HARPA: A TESTABILITY-DRIVEN, LITERATURE-GROUNDED FRAMEWORK FOR RESEARCH IDEATION

Anonymous authorsPaper under double-blind review

000

001

003

006

007

011

012

013

014

015

016

018

022

024

025

027

029

031

033 034

035

037

039

040 041

042

043

044

045

046

047

048

051

052

053

ABSTRACT

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

1 Introduction

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

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

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

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

060

061

063

065 066

067

068

069

071

072

074

076

077

078

079

080

083

084

086

087

089

090

091

092

095

097

099

100

101

102 103

104

105 106

107

108

109

Figure 1: Overview of HARPA. Starting from a source paper, 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≤0.05) and literature-grounding (+0.85, p≤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).

Automated ideation Computational frameworks such as Literature-Based Discovery (LBD) (Swanson, 1986) illustrate how disconnected literatures can be bridged to reveal hidden hypotheses. More recent systems (Radensky et al., 2024; Wang et al., 2023) focus on producing super-brief, noveltydriven research ideas typically assessed with human judgments rather than execution. Systems such as Chain of Ideas (Li et al., 2024b) and HypER (Vasu

System	1) Grounded ideas?	2) Domain- General?	3) Full proposal?	4) ASD Feasibility?	5) Adaptive?
GPT-5	×	✓	~	×	×
Scideator	✓	✓	×	×	×
Moose-Chem	✓	×	×	×	×
CodeScientist	✓	✓	✓	×	×
AI researcher	✓	✓	✓	×	×
HARPA (ours)	✓	✓	✓	✓	✓

Table 1: Comparison of ideation systems in terms of: 1) Are the ideas grounded in related work? 2) Can the ideator generate open-domain ideas? 3) Generates brief ideas or full proposal? 4) Does it consider feasibility w.r.t ASD agents? 5) Does it learn from prior experiments? (✓: yes, ×: no, ~: sometimes).²

et al., 2025) identify literature trends but generate ideas that are too high-level to be actionable, while Scideator (Radensky et al., 2024) generates diverse coarse-grained facets such as purpose, mechanism, or contribution, offering novelty but lacking operational clarity and require human refinement. IdeaSynth (Pu et al., 2024) transforms research ideas into proposals but demands substantial human-in-the-loop involvement, limiting scalability. Existing systems lack mechanisms to adapt their ideation in response to experimental feedback (Table 1). In this paper, we compare HARPA with the AI Researcher method (Si et al., 2024), which was custom-built for open-domain proposal generation and has demonstrated state-of-the-art performance on this task.

Bridging ideation and execution. Large-scale evaluations (Si et al., 2024; 2025) show that while AI-generated ideas may be perceived as more novel than expert-authored ones, they are often less feasible experimentally. Other ideation frameworks, including MLR-Copilot (Li et al., 2024c) and Agent Laboratory (Schmidgall et al., 2025), emphasize benchmark-guided or multi-agent settings but fall short of systematic experimental comparisons. Execution focused systems like CODE SCIENTIST (Jansen et al., 2025) and AI-Scientist (Lu et al., 2024) demonstrate end-to-end automated experimentation but assume hypotheses are already well-structured and feasible. HARPA complements these systems by generating structured, literature-grounded proposals. Unlike other ideators, HARPA integrates a reward model conditioned on ASD capabilities, making research hypotheses generation novel, grounded, and experimentally feasible. This makes it useful for human researchers positioning it as a building block toward the long-term vision of "robot scientists" (King et al., 2009). Table 1 compares HARPA with representative systems in the literature over different ideation attributes.

3 HARPA: Hypothesis And Research Proposal Assistant

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

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

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

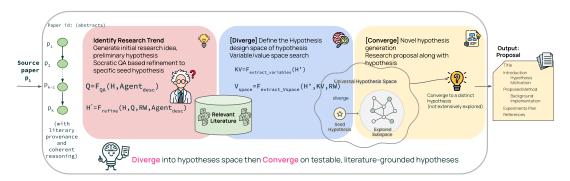


Figure 2: HARPA's Proposal Generator: Divergence and convergence to literature grounded novel proposals

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, \mathcal{Q} , that helps to navigate the specificity of this hypothesis (see Appendix L 10). This is denoted as $\mathcal{Q} = \mathcal{F}_{QA}(H, Agent_{desc})$. Then, given this set of \mathcal{Q} and the set of literature snippets RW relevant to H and $Agent_{desc}$, the language model can refine H by answering these \mathcal{Q} . We denote this using $H' = \mathcal{F}_{refine}(H, \mathcal{Q}, RW, Agent_{desc})$. 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 H times (see Appendix L 9) and each version of H is used as a query to collect H

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 $\mathcal{V}_{space} = \mathcal{F}_{extract_space}(H^{'}, KV, RW)$. 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 $H_{final} = \mathcal{F}_{generate}(H', \mathcal{V}_{space}, RW)$. 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 this idea is evolved as related work (detailed instruction in L14, Appendix F). All LLM function calls in this pipeline were backed by GPT-40 model.

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—

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).

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., 2024; 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:

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

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

4.3 AGENT-CENTRIC EVALUATION SETUP

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

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

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

A full end-to-end evaluation would be ideal, but it is too expensive and requires impractical expert annotation of random proposals. We instead combine expert review on a small set of relevant samples with large-scale testability on diverse ACL papers, leaving full evaluation to future work.

5 Main results

5.1 Human-centric Results

Figure 4 summarizes the expert evaluations of HARPA's proposal generator against the baseline across 11 dimensions. Nine dimensions define the research proposal quality (i.e., novelty, feasibility, expected effectiveness, excitement, grounding, specificity, coherence, motivation, and overall quality), while two meta-dimensions capture the user's familiarity with the proposal topic and their confidence in the judgment (complete proposal evaluation form in Appendix B). HARPA shows statistically significant gains in feasibility (+0.78, p < 0.05, bootstrap) and grounding (+0.85, p < 0.01, bootstrap). For specificity, motivation, and overall scores, HARPA shows a positive trend, although it does not rise to the level of statistical significance. For other metrics, HARPA performs comparably to the baseline (Appendix Table 6), showing that improvements in feasibility and grounding without sacrificing clarity or novelty. Novelty scores for HARPA averaged 5.98 ± 1.33 compared to 6.43 ± 1.32 for the baseline, with both systems rarely falling below the midpoint of the 10-point scale. This indicates that HARPA produces ideas perceived as incrementally novel. These findings align with our design goal that grounding research proposals in literature and refining hypotheses through a human-like workflow leads to more operational, testable research proposals. (Detailed rating distributions in Appendix A.1.)

Execution Success rates: We evaluated whether HARPA proposals more often succeed when executed by a typical ASD agent (here, CODESCIENTIST). For each user-evaluated proposal, we executed five independent runs labeled outcomes using the meta-analysis labeling scheme (Section 3.2). Figure 5a shows the raw distribution of execution outcomes across groups. HARPA produced a higher number of successful executions (20 vs. 11 for the baseline) and fewer outright failures than the baseline AI-researcher system. We further aggregated results into per-idea success rate, defined as the proportion of faithful executions per idea. To ensure fair comparison, we paired HARPA and baseline proposals generated from the same source paper and computed within-source paper differences in success rates. 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.

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

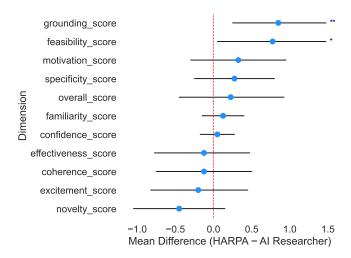
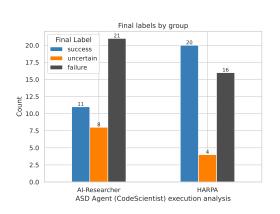
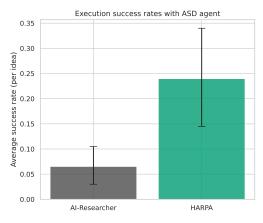


Figure 4: Mean difference between HARPA's proposal generator and AI-Researcher across nine evaluation dimensions. Also reporting the familiarity and confidence score differences. Points show average differences, horizontal bars indicate 95% bootstrap confidence intervals (10k resamples). Stars indicate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





- (a) Execution outcomes (counts) from CODESCIENTIST runs, labeled as success, uncertain, or failure.
- (b) Average per-idea success rates (95% bootstrap CIs) for HARPA vs. baseline AI-Researcher proposals.

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

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

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

System	Pairwise Consistency						Accuracy
	Execution Success	Complexity	Cost Efficiency	Hypothesis Validity	Interesting- ness	Faithfulness	(win)
Baseline	_	_	_	_	_	_	0.52
HARPA	0.80	0.67	0.73	0.69	0.55	0.79	0.81

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

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

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

5.2 AGENT-CENTRIC RESULTS

We next evaluated HARPA-scorer against a baseline untrained LLM scorer, a Qwen-7B-instruct (section 4.1). HARPA-scorer improves accuracy with a +0.28 absolute gain (a 53% relative improvement), with more balanced performance across classes. In addition, HARPA-scorer produces rubric-aligned reasoning traces with explicit scoring on feasibility, cost efficiency, and complexity, like the teacher model. In contrast, the baseline model produced unstructured free text (e.g., in Appendix L 6) that lacks actionable justifications and 4.84% unknown predictions. This alignment with oracle-style reasoning makes HARPA-scorer's judgement easier to interpret and more reliable for refining the research proposals.

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

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

6 CONCLUSION AND FUTURE WORK

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

ETHICS STATEMENT

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

THE USE OF LARGE LANGUAGE MODELS (LLMS)

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

REFERENCES

- Jinheon Baek, Sujay Kumar Jauhar, Silviu Cucerzan, and Sung Ju Hwang. Researchagent: Iterative research idea generation over scientific literature with large language models. *ArXiv*, abs/2404.07738, 2024a. URL https://api.semanticscholar.org/CorpusID:269042844.
- Jinheon Baek, Sujay Kumar Jauhar, Silviu Cucerzan, and Sung Ju Hwang. Researchagent: Iterative research idea generation over scientific literature with large language models. *arXiv* preprint *arXiv*:2404.07738, 2024b.
- Hui Chen, Miao Xiong, Yujie Lu, Wei Han, Ailin Deng, Yufei He, Jiaying Wu, Yibo Li, Yue Liu, and Bryan Hooi. Mlr-bench: Evaluating ai agents on open-ended machine learning research. *ArXiv*, abs/2505.19955, 2025a. URL https://api.semanticscholar.org/CorpusID:278911978.
- Lin Chen, Changshui Zhang, Feng Zhou, Qi Yang, Changming Xiao, Wanpeng Hu, and Haodi Liu. Socratic questioning: Learn to self-guide multimodal reasoning in the wild, 2025b. URL https://arxiv.org/abs/2501.02964.
- Xiusi Chen, Gaotang Li, Ziqi Wang, Bowen Jin, Cheng Qian, Yu Wang, Hongru Wang, Yu Zhang, Denghui Zhang, Tong Zhang, et al. Rm-r1: Reward modeling as reasoning. *arXiv preprint arXiv:2505.02387*, 2025c.
- Kevin Dunbar. How scientists think in the real world: Implications for science education. *Journal of applied developmental psychology*, 21(1):49–58, 2000.
- Jacob G Foster, Andrey Rzhetsky, and James A Evans. Tradition and innovation in scientists' research strategies. American sociological review, 80(5):875–908, 2015.
- Juraj Gottweis, Wei-Hung Weng, Alexander Daryin, Tao Tu, Anil Palepu, Petar Sirkovic, Artiom Myaskovsky, Felix Weissenberger, Keran Rong, Ryutaro Tanno, et al. Towards an ai co-scientist. arXiv preprint arXiv:2502.18864, 2025.
- Peter Jansen, Oyvind Tafjord, Marissa Radensky, Pao Siangliulue, Tom Hope, Bhavana Dalvi Mishra, Bodhisattwa Prasad Majumder, Daniel S Weld, and Peter Clark. Codescientist: End-to-end semi-automated scientific discovery with code-based experimentation. *arXiv preprint arXiv:2503.22708*, 2025.
- Ross D. King, Jem Rowland, Stephen G. Oliver, Michael Young, Wayne Aubrey, Emma Byrne, Maria Liakata, Magdalena Markham, Pinar Pir, Larisa N. Soldatova, Andrew Sparkes, Kenneth E. Whelan, and Amanda Clare. The automation of science. *Science*, 324(5923):85–89, 2009. doi: 10.1126/science.1165620. URL https://www.science.org/doi/abs/10.1126/science.1165620.
- David Klahr and Kevin Dunbar. Dual space search during scientific reasoning. *Cognitive science*, 12(1): 1–48, 1988.
- David Klahr and Herbert A Simon. Studies of scientific discovery: Complementary approaches and convergent findings. *Psychological Bulletin*, 125(5):524, 1999.
- Haitao Li, Qian Dong, Junjie Chen, Huixue Su, Yujia Zhou, Qingyao Ai, Ziyi Ye, and Yiqun Liu. Llms-as-judges: a comprehensive survey on llm-based evaluation methods. *arXiv preprint arXiv:2412.05579*, 2024a.

