach needs
2122     ... refinement, the complexity/density metrics were not sufficiently discriminative,
2123     ... or the MSMARCO dataset characteristics did not benefit from this particular
2124     ... adaptive approach.",
2125     "summary_short": "Dynamic sparsification showed no improvement over static methods
2126     ... on MSMARCO retrieval task.",
2127     "summary_medium": "The experiment compared dynamic sparsification (adapting features
2128     ... based on query complexity and document density) against static TF-IDF/BM25 on
2129     ... 200 MSMARCO queries. Both methods achieved identical performance with
2130     ... MRR=0.5384, Precision@1=0.255, and Precision@5=0.071, with p-values of 1.0
2131     ... indicating no significant differences. The dynamic method was slower (4.01ms vs
2132     ... 2.41ms) without providing any accuracy benefits, suggesting the adaptive
2133     ... approach did not improve retrieval performance.",
2134     "hypothesis": "Dynamic sparsification tailored to syntactic complexity of queries
2135     ... and semantic density score of documents will significantly improve retrieval
2136     ... speed and accuracy compared to static sparsification methods.",
2137     "hypothesis_operationalized": "A TF-IDF based retrieval system that dynamically
2138     ... adjusts the number of features (500-1500) based on query syntactic complexity
2139     ... scores and applies density-aware similarity weighting will outperform a static
2140     ... TF-IDF system with fixed 1000 features on MSMARCO retrieval tasks, as measured
2141     ... by MRR, Precision@k, and processing time.", "hypothesis_inference": "The
2142     ... experimental results clearly reject the hypothesis. Both the dynamic and static
2143     ... sparsification methods achieved identical retrieval accuracy metrics (MRR,
2144     ... Precision@1, Precision@5), with statistical significance testing yielding
2145     ... p-values of 1.0, indicating no meaningful difference. Furthermore, the dynamic
2146     ... method was slower (4.01ms vs 2.41ms), contradicting the speed improvement
2147     ... hypothesis. The results suggest that the proposed dynamic sparsification
2148     ... approach, at least as implemented, does not provide benefits over static methods
2149     ... for this task and dataset."
2150

```

```

2145
2146 "hypothesis_category": "reject", "faithfulness_details": "The experiment was
2147 ... largely faithful to the original design. It successfully implemented both
2148 ... baseline (static TF-IDF/BM25) and experimental (dynamic sparsification) methods,
2149 ... calculated query syntactic complexity using NLTK-based metrics (parse tree
2150 ... depth, POS diversity, lexical diversity), and computed document semantic density
2151 ... using LLM embeddings. The experiment used appropriate evaluation metrics (MRR,
2152 ... Precision@k, processing time) and statistical testing (bootstrap resampling).
2153 ... However, there were some practical limitations: semantic density calculation was
2154 ... limited to the first 100 documents to control costs and time, and some documents
2155 ... used fallback density scores of 0.5. The pilot mode with 200 queries was
2156 ... appropriate for initial testing. The implementation correctly followed the
2157 ... experimental design with proper data splits, metric calculations, and result
2158 ... analysis.",
2159 "faithfulness_category": "faithful",
2160 "interesting_results": false,
2161 "metadata_llm": {
2162   "tokens_prompt": 51967, "tokens_completion": 963, "tokens_reasoning": 0,
2163   "tokens_total": 52930, "cost": 0.170346, "model": "claude-sonnet-4-20250514",
2164   "temperature": 0.0, "max_tokens": 32000}, "errors": []},
2165   "execution_success": "success",
2166   "harpa_cost_efficiency": "Used 1.6066855500000001 out of 10 allowed cost.",
2167   "complexity_score": "5 out of 25 reflections used.",
2168   "agent_latest_issues_handled": [{"issues": [], "summary_of_changes": []}, {"issues": [
2169     ["ERROR: MSMARCO dataset structure is different than expected - 'passages'
2170     ... contains strings instead of dictionaries", "Need to add more debugging to
2171     ... understand the actual data structure", "Need to handle the actual MSMARCO data
2172     ... format correctly"],
2173     "summary_of_changes": ["Added extensive debugging to understand MSMARCO data
2174     ... structure", "Fixed data loading logic to handle actual dataset format", "Added
2175     ... error handling and fallback logic for data processing"]}]
2176   }
2177 }
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199

```

Listing 2: Example JSON snippet showing execution-derived factors from CodeScientist logs

E HARPA SCORER PROMPTS

In this section, we include all the prompts used for different tasks within the scorer harpa-rm pipeline.

2180 System prompt for oracle reasoning trace generation

```

2181
2182 Please act as an impartial evaluator and assess the testability or successful
2183 ... execution of the two research proposals generated by an Ideator to execute in an
2184 ... Automated Scientific Discovery (ASD) Agent.
2185
2186 ## START of description of ASD agent:
2187
2188 The ASD Agent is an automated discovery system that writes Python-based experiments,
2189 ... executes them in containers, and analyzes results|usually across multiple
2190 ... independent runs with a meta-analysis.
2191 More specifically, these automated scientific discovery systems operate by having
2192 ... code-based experimentation. They can generate code, run it, debug it, analyze
2193 ... results, create reports, and so forth.
2194
2195 What they can't do:
2196   1. They can't run physical experiments (e.g. wet-lab experiments).
2197   2. They can't perform anything that requires human involvement (e.g. a human
2198     ... manually creating or rating data), because this would not be
2199     ... fully-automatic, and is out of scope.
2200   3. Conduct user studies (e.g., surveys, interviews, usability testing)
2201   4. Depend on real-world deployment or user-facing validation

```

```

2200
2201      5. Require coordinated contributions from a team of experts to design, implement,
2202      ... and refine the idea.
2203      6. Require multiple rounds of expert thinking and intervention to make the idea
2204      ... work.

2205      ## END of description of ASD agent.

2206
2207      ### INTERNAL EXECUTION LOG - DO NOT REVEAL ###
2208      <exec_A>
2209      {{EXEC_META_A}}
2210      </exec_A>
2211      <exec_B>
2212      {{EXEC_META_B}}
2213      </exec_B>
2214      ### END INTERNAL LOG ###

2215      ## CRITICAL: How to Use Execution Logs for Evaluation

2216      The execution logs are PROVIDED FOR YOUR PRIVATE REASONING ONLY. You must:

2217      **DO**:
2218      1. **Validate Predictions**: Compare what the proposal suggested would happen vs.
2219      ... what actually happened
2220      2. **Identify Root Causes**: Understand WHY proposals succeeded or failed based on
2221      ... execution details
2222      3. **Learn Patterns**: Recognize proposal characteristics that correlate with
2223      ... execution outcomes
2224      4. **Generate Insights**: Explain what makes proposals more/less likely to succeed

2225      **NOT DO**:
2226      1. **Reveal Execution Details**: Do not mention specific logs, results, or outcomes
2227      ... in your response
2228      2. **Reference Errors**: Avoid discussing specific errors, successes, or
2229      ... implementation details from logs
2230      3. **Use Execution Phrases**: Don't use phrases like "based on execution" or "the
2231      ... experiment failed/succeeded"
2232      4. **Assume Knowledge**: Do not assume the reader has access to execution logs or
2233      ... results

2234      ## Evaluation steps:

2235      Please follow this process:
2236      1. First, review the execution logs silently and think through whether each
2237      ... experiment likely succeeded.
2238      2. Then score each rubric item, noting whether any execution failure or runtime
2239      ... issue affected it.
2240      3. Reflect whether complexity masked failure or whether execution flags were
2241      ... underweighted.
2242      - (Did the proposal fail because it was too complex for problems to be noticed
2243      ... clearly? Or were there signs of failure that were seen but not taken
2244      ... seriously enough?)
2245      4. Make a final decision based on overall score.

2246      Do not skip any step in this reasoning.

2247      1. Rate each proposal on these criteria (0.0 to 1.0):
2248      - Use the Fixed rubrics as the evaluation criteria tailored to the Client's question
2249      ... and Proposal, enclosed in <rubric> ... </rubric> tags.

2250      The Fixed rubric items are:
2251      1. Execution Success: Based on the proposal content, how likely is it to execute
2252      ... successfully in the ASD Agent? (1.0 = yes (completed status), 0.0 = no (with
2253      ... error or failed status))
2254

```

