DeepInnovator: Triggering the Innovative Capabilities of LLMs

Anonymous ACL submission

Abstract

The application of Large Language Models (LLMs) in accelerating scientific discovery has garnered increasing attention, with a key focus on constructing research agents endowed with innovative capability, i.e., the ability to autonomously generate novel and significant research ideas. Existing approaches predominantly rely on sophisticated prompt engineering and lack a systematic training paradigm. To address this, we propose DeepInnovator, a training framework designed to trigger the innovative capability of LLMs. Our approach comprises two core components. (1) *"Standing on the shoulders of giants"*. We construct an automated data extraction pipeline to extract and organize structured research knowledge from a vast corpus of unlabeled scientific literature. (2) *"Conjectures and refutations"*. We introduce a "Next Idea Prediction" training paradigm, which models the generation of research ideas as an iterative process of continuously predicting, evaluating, and refining plausible and novel next idea. Both automatic and expert evaluations demonstrate that our **DeepInnovator-14B** significantly outperforms untrained baselines, achieving win rates of 80.53%–93.81%, and attains performance comparable to that of current leading LLMs. This work provides a scalable training pathway toward building research agents with genuine, originative innovative capability, and will open-source the dataset to foster community advancement.

1 Introduction

Scientific discovery is fundamentally a process of building upon prior milestones. As Isaac Newton famously articulated, *"If I have seen further, it is by standing on the shoulders of giants"* (Newton, 1959). This insight suggests that the birth of a new idea is not an isolated event but a logical progression rooted in an existing body of work. To act as an effective innovator, one must systematically analyze how prior studies relate to each other by identifying shared objectives, conflicting results, and remaining gaps. This synthesis is essential for moving beyond incremental changes toward meaningful discovery.

However, a nascent idea is merely the beginning. The growth of scientific knowledge proceeds through *"conjectures and refutations"* (Popper, 2014). Innovation is not a one-off event but a process of iterative refinement. An innovator must not only synthesize existing literature but also possess the capability to critique its own proposals and leverage feedback to transform a rough concept into a crystallized research direction.

Large Language Models (LLMs) have recently demonstrated significant potential in the scientific domain. Existing agents aid in research tasks such as literature surveys (Xu and Peng, 2025), manuscript writing (Tang et al., 2025), or code generation (Weng et al., 2025; Novikov et al., 2025). However, a more fundamental question lies in the conceptualization phase: *Can LLMs autonomously generate novel and significant research ideas?* (Si et al., 2024). To systematically explore this innovative capability, we address two fundamental problems in training research agents: (1) how to effectively leverage vast corpora of unannotated scientific papers to structure existing knowledge; and (2) how to design a training pipeline that stimulates the research agent's capability for innovation, enabling it to autonomously generate and refine research ideas.

In this paper, we explore these problems through a clearly defined task: given a set of existing studies, determine whether LLMs can produce a logically coherent and directionally sound research idea. We propose DeepInnovator, an agent training framework designed to mimic the dual processes of idea generation and refinement. The framework consists of two main components. The first is **Automated Data Extraction and Synthesis**. This

module extracts key insights and inter-paper relationships from a vast corpus of unannotated scientific papers to construct a comprehensive research context. This process effectively allows the model to *"stand on the shoulders of giants"* by grounding generation in a structured understanding of prior work. The second component is a **"Next Idea Prediction"** training task. This approach requires the research agent to iteratively generate an improved idea based on the previous version and a generated comment, thereby eliciting the agent's innovative capability. We propose DeepInnovator-14B, a research agent built upon Qwen-14B-Instruct, via Reinforcement Learning (RL), utilizing process rewards to assess incremental improvements while employing a decoupled comment-reward mechanism to mitigate reward-hacking risks. This iterative optimization directly embodies the cycle of *"conjectures and refutations"*, enabling the agent to self-correct and refine its research idea.

To evaluate our approach, we conducted both automatic comparisons and expert evaluation. Our DeepInnovator-14B surpasses an 80% win rate against baselines across four distinct dimensions. Furthermore, despite being trained specifically on papers from mathematics, finance, statistics, and computer science, DeepInnovator demonstrates strong generalization capabilities. It generates research ideas in out-of-distribution fields, such as law and biotechnology, that significantly outperform those produced by the untrained base model and occasionally surpass GPT-4o. These results indicate that our carefully designed training methodology successfully enhances the innovative capability of research agents. To foster further research in scientific AI, we will publicly release this dataset to the community.