- Long Li, Weiwen Xu, Jiayan Guo, Ruochen Zhao, Xingxuan Li, Yuqian Yuan, Boqiang Zhang, Yuming
 Jiang, Yifei Xin, Ronghao Dang, et al. Chain of ideas: Revolutionizing research via novel idea
 development with llm agents. arXiv preprint arXiv:2410.13185, 2024b.
 - Ruochen Li, Teerth Patel, Qingyun Wang, and Xinya Du. Mlr-copilot: Autonomous machine learning research based on large language models agents, 2024c. URL https://arxiv.org/abs/2408.14033.
 - Chris Lu, Cong Lu, Robert Tjarko Lange, Jakob Foerster, Jeff Clune, and David Ha. The ai scientist: Towards fully automated open-ended scientific discovery. *arXiv preprint arXiv:2408.06292*, 2024.
 - Nancy J Nersessian. Creating scientific concepts. MIT press, 2010.
 - OpenAI. Gpt-5 system card. https://cdn.openai.com/gpt-5-system-card.pdf, 2025. Accessed: 2025-09-21.
 - Kevin Pu, KJ Feng, Tovi Grossman, Tom Hope, Bhavana Dalvi Mishra, Matt Latzke, Jonathan Bragg, Joseph Chee Chang, and Pao Siangliulue. Ideasynth: Iterative research idea development through evolving and composing idea facets with literature-grounded feedback. *arXiv preprint arXiv:2410.04025*, 2024.
 - Marissa Radensky, Simra Shahid, Raymond Fok, Pao Siangliulue, Tom Hope, and Daniel S Weld. Scideator: Human-Ilm scientific idea generation grounded in research-paper facet recombination. *arXiv* preprint arXiv:2409.14634, 2024.
 - Andrey Rzhetsky, Jacob G Foster, Ian T Foster, and James A Evans. Choosing experiments to accelerate collective discovery. *Proceedings of the National Academy of Sciences*, 112(47):14569–14574, 2015.
 - Samuel Schmidgall, Yusheng Su, Ze Wang, Ximeng Sun, Jialian Wu, Xiaodong Yu, Jiang Liu, Michael Moor, Zicheng Liu, and Emad Barsoum. Agent laboratory: Using llm agents as research assistants, 2025. URL https://arxiv.org/abs/2501.04227.
 - Chenglei Si, Diyi Yang, and Tatsunori Hashimoto. Can llms generate novel research ideas? a large-scale human study with 100+ nlp researchers. *arXiv preprint arXiv:2409.04109*, 2024.
 - Chenglei Si, Tatsunori Hashimoto, and Diyi Yang. The ideation-execution gap: Execution outcomes of llm-generated versus human research ideas. *arXiv preprint arXiv:2506.20803*, 2025.
 - Amanpreet Singh, Joseph Chee Chang, Dany Haddad, Aakanksha Naik, Jena D. Hwang, Rodney Kinney, Daniel S Weld, Doug Downey, and Sergey Feldman. Ai2 scholar QA: Organized literature synthesis with attribution. In Pushkar Mishra, Smaranda Muresan, and Tao Yu (eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pp. 513–523, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-253-4. URL https://aclanthology.org/2025.acl-demo.49/.
 - Don R Swanson. Fish oil, raynaud's syndrome, and undiscovered public knowledge. *Perspectives in biology and medicine*, 30(1):7–18, 1986.
 - Rosni Vasu, Chandrayee Basu, Bhavana Dalvi Mishra, Cristina Sarasua, Peter Clark, and Abraham Bernstein. Hyper: Literature-grounded hypothesis generation and distillation with provenance. *arXiv* preprint arXiv:2506.12937, 2025.
 - Qingyun Wang, Doug Downey, Heng Ji, and Tom Hope. Scimon: Scientific inspiration machines optimized for novelty. *arXiv preprint arXiv:2305.14259*, 2023.
 - Zonglin Yang, Wanhao Liu, Ben Gao, Tong Xie, Yuqiang Li, Wanli Ouyang, Soujanya Poria, Erik Cambria, and Dongzhan Zhou. Moose-chem: Large language models for rediscovering unseen chemistry scientific hypotheses. *arXiv preprint arXiv:2410.07076*, 2024.
 - Minjun Zhu, Qiujie Xie, Yixuan Weng, Jian Wu, Zhen Lin, Linyi Yang, and Yue Zhang. Ai scientists fail without strong implementation capability. *arXiv preprint arXiv:2506.01372*, 2025.

Table 4: Distribution of source papers across shared topics.

	Count
shared topic	
Graph Neural Networks and Graph Learning	3
Generative Models for Images	2
Recommender Systems with LLMs	2
Medical and Healthcare Applications	2
Differentiable Optimization	2
Bias and Fairness in NLP	2
Hallucination and Factuality in LLMs	2
Reinforcement Learning for Scheduling	2
Retrieval-Augmented Generation and Information Refinement	1
Continual Learning and Knowledge Distillation	1
Fake News Detection	1

A APPENDIX

A.1 HUMAN EVALUATION DETAILS

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

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

A.1.1 Source paper domains and topic distribution

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

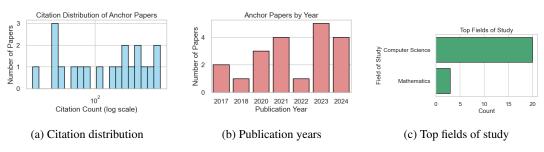


Figure 6: Aggregate statistics of source papers selected by experts.

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.

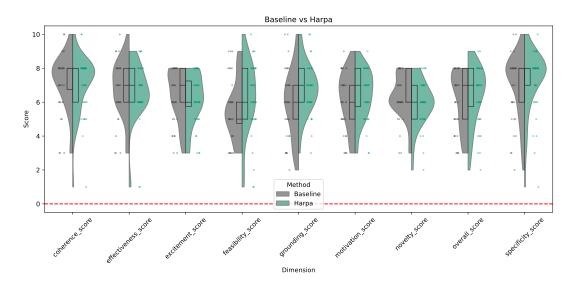


Figure 7: Distribution of expert ratings across nine dimensions for HARPA vs baseline. Shown for completeness (complementary to Fig. 4).

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

- **2. Novelty Score**: Whether the idea is creative and different from existing works on the topic, and brings fresh insights. You are encouraged to search for related works online. You should consider all papers that appeared online prior to July 2024 as existing work when judging the novelty.
 - 1. Not novel at all there are many existing ideas that are the same
 - 2.

- 3. Mostly not novel you can find very similar ideas
- 4.
- 5. Somewhat novel there are differences from existing ideas but not enough to turn into a new paper
- 6. Reasonably novel there are some notable differences from existing ideas and probably enough to turn into a new paper
- 7.
- 8. Clearly novel major differences from all existing ideas
- 9.
- 10. Very novel very different from all existing ideas in a very interesting and clever way

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

- **3. Feasibility Score**: How feasible it is to implement and execute this idea as a research project? Specifically, how feasible the idea is for a typical CS PhD student to execute within 1-2 months of time. You can assume that we have abundant OpenAI / Anthropic API access, but limited GPU compute.
 - Impossible: the idea doesn't make sense or the proposed experiments are flawed and cannot be implemented
 - 2.
 - 3. Very challenging: there are flaws in the proposed method or experiments, or the experiments require compute/human resources beyond any academic lab

 - Moderately feasible: It can probably be executed within the given time frame but would require careful planning, efficient use of APIs or some advanced computational strategies to overcome the limited GPU resources, and would require some modifications to the original proposal to make it work
 - 6. Feasible: Can be executed within the given constraints with some reasonable planning
 - 7.
 - 8. Highly Feasible: Straightforward to implement the idea and run all the experiments
 - 9.
 - Easy: The whole proposed project can be quickly executed within a few days without requiring advanced technical skills

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

- **4. Expected Effectiveness Score**: How likely the proposed idea is going to work well (e.g., better than existing baselines).
 - 1. Extremely Unlikely: The idea has major flaws and definitely won't work well
 - 2.

3. Low Effectiveness: The idea might work in some special scenarios but you don't expect it to work in general

5.	Somewhat ineffective: There might be some chance that the proposed idea can work better than existing baselines but the improvement will be marginal or inconsistent
6.	Somewhat effective: There is a decent chance that the proposed idea can beat existing baselines by moderate margins on a few benchmarks
7.	
8.	Probably Effective: The idea should offer some significant improvement over current methods on the relevant benchmarks
9.	
10.	Definitely Effective: You are very confident that the proposed idea will outperform existing methods by significant margins on many benchmarks
2-3 sente	d Effectiveness Rationale : Short justification for your score. (Your rationale should be at least ences.) <i>Hint: You must consider how the novelty of the idea relates to its excitement or impact—ea is not novel (e.g., already done before), it should generally not be rated as very exciting.</i>
	tement Score: How exciting and impactful this idea would be if executed as a full project. ne idea change the field and be very influential.
1.	Poor: You cannot identify the contributions of this idea, or it's not interesting at all and you would fight to have it rejected at any major AI conference
2.	
3.	Mediocre: this idea makes marginal contributions and is very incremental
4.	
5.	Leaning negative: it has interesting bits but overall not exciting enough
6.	Learning positive: exciting enough to be accepted at a major AI conference, but still has some weaknesses or somewhat incremental
7.	
8.	Exciting: would deepen the community's understanding or make major progress in this research direction
9.	
10.	Transformative: would change the research field profoundly and worth a best paper award at major AI conferences
sentence	nent Rationale: Short justification for your score. (Your rationale should be at least 2-3 s.) Hint: You must consider how the novelty of the idea relates to its excitement or impact — if is not novel (e.g., already done before), it should generally not be rated as very exciting.
strategie	ature Grounding : To what extent are the key components (e.g., model choice, tasks, evaluation es) grounded in existing scientific literature? You may also consider whether ideas reflect ablished domain knowledge or listed references.
1.	Not at all grounded: Mostly speculative or hallucinated; no support from literature or well-established concepts
2.	
3.	Weak grounding: A few connections to existing work, but most claims lack clear support from the listed references or alignment with well-known concepts
4.	
5.	Partially grounded: About half the components are linked to literature or reflect widely accepted ideas in the field
6.	Strong grounding: Most core elements are supported by the listed references or well-established concepts, with only minor gaps
7.	

879

with prior literature or accepted practices?

826	0.	widely accepted domain knowledge, though one or two key claims still lack clear support
827	9.	
828		
829 830	10.	Fully grounded: Every major concept and step is well-supported by listed references or clearly based on well-established domain knowledge; no unsupported claims remain.
831 832 833 834 835	2-3 sent reference assessm	ation: You should also provide a rationale for your score. (Your rationale should be at least tences.) Hint: If a claim is grounded in well-known concepts but not supported by the listed res, explain why it is reasonable based on your domain knowledge. Indicate whether your tent relies on (a) the proposal's reference list, (b) external sources you know, or (c) generally d field knowledge.
836 837 838		vation from Literature: Is the problem statement/overall idea clearly defined and motivated by ic, well-scoped research gap, or limitation identified in the widely recognized field knowledge?
839	1.	No clear motivation: idea feels arbitrary or disconnected
840	2.	
841 842		Weakly motivated: mentions general themes but lacks a compelling rationale
843		
844	4.	
845	5.	Somewhat motivated: a recognizable problem is present, but vague
846	6.	Well motivated: builds on a clear and relevant research direction
847	7.	
848	8.	Strongly motivated: clearly addresses a known issue or opportunity from existing work or
849		widely acknowledged field challenges
850	9.	
851 852		Exceptionally motivated: makes a compelling case for a timely and important problem grounded
853	10.	in the reference list or broadly recognized research needs
854		·
855 856		ation: You should also provide a rationale for your score. (Your rationale should be at least 2-3 es.) Also specify which part of the idea was most clearly linked to a literature-based motivation.
857 858		erence of Idea Composition: Are the combined components (problem, methods, tasks, and logically integrated and literature-informed?
859 860	1.	Incoherent: parts don't fit together; lacks logical or conceptual connection
861	2.	
862		Loosely connected: some rationale exists, but combination feels forced
863		•
864	4.	
865	5.	Reasonable fit: elements are compatible, though not deeply integrated
866	6.	Moderate coherence: combination makes general sense with limited justification
867	7.	
868 869	8.	Coherent and justified: combination makes sense and is literature-informed
870	9.	•
871		
872	10.	Highly coherent: seamless integration of ideas with strong literature basis
873	Explana	ation: You should also provide a rationale for your score. (Your rationale should be at least 2-3
874	sentence	es.) Hint: If the fit between components is strong, note which elements are well connected and
875		defined for implementation (e.g., problem-task pairing, method-metric match). If weak, specify
876	-	arts feel vague, disconnected, or hard to execute.
877 878		ificity of Proposed Method : How clearly does the proposed method present a testable research hypothesis? To what extent is it sufficiently detailed to be operationalized in a way that aligns

8. Very strong grounding: The vast majority of components are supported by listed references or

880	1	Extremely unclear: the method is explained in an extremely vague or ambiguous manner,
881	1.	making it impossible to understand or replicate the approach without additional information or
882		clarification.
883	2	
884	2.	
885	3.	Unclear: the method is described with some detail, but significant gaps in explanation or logic
886		leave the reader with considerable confusion and uncertainty about how to apply or replicate
887		the approach.
888	4.	
000	5	Somewhat clears method is described with sufficient detail to understand the basic enpressed

- 5. Somewhat clear: method is described with sufficient detail to understand the basic approach, but important elements remain vague or underdeveloped
- 6. Moderately clear: method is described with sufficient detail to understand the basic approach, but lacks the precision or specificity needed to fully replicate or grasp the nuances of the methodology without further guidance.

7.

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

9.

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

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.):

- 1. Critically flawed, trivial, or wrong, would be a waste of students' time to work on it
- 2. Strong rejection for major AI conferences
- 3. Clear rejection for major AI conferences
- 4. Ok but not good enough, rejection for major AI conferences
- 5. Decent idea but has some weaknesses or not exciting enough, marginally below the acceptance threshold of major AI conferences
- 6. Marginally above the acceptance threshold of major AI conferences
- 7. Good idea, would be accepted by major AI conferences
- 8. Top 50% of all published ideas on this topic at major AI conferences, clear accept
- 9. Top 15% of all published ideas on this topic at major AI conferences, strong accept
- 10. Top 5% of all published ideas on this topic at major AI conferences, will be a seminal paper

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:

- 1. Your evaluation is an educated guess
- 2. You are willing to defend the evaluation, but it is quite likely that you did not understand central parts of the paper
- 3. You are fairly confident that the evaluation is correct
- 4. You are confident but not absolutely certain that the evaluation is correct

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

Time: How many minutes did you spend on this task?

	Baseline			HARPA				
Dimension	Mean	Std	Min-Max	Median	Mean	Std	Min-Max	Median
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.

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

HARPA PROPOSAL

9/20/25, 9:20 AM Summary

Paper ID

3bfb5f836d944414c171f8f843eaf90cf5604243

1114 Title

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

Introduction

Problem Statement

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.

1136 Motivation

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

9/20/25, 9:20 AM Summary

Proposed Method

1155 1156

1164

1165

1168

1171

1175 1176 1177

1178 1179

1180 1181

1182

1183 1184

1185

1186 1187

1188

1189 1190

1191 1192

1193

1194 1195

1196

1197 1198

1199

1200 1201

1202 1203

1204 1205

1157 The research aims to explore the integration of stochastic softmax tricks with control 1158 variates to enhance variance reduction in spanning tree optimization problems. 1159 1160 Stochastic softmax tricks provide structured relaxations that allow for gradient 1161 estimation in combinatorial spaces, while control variates reduce the variance of these 1162 1163

gradient estimators by incorporating additional information. By combining these two techniques, the hypothesis posits that the model will achieve faster convergence and 1166 more stable performance. This approach addresses the gap in existing research where these techniques are typically applied in isolation. The expected outcome is a more efficient optimization process, particularly in graph-based problems like spanning trees, 1169 where maintaining the graph structure is crucial for accurate gradient computation. This combination is expected to reduce the variance of gradient estimates more effectively than either technique alone, leading to improved model performance metrics such as convergence speed and stability.