```

2255
2256     2. Complexity: Based on the proposal content, how complex would this be to
2257     ... implement? (1.0 = very complex (in terms of utilizing complete Reflection
2258     ... Budget), 0.0 = very simple (utilizing less reflection budget))
2259     3. Cost Efficiency: Based on the proposal content, how cost-efficient would this
2260     ... be to execute? (1.0 = very cost-efficient, 0.0 = very costly)
2261     4. Expected Hypothesis Validity: Based on the proposal content, how likely is
2262     ... the hypothesis to yield valid/conclusive results? (1.0 = very valid
2263     ... (support/reject), 0.0 = not valid (inconclusive))
2264     5. Expected Interestingness: Based on the proposal content, how interesting
2265     ... would the results be? (1.0 = very interesting, 0.0 = not interesting)
2266     6. Faithfulness: Based on the proposal content, how faithfully can the ASD agent
2267     ... be expected to execute the original intent? (1.0 = very faithful
2268     ... (faithful/deviations), 0.0 = not faithful (error))
2269
2270     2. Provide Justification
2271     - Assign reward scores using all available information
2272     - Inside <rubric>, include a <justify> ... </justify> section explaining why you
2273     ... chose those scores for the rubric criteria .
2274
2275     3. Compare both responses according to the rubric.
2276     4. Provide your evaluation inside <eval> ... </eval> tags, quoting or summarizing
2277     ... the Responses (Only Proposal Content) using the following tags:
2278
2279     - <quote_A> ... </quote_A> for direct quotes from Proposal A
2280     - <summary_A> ... </summary_A> for paraphrases of Proposal A
2281     - <quote_B> ... </quote_B> for direct quotes from Proposal B
2282     - <summary_B> ... </summary_B> for paraphrases of Proposal B
2283
2284     5. Final Judgment
2285     - End with your final judgment in the format: <answer>[[A]]</answer> or
2286     ... <answer>[[B]]</answer>
2287
2288     ## Important Notes:
2289     - You MAY read execution logs (if available) for private reasoning but MUST NOT
2290     ... reveal their contents
2291     - Only quote/summarize the proposal texts in your evaluation
2292     - Base your judgment on both proposal quality AND how well you can predict
2293     ... outcomes from proposal content
2294     - The execution logs (if available) are your "answer key" - use them to validate
2295     ... your reasoning
2296     - Do not let response order, length, or Response names affect your judgment.
2297     - Follow the response format strictly.
2298
2299     Your output must follow the formats below:
2300
2301     <rubric>
2302     detailed rubric items
2303     <justify> justification for the rubric </justify>
2304     </rubric>
2305
2306     <eval>
2307     include direct comparisons from proposal content supported by <quote_A>...</quote_A>
2308     ... or <summary_A>...</summary_A>, and <quote_B>...</quote_B>, or
2309     ... <summary_B>...</summary_B> tags
2310     </eval>
2311
2312     <answer>[[A/B]]</answer>
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2599
2600
2601
2602
2603
2604
2605
2606
2607
2608
2609
2610
2611
2612
2613
2614
2615
2616
2617
2618
2619
2620
2621
2622
2623
2624
2625
2626
2627
2628
2629
2630
2631
2632
2633
2634
2635
2636
2637
2638
2639
2640
2641
2642
2643
2644
2645
2646
2647
2648
2649
2650
2651
2652
2653
2654
2655
2656
2657
2658
2659
2660
2661
2662
2663
2664
2665
2666
2667
2668
2669
2670
2671
2672
2673
2674
2675
2676
2677
2678
2679
2680
2681
2682
2683
2684
2685
2686
2687
2688
2689
2690
2691
2692
2693
2694
2695
2696
2697
2698
2699
2700
2701
2702
2703
2704
2705
2706
2707
2708
2709
2710
2711
2712
2713
2714
2715
2716
2717
2718
2719
2720
2721
2722
2723
2724
2725
2726
2727
2728
2729
2730
2731
2732
2733
2734
2735
2736
2737
2738
2739
2740
2741
2742
2743
2744
2745
2746
2747
2748
2749
2750
2751
2752
2753
2754
2755
2756
2757
2758
2759
2760
2761
2762
2763
2764
2765
2766
2767
2768
2769
2770
2771
2772
2773
2774
2775
2776
2777
2778
2779
2780
2781
2782
2783
2784
2785
2786
2787
2788
2789
2790
2791
2792
2793
2794
2795
2796
2797
2798
2799
2800
2801
2802
2803
2804
2805
2806
2807
2808
2809
2810
2811
2812
2813
2814
2815
2816
2817
2818
2819
2820
2821
2822
2823
2824
2825
2826
2827
2828
2829
2830
2831
2832
2833
2834
2835
2836
2837
2838
2839
2840
2841
2842
2843
2844
2845
2846
2847
2848
2849
2850
2851
2852
2853
2854
2855
2856
2857
2858
2859
2860
2861
2862
2863
2864
2865
2866
2867
2868
2869
2870
2871
2872
2873
2874
2875
2876
2877
2878
2879
2880
2881
2882
2883
2884
2885
2886
2887
2888
2889
2890
2891
2892
2893
2894
2895
2896
2897
2898
2899
2900
2901
2902
2903
2904
2905
2906
2907
2908
2909
2910
2911
2912
2913
2914
2915
2916
2917
2918
2919
2920
2921
2922
2923
2924
2925
2926
2927
2928
2929
2930
2931
2932
2933
2934
2935
2936
2937
2938
2939
2940
2941
2942
2943
2944
2945
2946
2947
2948
2949
2950
2951
2952
2953
2954
2955
2956
2957
2958
2959
2960
2961
2962
2963
2964
2965
2966
2967
2968
2969
2970
2971
2972
2973
2974
2975
2976
2977
2978
2979
2980
2981
2982
2983
2984
2985
2986
2987
2988
2989
2990
2991
2992
2993
2994
2995
2996
2997
2998
2999
3000
3001
3002
3003
3004
3005
3006
3007
3008
3009
3010
3011
3012
3013
3014
3015
3016
3017
3018
3019
3020
3021
3022
3023
3024
3025
3026
3027
3028
3029
3030
3031
3032
3033
3034
3035
3036
3037
3038
3039
3040
3041
3042
3043
3044
3045
3046
3047
3048
3049
3050
3051
3052
3053
3054
3055
3056
3057
3058
3059
3060
3061
3062
3063
3064
3065
3066
3067
3068
3069
3070
3071
3072
3073
3074
3075
3076
3077
3078
3079
3080
3081
3082
3083
3084
3085
3086
3087
3088
3089
3090
3091
3092
3093
3094
3095
3096
3097
3098
3099
3100
3101
3102
3103
3104
3105
3106
3107
3108
3109
3110
3111
3112
3113
3114
3115
3116
3117
3118
3119
3120
3121
3122
3123
3124
3125
3126
3127
3128
3129
3130
3131
3132
3133
3134
3135
3136
3137
3138
3139
3140
3141
3142
3143
3144
3145
3146
3147
3148
3149
3150
3151
3152
3153
3154
3155
3156
3157
3158
3159
3160
3161
3162
3163
3164
3165
3166
3167
3168
3169
3170
3171
3172
3173
3174
3175
3176
3177
3178
3179
3180
3181
3182
3183
3184
3185
3186
3187
3188
3189
3190
3191
3192
3193
3194
3195
3196
3197
3198
3199
3200
3201
3202
3203
3204
3205
3206
3207
3208
3209
3210
3211
3212
3213
3214
3215
3216
3217
3218
3219
3220
3221
3222
3223
3224
3225
3226
3227
3228
3229
3230
3231
3232
3233
3234
3235
3236
3237
3238
3239
3240
3241
3242
3243
3244
3245
3246
3247
3248
3249
3250
3251
3252
3253
3254
3255
3256
3257
3258
3259
3260
3261
3262
3263
3264
3265
3266
3267
3268
3269
3270
3271
3272
3273
3274
3275
3276
3277
3278
3279
3280
3281
3282
3283
3284
3285
3286
3287
3288
3289
3290
3291
3292
3293
3294
3295
3296
3297
3298
3299
3300
3301
3302
3303
3304
3305
3306
3307
3308
3309
3310
3311
3312
3313
3314
3315
3316
3317
3318
3319
3320
3321
3322
3323
3324
3325
3326
3327
3328
3329
3330
3331
3332
3333
3334
3335
3336
3337
3338
3339
3340
3341
3342
3343
3344
3345
3346
3347
3348
3349
3350
3351
3352
3353
3354
3355
3356
3357
3358
3359
3360
3361
3362
3363
3364
3365
3366
3367
3368
3369
3370
3371
3372
3373
3374
3375
3376
3377
3378
3379
3380
3381
3382
3383
3384
3385
3386
3387
3388
3389
3390
3391
3392
3393
3394
3395
3396
3397
3398
3399
3400
3401
3402
3403
3404
3405
3406
3407
3408
3409
3410
3411
3412
3413
3414
3415
3416
3417
3418
3419
3420
3421
3422
3423
3424
3425
3426
3427
3428
3429
3430
3431
3432
3433
3434
3435
3436
3437
3438
3439
3440
3441
3442
3443
3444
3445
3446
3447
3448
3449
3450
3451
3452
3453
3454
3455
3456
3457
3458
3459
3460
3461
3462
3463
3464
3465
3466
3467
3468
3469
3470
3471
3472
3473
3474
3475
3476
3477
3478
3479
3480
3481
3482
3483
3484
3485
3486
3487
3488
3489
3490
3491
3492
3493
3494
3495
3496
3497
3498
3499
3500
3501
3502
3503
3504
3505
3506
3507
3508
3509
3510
3511
3512
3513
3514
3515
3516
3517
3518
3519
3520
3521
3522
3523
3524
3525
3526
3527
3528
3529
3530
3531
3532
3533
3534
3535
3536
3537
3538
3539
3540
3541
3542
3543
3544
3545
3546
3547
3548
3549
3550
3551
3552
3553
3554
3555
3556
3557
3558
3559
3560
3561
3562
3563
3564
3565
3566
3567
3568
3569
3570
3571
3572
3573
3574
3575
3576
3577
3578
3579
3580
3581
3582
3583
3584
3585
3586
3587
3588
3589
3590
3591
3592
3593
3594
3595
3596
3597
3598
3599
3600
3601
3602
3603
3604
3605
3606
3607
3608
3609
3610
3611
3612
3613
3614
3615
3616
3617
3618
3619
3620
3621
3622
3623
3624
3625
3626
3627
3628
3629
3630
3631
3632
3633
3634
3635
3636
3637
3638
3639
3640
3641
3642
3643
3644
3645
3646
3647
3648
3649
3650
3651
3652
3653
3654
3655
3656
3657
3658
3659
3660
3661
3662
3663
3664
3665
3666
3667
3668
3669
3670
3671
3672
3673
3674
3675
3676
3677
3678
3679
3680
3681
3682
3683
3684
3685
3686
3687
3688
3689
3690
3691
3692
3693
3694
3695
3696
3697
3698
3699
3700
3701
3702
3703
3704
3705
3706
3707
3708
3709
3710
3711
3712
3713
3714
3715
3716
3717
3718
3719
3720
3721
3722
3723
3724
3725
3726
3727
3728
3729
3730
3731
3732
3733
3734
3735
3736
3737
3738
3739
3740
3741
3742
3743
3744
3745
3746
3747
3748
3749
3750
3751
3752
3753
3754
3755
3756
3757
3758
3759
3760
3761
3762
3763
3764
3765
3766
3767
3768
3769
3770
3771
3772
3773
3774
3775
3776
3777
3778
3779
3780
3781
3782
3783
3784
3785
3786
3787
3788
3789
3790
3791
3792
3793
3794
3795
3796
3797
3798
3799
3800
3801
3802
3803
3804
3805
3806
3807
3808
3809
3810
3811
3812
3813
3814
3815
3816
3817
3818
3819
3820
3821
3822
3823
3824
3825
3826
3827
3828
3829
3830
3831
3832
3833
3834
3835
3836
3837
3838
3839
3840
3841
3842
3843
3844
3845
3846
3847
3848
3849
3850
3851
3852
3853
3854
3855
3856
3857
3858
3859
3860
3861
3862
3863
3864
3865
3866
3867
3868
3869
3870
3871
3872
3873
3874
3875
3876
3877
3878
3879
3880
3881
3882
3883
3884
3885
3886
3887
3888
3889
3890
3891
3892
3893
3894
3895
3896
3897
3898
3899
3900
3901
3902
3903
3904
3905
3906
3907
3908
3909
3910
3911
3912
3913
3914
3915
3916
3917
3918
3919
3920
3921
3922
3923
3924
3925
3926
3927
3928
3929
3930
3931
3932
3933
3934
3935
3936
3937
3938
3939
3940
3941
3942
3943
3944
3945
3946
3947
3948
3949
3950
3951
3952
3953
3954
3955
3956
3957
3958
3959
3960
3961
3962
3963
3964
3965
3966
3967
3968
3969
3970
3971
3972
3973
3974
3975
3976
3977
3978
3979
3980
3981
3982
3983
3984
3985
3986
3987
3988
3989
3990
3991
3992
3993
3994
3995
3996
3997
3998
3999
3999

```

Listing 3: Generate rubric-style reasoning trace given the proposal pairs and their execution metadata from an oracle model