2 Related Works

Agents for Research. The application of LLMs in scientific research has evolved from passive literature synthesis toward active ideation, yet a critical capability gap persists. Current systems primarily operate as *DeepResearch* assistants that orchestrate multi-step workflows to exhaustively retrieve and compress existing knowledge (Xu and Peng, 2025; OpenAI, 2025; Team et al., 2025b; Zhang et al., 2025). While effective at reducing cognitive load in literature review, these approaches fundamentally function as knowledge compressors that summarize what has already been done (Fan et al., 2025b; Jin et al., 2025). Efforts in automated scientific discovery fall into two categories: domain-specific solvers like AlphaFold (Jumper et al., 2021) and GNoME (Merchant et al., 2023), which optimize algorithms toward predefined objectives within their specialized domains; and end-to-end research agents such as AI Scientist (Yamada et al., 2025) and AI-Researcher (Tang et al., 2025) that automate workflows from ideation to manuscript writing but often prioritize engineering execution over ideation quality, relying on code-based validation that limits applicability to non-computational domains. They all represent implementations of a good research idea. Here, we aim to discuss a more general research agent designed for the generation of high-quality research ideas.

RL in Open-ended Domains. Reinforcement learning offers a pathway toward ideation but faces fundamental challenges in open-ended scientific domains. While RL has succeeded in deterministic settings with objective rewards (e.g., mathematics (Shao et al., 2024; Guo et al., 2025) and code generation (Fan et al., 2025a)), scientific innovation lacks oracle verifiers, necessitating subjective evaluation via LLM-as-a-Judge (Lee et al., 2024; Ma et al., 2024) or rubric-based frameworks (Gunjal et al., 2025; Viswanathan et al., 2025). These approaches, however, are vulnerable to reward hacking where models exploit surface-level judge preferences (Huang et al., 2025; Shao et al., 2025; Sharma et al., 2024). We design a method that decouples reward and comment, separating outcome scoring and improvement suggestions to mitigate reward-hacking risks and encourage the generation of intrinsically valuable ideas.

3 Problem Formulation for Innovation task

Many different "innovation tasks" have been undertaken by the community in the domain of building research agent. In existing work, different systems embody this concept in varying ways: AI-Researcher (Tang et al., 2025) and AI-Scientist (Yamada et al., 2025) define innovation tasks as having AI research assistants write research papers, using paper quality as a measure of innovative capability; whereas DeepScientist (Weng et al., 2025) and AlphaEvolve (Novikov et al., 2025) assess innovation by having AI assistants write code that improves algorithmic performance in specific domains. Despite these differing forms, such tasks fundamentally involve the realization and development of a research idea. Here, we aim to explore a more gen-

eral and fundamental form of innovation task: *how can foundation models be trained to autonomously generate high-quality research ideas?*

We formulate the innovation task as a sequential idea generation and refinement process: a base LLM acts as a research agent that, given contextual information C from a set of reference papers, autoregressively generates an initial research idea $y^{(0)}$ and iteratively refines it into increasingly sophisticated formulations. The training objective is to predict the "next improved idea" $y^{(i+1)}$ at each step i.

Formally, let $\mathcal{R} = \{r_1, r_2, \ldots, r_N\}$ denote a collection of reference papers. Their content is encoded into an initial context $C = \text{Encode}(\mathcal{R})$, which serves as the starting state for the generation process. The agent's policy π_θ operates autoregressively over a sequence of tokens, producing a trajectory of progressively refined ideas.

We define the action space to consist of a single generative action <idea></idea>, which generates a segment of idea text. The full generation is partitioned into a sequence of idea snapshots $y^{(0)}, y^{(1)}, \ldots, y^{(K)}$, where $y^{(0)}$ is the initial, coarse-grained research idea, each subsequent $y^{(k)}$ (for $k \geq 1$) represents a semantically complete and improved version of the idea, obtained by continuing token generation from the previous state. The final output $y^{(K)}$ is treated as the model's proposed innovative research idea, where K denotes the number of iterations.. Critically, this idea is not a direct paraphrase or extraction from $\mathcal{R}$, but rather emerges through an internal iterative refinement loop, where each new idea is conditioned on both the reference context and the model's own prior outputs, effectively simulating self-critique and creative evolution.

4 Training Methodology of DeepInnovator

A key challenge is turning LLMs' latent innovation potential, learned from human knowledge corpora, into behaviors that are guidable, evaluable, and optimizable. This challenge gives rise to three interrelated training challenges:

Challenge 1: Lack of Training Data. Unlike tasks in domains with well-defined correct answers, such as code generation, search, or mathematical reasoning, innovation tasks lack well-defined training data, readily available training objectives, and clear correctness criteria.