Background

Stochastic Softmax Tricks: Stochastic softmax tricks are used to create structured relaxations for combinatorial optimization problems, such as spanning trees. This involves using the Gumbel-Max trick to reparameterize distributions over one-hot binary vectors, allowing for gradient estimation in discrete distributions. The structured relaxation maintains the graph structure, enabling efficient gradient computation. This technique was selected for its ability to handle complex combinatorial spaces and its compatibility with gradient-based optimization methods.

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

Implementation

The proposed method involves integrating stochastic softmax tricks with control variates 1208 to optimize spanning tree problems. First, the stochastic softmax tricks are applied to

9/20/25, 9:20 AM Summary

1210

1215 1216

1217

1219

1220

1221

1224

1225 1226

1232 1233 1234

1239

1241

1242

1243

1245

1247

1248

1249

1250

1251

1253

1256

1257

1261

create a structured relaxation of the spanning tree problem, allowing for gradient estimation in a differentiable manner. This is achieved by representing the problem as a 1212 linear program and applying a softmax function to approximate the selection of edges. Next. control variates are introduced to further reduce the variance of the gradient estimators. This involves constructing a control variate based on an analytical linear approximation to the gradient estimator, which is then combined with the gradient estimates obtained from the stochastic softmax tricks. The integration occurs at the gradient computation stage, where the control variate is used to adjust the gradient estimates, leading to lower variance and improved convergence. The data flows from the structured relaxation to the control variate adjustment, with the final output being a more stable and efficient gradient estimate. This method is implemented using libraries that support automatic differentiation, such as TensorFlow or PyTorch, and is evaluated against baseline methods like the vanilla Gumbel-Softmax estimator.

Experiments Plan

Operationalization Information

Please implement an experiment to test 1240 the hypothesis that 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.

Experiment Overview

This experiment will compare three methods for spanning tree optimization: 1254

- 1. Baseline 1: Vanilla Gumbel-Softmax estimator
- 1258 2. Baseline 2: Stochastic softmax tricks without control variates 1260
 - 3. Experimental: Stochastic softmax tricks integrated with control variates

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

Implement structured relaxations for spanning trees:

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

3. Integrated Approach with Control Variates (Experimental)

9/20/25, 9:20 AM

The experiment should measure convergence speed (iterations to reach a predefined accuracy threshold) and 1267 stability (variance of predictions across different runs) for each method.

1271 1272

1265

1268

1270

1273 1274

1277

1278

1279

1280

1281

1283

1285

1286

1287

1290

1291 1292

1296

1299

1300

1303

1304

1305

1306

1307

Implementation Details

1275 Pilot Mode Settings 1276

Implement a global variable PILOT MODE with three possible settings: MINI_PILOT, PILOT, or FULL EXPERIMENT.

- 1282 MINI_PILOT: Use 5 small random graphs (10-15 nodes) and run 10 optimization iterations with 3 independent runs per method
- 1288 PILOT: Use 20 medium-sized random graphs (20-50 nodes) and run 50 optimization iterations with 10 independent runs per method
- FULL_EXPERIMENT: Use 100 graphs 1294 1295 of varying sizes (up to 100 nodes) and run 200 optimization iterations with 30 1298 independent runs per method

The experiment should first run in 1302 MINI_PILOT mode, then PILOT mode if successful, but stop before FULL EXPERIMENT (which will be manually triggered after human verification).

1308 1309 1310

Graph Dataset Generation

1311 1312

1315

1318

Use NetworkX to generate the following types of random graphs for the 1316 experiment:

1. Erdős-Rényi random graphs

Summary

Extend the stochastic softmax tricks implementation with control variates:

- Construct a control variate based on an analytical linear approximation to the gradient estimator
- Combine the control variate with the naïve gradient estimate from the stochastic softmax tricks
- Implement the optimal scaling parameter for the control variate
- Apply the adjusted gradient in the 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

9/20/25

5, 9:20	AM	Summary
	2. Barabási–Albert preferential	1. For each graph in the dataset:
1320	attachment graphs	a. Run each method (Baseline 1,
1321	• ,	•
1322 1323	Watts–Strogatz small-world graphs	Baseline 2, Experimental) multiple
1324	For each graph, agaign random adag	times with different random seeds
1325	For each graph, assign random edge	 b. Record the optimization
1326	weights from a uniform distribution [0.1,	trajectory (objective value vs.
1327	10.0].	iteration)
1328 1329		,
	Method Implementations	c. Measure the gradient variance at
1331	•	each iteration
1332	1. Vanilla Gumbel-Softmax Estimator	d. Calculate the final solution
1333		quality
1334 1335	(Baseline 1)	4 2
1336		1. Aggregate results across all graphs
1337	Implement the standard Gumbel-Softmax	and runs:
1338	trick for spanning tree optimization:	
1339	- Represent the graph as an edge	a. Calculate average convergence
1340 1341	selection problem	speed for each method
1342	·	 b. Calculate average stability for
1343	- Apply the Gumbel-Max trick to sample	each method
1344	spanning trees	c. Perform statistical significance
1345	- Use the softmax temperature parameter	_
1346 1347	to control the discreteness of the	tests (bootstrap resampling) to
1348	distribution	compare methods
1349	- Implement straight-through estimation	
1350		Visualization and Reporting
1351	for the backward pass	
1352 1353		1. Generate convergence plots
1354		showing objective value vs.
1355		iteration for each method
1356		
1357 1358		2. Create box plots showing the
1359		distribution of convergence speeds
1360		and stability metrics
1361		3. Generate tables with summary
1362		statistics for each method
1363 1364		
1365		Visualize example spanning trees
1366		produced by each method on
1367		selected graphs
1368		
1369 1370		Implementation Notes
1371		ļ
1372		
1373		

9/20/25, 9:20 AM Summary Use PyTorch for automatic 1375 differentiation and gradient 1376 computation 1377 1378 Use NetworkX for graph 1379 manipulation and visualization 1380 1381 · Implement proper seeding for 1382 reproducibility 1383 · Use a learning rate scheduler to 1384 1385 improve convergence 1386 · Save intermediate results to allow 1387 1388 for experiment resumption 1389 1390 1391 **Expected Output** 1392 1393 The experiment should produce: 1394 1395 1. A comprehensive report comparing the 1396 three methods 1397 1398 2. Convergence plots for each method 1399 3. Statistical analysis of the differences 1400 1401 between methods 1402 4. Visualizations of example spanning 1403 trees 1404 1405 5. Raw data for further analysis 1406 1407 Please implement this experiment and 1408 1409 run it first in MINI PILOT mode, then in 1410 PILOT mode if successful. Do not 1411 proceed to FULL EXPERIMENT mode 1412 1413 without human verification. 1414 1415 1416 1417 End Note: 1418 1419 1420 1421 The source paper is Paper 0: Learning with Differentiable Perturbed Optimizers (109 1422

citations, 2020). This idea draws upon a trajectory of prior work, as seen in the following 1424 sequence: <u>Paper 1</u> --> <u>Paper 2</u> --> <u>Paper 3</u> --> <u>Paper 4</u> --> <u>Paper 5</u>. The analysis 1425 reveals a consistent theme of addressing the high variance in gradient estimation for 1427 discrete latent variables, a challenge initially highlighted in the source paper. The

1428

9/20/25, 9:20 AM Summary

progression of research has introduced various techniques like stochastic softmax 1430 tricks, Rao-Blackwellization, and coupled gradient estimators to tackle this issue. 1432 However, these approaches often focus on specific applications or settings, such as 1433 combinatorial spaces or categorical variables. A novel research idea could involve 1435 developing a generalized framework that unifies these variance reduction techniques, 1436 making them adaptable to a broader range of discrete optimization problems. This 1438 would advance the field by providing a more versatile tool for training models with 1439 discrete components, addressing the limitations of existing methods that are often 1440 application-specific. 1441

1442 The initial trend observed from the progression of related work highlights a consistent research focus. However, the final hypothesis proposed here is not merely a 1445 continuation of that trend — it is the result of a deeper analysis of the hypothesis space. By identifying underlying gaps and reasoning through the connections between works, 1448 the idea builds on, but meaningfully diverges from, prior directions to address a more specific challenge.

References

1437

1444

1446

1447

1455 1456 1457

1458 1459

1460

1461 1462

1463

1464

1465 1466

1467

1468 1469

1470

1471 1472

1473

1474 1475

1476

1477

1478 1479

1480

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

9/20/25, 9:20 AM Summary

12. Partial Variance Reduction improves Non-Convex Federated learning on heterogeneous data (2022) 13. Improved Analysis and Rates for Variance Reduction under Without-replacement Sampling Orders (2021) 14. Stochastic Variance Reduction Methods for Policy Evaluation (2017) 15. Reducing Reparameterization Gradient Variance (2017)

BASELINE PROPOSAL

9/20/25, 9:20 AM Summary

Paper ID

3bfb5f836d944414c171f8f843eaf90cf5604243

Title

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

Introduction

1624 Problem Statement

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.

Motivation

Existing methods like fixed gradient clipping, adaptive learning rates, and normalization techniques often struggle to balance stability and convergence speed across different layers and tasks within a pipeline. Inspired by the success of noise injection in improving generalization and the adaptive nature of biological neural systems, we propose a method that dynamically adjusts gradient updates based on local statistics 1644 and stochastic perturbations. This approach allows for aggressive updates in stable regions while dampening oscillations in sensitive areas, all while introducing beneficial 1647 noise for improved exploration and generalization.

9/20/25, 9:20 AM Summary

Proposed Method

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

Experiments Plan

Step-by-Step Experiment Plan

Step 1: Implement ASGC

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

Step 2: Prepare Datasets

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

Step 3: Setup Baseline Models

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

9/20/25, 9:20 AM Summary

Step 4: Train Models with ASGC

1705 1706

Train the same model architectures using ASGC. Use a grid search to find optimal 1708 hyperparameters for ASGC, including the initial noise scale and clipping function parameters.

1710 1711 1712

1709

Step 5: Evaluate Performance

1713 1714

Compare ASGC against baselines on the following metrics: (1) Final test accuracy/BLEU score/game score. (2) Training time to reach a specific performance 1717 threshold. (3) Stability of training (measured by the variance of validation performance across epochs). (4) Generalization (measured by the gap between training and test ₁₇₂₀ performance).

1719 1721

Step 6: Analyze Robustness

1722 1723

Evaluate the robustness of ASGC to hyperparameter choices by training models with 1725 1726 randomly sampled hyperparameters and comparing the distribution of final performances against baselines.

1729 1730

Step 7: Visualize Gradient Statistics

1731

1732 Plot the distribution of gradient magnitudes before and after clipping for different layers and at different stages of training. Compare these distributions between ASGC and 1735 baseline optimizers.

1736 1737

1734

Step 8: Analyze Meta-Learned Parameters

1738 1740

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

1742 1743

Step 9: Ablation Studies

1744 1745 1746

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

1749 1750

1748

Step 10: Write Up Results

1751 1752 1753

Compile all results, visualizations, and analyses into a comprehensive report or paper 1755 draft.

1756 1757

9/20/25, 9:20 AM Summary

Test Case Examples

1760 1761 1762

Baseline Prompt Input

1763

1764 Train a ResNet-50 model on ImageNet using Adam optimizer with default 1765 hyperparameters. 1766

1767

Baseline Prompt Expected Output 1768

1769

1770 Final Top-1 Accuracy: 76.1%, Training Time: 90 hours, Stability (std dev of validation accuracy over last 10 epochs): 0.5% 1772

1773 1774

Proposed Prompt Input

1775 1776

Train a ResNet-50 model on ImageNet using ASGC optimizer with meta-learned 1778 hyperparameters.

1779

1780 **Proposed Prompt Expected Output**

1781 1782

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

1784 1785 1786

Explanation

1787

1789

1791

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

1792 1793 1794

Fallback Plan

1795 1796 1797

1799 1800

1801

1802 1803

1804

1805

1806 1807

1808

1809

If ASGC does not outperform baselines as expected, we can pivot the project to an indepth analysis of why adaptive stochastic methods struggle in certain scenarios. We would conduct a series of experiments to isolate the effects of gradient clipping, noise injection, and adaptive thresholds on different types of neural architectures and tasks. This could involve visualizing gradient flow through networks, analyzing the spectrum of the Hessian at different stages of training, and studying how different optimization techniques affect the loss landscape. We could also explore combining ASGC with other advanced optimization techniques like layer-wise adaptive rates or Hessian-based preconditioning. The goal would be to provide insights into the interplay between network architecture, task complexity, and optimization dynamics, potentially informing 1812 the development of next-generation optimization algorithms.

9/20/25, 9:20 AM Summary

1815		
1816	Pof	erences
1817	IZEI	erences
1818		
1819	1.	The Limited Multi-Label Projection Layer (2019)
1820 1821	2.	Optimizing Rank-Based Metrics With Blackbox Differentiation (2019)
1822	3.	Categorical Reparameterization with Gumbel-Softmax (2016)
1823		Differentiable Top-k Operator with Optimal Transport (2020)
1824 1825		
1826	5.	<u>Tackling Prevalent Conditions in Unsupervised Combinatorial Optimization:</u>
1827		<u>Cardinality, Minimum, Covering, and More</u> (2024)
1828	6.	<u>Differentiable Combinatorial Scheduling at Scale</u> (2024)
1829	7.	Fast Differentiable Sorting and Ranking (2020)
1830 1831		Differentiation of Blackbox Combinatorial Solvers (2019)
1832		• • •
1833		<u>Deep Network Flow for Multi-object Tracking</u> (2017)
1834	10.	<u>Learning Latent Trees with Stochastic Perturbations and Differentiable Dynamic</u>
1835		Programming (2019)
1836 1837		
1838		
1839		
1840		
1841		
1842		
1843 1844		
1845		
1846		
1847		
1848		
1849 1850		
1851		
1852		
1853		
1854		
1855		
1856 1857		
1858		
1859		
1860		
1861		
1862		
1863 1864		
1865		
1866		
1867		
1262		

System	Hypothesis/Problem Statement, Ratings, and Justification
HARPA	"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." 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.")
	Feasibility = 7 ("Using the softmax trick allows backpropagation/gradient estimation, it is a well known trick and the implementation is not so complicated, although thee control variates is not so clear how would be implemented (with
	examples) ")
	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 at "A generalized approximate control variate framework for multifidelity uncertainty quantification (2018)"
	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. ")
	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.") 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. ")
	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. ")
	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. ") 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. ") Confidence = 4
Baseline	"Gradient-based optimization in deep learning often suffers from instability and slow convergence, especially in
Daseille	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."
	Novelty = 5 ("From one perspective, I don't score high the novelty regarding this proposal since this would depend on some detais 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 novely characteristics by combining the statistics with stochastic perturbations.")
	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?") 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.")
	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.")
	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.")
	Motivation = 6 ("The section of motivation makes sense, although not grounded in the list of papers.")
	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).")
	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?")
	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.")
	Confidence = 4

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 openr1hf 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} .