```

2310
2311 System prompt for SFT dataset generation
2312
2313 Please act as an impartial evaluator and assess the testability or successful
2314 ... execution of the two research proposals generated by an Ideator to execute in an
2315 ... Automated Scientific Discovery (ASD) Agent.
2316
2317 ## START of description of ASD agent:
2318
2319 The ASD Agent is an automated discovery system that writes Python-based experiments,
2320 ... executes them in containers, and analyzes results|usually across multiple
2321 ... independent runs with a meta-analysis.
2322 More specifically, these automated scientific discovery systems operate by having
2323 ... code-based experimentation. They can generate code, run it, debug it, analyze
2324 ... results, create reports, and so forth.
2325
2326 What they can't do:
2327
2328 1. They can't run physical experiments (e.g. wet-lab experiments).
2329 2. They can't perform anything that requires human involvement (e.g. a human
2330 ... manually creating or rating data), because this would not be
2331 ... fully-automatic, and is out of scope.
2332 3. Conduct user studies (e.g., surveys, interviews, usability testing)
2333 4. Depend on real-world deployment or user-facing validation
2334 5. Require coordinated contributions from a team of experts to design, implement,
2335 ... and refine the idea.
2336 6. Require multiple rounds of expert thinking and intervention to make the idea
2337 ... work.
2338
2339 ## END of description of ASD agent.
2340
2341 ## Evaluation steps:
2342
2343 Please follow this process:
2344
2345 1. First, think through whether each experiment is likely to succeed.
2346 2. Then score each rubric item based on the proposal content.
2347 3. Reflect on whether complexity introduces challenges or obscures potential issues.
2348 - (Might the proposal fail because it is too complex for potential issues to be
2349 ... noticed clearly? Or are there signs of risk that may not have been taken
2350 ... seriously enough?)
2351 4. Make a final decision based on overall score.
2352
2353 Do not skip any step in this reasoning.
2354
2355 1. Rate each proposal on these criteria (0.0 to 1.0):
2356 - Use the Fixed rubrics as the evaluation criteria tailored to the Client's question
2357 ... and Proposal, enclosed in <rubric> ... </rubric> tags.
2358
2359 The Fixed rubric items are:
2360
2361 1. Execution Success: Based on the proposal content, how likely is it to execute
2362 ... successfully in the ASD Agent? (1.0 = yes (completed status), 0.0 = no (with
2363 ... error or failed status))
2364 2. Complexity: Based on the proposal content, how complex would this be to
2365 ... implement? (1.0 = very complex (in terms of utilizing complete Reflection
2366 ... Budget), 0.0 = very simple (utilizing less reflection budget))
2367 3. Cost Efficiency: Based on the proposal content, how cost-efficient would this
2368 ... be to execute? (1.0 = very cost-efficient, 0.0 = very costly)
2369 4. Expected Hypothesis Validity: Based on the proposal content, how likely is
2370 ... the hypothesis to yield valid/conclusive results? (1.0 = very valid
2371 ... (support/reject), 0.0 = not valid (inconclusive))
2372 5. Expected Interestingness: Based on the proposal content, how interesting
2373 ... would the results be? (1.0 = very interesting, 0.0 = not interesting)
2374 6. Faithfulness: Based on the proposal content, how faithfully can the ASD agent
2375 ... be expected to execute the original intent? (1.0 = very faithful
2376 ... (faithful/deviations), 0.0 = not faithful (error))
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2599
2600
2601
2602
2603
2604
2605
2606
2607
2608
2609
2610
2611
2612
2613
2614
2615
2616
2617
2618
2619
2620
2621
2622
2623
2624
2625
2626
2627
2628
2629
2630
2631
2632
2633
2634
2635
2636
2637
2638
2639
2640
2641
2642
2643
2644
2645
2646
2647
2648
2649
2650
2651
2652
2653
2654
2655
2656
2657
2658
2659
2660
2661
2662
2663
2664
2665
2666
2667
2668
2669
2670
2671
2672
2673
2674
2675
2676
2677
2678
2679
2680
2681
2682
2683
2684
2685
2686
2687
2688
2689
2690
2691
2692
2693
2694
2695
2696
2697
2698
2699
2700
2701
2702
2703
2704
2705
2706
2707
2708
2709
2710
2711
2712
2713
2714
2715
2716
2717
2718
2719
2720
2721
2722
2723
2724
2725
2726
2727
2728
2729
2730
2731
2732
2733
2734
2735
2736
2737
2738
2739
2740
2741
2742
2743
2744
2745
2746
2747
2748
2749
2750
2751
2752
2753
2754
2755
2756
2757
2758
2759
2760
2761
2762
2763
2764
2765
2766
2767
2768
2769
2770
2771
2772
2773
2774
2775
2776
2777
2778
2779
2780
2781
2782
2783
2784
2785
2786
2787
2788
2789
2790
2791
2792
2793
2794
2795
2796
2797
2798
2799
2800
2801
2802
2803
2804
2805
2806
2807
2808
2809
2810
2811
2812
2813
2814
2815
2816
2817
2818
2819
2820
2821
2822
2823
2824
2825
2826
2827
2828
2829
2830
2831
2832
2833
2834
2835
2836
2837
2838
2839
2840
2841
2842
2843
2844
2845
2846
2847
2848
2849
2850
2851
2852
2853
2854
2855
2856
2857
2858
2859
2860
2861
2862
2863
2864
2865
2866
2867
2868
2869
2870
2871
2872
2873
2874
2875
2876
2877
2878
2879
2880
2881
2882
2883
2884
2885
2886
2887
2888
2889
2890
2891
2892
2893
2894
2895
2896
2897
2898
2899
2900
2901
2902
2903
2904
2905
2906
2907
2908
2909
2910
2911
2912
2913
2914
2915
2916
2917
2918
2919
2920
2921
2922
2923
2924
2925
2926
2927
2928
2929
2930
2931
2932
2933
2934
2935
2936
2937
2938
2939
2940
2941
2942
2943
2944
2945
2946
2947
2948
2949
2950
2951
2952
2953
2954
2955
2956
2957
2958
2959
2960
2961
2962
2963
2964
2965
2966
2967
2968
2969
2970
2971
2972
2973
2974
2975
2976
2977
2978
2979
2980
2981
2982
2983
2984
2985
2986
2987
2988
2989
2990
2991
2992
2993
2994
2995
2996
2997
2998
2999
3000
3001
3002
3003
3004
3005
3006
3007
3008
3009
3010
3011
3012
3013
3014
3015
3016
3017
3018
3019
3020
3021
3022
3023
3024
3025
3026
3027
3028
3029
3030
3031
3032
3033
3034
3035
3036
3037
3038
3039
3040
3041
3042
3043
3044
3045
3046
3047
3048
3049
3050
3051
3052
3053
3054
3055
3056
3057
3058
3059
3060
3061
3062
3063
3064
3065
3066
3067
3068
3069
3070
3071
3072
3073
3074
3075
3076
3077
3078
3079
3080
3081
3082
3083
3084
3085
3086
3087
3088
3089
3090
3091
3092
3093
3094
3095
3096
3097
3098
3099
3100
3101
3102
3103
3104
3105
3106
3107
3108
3109
3110
3111
3112
3113
3114
3115
3116
3117
3118
3119
3120
3121
3122
3123
3124
3125
3126
3127
3128
3129
3130
3131
3132
3133
3134
3135
3136
3137
3138
3139
3140
3141
3142
3143
3144
3145
3146
3147
3148
3149
3150
3151
3152
3153
3154
3155
3156
3157
3158
3159
3160
3161
3162
3163
3164
3165
3166
3167
3168
3169
3170
3171
3172
3173
3174
3175
3176
3177
3178
3179
3180
3181
3182
3183
3184
3185
3186
3187
3188
3189
3190
3191
3192
3193
3194
3195
3196
3197
3198
3199
3200
3201
3202
3203
3204
3205
3206
3207
3208
3209
3210
3211
3212
3213
3214
3215
3216
3217
3218
3219
3220
3221
3222
3223
3224
3225
3226
3227
3228
3229
3230
3231
3232
3233
3234
3235
3236
3237
3238
3239
3240
3241
3242
3243
3244
3245
3246
3247
3248
3249
3250
3251
3252
3253
3254
3255
3256
3257
3258
3259
3260
3261
3262
3263
3264
3265
3266
3267
3268
3269
3270
3271
3272
3273
3274
3275
3276
3277
3278
3279
3280
3281
3282
3283
3284
3285
3286
3287
3288
3289
3290
3291
3292
3293
3294
3295
3296
3297
3298
3299
3300
3301
3302
3303
3304
3305
3306
3307
3308
3309
3310
3311
3312
3313
3314
3315
3316
3317
3318
3319
3320
3321
3322
3323
3324
3325
3326
3327
3328
3329
3330
3331
3332
3333
3334
3335
3336
3337
3338
3339
3340
3341
3342
3343
3344
3345
3346
3347
3348
3349
3350
3351
3352
3353
3354
3355
3356
3357
3358
3359
3360
3361
3362
3363
3364
3365
3366
3367
3368
3369
3370
3371
3372
3373
3374
3375
3376
3377
3378
3379
3380
3381
3382
3383
3384
3385
3386
3387
3388
3389
3390
3391
3392
3393
3394
3395
3396
3397
3398
3399
3400
3401
3402
3403
3404
3405
3406
3407
3408
3409
3410
3411
3412
3413
3414
3415
3416
3417
3418
3419
3420
3421
3422
3423
3424
3425
3426
3427
3428
3429
3430
3431
3432
3433
3434
3435
3436
3437
3438
3439
3440
3441
3442
3443
3444
3445
3446
3447
3448
3449
3450
3451
3452
3453
3454
3455
3456
3457
3458
3459
3460
3461
3462
3463
3464
3465
3466
3467
3468
3469
3470
3471
3472
3473
3474
3475
3476
3477
3478
3479
3480
3481
3482
3483
3484
3485
3486
3487
3488
3489
3490
3491
3492
3493
3494
3495
3496
3497
3498
3499
3500
3501
3502
3503
3504
3505
3506
3507
3508
3509
3510
3511
3512
3513
3514
3515
3516
3517
3518
3519
3520
3521
3522
3523
3524
3525
3526
3527
3528
3529
3530
3531
3532
3533
3534
3535
3536
3537
3538
3539
3540
3541
3542
3543
3544
3545
3546
3547
3548
3549
3550
3551
3552
3553
3554
3555
3556
3557
3558
3559
3560
3561
3562
3563
3564
3565
3566
3567
3568
3569
3570
3571
3572
3573
3574
3575
3576
3577
3578
3579
3580
3581
3582
3583
3584
3585
3586
3587
3588
3589
3590
3591
3592
3593
3594
3595
3596
3597
3598
3599
3600
3601
3602
3603
3604
3605
3606
3607
3608
3609
3610
3611
3612
3613
3614
3615
3616
3617
3618
3619
3620
3621
3622
3623
3624
3625
3626
3627
3628
3629
3630
3631
3632
3633
3634
3635
3636
3637
3638
3639
3640
3641
3642
3643
3644
3645
3646
3647
3648
3649
3650
3651
3652
3653
3654
3655
3656
3657
3658
3659
3660
3661
3662
3663
3664
3665
3666
3667
3668
3669
3670
3671
3672
3673
3674
3675
3676
3677
3678
3679
3680
3681
3682
3683
3684
3685
3686
3687
3688
3689
3690
3691
3692
3693
3694
3695
3696
3697
3698
3699
3700
3701
3702
3703
3704
3705
3706
3707
3708
3709
3710
3711
3712
3713
3714
3715
3716
3717
3718
3719
3720
3721
3722
3723
3724
3725
3726
3727
3728
3729
3730
3731
3732
3733
3734
3735
3736
3737
3738
3739
3740
3741
3742
3743
3744
3745
3746
3747
3748
3749
3750
3751
3752
3753
3754
3755
3756
3757
3758
3759
3760
3761
3762
3763
3764
3765
3766
3767
3768
3769
3770
3771
3772
3773
3774
3775
3776
3777
3778
3779
3780
3781
3782
3783
3784
3785
3786
3787
3788
3789
3790
3791
3792
3793
3794
3795
3796
3797
3798
3799
3800
3801
3802
3803
3804
3805
3806
3807
3808
3809
3810
3811
3812
3813
3814
3815
3816
3817
3818
3819
3820
3821
3822
3823
3824
3825
3826
3827
3828
3829
3830
3831
3832
3833
3834
3835
3836
3837
3838
3839
3840
3841
3842
3843
3844
3845
3846
3847
3848
3849
3850
3851
3852
3853
3854
3855
3856
3857
3858
3859
3860
3861
3862
3863
3864
3865
3866
3867
3868
3869
3870
3871
3872
3873
3874
3875
3876
3877
3878
3879
3880
3881
3882
3883
3884
3885
3886
3887
3888
3889
3890
3891
3892
3893
3894
3895
3896
3897
3898
3899
3900
3901
3902
3903
3904
3905
3906
3907
3908
3909
3910
3911
3912
3913
3914
3915
3916
3917
3918
3919
3920
3921
3922
3923
3924
3925
3926
3927
3928
3929
3930
3931
3932
3933
3934
3935
3936
3937
3938
3939
3940
3941
3942
3943
3944
3945
3946
3947
3948
3949
3950
3951
3952
3953
3954
3955
3956
3957
3958
3959
3960
3961
3962
3963
3964
3965
3966
3967
3968
3969
3970
3971
3972
3973
3974
3975
3976
3977
3978
3979
3980
3981
3982
3983
3984
3985
3986
3987
3988
3989
3990
3991
3992
3993
3994
3995
3996
3997
3998
3999
4000
4001
4002
4003
4004
4005
4006
4007
4008
4009
4010
4011
4012
4013
4014
4015
4016
4017
4018
4019
4020
4021
4022
4023
4024
4025
4026
4027
4028
4029
4030
4031
4032
4033
4034
4035
4036
4037
4038
4039
4040
4041
4042
4043
4044
4045
4046
4047
4048
4049
4050
4051
4052
4053
4054
4055
4056
4057
4058
4059
4060
4061
4062
4063
4064
4065
4066
4067
4068
4069
4070
4071
4072
4073
4074
4075
4076
4077
4078
4079
4080
4081
4082
4083
4084
4085
4086
4087
4088
4089
4090
4091
4092
4093
4094
4095
4096
4097
4098
4099
4100
4101
4102
4103
4104
4105
4106
4107
4108
4109
4110
4111
4112
4113
4114
4115
4116
4117
4118
4119
4120
4121
4122
4123
4124
4125
4126
4127
4128
4129
4130
4131
4132
4133
4134
4135
4136
4137
4138
4139
4140
4141
4142
4143
4144
4145
4146
4147
4148
4149
4150
4151
4152
4153
4154
4155
4156
4157
4158
4159
4160
4161
4162
4163
4164
4165
4166
4167
4168
4169
4170
4171
4172
4173
4174
4175
4176
4177
4178
4179
4180
4181
4182
4183
4184
4185
4186
4187
4188
4189
4190
4191
4192
4193
4194
4195
4196
4197
4198
4199
4200
4201
4202
4203
4204
4205
4206
4207
4208
4209
4210
4211
4212
4213
4214
4215
4216
4217
4218
4219
4220
4221
4222
4223
4224
4225
4226
4227
4228
4229
4230
4231
4232
4233
4234
4235
4236
4237
4238
4239
4240
4241
4242
4243
4244
4245
4246
4247
4248
4249
4250
4251
4252
4253
4254
4255
4256
4257
4258
4259
4260
4261
4262
4263
4264
4265
4266
4267
4268
4269
4270
4271
4272
4273
4274
4275
4276
4277
4278
4279
4280
4281
4282
4283
4284
4285
4286
4287
4288
4289
4290
4291
4292
4293
4294
4295
4296
4297
4298
4299
4300
4301
4302
4303
4304
4305
4306
4307
4308
4309
4310
4311
4312
4313
4314
4315
4316
4317
4318
4319
4320
4321
4322
4323
4324
4325
4326
4327
4328
4329
4330
4331
4332
4333
4334
4335
4336
4337
4338
4339
4340
4341
4342
4343
4344
4345
4346
4347
4348
4349
4350
4351
4352
4353
4354
4355
4356
4357
4358
4359
4360
4361
4362
4363
4364
4365
4366
4367
4368
4369
4370
4371
4372
4373
4374
4375
4376
4377
4378
4379
4380
4381
4382
4383
4384
4385
4386
4387
4388
4389
4390
4391
4392
4393
4394
4395
4396
4397
4398
4399
4400
4401
4402
4403
4404
4405
4406
4407
4408
4409
4410
4411
4412
4413
4414
4415
4416
4417
4418
4419
4420
4421
4422
4423
4424
4425
4426
4427
4428
4429
4430
4431
4432
4433
4434
4
```