Challenge 2: Inapplicability of Predefined Objectives in Research Tasks. Prior research-agent systems (Tang et al., 2025; Weng et al., 2025; Novikov et al., 2025) often rely on predefined objectives or hard-coded validation, yet defining such objectives presupposes a sound research idea; when the idea is still ambiguous and no validation pathway exists, these methods become ineffective.

Challenge 3: Reward Hacking in Open-Ended Evaluation. In such unbounded environments, relying solely on LLM-as-a-Judge to provide rewards is highly prone to reward hacking (Gao et al., 2023; Zhao et al., 2025; Hu et al., 2025), particularly incentivizing sycophantic responses that cater to the judge's presumed preferences rather than delivering insightful analysis of substantive value (Anthropic, 2025).

To address these challenges, we propose the following solutions: **(1)** We meticulously design a method for extracting training data from a large corpus of unannotated scientific papers (Sec. 4.1); **(2)** We formulate the Next Idea Prediction task to train DeepInnovator(Sec. 4.2); **(3)** We introduce a decoupled architecture that separates rewards from comments to mitigate reward-hacking risks (Sec. 4.3).

4.1 Automated Data Extraction and Synthesis

To bolster the innovative capabilities of agents in research tasks, we design an automated data synthesis pipeline to construct the "Next Idea Prediction" training task from arXiv. As shown in Fig. 1, given the references $\mathcal{R}$ of a target paper, the agent generates and iteratively refines the next research idea y using its own knowledge and the reference context C. This setup models knowledge integration and leapfrogging innovation in scientific research. For each target paper, we collect its references $\mathcal{R}$ as prior context. Directly using full reference texts is impractical: multiple papers can exceed the agent's context window, and redundant details obscure relations among key research insights.

To address this, we introduce a hierarchical abstraction pipeline that transforms the original citations into compact, structured representations of research ideas. Our pipeline comprises the following three key steps:

Idea Extraction: Using prompts, we extract the target paper's research idea for the reward model and formalize each cited paper's core idea as a structured statement, compressing information while preserving semantics.

Idea Clustering and Relationship Modeling:

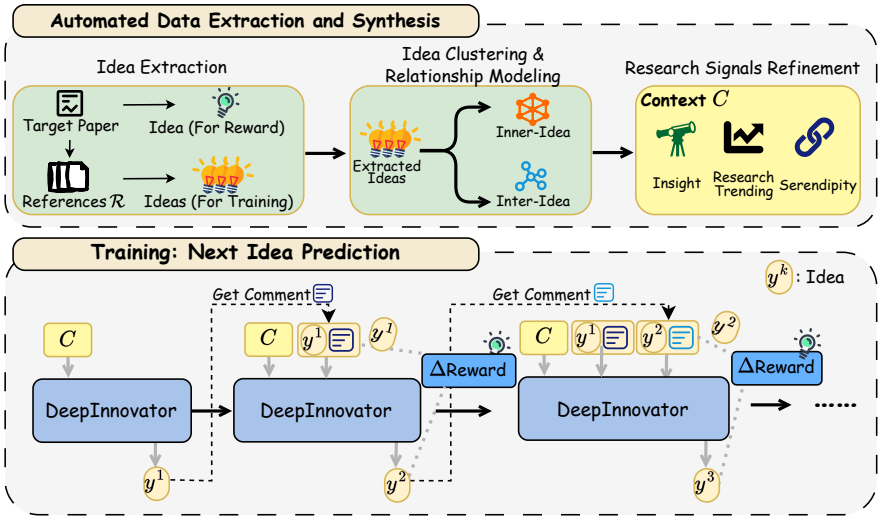

Figure 1: The DeepInnovator Framework. Top: We construct training data from arXiv through a carefully designed automated data extraction and synthesis pipeline (Sec. 4.1). Bottom: We perform RL training via a meticulously designed next-idea prediction task (4.2), coupled with a decoupled reward and critique mechanism (Sec. 4.3).

We perform semantic clustering on all extracted ideas and identify two types of critical relationships:

- *Inner-idea relations*: Evolutionary, variant, or refinement pathways among ideas within the same cluster;
- *Inter-idea relations*: Cross-cluster interactions such as integration, convergence, or conflict between ideas from different clusters.

This structured organization not only reveals the knowledge topology of prior work but also provides a logical scaffold for subsequent insight generation.