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

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

Example reasoning trace

<rubric>

- 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
- 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
 - 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
 - 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
 - \dots expectation that adaptive architectures should improve performance
 - 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
 - 6. Faithfulness:
 - Proposal A: 0.8 Clear operationalization with realistic simulation of \dots human-in-the-loop processes
 - Proposal B: 0.6 Complex integration requirements may lead to simplified ... implementations that deviate from intended design

<justify>

1980 1981 Proposal A scores higher on execution success due to its more straightforward ... implementation using established toxicity detection methods, while Proposal B's 1982 multiple complex components increase failure risk. Proposal A is less complex, 1983 ... focusing on a single domain (toxicity detection) with clear iterative 1984 ... improvement, whereas Proposal B requires implementing multiple RAG architectures 1985 with real-time and adaptive components. Cost efficiency favors Proposal A due to 1986 its focused scope and efficient use of pre-trained models. Both proposals have reasonable hypothesis validity, but Proposal B's expectation of improvement from 1987 ... adaptive architectures is slightly more logical. Proposal A offers higher 1988 ... interestingness due to its novel approach to multilingual safety alignment. 1989 ... Faithfulness is higher for Proposal A because its human-in-the-loop simulation ... is more realistic than Proposal B's complex integration requirements. 1990 </justify> 1991 </rubric> 1992 1993 1994 <eval> Both proposals address benchmark creation and evaluation but take different 1995 ... approaches. 1996 1997 <summary_A>Proposal A focuses on multilingual toxicity detection, comparing a 1998 baseline system using static data against an experimental system that integrates 1999 dynamic data collection with adversarial evaluation/summary_A>. <quote_A>"The ... experiment should compare a baseline system against an experimental system, with ... appropriate metrics to measure performance differences"</quote_A> and uses 2001 ... <quote_A>"compliance ratios in multilingual toxicity detection tasks and 2002 ... adversarial accuracy drop"</quote_A> as key metrics. 2003 2004 <summary_B>Proposal B targets retrieval-augmented generation systems, comparing ... three configurations: baseline RAG, real-time RAG, and dynamic RAG that combines 2005 both real-time retrieval and adaptive architectures</summary_B>. <quote_B>"This 2006 ... experiment will compare three RAG system configurations: 1. Baseline RAG: A 2007 ... standard RAG system with static retrieval strategies 2. Real-time RAG: A RAG 2008 system with real-time information retrieval 3. Dynamic RAG (Experimental): A RAG . . . system that integrates both real-time information retrieval and adaptive 2009 . . . retrieval architectures"</quote_B>. 2010 2011 In terms of implementation complexity, Proposal A presents a more manageable scope. <quote_A>"Use 100 examples per language from 5 languages (English, Spanish, 2013 French, German, Chinese) for training and 50 examples per language for testing. Run 5 iterations of the dynamic data collection process"</guote A> in pilot 2014 mode, with clear simulation strategies: <quote_A>"Since actual human annotators 2015 aren't available, simulate them by: 1. Creating a function that identifies model 2016 errors and generates challenging examples 2. Implementing different 'annotator 2017 personas' with varying expertise"</quote_A>. 2018 Proposal B involves significantly more complex integration requirements. 2019 <quote_B>"Implement a module that: - Dynamically accesses and integrates 2020 external knowledge bases during generation - Updates retrieval strategies based 2021 on the current generation state - Monitors the generation process to identify ... when new information is needed"</quote_B> and <quote_B>"Implement an 2022 ... architecture that: - Dynamically adjusts retrieval strategies based on task 2023 ... complexity and information needs - Uses a feedback mechanism to evaluate the 2024 quality of retrieved documents - Maintains multiple retrieval 2025 strategies"</quote_B>. . . . 2026 2027 Both proposals include appropriate pilot mode configurations, but Proposal A's ... approach is more straightforward to implement and debug. <summary_A>Proposal A 2028 uses established toxicity detection methods with clear metrics and realistic ... human-in-the-loop simulation</summary_A>, while <summary_B>Proposal B requires 2030

... coordinating multiple complex components including real-time retrieval, adaptive

architectures, and integration mechanisms</summary_B>.

2031

2032

2037

2039

2040

2041

2043

2045

2046

2047 2048

2049

2051

2052

2053

2057

2058

2059

2060

2061

2062

2063

2064

2065

2066

2067

2068

2069

2071

2074

2075

2076

2077

2079

2080

2081

2082

2083

2084

2086

2087

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

</eval>
</eval>
```

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

```
Example extracted execution trace
 "summary_results":
{"summary": "This experiment tested whether dynamic sparsification tailored to
    syntactic complexity of queries and semantic density of documents would improve
    retrieval speed and accuracy compared to static sparsification methods on the
    MSMARCO dataset. The experiment used 200 queries with 730 documents in pilot
    mode. The baseline method employed static TF-IDF with BM25 scoring using a fixed
    1000 features, while the experimental method dynamically adjusted the number of
    features (500-1500) based on query complexity scores and applied density-aware
. . .
    similarity weighting. However, the results showed identical performance between
. . .
    both methods: both achieved exactly the same MRR (0.5384), Precision@1 (0.255),
    and Precision@5 (0.071). The only difference was processing time, with the
... dynamic method being slower (4.01ms vs 2.41ms). Statistical significance testing
... using bootstrap resampling with 10,000 iterations yielded p-values of 1.0 for
... all metrics, indicating no significant differences. The experiment was
    well-implemented and faithful to the original design, successfully calculating
. . .
    query syntactic complexity using NLTK-based metrics and document semantic
. . .
    density using LLM embeddings for the first 100 documents. The lack of
    improvement suggests that either the dynamic sparsification approach needs
. . .
    refinement, the complexity/density metrics were not sufficiently discriminative,
. . .
... or the MSMARCO dataset characteristics did not benefit from this particular
... adaptive approach.",
 "summary_short": "Dynamic sparsification showed no improvement over static methods
    on MSMARCO retrieval task.",
 "summary_medium": "The experiment compared dynamic sparsification (adapting features
    based on query complexity and document density) against static TF-IDF/BM25 on
    200 MSMARCO queries. Both methods achieved identical performance with
... MRR=0.5384, Precision@1=0.255, and Precision@5=0.071, with p-values of 1.0
... indicating no significant differences. The dynamic method was slower (4.01ms vs
... 2.41ms) without providing any accuracy benefits, suggesting the adaptive
    approach did not improve retrieval performance.",
 "hypothesis": "Dynamic sparsification tailored to syntactic complexity of queries
    and semantic density score of documents will significantly improve retrieval
    speed and accuracy compared to static sparsification methods.",
 "hypothesis_operationalized": "A TF-IDF based retrieval system that dynamically
    adjusts the number of features (500-1500) based on query syntactic complexity
    scores and applies density-aware similarity weighting will outperform a static
    TF-IDF system with fixed 1000 features on MSMARCO retrieval tasks, as measured
. . .
    by MRR, Precision@k, and processing time.", "hypothesis_inference": "The
    experimental results clearly reject the hypothesis. Both the dynamic and static
    sparsification methods achieved identical retrieval accuracy metrics (MRR,
. . .
... Precision@1, Precision@5), with statistical significance testing yielding
... p-values of 1.0, indicating no meaningful difference. Furthermore, the dynamic
... method was slower (4.01ms vs 2.41ms), contradicting the speed improvement
\ldots hypothesis. The results suggest that the proposed dynamic sparsification
    approach, at least as implemented, does not provide benefits over static methods
    for this task and dataset.",
```

```
2090
2091
           "hypothesis_category": "reject", "faithfullness_details": "The experiment was
              largely faithful to the original design. It successfully implemented both
               baseline (static TF-IDF/BM25) and experimental (dynamic sparsification) methods,
          ... calculated query syntactic complexity using NLTK-based metrics (parse tree
          ... depth, POS diversity, lexical diversity), and computed document semantic density
2095
          ... using LLM embeddings. The experiment used appropriate evaluation metrics (MRR,
2096
               Precision@k, processing time) and statistical testing (bootstrap resampling).
              However, there were some practical limitations: semantic density calculation was
          ... limited to the first 100 documents to control costs and time, and some documents
2098
          ... used fallback density scores of 0.5. The pilot mode with 200 queries was
          ... appropriate for initial testing. The implementation correctly followed the
2100
          ... experimental design with proper data splits, metric calculations, and result
               analysis.",
2101
           "faithfullness_category": "faithful",
2102
           "interesting_results": false,
2103
           "metadata_llm":
2104
           {"tokens_prompt": 51967, "tokens_completion": 963, "tokens_reasoning": 0,
               "tokens_total": 52930, "cost": 0.170346, "model": "claude-sonnet-4-20250514",
2105
               "temperature": 0.0, "max_tokens": 32000}, "errors": []},
2106
           "execution_success": "success",
2107
           "harpa_cost_efficiency":
2108
           "Used 1.6066855500000001 out of 10 allowed cost."
2109
           "complexity_score": "5 out of 25 reflections used.",
           "agent_latest_issues_handled": [{"issues": [], "summary_of_changes": []}, {"issues":
2110
          ... ["ERROR: MSMARCO dataset structure is different than expected - 'passages'
2111
          ... contains strings instead of dictionaries", "Need to add more debugging to
2112
          ... understand the actual data structure", "Need to handle the actual MSMARCO data
2113
               format correctly"],
           "summary_of_changes": ["Added extensive debugging to understand MSMARCO data
2114
              structure", "Fixed data loading logic to handle actual dataset format", "Added
2115
               error handling and fallback logic for data processing"]}]
2116
           }
2117
           }
2118
```

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

E HARPA SCORER PROMPTS

2119 2120

2121 2122

2123

2124 2125

2126 2127

2128

2130

21312132

2133

2134

2135

2136

2137 2138

2139

2140

2141

2142

2143

2144

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

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

```
2145
2146
               5. Require coordinated contributions from a team of experts to design, implement,
               ... and refine the idea.
2147
               6. Require multiple rounds of expert thinking and intervention to make the idea
2148
               ... work.
2149
2150
           ## END of description of ASD agent.
2151
           ### INTERNAL EXECUTION LOG - DO NOT REVEAL ###
2152
           <exec A>
2153
           {{EXEC_META_A}}
2154
           </exec_A>
2155
           <exec_B>
           {{EXEC_META_B}}
2156
           </exec_B>
2157
           ### END INTERNAL LOG ###
2158
2159
           ## CRITICAL: How to Use Execution Logs for Evaluation
2160
           The execution logs are PROVIDED FOR YOUR PRIVATE REASONING ONLY. You must:
2161
2162
           **D0**:
2163
           1. **Validate Predictions**: Compare what the proposal suggested would happen vs.
2164
           ... what actually happened
           2. **Identify Root Causes**: Understand WHY proposals succeeded or failed based on
2165
          ... execution details
2166
           3. **Learn Patterns**: Recognize proposal characteristics that correlate with
2167
          ... execution outcomes
2168
           4. **Generate Insights**: Explain what makes proposals more/less likely to succeed
2169
           **NOT DO**
2170
           1. **Reveal Execution Details**: Do not mention specific logs, results, or outcomes
2171
          ... in your response
2172
           2. **Reference Errors**: Avoid discussing specific errors, successes, or
2173
          ... implementation details from logs
           3. **Use Execution Phrases**: Don't use phrases like "based on execution" or "the
2174
          ... experiment failed/succeeded"
2175
           4. **Assume Knowledge**: Do not assume the reader has access to execution logs or
2176
          ... results
2177
2178
           ## Evaluation steps:
2179
           Please follow this process:
2180
           1. First, review the execution logs silently and think through whether each
2181
               experiment likely succeeded.
2182
           2. Then score each rubric item, noting whether any execution failure or runtime
          ... issue affected it.
2183
           3. Reflect whether complexity masked failure or whether execution flags were
2184
          ... underweighted.
2185
               - (Did the proposal fail because it was too complex for problems to be noticed
2186
              ... clearly? Or were there signs of failure that were seen but not taken
2187
               ... seriously enough?)
           4. Make a final decision based on overall score.
2188
2189
           Do not skip any step in this reasoning.
2190
2191
           1. Rate each proposal on these criteria (0.0 to 1.0):
2192
           - Use the Fixed rubrics as the evaluation criteria tailored to the Client's question
          ... and Proposal, enclosed in <rubric> ... </rubric> tags.
2193
2194
           The Fixed rubric items are:
2195
               1. Execution Success: Based on the proposal content, how likely is it to execute
2196
              ... successfully in the ASD Agent? (1.0 = yes (completed status), 0.0 = no (with
                   error or failed status))
2197
```