```

2365
2366 2. Provide Justification
2367   - Assign reward scores using all available information
2368   - Inside <rubric>, include a <justify> ... </justify> section explaining why you
2369   ... chose those scores for the rubric criteria .
2370
2371 3. Compare both responses according to the rubric.
2372 4. Provide your evaluation inside <eval> ... </eval> tags, quoting or summarizing
2373 ... the Responses (Only Proposal Content) using the following tags:
2374   - <quote_A> ... </quote_A> for direct quotes from Proposal A
2375   - <summary_A> ... </summary_A> for paraphrases of Proposal A
2376   - <quote_B> ... </quote_B> for direct quotes from Proposal B
2377   - <summary_B> ... </summary_B> for paraphrases of Proposal B
2378
2379 5. Final Judgment
2380   - End with your final judgment in the format: <answer>[[A]]</answer> or
2381 ... <answer>[[B]]</answer>
2382
2383 ## Important Notes:
2384   - Base your judgment on the the Fixed rubrics as the evaluation criteria
2385   - Only quote/summarize the proposal texts in your evaluation
2386   - Base your judgment on both proposal quality AND how well you can predict
2387   ... outcomes from proposal content
2388   - Do not let response order, length, or Response names affect your judgment.
2389   - Follow the response format strictly.
2390
2391 Your output must follow the formats below:
2392
2393 <rubric>
2394 detailed rubric items
2395 <justify> justification for the rubric </justify>
2396 </rubric>
2397
2398 <eval>
2399 include direct comparisons from proposal content supported by <quote_A>...</quote_A>
2400 ... or <summary_A>...</summary_A>, and <quote_B>...</quote_B>, or
2401 ... <summary_B>...</summary_B> tags
2402 </eval>
2403
2404 <answer>[[A/B]]</answer>
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419

```

Listing 4: Generate SFT dataset with reasoning trace generation instruction without metadata information from CODESCIENTIST execution traces.

Topic extraction from abstracts

You are a research assistant. I will give you a list of research paper abstracts and
... metadata.

Your task is to:

1. Identify a small set of **shared, broad research topics** (e.g., "Prompt
... Learning", "Multimodal Learning", "NLP Applications", etc.)
2. **Assign exactly one topic label to each abstract**, using the same topic for
... similar papers.

Important rules:

- You must return **one topic assignment per paper**.
- Use **shared topics** across multiple papers. Avoid overly specific or unique
... topics.
- There should be **at least 5{10 papers per topic**}, ideally more.
- Do not return a list of only the topics | we need an explicit mapping for each
... paper.

```

2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
    ---  

    Format your response as a JSON list:  

    ...  

    {  

        "topics": [  

            {{ "index": 0, "topic": "Prompt Learning" }},  

            {{ "index": 1, "topic": "Knowledge Distillation" }},  

            ...  

        ]  

    }  

    ...  

    Here are the abstracts and metadata:  

    """ + "\n\n".join([f"{i}. {text.strip()}" for i, text in enumerate(source_texts)])

```

Listing 5: Extract source paper abstract topics to construct preference pairs across same topics

Baseline predicted reasoning trace

<rubric>

1. Execution Success: Based on the proposal content, how likely is it to execute ... successfully in the ASD Agent?
2. Complexity: Based on the proposal content, how complex would this be to implement?
3. Cost Efficiency: Based on the proposal content, how cost-efficient would this be ... to execute?
4. Expected Hypothesis Validity: Based on the proposal content, how likely is the ... hypothesis to yield valid/conclusive results?
5. Expected Interestingness: Based on the proposal content, how interesting would ... the results be?
6. Faithfulness: Based on the proposal content, how faithfully can the ASD agent be ... expected to execute the original intent?

<justify>

1. **Execution Success**: Both proposals are detailed and involve clear steps that ... can be automated, but Proposal A has a more structured experimental design with ... multiple controls and well-defined variables, potentially leading to clearer ... results.
2. **Complexity**: Proposal A is more complex, involving multiple adaptive ... strategies (CAR and PATS) and a detailed operationalization plan. However, ... Proposal B involves integrating a dynamic reward model with GFlowNets, which ... might also be complex.
3. **Cost Efficiency**: Both proposals are likely to be fairly costly due to the ... need for in-depth experiments, but Proposal A may benefit from modular ... implementation, making it potentially more cost-efficient.
4. **Expected Hypothesis Validity**: Proposal A is stronger in terms of hypothesis ... validity as it involves a direct comparison of multiple conditions, providing a ... robust framework to test the hypothesis.
5. **Expected Interestingness**: Both proposals are interesting, but Proposal A ... offers a clear pathway to both enhanced accuracy and efficiency, which could be ... more fascinating to researchers.
6. **Faithfulness**: Both proposals are faithful to their intentions, but Proposal A ... provides a more detailed and structured implementation plan, which can be more ... closely followed by the ASD agent.

</justify>

</rubric>

<eval>

2475
2476 `<quote_A>`The experiment should include three conditions: (1) a baseline with fixed
2477 ... reasoning strategy, (2) CAR-only implementation, and (3) integrated CAR+PATS
2478 ... implementation. The experiment should be structured to clearly demonstrate
2479 ... whether the integrated CAR+PATS approach outperforms both the baseline and
2480 ... CAR-only approaches in terms of accuracy and computational efficiency.`</quote_A>`
2481 `<quote_B>`This experiment will compare three approaches for mathematical reasoning:
2482 ... 1. **Baseline 1 (Static Reward Model)**, 2. **Baseline 2 (PPO)**, and 3.
2483 ... **Experimental (Dynamic Reward GFlowNets)**. The Dynamic Reward GFlowNets
2484 ... approach is expected to enhance both reasoning accuracy and diversity
2485 ... significantly compared to traditional methods.`</quote_B>`
2486
2487 Proposal A is more structured with clear conditions and detailed steps for each
2488 ... phase of the experiment, ensuring a faithful and successful execution. Proposal
2489 ... B, while interesting, lacks a detailed structured approach and relies more on a
2490 ... high-level comparison, which might lead to less clarity in results.
2491
2492 In terms of complexity, both proposals are complex, but Proposal A is more modular
2493 ... and structured, making it easier to implement each component in the ASD Agent.
2494 ... The cost efficiency is similar for both, but Proposal A might be slightly more
2495 ... cost-efficient due to its structured approach, simplifying the execution
2496 ... process.
2497
2498 Hypothesis validity is stronger in Proposal A due to the direct comparison of
2499 ... conditions, providing a clear framework to test the hypothesis. The expected
2500 ... interestingness is high for both, but Proposal A may be marginally more
2501 ... interesting due to its enhanced focus on computational efficiency alongside
2502 ... accuracy.
2503
2504 `</eval>`
2505
2506 `<answer>[[A]]</answer>`

2507 Listing 6: Example of rubric-style reasoning trace generated by the baseline LLM scorer. Note that
2508 while formatted as a rubric, the reasoning is unstructured, fails to follow instructions, and does not
2509 provide actionable justifications, making it unsuitable for guiding proposal refinement.

2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529

2530

F HARPA PROMPTS

2531

In this section, we include all the prompts used for different tasks within the HARPA pipeline.

2532

2533

2534

Example part of HARPA artefact illustrating key variable and value space

2535

2536

2537

2538

2539

2540

2541

2542

2543

2544

2545

2546

2547

2548

2549

2550

2551

2552

2553

2554

2555

2556

2557

2558

2559

2560

2561

2562

2563

2564

2565

2566

2567

2568

2569

2570

2571

2572

2573

2574

2575

2576

2577

2578

2579

2580

2581

2582

2583

2584

```
{
  "key_variables": [
    {
      "name": "Energy Consumption",
      "source_paper": "An Enhanced Multi-Objective Evolutionary Algorithm with Reinforcement Learning for Energy-Efficient Scheduling in the Flexible Job Shop",
      "type": "measurable",
      "definition": "The total amount of energy consumed by the job shop operations during the scheduling process.",
      "importance": "This variable is central to the hypothesis as it directly relates to the goal of reducing energy usage through the proposed framework.",
      "specific_details": "Energy consumption can be measured in kilowatt-hours (kWh) and evaluated using sensors or energy meters attached to machines. The evaluation can include machine energy consumption, workshop energy consumption, and can be benchmarked against traditional methods that do not optimize for energy efficiency."
    },
    {
      "name": "Makespan",
      "source_paper": "A Q-Learning Rescheduling Approach to the Flexible Job Shop Problem Combining Energy and Productivity Objectives",
      "type": "measurable",
      "definition": "The total time required to complete all scheduled jobs in the job shop environment.",
      "importance": "Makespan is a critical performance metric in scheduling, reflecting the efficiency of the scheduling method in minimizing completion time.",
      "specific_details": "Makespan is typically measured in hours or minutes from the start of the first job to the completion of the last job. It can be evaluated using scheduling software or simulation tools, and compared against benchmarks from traditional scheduling methods."
    },
    {
      "name": "Multi-Agent Reinforcement Learning Framework",
      "source_paper": "A multi objective collaborative reinforcement learning algorithm for flexible job shop scheduling",
      "type": "design-choice",
      "definition": "A framework that uses multiple agents to learn and make scheduling decisions in a flexible job shop environment.",
      "importance": "This framework is essential for implementing the hypothesis, as it is proposed to dynamically optimize scheduling for energy efficiency and makespan reduction.",
      "specific_details": "The framework can be implemented using algorithms like Q-learning, PPO, or D3QN. Agents can be designed to handle job and machine selection, with state features represented using disjunctive graphs. The framework's effectiveness can be evaluated through simulation and comparison with traditional methods."
    },
    {
      "name": "Traditional Scheduling Methods",
      "source_paper": "A Multi-Agent Reinforcement Learning Approach to the Dynamic Job Shop Scheduling Problem",
      "type": "design-choice",
      "definition": "Conventional scheduling methods that do not incorporate energy efficiency as a primary objective.",
      "importance": "These methods serve as a baseline for comparison to evaluate the effectiveness of the proposed framework."
    }
  ]
}
```

```

2585
2586
2587
2588
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2599
2600
2601
2602
2603
2604
2605
2606
2607
2608
2609
2610
2611
2612
2613
2614
2615
2616
2617
2618
2619
2620
2621
2622
2623
2624
2625
2626
2627
2628
2629
2630
2631
2632
2633
2634
2635
2636
2637
2638
2639

    "specific_details": "Traditional methods may include first-in-first-out, ... shortest processing time, or earliest due date rules. These can be ... implemented in simulation environments to provide a benchmark for ... evaluating the proposed multi-agent reinforcement learning framework."
  },
  {
    "name": "Dynamic Flexible Job Shop Environment",
    "source_paper": "Dynamic Agent-based Bi-objective Robustness for Tardiness ... and Energy in a Dynamic Flexible Job Shop",
    "type": "design-choice",
    "definition": "A job shop environment characterized by variability and ... uncertainty in job arrivals and machine availability.",
    "importance": "This environment is crucial for testing the hypothesis, as it ... reflects real-world conditions where the proposed framework is expected ... to perform.",
    "specific_details": "The environment can be simulated using discrete-event ... simulation tools, incorporating elements like random job arrivals, ... machine failures, and dynamic job priorities. The setup allows for ... testing the adaptability and robustness of the scheduling framework."
  }
],
"variable_space": [
  {
    "Energy Consumption": [
      {
        "value_name": "Machine Energy Consumption",
        "source_paper": "An Enhanced Multi-Objective Evolutionary Algorithm with ... Reinforcement Learning for Energy-Efficient Scheduling in the ... Flexible Job Shop",
        "confidence": "High",
        "is_alternative": false,
        "specific_details": "Machine energy consumption refers to the energy ... used by individual machines during their operation in the job shop. ... This can be measured using energy meters attached to each machine, ... which track the kilowatt-hours (kWh) consumed. The study by Lu et al. ... established a multi-objective integer programming model that ... includes machine energy consumption as a key objective. The model ... aims to minimize this consumption by optimizing the scheduling of ... tasks across machines, considering factors like machine start-up and ... shutdown times. Compatible models include those that can integrate ... with energy meters and support real-time data collection, such as ... systems using IoT-enabled devices. The baseline comparator for this ... value is traditional scheduling methods that do not account for ... energy efficiency, typically resulting in higher energy usage."
      },
      {
        "value_name": "Workshop Energy Consumption",
        "source_paper": "An Enhanced Multi-Objective Evolutionary Algorithm with ... Reinforcement Learning for Energy-Efficient Scheduling in the ... Flexible Job Shop",
        "confidence": "High",
        "is_alternative": false,
      }
    ]
  }
]

```

```

2640
2641
2642
2643
2644
2645
2646
2647
2648
2649
2650
2651
2652
2653
2654
2655
2656
2657
2658
2659
2660
2661
2662
2663
2664
2665
2666
2667
2668
2669
2670
2671
2672
2673
2674
2675
2676
2677
2678
2679
2680
2681
2682
2683
2684
2685
2686
2687
2688
2689
2690
2691
2692
2693
2694

    "specific_details": "Workshop energy consumption encompasses the total
    ... energy used by all machines and processes within the job shop. This
    ... includes both the operational energy of machines and the energy used
    ... for auxiliary processes like lighting and climate control. The study
    ... proposes a model that aims to minimize workshop energy consumption
    ... by optimizing the overall scheduling strategy, using reinforcement
    ... learning to dynamically adjust parameters and improve energy
    ... efficiency. Measurement techniques involve aggregating data from
    ... multiple energy meters and sensors throughout the workshop.
    ... Compatible models are those that can handle large-scale data
    ... integration and real-time adjustments, such as systems using
    ... advanced analytics platforms. The baseline comparator is again
    ... traditional scheduling methods that do not optimize for energy
    ... efficiency, leading to higher overall energy consumption."
},
//...
],
//...
{
  "Traditional Scheduling Methods": [
    {
      "value_name": "First-In-First-Out (FIFO)",
      "source_paper": "A Multi-Agent Reinforcement Learning Approach to the
      ... Dynamic Job Shop Scheduling Problem",
      "confidence": "High",
      "is_alternative": false,
      "specific_details": "FIFO is a traditional scheduling method where the
      ... jobs are processed in the order they arrive at the job shop. This
      ... method does not consider job priority or energy consumption, making
      ... it a straightforward but potentially inefficient approach. In the
      ... context of job shop scheduling, FIFO serves as a baseline for
      ... evaluating more advanced scheduling techniques. The method is
      ... typically implemented in simulation environments to provide a
      ... benchmark for comparison. Compatible models include any discrete
      ... event simulation model that can handle job arrival and processing
      ... sequences. The baseline comparator for FIFO is often more
      ... sophisticated scheduling algorithms that incorporate dynamic
      ... decision-making and energy efficiency considerations."
    },
    {
      "value_name": "Shortest Processing Time (SPT)",
      "source_paper": "A Multi-Agent Reinforcement Learning Approach to the
      ... Dynamic Job Shop Scheduling Problem",
      "confidence": "High",
      "is_alternative": false,
      "specific_details": "SPT prioritizes jobs with the shortest processing
      ... time, aiming to minimize the average job completion time. This
      ... method does not account for energy consumption or job arrival times,
      ... focusing solely on processing efficiency. In practice, SPT can be
      ... implemented using a priority queue where jobs are sorted by their
      ... processing time. This method is often used as a benchmark in
      ... scheduling studies to compare against more complex algorithms that
      ... incorporate additional objectives like energy efficiency.
      ... Compatible models include those that can dynamically sort and
      ... prioritize jobs based on processing time. The baseline comparator is
      ... typically a more comprehensive scheduling strategy that considers
      ... multiple objectives."
    },
    {
      "value_name": "Earliest Due Date (EDD)",
      "source_paper": "A Multi-Agent Reinforcement Learning Approach to the
      ... Dynamic Job Shop Scheduling Problem",
      "confidence": "High",
      "is_alternative": false,
    }
  ]
}

```

```

2695
2696
2697
2698
2699
2700
2701
2702
2703
2704
2705
2706
2707
2708
2709
2710
2711
2712
2713
2714
2715
2716
2717
2718
2719
2720
2721
2722
2723
2724
2725
2726
2727
2728
2729
2730
2731
2732
2733
2734
2735
2736
2737
2738
2739
2740
2741
2742
2743
2744
2745
2746
2747
2748
2749

    "specific_details": "EDD schedules jobs based on their due dates, with
    ... the goal of minimizing tardiness. This method does not consider
    ... energy consumption or processing time, focusing instead on meeting
    ... deadlines. EDD can be implemented using a scheduling algorithm that
    ... sorts jobs by their due dates and assigns them to machines
    ... accordingly. This method is often used in environments where meeting
    ... delivery deadlines is critical. Compatible models include those that
    ... can handle job prioritization based on due dates. The baseline
    ... comparator is typically a scheduling method that incorporates
    ... additional factors such as energy consumption and processing
    ... efficiency."
},
//...
],
//...
}
]
}

```

Listing 7: Example JSON snippet showing some key variables and values with detailed information extracted by the HARPA proposal generator

Generate preliminary hypothesis with rationale

```

agent_capabilities = """
    The ASD Agent is an automated discovery system that writes Python-based
    ... experiments, executes them in containers, and analyzes results|usually
    ... across five independent runs with a meta-analysis.

    ASD agent's goal is to downscope the idea to something an undergrad or MSc
    ... student or PhD student could realistically implement, while retaining
    ... novelty and scientific rigour. The result should be suitable for a
    ... conference paper.

    AGENT CONSTRAINTS & CAPABILITIES:
    - The ASD Agent writes Python-based experiments and executes them in
    ... containers
    - Typically runs 5 independent experiments with meta-analysis
    - Target audience: Undergrad/MSc/PhD student implementation level
    - Output should be suitable for workshop or conference paper submission
    - NO manual human ratings (considered 'external major effort')
    - NO model fine-tuning or pretraining
    - NO access to external or private datasets
    - Must use only existing codeblocks and buildable logic
    - All experiments must be fully implementable in Python

"""

system_message = """
    You are a clever AI research scientist with limited
    ... resources, whose primary goal is to identify promising, new, and key
    ... scientific
    problems based on existing scientific literature, in order to aid researchers in
    ... discovering novel
    and significant research opportunities that can advance the field."""
user_message = """
    You are a clever AI research scientist with limited resources
    ... tasked with generating novel research problems based on existing scientific
    ... literature. Your goal is to aid an autonomous discovery agent in identifying
    ... significant research opportunities that can advance the field.

    You are going to generate a research problem that should be original, clear,
    ... feasible, relevant, and significant to its field. This will be based on the
    ... title and abstract of the source paper, those of {len(citing_paper_list)}
    ... related papers in the existing literature.

```

2750
 2751
 2752 IMPORTANT: When evaluating feasibility and outlining the testing approach,
 2753 ... consider the following agent-specific information:
 2754 ```{agent_capabilities}```
 2755 Now, let's start with the research problem generation task.
 2756 1. Understanding of the source paper, and the related papers is essential:
 2757 - The source paper is the primary research study you aim to enhance or build
 2758 ... upon through future
 2759 research, serving as the central source and focus for identifying and developing
 2760 ... the specific
 2761 research problem.
 2762 - The related papers are arranged in temporal order of citation, such that paper
 2763 ... 2 cites paper 1 and
 2764 paper 3 cites paper 2 and so on. The relevant papers provide additional context
 2765 ... and insights that are essential for
 2766 understanding and expanding upon the source paper. However, all the papers in
 2767 ... the list may not be relevant to the primary
 2768 research you are focusing on.
 2769 2. Your approach should be systematic:
 2770 - Start by thoroughly reading the title and abstract of the source paper to
 2771 ... understand its core focus.
 2772 - Next, proceed to read the titles and abstracts of the related papers in the
 2773 ... order in which they appear in the list. Each related paper is accompanied by
 2774 ... an explanation of its relevance to the previous paper, with the first
 2775 ... related paper considering the source paper as the previous paper.
 2776 Identify the papers that form a logical reasoning chain starting from the source
 2777 ... paper.
 2778 - Use only these papers to gain a broader perspective about the progression of
 2779 ... the primary research topic over time.
 2780
 2781 Note that your research idea and hypothesis MUST be testable using the AGENT with
 2782 ... these specific capabilities:
 2783 When evaluating feasibility and outlining the testing approach, consider the
 2784 ... following agent-specific information. Manual human ratings in the research
 2785 ... (e.g. human rating of the quality of generated text from an experiment) is
 2786 ... considered an `external` resource of `major` effort, for the purposes of the
 2787 ... potential research experiments, and should generally be avoided (unless
 2788 ... absolutely required for the research).
 2789
 2790 IMPORTANT: The hypothesis should be implementable in Python, using the above or
 2791 ... other functions. Don't suggest a task that requires skills that cannot be
 2792 ... implemented, e.g., human studies. Don't suggest a task that requires access
 2793 ... to external datasets, as you do not have access to them. Do not suggest tasks
 2794 ... that involve pretraining or fine-tuning models, as you do not have the
 2795 ... resources for such experiments.
 2796
 2797 Now, I am going to provide the source paper and related papers as an enumerated
 2798 ... list of Title, Abstract and Year of publication
 2799 ... triple, as follows:
 2800 Source paper title: {source_paper['title']}

Source paper abstract: {source_paper['abstract']}

Source paper year of publication: {source_paper['year']}

Related papers: {citing_paper_list}

With the provided source paper, and the related papers, your objective now is to
 ... formulate a
 ... research problem that not only builds upon these existing studies but also
 ... strives to be original, clear, feasible, relevant, and significant. Before
 ... crafting the research problem, revisit the title and abstract of the target
 ... paper, to ensure it remains the focal point of your research problem
 ... identification process.

2805
 2806 Now convert this idea into a concrete testable hypothesis. Remember hypothesis
 2807 ... is a declarative statement expressing a
 2808 relationship between two variables like independent or dependent variables or
 2809 ... left group and right group in a given context.
 2810 Your hypothesis should contain the key variable or variables from your research
 2811 ... idea.
 2812
 2813 Source paper title: {source_paper['title']}
 2814 Source paper abstract: {source_paper['abstract']}

2815 Remember that a hypothesis is a declarative statement expressing a relationship
 2816 ... between two variables (e.g., independent and dependent variables) in a given
 2817 ... context. Your refined hypothesis should contain the key variables from your
 2818 ... research idea.

2819 Then, following your review of the above content, please proceed to analyze the
 2820 ... progression of the research topic. Now output this analysis, the research
 2821 ... idea and hypothesis with the rationale.
 2822 Your output should be a valid JSON with the following fields.
 2823 Output a JSON object in the following format:
 2824 ```json
 2825 {
 2826 "Analysis": {{Output a dictionary with each paper in the Related Papers as a key.
 2827 ... For each key (paper) analyze how this paper builds upon the previous papers
 2828 ... in the list. For example, how Paper 0 builds upon source paper and Paper 1
 2829 ... builds upon the concepts in Paper 0 and so on. Elaborate on specific
 2830 ... advancements made, including the explanation behind their effectiveness in
 2831 ... addressing previous challenges. Apply this analytical approach to each valid
 2832 ... paper in the sequence, adding the analysis as the value for each key in a
 2833 ... few sentences. Ignore papers that do not build upon the previous papers and
 2834 ... diverge from the original source paper's topic significantly.}},
 2835 "Rationale": "Summarize the above analysis and explain how you would come up
 2836 ... with a research idea that will advance the field of work while addressing
 2837 ... the limitations of previous work and building upon the existing work.",
 2838 "Research idea": "Delineate an elaborate research problem here including the key
 2839 ... variables.",
 2840 "Hypothesis": "Provide a concrete testable hypothesis that follows from the
 2841 ... above research problem here"
 2842 }}
 2843 ...
 2844 This JSON will be automatically parsed, so ensure the format is precise.

2841
 2842 Listing 8: Generate preliminary hypothesis with rationale after analyzing trends from temporal reasoning
 2843 paper chains

2844
 2845 Generalize Hypothesis for Literature Search

2846
 2847 This is an automated scientific discovery task, with the overall goal of trying to
 2848 ... assess the novelty of scientific claims.
 2849 # Background
 2850 If you think about it, nearly every experiment could be considered novel if you make
 2851 ... the claims specific enough -- for example, performing a well-known experiment on
 2852 ... a specific day, or getting very specific values from the experiment.
 2853 The purpose of your task is to take an input claim, and progressively rewrite it as
 2854 ... several (progressively more general) claims.
 2855 Another system will assess the novelty of these generalized claims, allowing us to
 2856 ... detect not simply whether a claim is novel or not, but how specific a claim has
 2857 ... to be before it's considered novel.
 2858
 2859 # Specific task
 2860 You will be given a claim (below), and your task will be to generate 4 progressively
 2861 ... more generalized versions of that claim.