```
2201
               2. Complexity: Based on the proposal content, how complex would this be to
              ... implement? (1.0 = very complex (in terms of utlizing complete Reflection
                   Budget), 0.0 = very simple (utilizing less reflection budget))
               3. Cost Efficiency: Based on the proposal content, how cost-efficient would this
2204
              ... be to execute? (1.0 = \text{very cost-efficient}, 0.0 = \text{very costly})
2205
               4. Expected Hypothesis Validity: Based on the proposal content, how likely is
2206
                   the hypothesis to yield valid/conclusive results? (1.0 = very valid
                   (support/reject), 0.0 = not valid (inconclusive))
               5. Expected Interestingness: Based on the proposal content, how interesting
2208
              ... would the results be? (1.0 = \text{very interesting}, 0.0 = \text{not interesting})
               6. Faithfulness: Based on the proposal content, how faithfully can the ASD agent
2210
              ... be expected to execute the original intent? (1.0 = very faithful
              ... (faithful/deviations), 0.0 = not faithful (error))
2211
2212
           2. Provide Justification
2213
               - Assign reward scores using all available information
2214
               - Inside <rubric>, include a <justify> ... </justify> section explaining why you
2215
              ... chose those scores for the rubric criteria .
2216
           3. Compare both responses according to the rubric.
2217
           4. Provide your evaluation inside <eval> ... </eval> tags, quoting or summarizing
2218
          ... the Responses (Only Proposal Content) using the following tags:
2219
               - <quote_A> ... </quote_A> for direct quotes from Proposal A
               - <summary_A> ... </summary_A> for paraphrases of Proposal A
2221
               - <quote_B> ... </quote_B> for direct quotes from Proposal B
2222
               - <summary_B> ... </summary_B> for paraphrases of Proposal B
2223
2224
           5. Final Judgment
           - End with your final judgment in the format: <answer>[[A]]</answer> or
          ... <answer>[[B]]</answer>
2227
           ## Important Notes:
2228
               - You MAY read execution logs (if available) for private reasoning but MUST NOT
              ... reveal their contents
2229
               - Only quote/summarize the proposal texts in your evaluation
               - Base your judgment on both proposal quality AND how well you can predict
2231
              ... outcomes from proposal content
               - The execution logs (if available) are your "answer key" - use them to validate
              ... your reasoning
               - Do not let response order, length, or Response names affect your judgment.
2234
               - Follow the response format strictly.
2235
           Your output must follow the formats below:
           <rubric>
           detailed rubric items
           <justify> justification for the rubric </justify>
2240
           </rubric>
2241
2242
           <eval>
           include direct comparisons from proposal content supported by <quote_A>...</quote_A>
2243
          ... or <summary_A>...</summary_A>, and <quote_B>...</quote_B>, or
2244
          ... <summary_B>...</summary_B> tags
2245
           </eval>
2246
2247
           <answer>[[A/B]]</answer>
2248
```

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

2251

2255 System prompt for SFT dataset generation 2256 2257 Please act as an impartial evaluator and assess the testability or successful execution of the two research proposals generated by an Ideator to execute in an ... Automated Scientific Discovery (ASD) Agent. 2260 ## START of description of ASD agent: 2261 2262 The ASD Agent is an automated discovery system that writes Python-based experiments, 2263 executes them in containers, and analyzes results usually across multiple independent runs with a meta-analysis. More specifically, these automated scientific discovery systems operate by having 2265 ... code-based experimentation. They can generate code, run it, debug it, analyze 2266 results, create reports, and so forth. 2267 2268 What they can't do: 1. They can't run physical experiments (e.g. wet-lab experiments). 2269 2. They can't perform anything that requires human involvement (e.g. a human 2270 ... manually creating or rating data), because this would not be 2271 ... fully-automatic, and is out of scope. 2272 3. Conduct user studies (e.g., surveys, interviews, usability testing) 2273 4. Depend on real-world deployment or user-facing validation 5. Require coordinated contributions from a team of experts to design, implement, 2274 ... and refine the idea. 2275 6. Require multiple rounds of expert thinking and intervention to make the idea 2276 ... work. 2277 ## END of description of ASD agent. 2278 2279 ## Evaluation steps: 2280 Please follow this process: 2282 1. First, think through whether each experiment is likely to succeed. 2. Then score each rubric item based on the proposal content. 2283 3. Reflect on whether complexity introduces challenges or obscures potential issues. 2284 - (Might the proposal fail because it is too complex for potential issues to be 2285 ... noticed clearly? Or are there signs of risk that may not have been taken 2286 seriously enough?) 2287 4. Make a final decision based on overall score. Do not skip any step in this reasoning. 2289 2290 1. Rate each proposal on these criteria (0.0 to 1.0): 2291 - Use the Fixed rubrics as the evaluation criteria tailored to the Client's question ... and Proposal, enclosed in <rubric> ... </rubric> tags. 2292 The Fixed rubric items are: 2294 1. Execution Success: Based on the proposal content, how likely is it to execute 2295 ... successfully in the ASD Agent? (1.0 = yes (completed status), 0.0 = no (with 2296 ... error or failed status)) 2. Complexity: Based on the proposal content, how complex would this be to 2297 ... implement? (1.0 = very complex (in terms of utlizing complete Reflection Budget), 0.0 = very simple (utilizing less reflection budget)) 2299 3. Cost Efficiency: Based on the proposal content, how cost-efficient would this 2300 be to execute? (1.0 = very cost-efficient, 0.0 = very costly) 2301 4. Expected Hypothesis Validity: Based on the proposal content, how likely is ... the hypothesis to yield valid/conclusive results? (1.0 = very valid ... (support/reject), 0.0 = not valid (inconclusive)) 2303 5. Expected Interestingness: Based on the proposal content, how interesting 2304 ... would the results be? (1.0 = very interesting, 0.0 = not interesting)6. Faithfulness: Based on the proposal content, how faithfully can the ASD agent ... be expected to execute the original intent? (1.0 = very faithful 2306 ... (faithful/deviations), 0.0 = not faithful (error)) 2307

2346

2347

23492350

2351

2352 2353

2354

2355

2356

23572358

2359

2361

2362

```
2310
2311
           2. Provide Justification
               - Assign reward scores using all available information
2312
               - Inside <rubric>, include a <justify> ... </justify> section explaining why you
               ... chose those scores for the rubric criteria .
2314
2315
           3. Compare both responses according to the rubric.
2316
           4. Provide your evaluation inside <eval> ... </eval> tags, quoting or summarizing
          ... the Responses (Only Proposal Content) using the following tags:
2318
               - <quote_A> ... </quote_A> for direct quotes from Proposal A
               - <summary_A> ... </summary_A> for paraphrases of Proposal A
               - <quote_B> ... </quote_B> for direct quotes from Proposal B
2320
               - <summary_B> ... </summary_B> for paraphrases of Proposal B
2321
2322
           5. Final Judgment
           - End with your final judgment in the format: <answer>[[A]]</answer> or
2324
          ... <answer>[[B]]</answer>
           ## Important Notes:
2326
               - Base your judgment on the the Fixed rubrics as the evaluation criteria
2327
               - Only quote/summarize the proposal texts in your evaluation
2328
               - Base your judgment on both proposal quality AND how well you can predict
               ... outcomes from proposal content
               - Do not let response order, length, or Response names affect your judgment.
               - Follow the response format strictly.
2331
2332
           Your output must follow the formats below:
2333
2334
           <rubric>
           detailed rubric items
           <justify> justification for the rubric </justify>
           </rubric>
2337
2338
           <eval>
           include direct comparisons from proposal content supported by <quote_A>...</quote_A>
2339
          ... or <summary_A>...</summary_A>, and <quote_B>...</quote_B>, or
          ... <summary_B>...</summary_B> tags
2341
           </eval>
2342
2343
           <answer>[[A/B]]</answer>
2344
```

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.
```

2369 2370

2371

2373 2374

2375

2376

2377 2378

2379

2381 2382

2383

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

2386 Baseline predicted reasoning trace 2387 2388 <rubric> 2389 1. Execution Success: Based on the proposal content, how likely is it to execute 2390 ... successfully in the ASD Agent? 2391 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 2392 ... to execute? 2393 4. Expected Hypothesis Validity: Based on the proposal content, how likely is the 2394 hypothesis to yield valid/conclusive results? 5. Expected Interestingness: Based on the proposal content, how interesting would 2396 ... the results be? 6. Faithfulness: Based on the proposal content, how faithfully can the ASD agent be 2397 ... expected to execute the original intent? <iustifv> 2399 1. **Execution Success**: Both proposals are detailed and involve clear steps that 2400 ... can be automated, but Proposal A has a more structured experimental design with ... multiple controls and well-defined variables, potentially leading to clearer 2401 2402 2. **Complexity**: Proposal A is more complex, involving multiple adaptive 2403 ... strategies (CAR and PATS) and a detailed operationalization plan. However, 2404 \dots Proposal B involves integrating a dynamic reward model with GFlowNets, which 2405 ... might also be complex. 3. **Cost Efficiency**: Both proposals are likely to be fairly costly due to the 2406 ... need for in-depth experiments, but Proposal A may benefit from modular 2407 ... implementation, making it potentially more cost-efficient. 2408 4. **Expected Hypothesis Validity**: Proposal A is stronger in terms of hypothesis 2409 ... validity as it involves a direct comparison of multiple conditions, providing a ... robust framework to test the hypothesis. 2410 5. **Expected Interestingness**: Both proposals are interesting, but Proposal A 2411 ... offers a clear pathway to both enhanced accuracy and efficiency, which could be 2412 ... more fascinating to researchers. 2413 6. **Faithfulness**: Both proposals are faithful to their intentions, but Proposal A 2414 ... provides a more detailed and structured implementation plan, which can be more ... closely followed by the ASD agent. 2415 </justify> 2416 </rubric> 2417 2418 <eval> 2419

```
2420
2421
           <quote_A>The experiment should include three conditions: (1) a baseline with fixed
               reasoning strategy, (2) CAR-only implementation, and (3) integrated CAR+PATS
2422
               implementation. The experiment should be structured to clearly demonstrate
               whether the integrated CAR+PATS approach outperforms both the baseline and
2424
               CAR-only approaches in terms of accuracy and computational efficiency.</quote_A>
2425
           <quote_B>This experiment will compare three approaches for mathematical reasoning:
2426
               1. **Baseline 1 (Static Reward Model)**, 2. **Baseline 2 (PPO)**, and 3.
               **Experimental (Dynamic Reward GFlowNets)**. The Dynamic Reward GFlowNets
2427
               approach is expected to enhance both reasoning accuracy and diversity
2428
               significantly compared to traditional methods.</quote_B>
2429
2430
           Proposal A is more structured with clear conditions and detailed steps for each
               phase of the experiment, ensuring a faithful and successful execution. Proposal
2431
               B, while interesting, lacks a detailed structured approach and relies more on a
2432
               high-level comparison, which might lead to less clarity in results.
2433
2434
           In terms of complexity, both proposals are complex, but Proposal A is more modular
2435
               and structured, making it easier to implement each component in the ASD Agent.
               The cost efficiency is similar for both, but Proposal A might be slightly more
2436
               cost-efficient due to its structured approach, simplifying the execution
          . . .
2437
               process.
2438
2439
           Hypothesis validity is stronger in Proposal A due to the direct comparison of
               conditions, providing a clear framework to test the hypothesis. The expected
               interestingness is high for both, but Proposal A may be marginally more
2441
               interesting due to its enhanced focus on computational efficiency alongside
2442
              accuracy.
          . . .
2443
2444
           Faithfulness to the original intent is high for both, but Proposal A provides a
          ... detailed operationalization plan, making it more easily executed by the ASD
2445
              Agent.
2446
           </eval>
2447
2448
           <answer>[[A]]</answer>
2449
```

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

F HARPA PROMPTS

2475

2476

2477

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

```
2478
           Example part of HARPA artefact illustrating key variable and value space
2479
2480
           {
  "key_variables": [
2481
2482
                   "name": "Energy Consumption",
2483
                   "source_paper": "An Enhanced Multi-Objective Evolutionary Algorithm with
2484
                  ... Reinforcement Learning for Energy-Efficient Scheduling in the Flexible
2485
                  ... Job Shop".
2486
                   "type": "measurable",
                    "definition": "The total amount of energy consumed by the job shop
2487
                  ... operations during the scheduling process."
2488
                   "importance": "This variable is central to the hypothesis as it directly
2489
                  ... relates to the goal of reducing energy usage through the proposed
2490
                   ... framework."
2491
                    "specific_details": "Energy consumption can be measured in kilowatt-hours
2492
                  ... (kWh) and evaluated using sensors or energy meters attached to machines.
                       The evaluation can include machine energy consumption, workshop energy
2493
                       consumption, and can be benchmarked against traditional methods that do
2494
                       not optimize for energy efficiency."
                  . . .
2495
                 },
2496
                   "name": "Makespan",
2497
                   "source_paper": "A Q-Learning Rescheduling Approach to the Flexible Job Shop
2498
                  ... Problem Combining Energy and Productivity Objectives",
2499
                   "type": "measurable".
2500
                   "definition": "The total time required to complete all scheduled jobs in the
2501
                   ... job shop environment.",
                   "importance": "Makespan is a critical performance metric in scheduling,
2502
                  ... reflecting the efficiency of the scheduling method in minimizing
                  ... completion time."
2504
                   "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
2506
                       against benchmarks from traditional scheduling methods.
2507
                 },
                   "name": "Multi-Agent Reinforcement Learning Framework",
                    "source_paper": "A multi objective collaborative reinforcement learning
2510
                  ... algorithm for flexible job shop scheduling",
2511
                   "type": "design-choice",
                   "definition": "A framework that uses multiple agents to learn and make
2513
                    . scheduling decisions in a flexible job shop environment."
2514
                   "importance": "This framework is essential for implementing the hypothesis,
                  ... as it is proposed to dynamically optimize scheduling for energy
2515
                       efficiency and makespan reduction."
                   "specific_details": "The framework can be implemented using algorithms like
2517
                  ... Q-learning, PPO, or D3QN. Agents can be designed to handle job and
2518
                  ... machine selection, with state features represented using disjunctive
2519
                  ... graphs. The framework's effectiveness can be evaluated through
                       simulation and comparison with traditional methods.'
2520
                  . . .
                 },
2521
                 {
                   "name": "Traditional Scheduling Methods",
                   "source_paper": "A Multi-Agent Reinforcement Learning Approach to the
2524
                  ... Dynamic Job Shop Scheduling Problem",
                   "type": "design-choice",
                   "definition": "Conventional scheduling methods that do not incorporate
2526
                   ... energy efficiency as a primary objective.",
2527
                   "importance": "These methods serve as a baseline for comparison to evaluate
                  ... the effectiveness of the proposed framework.",
```

```
2530
2531
                    "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."
                 },
2535
                 {
                   "name": "Dynamic Flexible Job Shop Environment",
2536
                    "source_paper": "Dynamic Agent-based Bi-objective Robustness for Tardiness
                   ... and Energy in a Dynamic Flexible Job Shop",
2538
                   "type": "design-choice",
2539
                   "definition": "A job shop environment characterized by variability and
2540
                    .. uncertainty in job arrivals and machine availability.",
                   "importance": "This environment is crucial for testing the hypothesis, as it
2541
                  ... reflects real-world conditions where the proposed framework is expected
2542
                   ... to perform.",
2543
                   "specific_details": "The environment can be simulated using discrete-event
2544
                  ... simulation tools, incorporating elements like random job arrivals,
2545
                  ... machine failures, and dynamic job priorities. The setup allows for
                  ... testing the adaptability and robustness of the scheduling framework."
2546
2547
2548
                "variable_space": [
                   "Energy Consumption": [
2551
                       "value_name": "Machine Energy Consumption",
2552
                       "source_paper": "An Enhanced Multi-Objective Evolutionary Algorithm with
2553
                       ... Reinforcement Learning for Energy-Efficient Scheduling in the
2554
                           Flexible Job Shop",
                       "confidence": "High",
2555
                       "is_alternative": false,
                       "specific_details": "Machine energy consumption refers to the energy
2557
                       ... used by individual machines during their operation in the job shop.
2558
                           This can be measured using energy meters attached to each machine,
                            which track the kilowatt-hours (kWh) consumed. The study by Lu et al.
2559
                            established a multi-objective integer programming model that
                           includes machine energy consumption as a key objective. The model
2561
                           aims to minimize this consumption by optimizing the scheduling of
                           tasks across machines, considering factors like machine start-up and
2563
                           shutdown times. Compatible models include those that can integrate
                           with energy meters and support real-time data collection, such as
2564
                            systems using IoT-enabled devices. The baseline comparator for this
                       . . .
2565
                            value is traditional scheduling methods that do not account for
                            energy efficiency, typically resulting in higher energy usage."
                       . . .
2567
                     },
                     {
                       "value_name": "Workshop Energy Consumption",
                       "source_paper": "An Enhanced Multi-Objective Evolutionary Algorithm with
2570
                       ... Reinforcement Learning for Energy-Efficient Scheduling in the
                       ... Flexible Job Shop",
                       "confidence": "High",
2572
                       "is_alternative": false,
2573
2574
```

```
2585
                        "specific_details": "Workshop energy consumption encompasses the total
2586
                           energy used by all machines and processes within the job shop. This
2587
                           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
2590
                           by optimizing the overall scheduling strategy, using reinforcement
2591
                           learning to dynamically adjust parameters and improve energy
                           efficiency. Measurement techniques involve aggregating data from
                           multiple energy meters and sensors throughout the workshop.
2593
                           Compatible models are those that can handle large-scale data
2594
                           integration and real-time adjustments, such as systems using
2595
                           advanced analytics platforms. The baseline comparator is again
                           traditional scheduling methods that do not optimize for energy
2596
                           efficiency, leading to higher overall energy consumption."
2597
                     },
2598
                     //...
                     ],
2601
                    "Traditional Scheduling Methods": [
2602
2603
                        "value_name": "First-In-First-Out (FIFO)",
                       "source_paper": "A Multi-Agent Reinforcement Learning Approach to the
2604
                       ... Dynamic Job Shop Scheduling Problem",
2605
                        "confidence": "High"
2606
                        "is_alternative": false,
2607
                        "specific_details": "FIFO is a traditional scheduling method where the
2608
                           jobs are processed in the order they arrive at the job shop. This
2609
                           method does not consider job priority or energy consumption, making
                           it a straightforward but potentially inefficient approach. In the
2610
                           context of job shop scheduling, FIFO serves as a baseline for
2611
                           evaluating more advanced scheduling techniques. The method is
2612
                           typically implemented in simulation environments to provide a
2613
                           benchmark for comparison. Compatible models include any discrete
                           event simulation model that can handle job arrival and processing
2614
                           sequences. The baseline comparator for FIFO is often more
2615
                           sophisticated scheduling algorithms that incorporate dynamic
2616
                           decision-making and energy efficiency considerations."
                      . . .
2617
                     },
2618
                       "value_name": "Shortest Processing Time (SPT)",
2619
                        "source_paper": "A Multi-Agent Reinforcement Learning Approach to the
2620
                       ... Dynamic Job Shop Scheduling Problem",
2621
                       "confidence": "High"
2622
                       "is_alternative": false,
                       "specific_details": "SPT prioritizes jobs with the shortest processing
2623
                           time, aiming to minimize the average job completion time. This
2624
                           method does not account for energy consumption or job arrival times,
2625
                           focusing solely on processing efficiency. In practice, SPT can be
                           implemented using a priority queue where jobs are sorted by their
2627
                           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.
2629
                           Compatible models include those that can dynamically sort and
2630
                           prioritize jobs based on processing time. The baseline comparator is
2631
                           typically a more comprehensive scheduling strategy that considers
                      . . .
2632
                           multiple objectives."
                     },
2633
2634
                        "value_name": "Earliest Due Date (EDD)",
2635
                        "source_paper": "A Multi-Agent Reinforcement Learning Approach to the
2636
                       ... Dynamic Job Shop Scheduling Problem",
                        "confidence": "High"
2637
                        "is_alternative": false,
2638
2639
```