```

2860
2861
2862 # 7 Examples of the Generalization Process
2863 Below are 7 examples of the generalization process (represented in JSON), to help
2864 ... you understand the task.
2865 - The keys represent names for the 7 different claim examples.
2866 - The value is a list of the (progressively more generalized) claims.
2867 - The 'generalization' key represents the level of generalization (0 is the original
2868 ... claim).
2869 - The 'claim' key represents the claim itself.
2870
2871 <Add here few-shot examples>
2872
2873
2874
2875
2876 # Claim to generalize
2877 The claim to generalize is:
2878 <original_claim>
2879
2880
2881
2882
2883
2884
2885
2886
2887
2888
2889
2890
2891
2892
2893
2894
2895
2896
2897
2898
2899
2900
2901
2902
2903
2904
2905
2906
2907
2908
2909
2910
2911
2912
2913
2914

```

What should I do if the claim above has multiple claims?
- If the claim above has multiple claims, you should pick the single most salient
... claim, and generalize it.

Output format:
- Output in JSON format, as above
- You should output a dictionary with a single key (a few-word summarized version of
... the claim)
- The value should be a list of 4 progressively more generalized versions of the
... claim
- The 'generalization' key should be an integer from 0 to 3, representing the level
... of generalization (0 is the original claim)
- The 'claim' key should be the claim itself

Please output your JSON response between a single code block (```), as it will be
... automatically extracted. You can write any text before or after the code block
... to help you think, but the text in the code block must be exclusively valid JSON.

Listing 9: Generalized H to progressive 4 levels of claims used for literature search

2897 Generate hypothesis specific questions

2898 You are an AI research assistant. Your task is to analyze the following hypothesis
... and generate insightful, targeted questions that will help researchers refine
... it into something testable, implementable, and scientifically valid.

2899 The hypothesis is currently vague and underspecified. Much of the critical
... information required to implement it | such as variables, evaluation
... metrics, tasks, or assumptions | is missing or unclear.

2900 Your goal is to help move this hypothesis toward implementation. If you could
... ask the author of the hypothesis some questions to clarify or sharpen it,
... what would they be?

2901 First, carefully read the following hypothesis:
{hypothesis}

2902 Now, consider the available capabilities for this research:
{agent_capabilties}

2915
 2916 Your goal is to efficiently analyze the hypothesis and generate 20 concise,
 2917 ... focused questions that will help researchers refine and operationalize it
 2918 ... into something implementable and testable. Each question should clearly
 2919 ... target a part of the hypothesis (e.g., variable, measure, assumption, or
 2920 ... outcome). Mention which part you're refining (e.g., IV, DV, comparison
 2921 ... group, comparison variable, operationalization, feasibility).
 2922
 2923 You can make the QA generation more useful by asking the model to *aim* each
 2924 ... question at helping answer/refine one of these:
 2925 ... - `refined_hypothesis`
 2926 ... - `key_variables`
 2927 ... - `research_idea_required_code_and_resources`
 2928 ... - `research_idea_external_requirements`
 2929 ... - `testing_approach`
 2930
 2931 Before generating your 20 questions, reflect on the hypothesis using these
 2932 ... guiding prompts:
 2933
 2934 1. What are the key terms and variables involved?
 2935 2. How can each component be operationalized and measured?
 2936 3. What capabilities from the system are most relevant?
 2937 4. What design setups or tasks could support testing?
 2938 5. What might hinder testing | e.g., feasibility, confounds, or constraints?
 2939 6. What would success look like, and how could it be quantified?
 2940 7. What ethical or resource considerations exist?
 2941
 2942 Use these reflections to inform the questions you write, ensuring they are
 2943 ... well-grounded and cover diverse aspects of hypothesis development and
 2944 ... testing.
 2945
 2946 Where possible, generate questions that will later help produce values for:
 2947 ... - a more specific and testable `refined_hypothesis`
 2948 ... - a list of `key_variables` (IVs, DVs, controls, comparison group, comparison
 2949 ... variables, etc.)
 2950 ... - a list of code/resources in `research_idea_required_code_and_resources`
 2951 ... - package or library requirements
 2952 ... - testing/evaluation structure (`testing_approach`)
 2953
 2954 Present your questions in the following format:
 2955
 2956

```
```json
 2957 {{ "questions": [
 2958 {
 2959 "question": "[Your first question here]"
 2960 },
 2961 {
 2962 "question": "[Your second question here]"
 2963 },
 2964 ...
 2965 {
 2966 "question": "[Your twentieth question here]"
 2967 }
 2968]}}
```

 2969  
 2970 Remember, your analysis and questions should be designed to provide researchers with  
 2971 ... the necessary information to design and implement a robust study testing the  
 2972 ... given hypothesis. Strive for clarity and conciseness in both your analysis and  
 2973 ... questions to ensure the task and results are crisp and easily actionable.

Listing 10: Generate atleast 20 questions to refine the preliminary hypothesis  $H$  to  $H'$

2970

2971

2972

2973

2974

2975

2976

2977

2978

2979

2980

2981

2982

2983

2984

2985

2986

2987

2988

2989

2990

2991

2992

2993

2994

2995

2996

2997

2998

2999

3000

3001

3002

3003

3004

3005

3006

3007

3008

3009

3010

3011

3012

3013

3014

3015

3016

3017

3018

3019

3020

3021

3022

3023

3024

## Refine Hypothesis based on Socratic QA

You are an expert scientific researcher tasked with refining a given hypothesis to ... make it more specific, easily testable, and practically feasible. This process ... is crucial in scientific research as it helps in designing experiments and ... studies that can effectively validate or invalidate the hypothesis.

Here is the original hypothesis you need to refine:

Initial Hypothesis: {initial\_hypothesis}

Here are related papers with title and key passages that may directly inform or ... relate to the hypothesis.

Provenance papers: {relevant\_paper\_list}

Your task is to come up with new refined research hypothesis, and follow-on research ... ideas, based on the research questions, research programs, hypotheses, ... operationalizations of experiments, or any other information provided in these ... paper excerpts.

You can use content from one paper, or combine content from multiple papers to ... generate new ideas.

Similar papers: {similar\_paper\_list}

Your task is to refine this hypothesis by making it more specific, ensuring it is ... testable, and evaluating its practical feasibility.

Answer the following 20 clarifying questions to help sharpen the hypothesis:  
{questions}

Use the insights from the provenance and similar paper excerpts to support and ... justify your answers wherever applicable. Before providing your final output, ... wrap your thought process in 'thoughts' of the output JSON. Include the ... following subsections:

1. **Initial Analysis** | Break down the hypothesis: variables, assumptions, and ... relationships.
2. **Related Literature** | Quote and summarize relevant insights from similar ... papers. List testable variables from them.
3. **Specificity** | Suggest ways to make the hypothesis more concrete. Rank by ... specificity.
4. **Testability** | Propose 2-3 test designs, list what to measure and possible ... challenges.
5. **Feasibility** | For key variables, suggest how to measure them and rate ... feasibility. Address compute limits, ethics, and practical agent constraints. ... Also, list any code resources, models, datasets, or tools required | these ... should map directly into your `research\_idea\_required\_code\_and\_resources` field.
6. **Testing Approach** | Outline how the hypothesis could be tested using available ... agent tools only (no external data, no human evals, no fine-tuning, no ... model-training).
7. **Final Refinement** | Synthesize the answers of clarifying questions and above ... considerations to create a refined, specific, testable, and feasible version of ... initial hypothesis.

IMPORTANT: When evaluating feasibility and outlining the testing approach, consider ... the following agent-specific information:

{agent\_capabilities}

IMPORTANT: The hypothesis should be implementable in Python, using the above or ... other functions. Don't suggest a task that requires skills that cannot be ... implemented, e.g., human studies. Don't suggest a task that requires access to ... external datasets, as you do not have access to them. Do not suggest tasks that ... involve pretraining or fine-tuning models, as you do not have the resources for ... such experiments.

```

3025
3026 Remember that a hypothesis is a declarative statement expressing a relationship
3027 ... between two variables (e.g., independent and dependent variables) in a given
3028 ... context. Your refined hypothesis should contain the key variables from your
3029 ... research idea.
3030
3031 Ensure each answer is supported by information from the hypothesis, agent
3032 ... capabilities, or provided papers. If an answer cannot be derived, explain what
3033 ... information is missing.
3034
3035 Example output structure (this is a generic example to illustrate the format):
3036
3037 ````json
3038 {{ "thoughts": {{ "Initial Analysis": "...", "Similar Papers": "...", "Specificity Improvements": "...", "Testability Considerations": "...", "Measurability and Feasibility": "...", "Testing Approach": "...", "Final Refinement": "...", "Clarifying Questions & Answers": {{ "Q1": "Answer to question 1", "Q2": "Answer to question 2", "...", "Q20": "Answer to question 20" }} }}, "refined_hypothesis": "Provide a concrete testable hypothesis", "key_variables": [list of key variables], "research_idea_required_code_and_resources": [{{ "name": "Example Resource", "description": "Brief description of the resource", "where": "One of: 'existing codeblock', 'external', or 'build'", "effort": "One of: 'minor', 'moderate', or 'major'" }], "research_idea_external_requirements": ["example_package (for specific purpose)"], }}, ````.
3040
3041
3042
3043
3044
3045
3046
3047
3048
3049
3050
3051
3052
3053
3054
3055
3056
3057
3058
3059
3060
3061
3062
3063
3064
3065
3066
3067
3068
3069
3070
3071
3072
3073
3074
3075
3076
3077
3078
3079

```

Listing 11: Refine the preliminary hypothesis  $H$  to  $H'$  by answering Socratic questions and making it more specific

#### Key Concepts Extraction

You are an expert scientific researcher tasked with analyzing a given hypothesis and ... extracting key information from related papers. Your goal is to identify key ... variables, their possible value options, and rate these options for specificity, ... testability, and feasibility.

Here is the hypothesis you need to analyze:

Hypothesis: {hypothesis}

To assist you in this task, here are related papers with titles and corresponding ... passages that might be relevant to the given hypothesis:

Similar Paper Context:

3080  
 3081 {similar\_retreived\_papers}  
 3082  
 3083 Your task is to analyze this hypothesis and the related papers to extract key  
 3084 ... variables. Follow these steps:  
 3085 1. Analyze the hypothesis explicitly and systematically extract key variables:  
 3086 - Clearly identify every explicitly mentioned variable or design-level choices  
 3087 ... within the hypothesis as a distinct key variable. This includes quantifiable  
 3088 ... variables and design-level choices.  
 3089 - Convert any implicit or abstract concepts (e.g., performance, reliability,  
 3090 ... robustness) into clearly defined and measurable variables or implementable  
 3091 ... design choices. Do not include vague or unmeasurable conceptual ideas unless  
 3092 ... they are clearly defined in operational terms and when they are central to the  
 3093 ... hypothesis.  
 3094 - Ensure key variables have either measurable, quantifiable properties, such as  
 3095 ... "Model Training Time (seconds)," "Error Rate (%)," or "Knowledge Retention  
 3096 ... Score." Or a design choice that affects implementation or evaluation (e.g., "Use  
 3097 ... of pretraining dataset X", "Fine-tuning vs. zero-shot prompting")  
 3098 - Provide a precise, measurable definition (one sentence) for each identified  
 3099 ... key variable. Explicitly define how each key variable should be measured,  
 3100 ... stating exact metrics, evaluation criteria, or assessment methods clearly  
 3101 ... and concisely.  
 3102 - For design choices, define what the choice is, its implications, and how it  
 3103 ... would be implemented or varied in an experiment.  
 3104 - Example design choices include memory architecture (e.g., episodic memory,  
 3105 ... fact-memory modules), prompt strategy (e.g., few-shot, chain-of-thought),  
 3106 ... retrieval method (e.g., top-k, semantic retrieval), narrative control  
 3107 ... mechanism (e.g., branching storylets, story graphs), or model integration  
 3108 ... choices (e.g., use of fine-tuned GPT-3 vs. GPT-4). These should be specific  
 3109 ... and tied to actual implementation decisions that can affect the system  
 3110 ... behavior or experimental outcome.  
 3111 - Include relevant experiment-level factors (e.g., dataset choice, baseline models,  
 3112 ... training configurations) as variables if they impact testing the hypothesis  
 3113 - Do not omit any explicitly mentioned concept from the hypothesis.  
 3114 2. Review the similar papers:  
 3115 - Extract relevant quotes.  
 3116 - Analyze how each quote relates to the hypothesis.  
 3117 - Identify specific and testable variables or design choices from the quotes.  
 3118 3. For each key variable:  
 3119 - Clearly define how it should be measured or implemented (in `specific\_details`).  
 3120 - Indicate the type of variable using "type": "measurable" or "type":  
 3121 ... "design-choice" in the output.  
 3122 - Determine whether the key variable is **\*\*explicitly mentioned\*\*** in related work or  
 3123 ... if it is inferred.  
 3124 a. Mark variables found in paper excerpts with their exact paper title and  
 3125 ... include page/section if available  
 3126 b. Mark variables as 'LLM-recommended' only if not supported by provided papers  
 3127 - In specific\_details, provide:  
 3128 a. For measurable variables: metrics, methods of evaluation, potential value  
 3129 ... ranges, and example benchmarks  
 3130 b. For design choices: the specific options or configurations, how they can be  
 3131 ... varied, how they impact implementation, and any relevant examples or  
 3132 ... baselines  
 3133  
 3134 Remember to focus solely on analyzing the given hypothesis, identifying key  
 3135 ... variables, and extracting specific value options from the similar papers. Do not  
 3136 ... attempt to refine or improve the hypothesis.  
 3137  
 3138 Your final output should be structured clearly and explicitly to enhance  
 3139 ... interpretability. Follow this JSON format strictly:

```

3135
3136 ````json
3137 {{{
3138 "hypothesis": "state the hypothesis as given",
3139 "list_key_variables": ["variable_1", "variable_2", "..."],
3140 "key_variables": [
3141 {{{
3142 "name": "concise Variable Name",
3143 "source_paper": "Paper Title or 'LLM-recommended'",
3144 "type": "measurable" or "design-choice",
3145 "definition": "Precise, measurable definition of the variable.",
3146 "importance": "Brief explanation of why this variable matters to the
3147 ... hypothesis.",
3148 "specific_details": "Detailed information on measurement techniques,
3149 ... potential value ranges, and specific examples of implementation,
3150 ... elaborated with information from related passages."
3151 }}}
3152]
3153 }}}
3154 ````
```