2658

```
2640
2641
                        "specific_details": "EDD schedules jobs based on their due dates, with
                           the goal of minimizing tardiness. This method does not consider
2642
                            energy consumption or processing time, focusing instead on meeting
                            deadlines. EDD can be implemented using a scheduling algorithm that
2644
                            sorts jobs by their due dates and assigns them to machines
2645
                            accordingly. This method is often used in environments where meeting
2646
                            delivery deadlines is critical. Compatible models include those that
                            can handle job prioritization based on due dates. The baseline
2647
                            comparator is typically a scheduling method that incorporates
2648
                            additional factors such as energy consumption and processing
                            efficiency."
                       . . .
2650
                      },
                      //...
2651
                      ],
2652
                      //...
2653
               }
2654
               ]
2655
           }
2656
```

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
2661
2662
           agent_capabilties = f"""
2663
                   The ASD Agent is an automated discovery system that writes Python-based
                       experiments, executes them in containers, and analyzes results usually
2665
                       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
2668
                       novelty and scientific rigour. The result should be suitable for a
2669
                       conference paper.
2670
                   AGENT CONSTRAINTS & CAPABILITIES:
2671
                   - The ASD Agent writes Python-based experiments and executes them in
2672
                       containers
2673
                   - Typically runs 5 independent experiments with meta-analysis
2674
                   - Target audience: Undergrad/MSc/PhD student implementation level
2675
                   - Output should be suitable for workshop or conference paper submission
                   - NO manual human ratings (considered 'external major effort')
2676
                   - NO model fine-tuning or pretraining
                   - NO access to external or private datasets
2678
                   - Must use only existing codeblocks and buildable logic
2679
                   - All experiments must be fully implementable in Python
2680
               system_message = f"""You are a clever AI research scientist with limited
2682
                   resources, whose primary goal is to identify promising, new, and key
2684
               problems based on existing scientific literature, in order to aid researchers in
                   discovering novel
2685
               and significant research opportunities that can advance the field."""
2686
               user_message = f"""You are a clever AI research scientist with limited resources
2687
                   tasked with generating novel research problems based on existing scientific
                   literature. Your goal is to aid an autonomous discovery agent in identifying
2689
                   significant research opportunities that can advance the field.
               You are going to generate a research problem that should be original, clear,
2691
                   feasible, relevant, and significant to its field. This will be based on the
2692
                   title and abstract of the source paper, those of {len(citing_paper_list)}
2693
                   related papers in the existing literature.
2694
```

```
2695
2696
               IMPORTANT: When evaluating feasibility and outlining the testing approach,
2697
              ... consider the following agent-specific information:
2698
               ```{agent_capabilties}`
2699
2700
 Now, let's start with the research problem generation task.
2701
 1. Understanding of the source paper, and the related papers is essential:
 - The source paper is the primary research study you aim to enhance or build
2702
 ... upon through future
2703
 research, serving as the central source and focus for identifying and developing
2704
 ... the specific
2705
 research problem.
 - The related papers are arranged in temporal order of citation, such that paper
2706
 ... 2 cites paper 1 and
2707
 paper 3 cites paper 2 and so on. The relevant papers provide additional context
2708
 and insights that are essential for
2709
 understanding and expanding upon the source paper. However, all the papers in
2710
 ... the list may not be relevant to the primary
 research you are focusing on.
2711
2712
 2. Your approach should be systematic:
2713
 - Start by thoroughly reading the title and abstract of the source paper to
 ... understand its core focus.
2714
 - Next, proceed to read the titles and abstracts of the related papers in the
2715
 ... order in which they appear in the list. Each related paper is accompanied by
2716
 ... an explanation of its relevance to the previous paper, with the first
2717
 ... related paper considering the source paper as the previous paper.
2718
 Identify the papers that form a logical reasoning chain starting from the source
2719
 ... paper.
 - Use only these papers to gain a broader perspective about the progression of
2720
 ... the primary research topic over time.
2721
2722
 Note that your research idea and hypothesis MUST be testable using the AGENT with
2723
 ... these specific capabilities:
 When evaluating feasibility and outlining the testing approach, consider the
2724
 ... following agent-specific information. Manual human ratings in the research
2725
 ... (e.g. human rating of the quality of generated text from an experiment) is
2726
 ... considered an `external` resource of `major` effort, for the purposes of the
2727
 potential research experiments, and should generally be avoided (unless
 ... absolutely required for the research).
2728
2729
 IMPORTANT: The hypothesis should be implementable in Python, using the above or
2730
 ... other functions. Don't suggest a task that requires skills that cannot be
2731
 implemented, e.g., human studies. Don't suggest a task that requires access
2732
 ... 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
2733
 ... resources for such experiments.
2734
2735
 Now, I am going to provide the source paper and related papers as an enumerated
2736
 ... list of Title, Abstract and Year of publication
2737
 triple, as follows:
 Source paper title: {source_paper['title']}
 Source paper abstract: {source_paper['abstract']}
2739
 Source paper year of publication: {source_paper['year']}
2740
 Related papers: {citing_paper_list}
2741
 With the provided source paper, and the related papers, your objective now is to
 ... formulate a
2742
 research problem that not only builds upon these existing studies but also
2743
 ... strives to be original, clear, feasible, relevant, and significant. Before
2744
 ... crafting the research problem, revisit the title and abstract of the target
2745
 ... paper, to ensure it remains the focal point of your research problem
2746
 ... identification process.
2747
```

```
2750
2751
 Now convert this idea into a concrete testable hypothesis. Remember hypothesis
 ... is a declarative statement expressing a
2752
 relationship between two variables like independent or dependent variables or
 ... left group and rigt group in a given context.
2754
 Your hypothesis should contain the key variable or variables from your research
2755
2756
 Source paper title: {source_paper['title']}
 Source paper abstract: {source_paper['abstract']}
2758
2759
 Remember that a hypothesis is a declarative statement expressing a relationship
2760
 \dots between two variables (e.g., independent and dependent variables) in a given
 context. Your refined hypothesis should contain the key variables from your
2761
 research idea.
2762
2763
 Then, following your review of the above content, please proceed to analyze the
2764
 ... progression of the research topic. Now output this analysis, the research
2765
 ... idea and hypothesis with the rationale.
 Your output should be a valid JSON with the following fields.
2766
 Output a JSON object in the following format:
2767
 json
2768
 {{
2769
 "Analysis": {{Output a dictionary with each paper in the Related Papers as a key.
 For each key (paper) analyze how this paper builds upon the previous papers
2770
 ... in the list. For example, how Paper 0 builds upon source paper and Paper 1
2771
 ... builds upon the concepts in Paper 0 and so on. Elaborate on specific
2772
 ... advancements made, including the explanation behind their effectiveness in
2773
 ... addressing previous challenges. Apply this analytical approach to each valid
2774
 paper in the sequence, adding the analysis as the value for each key in a
 ... few sentences. Ignore papers that do not build upon the previous papers and
2775
 ... diverge from the original source paper's topic significantly.}},
2776
 "Rationale": "Summarize the above analysis and explain how you would come up
2777
 ... with a research idea that will advance the field of work while addressing
2778
 ... the limitations of previous work and building upon the existing work.",
 "Research idea": "Delineate an elaborate research problem here including the key
2779
 ... variables."
2780
 "Hypothesis": "Provide a concrete testable hypothesis that follows from the
2781
 ... above research problem here"
2782
 }}
2783
 This JSON will be automatically parsed, so ensure the format is precise.
2784
2785
```

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

### Generalize Hypothesis for Literature Search

2786

2787

2788 2789

2790 2791

2792

2793

2794

2795

2796

2797

2798

2799

2801

2802

2803

```
This is an automated scientific discovery task, with the overall goal of trying to ... assess the novelty of scientific claims.

Background

If you think about it, nearly every experiment could be considered novel if you make ... the claims specific enough -- for example, performing a well-known experiment on ... a specific day, or getting very specific values from the experiment.

The purpose of your task is to take an input claim, and progressively rewrite it as ... several (progressively more general) claims.

Another system will assess the novelty of these generalized claims, allowing us to ... detect not simply whether a claim is novel or not, but how specific a claim has ... to be before it's considered novel.

Specific task
You will be given a claim (below), and your task will be to generate 4 progressively ... more generalied versions of that claim.
```

```
2805
2806
 # 7 Examples of the Generalization Process
2807
 Below are 7 examples of the generalization process (represented in JSON), to help
 ... you understand the task.
2809
 - The keys represent names for the 7 different claim examples.
2810
 - The value is a list of the (progressively more generalized) claims.
2811
 - The 'generalization' key represents the level of generalization (0 is the original
 ... claim).
2812
 - The 'claim' key represents the claim itself.
2813
2814
 <Add here few-shot examples>
2815
2816
 # Claim to generalize
2817
 The claim to generalize is:
2818
 <original_claim>
2819
 # What should I do if the claim above has multiple claims?
2821
 - If the claim above has multiple claims, you should pick the single most salient
2822
 ... claim, and generalize it.
2823
2824
 # Output format:
 - Output in JSON format, as above
 - You should output a dictionary with a single key (a few-word summarized version of
2826
 ... the claim)
2827
 - The value should be a list of 4 progressively more generalized versions of the
2828
2829
 - The 'generalization' key should be an integer from 0 to 3, representing the level
 of generalization (0 is the original claim)
2830
 - The 'claim' key should be the claim itself
2832
 Please output your JSON response between a single code block (```), as it will be
2833
 ... 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.
2834
```

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

## Generate hypothesis specific questions

2836 2837

2845

2847

2849

2850

2851

2853 2854

2856

```
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.

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.

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?