Listing 12: Extraction of key variables or concepts

### Exploring Variable Space

Your goal is to identify specific variable values for a given variable from a given  
 ... hypothesis and the provided relevant literature excerpts as context.

Here is the hypothesis you need to analyze:

Hypothesis: {hypothesis}

Now, the value options you need to extract is for the key variable provided here:

Key variable information: {variable\_info}

To assist you in this task, here are related papers with titles and corresponding  
 ... passages that might be relevant to the given hypothesis and the key variables:

Similar Paper Context:

{similar\_retrieved\_papers}

Your task is to analyze the hypothesis and related papers to extract  
 ... **implementation-relevant, distinct, and non-redundant** values for the given  
 ... key variable. Follow these rules:

-----

1. **Determine the nature of the key variable**

- First, determine if the key variable is itself a metric/outcome measure (e.g.,  
 ... "Task Completion Rate", "Accuracy")

- If it IS a metric/outcome measure:

- a. DO NOT extract implementation environments or frameworks as values
  - b. Instead, identify specific and quantifiable alternative metrics that  
 ... could directly replace this key variable
  - c. Examples: Instead of "Accuracy", alternatives include Precision, Recall,  
 ... F1-score, etc.

- If it is NOT a metric/outcome measure: Identify a minimum of 15 distinct  
 ... variable values from the papers

- Extract values that are (1) specific design choices (e.g. architectures,  
 ... training settings, prompt formats, toolkits), (2) implementation  
 ... strategies (e.g. planning mechanisms, memory structures), or (3)  
 ... quantifiable outcome metrics where applicable.
  - **For ALL identified values/alternatives**

3190  
 3191 a. Prioritize the most relevant values to the hypothesis if there are  
 3192 ... many (>15) options  
 3193 b. Mark variable values found in paper excerpts with their exact paper  
 3194 ... title and include page/section if available  
 3195 c. Mark values as `LLM-recommended` only if not clearly supported by  
 3196 ... provided papers  
 3197 d. Prioritize values directly sourced from provided papers over  
 3198 ... LLM-generated suggestions  
 3199 e. Assign confidence levels using these criteria:  
 3200   - High: Values explicitly mentioned in papers with detailed  
 3201   ... implementation information available  
 3202   - Medium: Values that can be reasonably inferred from the papers but  
 3203   ... aren't explicitly stated  
 3204   - Low: Values that may be applicable based on general domain  
 3205   ... knowledge but aren't explicitly mentioned in papers  
 3206 f. Include concrete examples or parameter ranges for specificity  
 3207 g. DO NOT extract vague concepts, AI frameworks, or general  
 3208 ... methodologies (e.g., "Reinforcement Learning") as variable values.  
 3209 h. Do not extract values that are purely numerical performance metrics  
 3210 ... (e.g., "67% task completion", "80% accuracy") | even if they differ  
 3211 ... across models or setups. Your task is to extract design decisions,  
 3212 ... implementation structures, and qualitative strategies | not  
 3213 ... performance outcomes or numeric results. Values like "75% task  
 3214 ... success" or "F1 score 0.88" are not allowed under any condition. If  
 3215 ... they appear in the paper, ignore or summarize them in  
 3216 ... specific\_details if useful.  
 3217   - You may additionally propose up to 3 novel, plausible variable values (as  
 3218 ... `LLM-recommended`) using your domain knowledge and the provided context.  
 3219  
 3220 2. **Extract relevant alternatives:**  
 3221 - If the papers mention alternative approaches or techniques that could substitute  
 3222 ... for the key variable, include these as well.  
 3223   a. For example, if the key variable is "Q-learning integration", include other  
 3224 ... reinforcement learning techniques mentioned in the papers  
 3225   b. Clearly indicate that these are alternatives to the main variable  
 3226   c. Apply the same source attribution and confidence levels as for direct  
 3227 ... variable values  
 3228 - If you cannot find sufficient values (at least 3) from the provided papers, state  
 3229 ... this clearly before providing your recommendations.  
 3230 - If the key variable is itself a variable value (e.g., "Task Completion Rate",  
 3231 ... "Accuracy", "Success Rate"), then DO NOT extract variable values. As relevant  
 3232 ... alternative, enumerate all possible alternative **variables** that directly  
 3233 ... replace this key variables. These should be described as variable values with  
 3234 ... detailed technical explanations | not as outcomes or statistical results.  
 3235  
 3236 3. **Additional requirements for ensuring specificity and measurability:**  
 3237 - For each extracted variable value, generate an enriched `specific_details` field by  
 3238 ... elaborating how the value is implemented in practice.  
 3239   Include precise, implementation-level information based on the paper excerpts.  
 3240 - Strictly use the Similar Paper Context to guide your response.  
 3241 - Your elaboration should be specific and use implementation-relevant language.  
 3242 ... Avoid short summaries. Each `specific_details` must be at least 5 sentences and  
 3243 ... include concrete implementation mechanisms such as model type, prompt  
 3244 ... strategies, tuning parameters, evaluation setups, or data collection protocols.  
 3245 ... If not in the text, infer plausible methods and label them as inferred.  
 3246  
 3247 -----  
 3248  
 3249 In your elaboration, include as many of the following implementation details as are  
 3250 ... meaningfully associated with the specific variable value:  
 3251 - Architecture or model used (e.g., transformer, GPT, story graph)  
 3252 - Hyperparameters or training settings (e.g., learning rate, temperature, top-k)  
 3253 - Implementation methods (e.g., prompt templates, retrieval techniques, scoring  
 3254 ... functions)

```

3245
3246 - Evaluation metrics (e.g., accuracy, user ratings, engagement frequency)
3247 - Experimental conditions (e.g., number of participants, dataset used, baseline
3248 ... comparisons)
3249 - Any specific mechanics (e.g., branching storylets, memory modules, dialogue
3250 ... control)
3251 - Optional: any results or findings showing impact or performance
3252 - Do not write vague or conceptual explanations like \this allows more freedom" or
3253 ... \this improves engagement." Instead, explain how the value is implemented | e.g.,
3254 ... \This was achieved using GPT-3 with zero-shot prompts and a node-graph
3255 ... controller to support real-time narrative updates based on player input."
3256 - Do not include result percentages or numeric task scores as values | describe how
3257 ... the system works, not how well it scored.

3258 Your final output should extract the variable name from the "key variable
3259 ... information" provided and use it in place of VARIABLE_NAME in the JSON format
3260 ... below:
3261
3262 Each entry in the list should describe a **specific measurable value or design
3263 ... choice** relevant to the key variable. Both types are valid:
3264 - Measurable values refer to quantifiable parameters, metric types, or behavioral
3265 ... outcomes that can be empirically tracked or computed (e.g., accuracy, latency,
3266 ... F1 score, response time, number of steps).
3267 - Design choices refer to implementation decisions that define system behavior, such
3268 ... as model type, architecture, prompting strategies, memory systems, or dataset
3269 ... selection.

3270
3271
3272
3273
3274
3275
3276
3277
3278
3279
3280
3281
3282
3283
3284
3285
3286
3287
3288
3289
3290
3291
3292
3293
3294
3295
3296
3297
3298
3299
```json
{
  "VARIABLE_NAME": [
    {
      "value_name": "Name of this variable value",
      "source_paper": "Paper Title or 'LLM-recommended'",
      "confidence": "High/Medium/Low",
      "is_alternative": false,
      "specific_details": "Detailed paragraph on measurement techniques, potential
        ... value ranges, and specific examples of implementation, elaborated with
        ... information from related passages."
    },
    {
      "value_name": "Name of this alternative variable value",
      "source_paper": "Paper Title or 'LLM-recommended'",
      "confidence": "High/Medium/Low",
      "is_alternative": true,
      "specific_details": "Detailed paragraph on measurement techniques, potential
        ... value ranges, and specific examples of implementation, elaborated with
        ... information from related passages."
    }
  ]
}
```

```

Listing 13: Exploring Variable Value Space given the set of key variables or concepts

## Final hypothesis and research proposal

You are an expert scientific researcher tasked with refining a given hypothesis  
 ... into a more specific and testable form. Your goal is to generate novel  
 ... hypotheses that:  
 - Are strictly based on the given variable options (no new variables should  
 ... be introduced).

```

3300
3301 - Focus solely on the key variable and its concrete variable values or
3302 ... implementations or alternatives, STRICTLY avoid any ambiguous phrasing
3303 - Use novel variable combinations that have not been extensively explored in
3304 ... similar papers.
3305 - Avoid including specific numerical outcomes (e.g., '\45% improvement') in
3306 ... the hypothesis phrasing.
3307 - Provide a detailed theoretical and practical justification for why the
3308 ... refined hypothesis is an important and promising research direction.

3309 ---
3310
3311 ### **Step 1: Understand the Context**
3312
3313 - **Initial Hypothesis:**

3314 `'{hypothesis}`

3315 - **Available Variables and Value Options:**

3316 `'{variable_info}`

3317 - **Similar Papers (to avoid overlap):**

3318 `'{similar_paper_list}`

3319 (Each item includes paper title, citation count, and year - use this metadata to
3320 ... assess which papers are foundational vs. fringe or outdated. Avoid redoing
3321 ... what's already exists unless you're offering a clear novel twist.)

3322 ---
3323
3324 ### Step 1.5: Plan Your Reasoning
3325 Before generating the specific testable hypothesis, outline the logical
3326 ... reasoning process to **Ensure Novelty and Relevance**:
3327 - What is the main contribution of the initial hypothesis?
3328 - Which variables are most critical?
3329 - Carefully review the `similar_paper_list` to identify variable combinations or
3330 ... configurations **already explored**.
3331 - For each similar paper, consider its citation count and publication year to
3332 ... avoid overlaps with highly cited or recent papers unless offering a clearly
3333 ... novel twist, and to spot works worth revisiting.
3334 - Identify gaps in existing research that your hypothesis can address. The
3335 ... hypothesis should explore NEW VARIABLE COMBINATIONS or CONDITIONS or DESIGN
3336 ... CHOICES that were NOT EXTENSIVELY tested in similar papers.
3337 - The research idea space is vast - prioritize hypotheses that seem explanatory,
3338 ... surprising, or tied to concrete downstream benefits. Not all combinations
3339 ... are equally promising. Ask: *Why is this idea worth testing over 999
3340 ... others?* What gap or uncertainty does it address?
3341 - Avoid trivial permutations (e.g., swapping known modules without meaningful
3342 ... interaction).
3343 - Ensure the integration logic is **not only novel** but **precisely**
3344 ... describable**|how the components work together must be clearly traceable
3345 ... from input to output.

3346 ---
3347
3348 ### Step 2: Generate a Specific Testable Hypothesis
3349
3350 - Analyse the initial hypothesis and generate a specific testable hypothesis by
3351 ... making the key variables from the hypothesis as specific as possible using
3352 ... the variable value options and the similar paper excerpts provided.

3353 Before we begin the refinement process, let's consider the some of the
3354 ... capabilities and description of the autonomous discovery agent that will be
3355 ... testing this hypothesis:
3356
3357 IMPORTANT: When evaluating feasibility and testability of the hypothesis,
3358 ... consider the following agent-specific information:
3359
3360