First, carefully read the following hypothesis: {hypothesis}

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

```
2861
 Your goal is to efficiently analyze the hypothesis and generate 20 concise,
 focused questions that will help researchers refine and operationalize it
 into something implementable and testable. Each question should clearly
 target a part of the hypothesis (e.g., variable, measure, assumption, or
 outcome). Mention which part you're refining (e.g., IV, DV, comparison
2865
 group, comparison variable, operationalization, feasibility).
2866
 You can make the QA generation more useful by asking the model to *aim* each
 ... question at helping answer/refine one of these:
2868
 refined_hypothesis`
 - `key_variables`
2870
 - `research_idea_required_code_and_resources`
 - `research_idea_external_requirements`
2871
 - `testing_approach`
2872
2873
 Before generating your 20 questions, reflect on the hypothesis using these
2874
 ... guiding prompts:
2875
 1. What are the key terms and variables involved?
2876
 2. How can each component be operationalized and measured?
2877
 3. What capabilities from the system are most relevant?
2878
 4. What design setups or tasks could support testing?
2879
 5. What might hinder testing \mid e.g., feasibility, confounds, or constraints?
 6. What would success look like, and how could it be quantified?
 7. What ethical or resource considerations exist?
2881
2882
 Use these reflections to inform the questions you write, ensuring they are
2883
 well-grounded and cover diverse aspects of hypothesis development and
 testing.
2885
 Where possible, generate questions that will later help produce values for:
 a more specific and testable `refined_hypothesis`
2887
 - a list of `key_variables` (IVs, DVs, controls, comparison group, comparison
2888
 ... variables, etc.)
 - a list of code/resources in `research_idea_required_code_and_resources`
2889

 package or library requirements

 - testing/evaluation structure (`testing_approach`)
2891
 Present your questions in the following format:
               ```json
2894
               {{
                   "questions": [
                       {{
                            "question": "[Your first question here]"
                       }},
                       {{
                            "question": "[Your second question here]"
2900
                       }},
2901
2902
                       {{
                           "question": "[Your twentith question here]"
                       }}
2904
                       ٦
2905
               }}
2906
2907
           Remember, your analysis and questions should be designed to provide researchers with
2908
               the necessary information to design and implement a robust study testing the
2909
               given hypothesis. Strive for clarity and conciseness in both your analysis and
2910
               questions to ensure the task and results are crisp and easily actionable.
2911
2912
```

Listing 10: Generate at least 20 questions to refine the preliminary hypothesis H to H'

2915 Refine Hypothesis based on Socratic QA 2916 2917 You are an expert scientific researcher tasked with refining a given hypothesis to 2918 ... make it more specific, easily testable, and practically feasible. This process 2919 ... is crucial in scientific research as it helps in designing experiments and ... studies that can effectively validate or invalidate the hypothesis. 2920 2921 Here is the original hypothesis you need to refine: 2922 2923 Initial Hypothesis: {initial_hypothesis} 2924 Here are related papers with title and key passages that may directly inform or 2925 ... relate to the hypothesis. 2926 Provenance papers: {relevant_paper_list} 2927 2928 Your task is to come up with new refined research hypothesis, and follow-on research ... ideas, based on the research questions, research programs, hypotheses, 2929 operationalizations of experiments, or any other information provided in these ... paper excerpts. 2931 You can use content from one paper, or combine content from multiple papers to 2932 ... generate new ideas. 2933 Similar papers: {similar_paper_list} 2934 Your task is to refine this hypothesis by making it more specific, ensuring it is 2935 ... testable, and evaluating its practical feasibility. 2937 Answer the following 20 clarifying questions to help sharpen the hypothesis: 2938 {questions} 2939 Use the insights from the provenance and similar paper excerpts to support and 2940 justify your answers wherever applicable. Before providing your final output, 2941 wrap your thought process in 'thoughts' of the output JSON. Include the 2942 ... following subsections: 1. **Initial Analysis** | Break down the hypothesis: variables, assumptions, and 2943 ... relationships. 2944 2. **Related Literature** | Quote and summarize relevant insights from similar 2945 ... papers. List testable variables from them. 2946 3. **Specificity** | Suggest ways to make the hypothesis more concrete. Rank by 2947 ... specificity. 4. **Testability** | Propose 2-3 test designs, list what to measure and possible 2948 ... challenges. 2949 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. 2951 2952 6. **Testing Approach** | Outline how the hypothesis could be tested using available 2953 ... agent tools only (no external data, no human evals, no fine-tuning, no 2954 ... model-training). 2955 7. **Final Refinement** | Synthesize the answers of clarifying questions and above ... considerations to create a refined, specific, testable, and feasible version of 2956 ... initial hypothesis. 2957 2959 IMPORTANT: When evaluating feasibility and outlining the testing approach, consider 2960 ... the following agent-specific information: 2961 {agent_capabilties} 2962 IMPORTANT: The hypothesis should be implementable in Python, using the above or 2963 other functions. Don't suggest a task that requires skills that cannot be 2964 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 2965 ... involve pretraining or fine-tuning models, as you do not have the resources for 2966 ... such experiments.

3012

3014

3015

3016 3017

3018 3019

3022 3023

3024

```
2970
2971
            Remember that a hypothesis is a declarative statement expressing a relationship
                between two variables (e.g., independent and dependent variables) in a given
2972
                 context. Your refined hypothesis should contain the key variables from your
                research idea.
2974
2975
            Ensure each answer is supported by information from the hypothesis, agent
2976
                 capabilities, or provided papers. If an answer cannot be derived, explain what
                information is missing.
2978
            Example output structure (this is a generic example to illustrate the format):
2979
            ```json
2980
 {{
 "thoughts": {{
2981
2982
 "Initial Analysis": "...",
2983
 "Similar Papers": "...",
 "Specificity Improvements": "...",
"Testability Considerations": "...",
 "Measurability and Feasibility": "...",
2986
 "Testing Approach": "...", "Final Refinement": "...",
2987
2988
 "Clarifying Questions & Answers": \{\{
 "Q1": "Answer to question 1"
 "Q2": "Answer to question 2",
2991
 "Q20": "Answer to question 20"
2992
 }}
2993
 }},
 "refined_hypothesis": "Provide a concrete testable hypothesis",
 "key_variables": [list of key variables],
2995
 "research_idea_required_code_and_resources": [
 {{
2997
 "name": "Example Resource",
 "description": "Brief description of the resource",
2998
 "where": "One of: 'existing codeblock', 'external', or 'build'", "effort": "One of: 'minor', 'moderate', or 'major'"
2999
 }}
3001
],
 "research_idea_external_requirements": [
 "example_package (for specific purpose)"
3004
 ٦.
 }}
3007
```

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

# 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:

```
3025
3026
 {similar_retreived_papers}
3027
 Your task is to analyze this hypothesis and the related papers to extract key
3028
 ... variables. Follow these steps:
3030
 1. Analyze the hypothesis explicitly and systematically extract key variables:
3031
 - Clearly identify every explicitly mentioned variable or design-level choices
 within the hypothesis as a distinct key variable. This includes quantifiable
3032
 variables and design-level choices.
3033
 - Convert any implicit or abstract concepts (e.g., performance, reliability,
3034
 ... robustness) into clearly defined and measurable variables or implementable
3035
 ... design choices. Do not include vague or unmeasurable conceptual ideas unless
 they are clearly defined in operational terms and when they are central to the
3036
 . . .
 hypothesis.
3037
 - Ensure key variables have either measurable, quantifiable properties, such as
3038
 ... "Model Training Time (seconds)," "Error Rate (%)," or "Knowledge Retention
3039
 ... Score." Or a design choice that affects implementation or evaluation (e.g., "Use
 ... of pretraining dataset X", "Fine-tuning vs. zero-shot prompting")
3040
 - Provide a precise, measurable definition (one sentence) for each identified
3041
 ... key variable. Explicitly define how each key variable should be measured,
3042
 stating exact metrics, evaluation criteria, or assessment methods clearly
3043
 ... and concisely.
3044
 - For design choices, define what the choice is, its implications, and how it
 ... would be implemented or varied in an experiment.
 - Example design choices include memory architecture (e.g., episodic memory,
 ... fact-memory modules), prompt strategy (e.g., few-shot, chain-of-thought),
3047
 ... retrieval method (e.g., top-k, semantic retrieval), narrative control
3048
 ... mechanism (e.g., branching storylets, story graphs), or model integration
 choices (e.g., use of fine-tuned GPT-3 vs. GPT-4). These should be specific
3049
 \dots and tied to actual implementation decisions that can affect the system
3050
 ... behavior or experimental outcome.
 - Include relevant experiment-level factors (e.g., dataset choice, baseline models,
3052
 ... training configurations) as variables if they impact testing the hypothesis
3053
 - Do not omit any explicitly mentioned concept from the hypothesis.
3054
 2. Review the similar papers:
3055
 - Extract relevant quotes.
3056
 - Analyze how each quote relates to the hypothesis.
3057
 - Identify specific and testable variables or design choices from the quotes.
3058
 3. For each key variable:
3059
 - Clearly define how it should be measured or implemented (in `specific_details`).
3060
 - Indicate the type of variable using "type": "measurable" or "type":
3061
 "design-choice" in the output.
3062
 - Determine whether the key variable is **explicitly mentioned** in related work or
3063
 ... if it is inferred.
 a. Mark variables found in paper excerpts with their exact paper title and
3064
 ... include page/section if available
3065
 b. Mark variables as 'LLM-recommended' only if not supported by provided papers
3066
 - In specific_details, provide:
3067
 a. For measurable variables: metrics, methods of evaluation, potential value
 ... ranges, and example benchmarks
3068
 b. For design choices: the specific options or configurations, how they can be
3069
 ... varied, how they impact implementation, and any relevant examples or
3070
 ... baselines
3071
 Remember to focus solely on analyzing the given hypothesis, identifying key
 variables, and extracting specific value options from the similar papers. Do not
3073
 attempt to refine or improve the hypothesis.
3074
3075
 Your final output should be structured clearly and explicitly to enhance
3076
 ... interpretability. Follow this JSON format strictly:
3077
3078
```

3081

3085

3086

3088

3090

3091

3092

3093

3096 3097 3098

3100 3101

3102 3103

3104

3105 3106

3107

3108 3109

3110

3111 3112

3113

3114

3116 3117

3118

3119

3120

3121

3122

3123

3124

3125

3126

3127

3128

3129

3130

3131

3132

3133

3134

```
```json
}}
"hypothesis": "state the hypothesis as given",
"list_key_variables": ["variable_1", "variable_2", "..."],
"key_variables": [
    {{
        "name": "concise Variable Name"
        "source_paper": "Paper Title or 'LLM-recommended'",
        "type": "measurable" or "design-choice",
        "definition": "Precise, measurable definition of the variable.",
        "importance": "Brief explanation of why this variable matters to the
       ... hypothesis."
        "specific_details": "Detailed information on measurement techniques,
       ... potential value ranges, and specific examples of implementation,
            elaborated with information from related passages."
   }}
]
}}
```

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**

```
3135
3136
                       a. Prioritize the most relevant values to the hypothesis if there are
                       ... many (>15) options
3137
                       b. Mark variable values found in paper excerpts with their exact paper
3138
                       ... title and include page/section if available
3139
                       c. Mark values as `LLM-recommended` only if not clearly supported by
3140
                       ... provided papers
3141
                       d. Prioritize values directly sourced from provided papers over
                       ... LLM-generated suggestions
3142
                       e. Assign confidence levels using these criteria:
3143
                           - High: Values explicitly mentioned in papers with detailed
3144
                          ... implementation information available
3145
                           - Medium: Values that can be reasonably inferred from the papers but
                          ... aren't explicitly stated
3146
                           - Low: Values that may be applicable based on general domain
3147
                           ... knowledge but aren't explicitly mentioned in papers
3148
                       f. Include concrete examples or parameter ranges for specificity
3149
                       g. DO NOT extract vague concepts, AI frameworks, or general
                       ... methodologies (e.g., "Reinforcement Learning") as variable values.
3150
                       h. Do not extract values that are purely numerical performance metrics
3151
                      ... (e.g., "67% task completion", "80% accuracy") | even if they differ
3152
                           across models or setups. Your task is to extract design decisions,
3153
                           implementation structures, and qualitative strategies | not
                      ... performance outcomes or numeric results. Values like "75% task
3154
                           success" or "F1 score 0.88" are not allowed under any condition. If
3155
                           they appear in the paper, ignore or summarize them in
3156
                       ... specific_details if useful.
3157
                   - You may additionally propose up to 3 novel, plausible variable values (as
3158
                       `LLM-recommended`) using your domain knowledge and the provided context.
3159
           2. **Extract relevant alternatives:**
3160
           - If the papers mention alternative approaches or techniques that could substitute
3161
          ... for the key variable, include these as well.
3162
               a. For example, if the key variable is "Q-learning integration", include other
3163
              ... reinforcement learning techniques mentioned in the papers
               b. Clearly indicate that these are alternatives to the main variable
3164
               c. Apply the same source attribution and confidence levels as for direct
3165
              ... variable values
3166
           - If you cannot find sufficient values (at least 3) from the provided papers, state
3167
          ... this clearly before providing your recommendations.
3168
           - If the key variable is itself a variable value (e.g., "Task Completion Rate",
          ... "Accuracy", "Success Rate"), then DO NOT extract variable values. As relevant
3169
              alternative, enumerate all possible alternative **variables** that directly
3170
               replace this key variables. These should be described as variable values with
3171
               detailed technical explanations | not as outcomes or statistical results.
3172
           3. **Additional requirements for ensuring specificity and measurability:**
3173
           - For each extracted variable value, generate an enriched specific_details field by
3174
               elaborating how the value is implemented in practice.
3175
           Include precise, implementation-level information based on the paper excerpts.
3176
           - Strictly use the Similar Paper Context to guide your response.
3177
           - Your elaboration should be specific and use implementation-relevant language.
          ... Avoid short summaries. Each specific_details must be at least 5 sentences and
3178
          ... include concrete implementation mechanisms such as model type, prompt
3179
              strategies, tuning parameters, evaluation setups, or data collection protocols.
3180
          ... If not in the text, infer plausible methods and label them as inferred.
3181
3182
3183
           In your elaboration, include as many of the following implementation details as are
3184
          ... meaningfully associated with the specific variable value:
3185
           - Architecture or model used (e.g., transformer, GPT, story graph)
3186
           - Hyperparameters or training settings (e.g., learning rate, temperature, top-k)
           - Implementation methods (e.g., prompt templates, retrieval techniques, scoring
3187
          ... functions)
3188
```

```
3190
3191
           - Evaluation metrics (e.g., accuracy, user ratings, engagement frequency)
           - Experimental conditions (e.g., number of participants, dataset used, baseline
3192
          ... comparisons)
3193
           - Any specific mechanics (e.g., branching storylets, memory modules, dialogue
3194
          ... control)
3195
           - Optional: any results or findings showing impact or performance
3196
           - Do not write vague or conceptual explanations like \this allows more freedom" or
              \this improves engagement." Instead, explain how the value is implemented | e.g.,
3197
          ... \This was achieved using GPT-3 with zero-shot prompts and a node-graph
3198
               controller to support real-time narrative updates based on player input."
           - Do not include result percentages or numeric task scores as values | describe how
3200
          \dots the system works, not how well it scored.
3201
           Your final output should extract the variable name from the "key variable
3202
          ... information" provided and use it in place of VARIABLE_NAME in the JSON format
3203
               below:
3204
           Each entry in the list should describe a **specific measurable value or design
          ... choice** relevant to the key variable. Both types are valid:
3206
           - Measurable values refer to quantifiable parameters, metric types, or behavioral
3207
          ... outcomes that can be empirically tracked or computed (e.g., accuracy, latency,
3208
          ... F1 score, response time, number of steps).
           - Design choices refer to implementation decisions that define system behavior, such
          ... as model type, architecture, prompting strategies, memory systems, or dataset
          \dots selection.
3211
3212
           ```json
3213
3214
 "VARIABLE_NAME": [
3215
 }}
 "value_name": "Name of this variable value",
3217
 "source_paper": "Paper Title or 'LLM-recommended'",
 "confidence": "High/Medium/Low",
3218
 "is_alternative": false,
3219
 "specific_details": "Detailed paragraph on measurement techniques, potential
3220
 ... value ranges, and specific examples of implementation, elaborated with
3221
 information from related passages."
 }},
 {{
 'value_name": "Name of this alternative variable value",
3224
 "source_paper": "Paper Title or 'LLM-recommended'",
3225
 "confidence": "High/Medium/Low",
3226
 "is_alternative": true,
 "specific_details": "Detailed paragraph on measurement techniques, potential
 ... value ranges, and specific examples of implementation, elaborated with
 information from related passages."
 }},
3230
 // More variable values or alternatives
3231
]
 }}
3232
3234
```

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

### Final hypothesis and research proposal

3235

3236 3237

3238 3239

3240

3241

3242

3243

```
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).
```

```
3245
3246
 - Focus solely on the key variable and its concrete variable values or
 ... implementations or alternatives, STRICTLY avoid any ambiguous phrasing
3247
 - Use novel variable combinations that have not been extensively explored in
 ... similar papers.
 - Avoid including specific numerical outcomes (e.g., \45% improvement") in
3250
 ... the hypothesis phrasing.
3251
 - Provide a detailed theoretical and practical justification for why the
 ... refined hypothesis is an important and promising research direction.
3252
3253
 ### **Step 1: Understand the Context**
3256
 - **Initial Hypothesis:**
3257
 `{hypothesis}`
3258
3259
 - **Available Variables and Value Options:**
3260
 `{variable_info}`
3261
 - **Similar Papers (to avoid overlap):**
3262
 `{similar_paper_list}
3263
3264
 (Each item includes paper title, citation count, and year - use this metadata to
 assess which papers are foundational vs. fringe or outdated. Avoid redoing
3265
 what's already exists unless you're offering a clear novel twist.)
3266
3267
3268
 ### Step 1.5: Plan Your Reasoning
3269
 Before generating the specific testable hypothesis, outline the logical
 ... reasoning process to **Ensure Novelty and Relevance**:
3270
 - What is the main contribution of the initial hypothesis?
3271
 - Which variables are most critical?
3272
 - Carefully review the `similar_paper_list` to identify variable combinations or
3273
 configurations **already explored**.
 - For each similar paper, consider its citation count and publication year to
3274
 ... avoid overlaps with highly cited or recent papers unless offering a clearly
 ... novel twist, and to spot works worth revisiting.
3276
 - Identify gaps in existing research that your hypothesis can address. The
3277
 ... hypothesis should explore NEW VARIABLE COMBINATIONS or CONDITIONS or DESIGN
 CHOICES that were NOT EXTENSIVELY tested in similar papers.
 - The research idea space is vast - prioritize hypotheses that seem explanatory,
3279
 surprising, or tied to concrete downstream benefits. Not all combinations
3280
 are equally promising. Ask: *Why is this idea worth testing over 999
3281
 others?* What gap or uncertainty does it address?
3282
 - Avoid trivial permutations (e.g., swapping known modules without meaningful
 interaction).
 - Ensure the integration logic is **not only novel** but **precisely
 describable**|how the components work together must be clearly traceable
3285
 from input to output.
 . . .
3286
3287
 ### Step 2: Generate a Specific Testable Hypothesis
3290
 - Analyse the initial hypothesis and generate a specific testable hypothesis by
3291
 ... making the key variables from the hypothesis as specific as possible using
3292
 the variable value options and the similar paper excerpts provided.
3293
 Before we begin the refinement process, let's consider the some of the
 capabilities and description of the autonomous discovery agent that will be
 testing this hypothesis:
3296
 IMPORTANT: When evaluating feasibility and testability of the hypothesis,
3297
 ... consider the following agent-specific information:
```

```
3300
3301
 ### Agent description:
3302
 {agent_description}
 - **For every variable and process mentioned in your hypothesis**, explicitly
3305
3306
 - The required code, resource, model, or tool.
 - Source: "existing codeblock" (if in the codeblock library), "external", or
 "build" (if needs to be created).
3308
 - Effort: `"minor"`, `"moderate"`, `"major"`.
 - If a component is not found in the available resources, mark as `"build"` or
 `"external"`.
3310

 This mapping is **critical** for experiment feasibility|*missing or incorrect

3311
 entries are a critical error*.
3312
3313
3314
 Before proceeding, you must strictly follow the following tiered guideline:
3316
 #### MANDATORY
3317
 - Strictly use the provided variable options. Do not introduce external
3318
 ... variables.
3319
 - Focus strictly on the key variable and its concrete variable values with
 ... implementations or alternatives, and AVOID any ambiguous phrasing.
3320
 - If applicable, make it simple and easy to understand. The hypothesis
3321
 ... should explore NEW VARIABLE COMBINATIONS or CONDITIONS or DESIGN CHOICES
3322
 that were NOT EXTENSIVELY tested in similar papers.
3323
 - Make it highly specific and testable. Clearly define the condition, the
3324
 expected measurable outcome, a control or comparative condition (if
 applicable)
3325
 - Ensure originality. The hypothesis should explore NEW VARIABLE
 ... COMBINATIONS or CONDITIONS or DESIGN CHOICES that were not extensively
3327
 ... tested in similar papers, but also technically CORRECT.
3328
 - Make sure the combination is not just NOVEL, but also PURPOSEFUL. Why
 ... do these components logically belong together? What capability does
3329
 one component enable or enhance in the other?
 - The research idea space is vast - prioritize hypotheses that seem \,
3331
 ... explanatory, surprising, or tied to concrete downstream benefits.
 ... Not all combinations are equally promising. Ask: *Why is this idea
 ... worth testing over 999 others from this space?* What gap or
 ... uncertainty does it address?
3334
 - Do not include exact numerical claims (e.g., "45% improvement", "2.1x .. increase"). Use comparative phrasing like "reduced," "improved,"
3335
 "higher," "significantly more/less" instead. Specific metrics should
 ... appear in the evaluation section, not in the hypothesis itself.
 - Provide a fully aligned and exhaustive
 `research_idea_required_code_and_resources`.
 - Include **detailed, step-by-step theoretical justification** and
3340
 ... **expected synergy** between components.
3341
3342
 #### RECOMMENDED PRACTICES
 - Use simple, readable phrasing.
3343
 - Favor comparative wording ("higher", "improved") over numeric claims.
3344
 - Keep pilot-friendly scope: small data or short episodes.
3345
3346
 #### PROHIBITED
 - No external/unlisted variables.
3347
 - No specific numeric performance outcomes in hypotheses.
3348
 - No model **FINE_TUNING, PRETRAINING**, or internal **parameter updates**.
3349
 - AVOID human evaluation unless marked external/major.
3350
 - Do not omit any mentioned implementation from resource lists.
3351
 #### FINAL SELF-CHECK
3352
 - [] All variables are from the given space
3353
3354
```

```
3355
3356
 -[] Hypothesis is clear, testable, and comparative
 - [] No numeric performance claims in the hypothesis.
3357
 - [] No model fine-tuning or human studies unless justified
 - [] Resources list is complete and properly tagged (where + effort)
 -[] Hypothesis is implementable with codeblocks or buildable logic
3361
 Remember that a hypothesis is a declarative statement expressing a relationship
3363
 between two variables (e.g., independent and dependent variables) in a given
 context. Your refined hypothesis should contain the key variables from your
3365
 research idea.
3366
3367
3368
 ### Step 3: Litmus Test: Is Your Hypothesis Understandable?
 Try this test:
3371
 Ask: Could an MSc student with no background in the specific technique
 ... **understand and implement** your hypothesis just from reading the
3373
 research_idea_long_description?
3374
 If not | explain the terms more clearly. If any key term or technique may not be
 ... intuitive, include a brief, concrete example of how it works in practice.
3376
 Ask: Would a technically trained MSc student be able to reconstruct why and how
3377
 ... these techniques fit together just by reading this?
3378
 If not, the `theoretical_justification` is too shallow.
 Remember: A technically trained MSc student must be able to understand each
3380
 component and how they fit together. Avoid unexplained jargon. If a method
 is mentioned (e.g., \multi-arm bandit" or \binary token"), explain what it
 means, why it's used, and how it works in this experiment.
3383
3384
 ### Step 4: Structure Your Output in JSON Format
3386
 Based on your analysis, generate a refined hypothesis and provide the following
3387
 ... information in JSON format:
3388
               ```json
               {{
3391
                   "research_gap": "In 1-2 sentences, clearly state the specific gap or
                  ... limitation in similar paper list or prior work that this hypothesis
                  ... addresses. Use plain language. Focus on what has not been tried or is
                       still unclear (e.g., 'No prior work tested X under noisy supervision' or
                       'Existing models overlook interaction between A and B'). Avoid vague
3395
                       claims like 'this is underexplored'. What has not been tested, why is
                       that important, and how will this hypothesis help fill that gap?"
                   "research_question": "A clear, testable research question that can be \,
3397
                  ... addressed using the refined hypothesis. It should reflect the causal or
                       comparative relationship proposed, reuse key variable or method terms,
3399
                       and be answerable using the system's capabilities. Frame it in
3400
                  \dots open-ended scientific language (e.g., 'Does...', 'How does...',
3401
                       effect does...').",
                   "research_idea_hypothesis": "Provide a concrete testable hypothesis",
3402
                   "research_idea_long_description": {{
3403
3404
```

```
3411
                       "description": "A clear paragraph explaining the complete research idea
                           including what will be tested, how it will be implemented, and
3412
                           expected outcomes. Clearly explain the motivation, purpose, and
3413
                           expected outcomes. Use the selected variable values to describe how
3414
                           each component contributes individually, why their combination is
3415
                           expected to work synergistically, and how this addresses gaps or
3416
                           limitations in prior work (as reflected in the similar paper
                           excerpts). If any mechanism or interaction may be unclear, add a
3417
                           simple, task-specific example to illustrate how it works in practice
3418
                           (e.g., 'when a symptom keyword is detected, a query to the memory
3419
                           module is triggered'). Also explain why the chosen evaluation domain
3420
                           is appropriate. Justify clearly. Tie your reasoning to specific
                           characteristics of the task or evaluation environment, and avoid
3421
                           vague statements|be specific about what performance improvements are
3422
                           expected and why. (200-400 words)",
3423
3424
                       "research_idea_variables": {{
                           "concise Variable Name": "Begin by clearly defining what the selected
3425
                               value represents|whether it's an architecture, strategy, metric,
3426
                                dataset, or baseline condition. Describe exactly how this
3427
                                variable will be configured, used, or operationalized in the
3428
                                experiment; for example, specify how a module is implemented,
3429
                                how a metric is calculated, or how a strategy is triggered.
                                Explain why this specific value was selected over alternatives,
3430
                                including its advantages, novelty, or relevance to the
3431
                               hypothesis. Describe the expected role this variable plays in
3432
                                the research problem-what outcome it directly influences or
3433
                                enables. If the variable is measurable, explicitly define how it
3434
                               will be assessed, including the metric used, how it's
                               calculated, and what range of values or thresholds would
3435
                                indicate a successful outcome. If the concept is non-obvious,
3436
                                include a **simple illustrative example** to aid understanding.
3437
                               Your explanation should be grounded in the context of the
3438
                               hypothesis and tied directly to experimental design choices and
                                evaluation logic.(200-400 words)",
3439
                           //Define each non-obvious technique, strategy, or mechanism used in
3440
                               the hypothesis, include a 1-2 sentence example of how it would
3441
                               behave in a sample input scenario.. Add detailed defintion and
3442
                               description of every independent, dependent, comparable groups,
3443
                               comparative variables, and control variables in simple format.
                       }},
3444
3445
                        "research_idea_design_prompt": "Describe in detail how the hypothesis
3446
                           will be implemented using the agent's capabilities. If any new logic
3447
                           must be built (i.e., not available as an existing codeblock),
                           explicitly describe how it will work at a data and control-flow
3448
                           level. Explain what the new module does (e.g., filters, ranks,
3449
                           reweights, scores), how it fits between existing components, and
3450
                           what rules, heuristics, or computations it will use. Describe
3451
                           exactly how their outputs are linked, how data flows from one to
3452
                           another, and what transformations occur at each step. + If multiple
                           modules or strategies are combined, explain where and how the
                           integration happens|in logic, in inputs/outputs, or in processing
3454
                           flow. Aim for clarity so that a ASD agent could build it based on
3455
                           your explanation. Include all setup steps, model configurations,
3456
                           inputs/outputs expected, and how the hypothesis will be realized
3457
                           end-to-end in code. (500-1000 words)",
3458
3459
```

```
3465
3466
                       "research_idea_metric": "Primary and secondary metrics that will be used
                           to evaluate the hypothesis. Explain how the hypothesis will be
3467
                            tested using concrete metrics and comparative setups. Identify the
                           benchmark tasks or datasets to be used, the control condition (e.g.,
3469
                           a baseline agent without the component being tested), and the exact
3470
                           performance metrics (e.g., task success rate, reasoning accuracy,
3471
                            number of valid steps). Define how improvement or success will be
                           interpreted, including thresholds, number of runs, or statistical
3472
                           confidence if relevant. If qualitative evaluations are involved,
3473
                           explain how they will be derived. Ensure that all evaluations are
3474
                            feasible using the agent's capabilities.(200-400 words)"
3475
                   }},
                    "research_idea_name": "A short, descriptive name for the research idea (3-5
3476
                   ... words)"
3477
                    "research_idea_short_description": "A single concise sentence summarizing
3478
                    .. the core idea (15-25 words)"
3479
                   "research_baselines": "Simple list of baseline approaches to compare
3480
                   .. against",
                    "research_idea_pilot": "Brief description of an initial small-scale test to
3481
                   ... validate the approach",
3482
                    "research_idea_required_code_and_resources": [
3483
                       {{
3484
                        "name": "Example Resource",
                        "description": "Brief description of the resource",
                        "where": "One of: 'existing codeblock', 'external', or 'build'",
3486
                        "effort": "One of: 'minor', 'moderate', or 'major'"
3487
                       }}.
3488
                       // EXHAUSTIVE list of ALL required CODE, RESOURCES, MODELS, etc.
                       ... mentioned in the ENTIRE RESEARCH IDEA
                          ],
3490
3491
                    "research_idea_external_requirements": [
3492
                        "example_package (for specific purpose)"
3493
3494
                    "explanation": {{
3495
                        "difference": "How it differs from the initial hypothesis",
3496
                        "novelty": "Explain exactly what is new in this configuration. Compare
3497
                       ... it to setups or strategies found in the similar paper list. Clarify
3498
                       ... what has not been explored and why this combination is interesting
                           or promising. Be specific and concise - avoid vague claims like
3499
                            'this hasn't been done before'."
                       "specificity": "How is it more specific, testable, and feasible"
3501
                       "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
3505
                           well-matched to the hypothesis and setup. (200-400 words)",
                        "expected_synergies": "Be precise: What output from Component A is used
3507
                       ... 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)"
3510
                   }}
3511
               }}
3512
3513
3514
```

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