```

```

3355
3356
3357 ### Agent description:
3358 {agent_description}
3359
3360 - **For every variable and process mentioned in your hypothesis**, explicitly
3361 ... list:
3362 - The required code, resource, model, or tool.
3363 - Source: `existing codeblock` (if in the codeblock library), `external`, or
3364 ... `build` (if needs to be created).
3365 - Effort: `minor`, `moderate`, `major`.
3366 - If a component is not found in the available resources, mark as `build` or
3367 ... `external`.
3368 - This mapping is **critical** for experiment feasibility|*missing or incorrect
3369 ... entries are a critical error*.
3370
3371 ---
3372
3373 Before proceeding, you must strictly follow the following tiered guideline:
3374
3375 ##### MANDATORY
3376 - Strictly use the provided variable options. Do not introduce external
3377 ... variables.
3378 - Focus strictly on the key variable and its concrete variable values with
3379 ... implementations or alternatives, and AVOID any ambiguous phrasing.
3380 - If applicable, make it simple and easy to understand. The hypothesis
3381 ... should explore NEW VARIABLE COMBINATIONS or CONDITIONS or DESIGN CHOICES
3382 ... that were NOT EXTENSIVELY tested in similar papers.
3383 - Make it highly specific and testable. Clearly define the condition, the
3384 ... expected measurable outcome, a control or comparative condition (if
3385 ... applicable)
3386 - Ensure originality. The hypothesis should explore NEW VARIABLE
3387 ... COMBINATIONS or CONDITIONS or DESIGN CHOICES that were not extensively
3388 ... tested in similar papers, but also technically CORRECT.
3389 - Make sure the combination is not just NOVEL, but also PURPOSEFUL. Why
3390 ... do these components logically belong together? What capability does
3391 ... one component enable or enhance in the other?
3392 - The research idea space is vast - prioritize hypotheses that seem
3393 ... explanatory, surprising, or tied to concrete downstream benefits.
3394 ... Not all combinations are equally promising. Ask: *Why is this idea
3395 ... worth testing over 999 others from this space?* What gap or
3396 ... uncertainty does it address?
3397 - Do not include exact numerical claims (e.g., "45% improvement", "2.1x
3398 ... increase"). Use comparative phrasing like "reduced," "improved,"
3399 ... "higher," "significantly more/less" instead. Specific metrics should
3400 ... appear in the evaluation section, not in the hypothesis itself.
3401 - Provide a fully aligned and exhaustive
3402 ... `research_idea_required_code_and_resources`.
3403 - Include **detailed, step-by-step theoretical justification** and
3404 ... **expected synergy** between components.
3405
3406 ##### RECOMMENDED PRACTICES
3407 - Use simple, readable phrasing.
3408 - Favor comparative wording ("higher", "improved") over numeric claims.
3409 - Keep pilot-friendly scope: small data or short episodes.
3410
3411 ##### PROHIBITED
3412 - No external/unlisted variables.
3413 - No specific numeric performance outcomes in hypotheses.
3414 - No model **FINE_TUNING, PRETRAINING**, or internal **parameter updates**.
3415 - AVOID human evaluation unless marked external/major.
3416 - Do not omit any mentioned implementation from resource lists.
3417
3418 ##### FINAL SELF-CHECK
3419 - [] All variables are from the given space

```

```

3410
3411 - [] Hypothesis is clear, testable, and comparative
3412 - [] No numeric performance claims in the hypothesis.
3413 - [] No model fine-tuning or human studies unless justified
3414 - [] Resources list is complete and properly tagged (where + effort)
3415 - [] Hypothesis is implementable with codeblocks or buildable logic
3416
3417 ---
3418 Remember that a hypothesis is a declarative statement expressing a relationship
3419 ... between two variables (e.g., independent and dependent variables) in a given
3420 ... context. Your refined hypothesis should contain the key variables from your
3421 ... research idea.
3422
3423 ---
3424 ### Step 3: Litmus Test: Is Your Hypothesis Understandable?
3425
3426 Try this test:
3427
3428 Ask: Could an MSc student with no background in the specific technique
3429 ... **understand and implement** your hypothesis just from reading the
3430 ... research_idea_long_description?
3431 If not | explain the terms more clearly. If any key term or technique may not be
3432 ... intuitive, include a brief, concrete example of how it works in practice.
3433
3434 Ask: Would a technically trained MSc student be able to reconstruct why and how
3435 ... these techniques fit together just by reading this?
3436 If not, the `theoretical_justification` is too shallow.
3437
3438 Remember: A technically trained MSc student must be able to understand each
3439 ... component and how they fit together. Avoid unexplained jargon. If a method
3440 ... is mentioned (e.g., \multi-arm bandit" or \binary token"), explain what it
3441 ... means, why it's used, and how it works in this experiment.
3442
3443 ---
3444
3445 ### Step 4: Structure Your Output in JSON Format
3446 Based on your analysis, generate a refined hypothesis and provide the following
3447 ... information in JSON format:
3448
3449
3450 ```json
3451 {{ "research_gap": "In 1-2 sentences, clearly state the specific gap or
3452 ... limitation in similar paper list or prior work that this hypothesis
3453 ... addresses. Use plain language. Focus on what has not been tried or is
3454 ... still unclear (e.g., 'No prior work tested X under noisy supervision' or
3455 ... 'Existing models overlook interaction between A and B'). Avoid vague
3456 ... claims like 'this is underexplored'. What has not been tested, why is
3457 ... that important, and how will this hypothesis help fill that gap?", "research_question": "A clear, testable research question that can be
3458 ... addressed using the refined hypothesis. It should reflect the causal or
3459 ... comparative relationship proposed, reuse key variable or method terms,
3460 ... and be answerable using the system's capabilities. Frame it in
3461 ... open-ended scientific language (e.g., 'Does...', 'How does...', 'What
3462 ... effect does...').", "research_idea_hypothesis": "Provide a concrete testable hypothesis", "research_idea_long_description": {} }}}
3463
3464

```

```

3465
3466 "description": "A clear paragraph explaining the complete research idea
3467 ... including what will be tested, how it will be implemented, and
3468 ... expected outcomes. Clearly explain the motivation, purpose, and
3469 ... expected outcomes. Use the selected variable values to describe how
3470 ... each component contributes individually, why their combination is
3471 ... expected to work synergistically, and how this addresses gaps or
3472 ... limitations in prior work (as reflected in the similar paper
3473 ... excerpts). If any mechanism or interaction may be unclear, add a
3474 ... simple, task-specific example to illustrate how it works in practice
3475 ... (e.g., 'when a symptom keyword is detected, a query to the memory
3476 ... module is triggered'). Also explain why the chosen evaluation domain
3477 ... is appropriate. Justify clearly. Tie your reasoning to specific
3478 ... characteristics of the task or evaluation environment, and avoid
3479 ... vague statements|be specific about what performance improvements are
3480 ... expected and why. (200-400 words)",
3481
3482 "research_idea_variables": {
3483 "concise Variable Name": "Begin by clearly defining what the selected
3484 ... value represents|whether it's an architecture, strategy, metric,
3485 ... dataset, or baseline condition. Describe exactly how this
3486 ... variable will be configured, used, or operationalized in the
3487 ... experiment; for example, specify how a module is implemented,
3488 ... how a metric is calculated, or how a strategy is triggered.
3489 ... Explain why this specific value was selected over alternatives,
3490 ... including its advantages, novelty, or relevance to the
3491 ... hypothesis. Describe the expected role this variable plays in
3492 ... the research problem-what outcome it directly influences or
3493 ... enables. If the variable is measurable, explicitly define how it
3494 ... will be assessed, including the metric used, how it's
3495 ... calculated, and what range of values or thresholds would
3496 ... indicate a successful outcome. If the concept is non-obvious,
3497 ... include a **simple illustrative example** to aid understanding.
3498 ... Your explanation should be grounded in the context of the
3499 ... hypothesis and tied directly to experimental design choices and
3500 ... evaluation logic.(200-400 words)",
3501 //Define each non-obvious technique, strategy, or mechanism used in
3502 ... the hypothesis, include a 1-2 sentence example of how it would
3503 ... behave in a sample input scenario.. Add detailed defintion and
3504 ... description of every independent, dependent, comparable groups,
3505 ... comparative variables, and control variables in simple format.
3506 },
3507
3508 "research_idea_design_prompt": "Describe in detail how the hypothesis
3509 ... will be implemented using the agent's capabilities. If any new logic
3510 ... must be built (i.e., not available as an existing codeblock),
3511 ... explicitly describe how it will work at a data and control-flow
3512 ... level. Explain what the new module does (e.g., filters, ranks,
3513 ... reweights, scores), how it fits between existing components, and
3514 ... what rules, heuristics, or computations it will use. Describe
3515 ... exactly how their outputs are linked, how data flows from one to
3516 ... another, and what transformations occur at each step. + If multiple
3517 ... modules or strategies are combined, explain where and how the
3518 ... integration happens|in logic, in inputs/outputs, or in processing
3519 ... flow. Aim for clarity so that a ASD agent could build it based on
3519 ... your explanation. Include all setup steps, model configurations,
3519 ... inputs/outputs expected, and how the hypothesis will be realized
3519 ... end-to-end in code. (500-1000 words)",

```

```

3520
3521
3522
3523
3524
3525
3526
3527
3528
3529
3530
3531
3532
3533
3534
3535
3536
3537
3538
3539
3540
3541
3542
3543
3544
3545
3546
3547
3548
3549
3550
3551
3552
3553
3554
3555
3556
3557
3558
3559
3560
3561
3562
3563
3564
3565
3566
3567
3568
3569
3570
3571
3572
3573
3574

 "research_idea_metric": "Primary and secondary metrics that will be used
 ... to evaluate the hypothesis. Explain how the hypothesis will be
 ... tested using concrete metrics and comparative setups. Identify the
 ... benchmark tasks or datasets to be used, the control condition (e.g.,
 ... a baseline agent without the component being tested), and the exact
 ... performance metrics (e.g., task success rate, reasoning accuracy,
 ... number of valid steps). Define how improvement or success will be
 ... interpreted, including thresholds, number of runs, or statistical
 ... confidence if relevant. If qualitative evaluations are involved,
 ... explain how they will be derived. Ensure that all evaluations are
 ... feasible using the agent's capabilities.(200-400 words)"
 },
 "research_idea_name": "A short, descriptive name for the research idea (3-5
 ... words)",
 "research_idea_short_description": "A single concise sentence summarizing
 ... the core idea (15-25 words)",
 "research_baselines": "Simple list of baseline approaches to compare
 ... against",
 "research_idea_pilot": "Brief description of an initial small-scale test to
 ... validate the approach",
 "research_idea_required_code_and_resources": [
 {
 "name": "Example Resource",
 "description": "Brief description of the resource",
 "where": "One of: 'existing codeblock', 'external', or 'build'",
 "effort": "One of: 'minor', 'moderate', or 'major'"
 },
 // EXHAUSTIVE list of ALL required CODE, RESOURCES, MODELS, etc.
 ... mentioned in the ENTIRE RESEARCH IDEA
],
 "research_idea_external_requirements": [
 "example_package (for specific purpose)"
],
 "explanation": {
 "difference": "How it differs from the initial hypothesis",
 "novelty": "Explain exactly what is new in this configuration. Compare
 ... it to setups or strategies found in the similar paper list. Clarify
 ... what has not been explored and why this combination is interesting
 ... or promising. Be specific and concise - avoid vague claims like
 ... 'this hasn't been done before'.",
 "specificity": "How is it more specific, testable, and feasible",
 "theoretical_justification": "Explain what each component does in this
 ... experiment and why it's useful on its own. Use **concrete,
 ... task-relevant examples**, not general claims. For instance: 'Rotary
 ... embeddings improve recall by preserving positional clues in long
 ... legal clauses.' Explain why any specific evaluation domain is
 ... well-matched to the hypothesis and setup.(200-400 words)",
 "expected_synergies": "Be precise: What output from Component A is used
 ... by Component B? Why in Condition C? At what stage? In what format?
 ... At what decision point? E.g., 'The emotion score from module A
 ... weights the retrieval candidates in module B before ranking.'
 ... (200-400 words)"
 }
}
```

```

Listing 14: Converging to a novel and testable research hypothesis given the hypothesis space