Higher-Order Research Signal Refinement: Building upon the idea relationships, we further distill three categories of higher-order research signals:

- *Insight*: Non-obvious patterns or contradictions inferred from multiple citation ideas (e.g., "existing methods neglect temporal consistency in cross-modal alignment");
- *Research Trending*: Emerging or declining research directions identified through progressive relationships among ideas;
- *Serendipity*: Latent connections between seemingly unrelated domains (e.g., "transferring exploration mechanisms from reinforcement learning to neural architecture search").

We model these signals to reduce context length and capture three cognitive mechanisms in scientific reasoning: *inductive inference* (Insight) (Newell et al., 2001; Harvey, 2014), *prospective judgment* (Research Trending) (Chen, 2006; Min et al., 2021), and *cross-domain association* (Serendipity) (Kennedy et al., 2022; Kang et al., 2022; Gentner, 2011), which are core drivers of breakthrough innovation. By formalizing these cognitive primitives into computable intermediate representations, we provide the research agent with a scaffold that exposes recurring relations among prior studies, thereby helping it generate innovative and forward-looking predictions as well as grounded recombinations and refinements rather than relying only on unstructured summaries.

4.2 Reward Signal Design

Inspired by contrastive delta rewards (Seed et al., 2025) and long-horizon agent training (Wang et al., 2025), we design a **process-oriented delta reward** for scientific tasks without ground-truth answers or predefined objectives.

Rather than only judging whether the final idea approximates the actual research idea, this mechanism rewards improvement at each refinement step. Since scientific breakthroughs often emerge through trial, reflection, and recombination, the reward encourages exploratory reasoning and cognitive progress across the trajectory.

Using Group Relative Policy Optimization (GRPO) (Shao et al., 2024), we optimize the agent policy π_θ to generate multi-turn research ideas with sustained cognitive progress conditioned on citation context C. For each C, we sample G complete trajectories $o_i = (y_i^{(0)}, y_i^{(1)}, \ldots, y_i^{(K)})$, each rewarded by accumulated stepwise improvement signals from an external reward LLM `Reward`.

At step k, `Reward` scores the previous and current ideas, y^{k-1} and y^k, against the real idea; their

score difference defines the step reward. The final reward $R(o_i)$ can be formulated as:

$$R(o_i) = \sum_{k=1}^{K} \texttt{Reward}\big(y_i^{(k-1)}, y_i^{(k)}; q\big), \quad (1)$$

Based on this, we subsequently compute the normalized intra-group advantage estimate $\hat{A}_i$. The final GRPO loss function is:

$$\mathcal{L}_{\text{GRPO}}(\theta) = -\mathbb{E}_{q\sim P(Q)}\Bigg[\frac{1}{G}\sum_{i=1}^{G}\frac{1}{|o_i|}\sum_{t=1}^{|o_i|} \min\Big(r_{i,t}(\theta)\hat{A}_i,\ \text{clip}\left(r_{i,t}(\theta), 1-\varepsilon, 1+\varepsilon\right)\hat{A}_i\Big) + \beta\,\text{KL}\Big(\pi_\theta \big\| \pi_{\text{ref}}\Big)\Bigg]. \quad (2)$$

where $|o_i|$ is the trajectory length, $\hat{A}_i$ is derived from $R(o_i)$, $r_{i,t}(\theta) = \dfrac{\pi_\theta(a_{i,t} \mid s_{i,t})}{\pi_{\text{old}}(a_{i,t} \mid s_{i,t})}$ is the new-to-old policy ratio for action $a_{i,t}$ under state $s_{i,t}$, and clip$(\cdot)$ bounds the policy update.

Through this process-oriented formulation, the reward avoids reliance on a single "ground-truth" target, which is difficult for innovation tasks, and instead emphasizes the agent's evolving reasoning process.

4.3 Decomposing Reward and Comment in Open-ended Tasks

In RL frameworks that rely solely on reward models, agents may hack the reward by generating text that superficially exhibits high-scoring characteristics but actually deviates from human intent. To mitigate this pressure, we introduce text-guided improvement directions (*comments*) that identify issues in the current idea and suggest corrections. The architecture decouples this guidance from evaluation: the agent receives suggestions from `Comment`, while effectiveness is judged only by `Reward`.

Specifically, during training, for each draft idea $y^{(k)}$ generated by the agent, we invoke a comment LLM `Comment` to analyze its performance within the context of relevant literature C, and output structured suggestions—for example: "fails to address temporal consistency" or "attempts to solve a problem already addressed in existing literature". The agent then uses this feedback from `Comment` to produce an improved version $y^{(k+1)}$, thereby simulating an authentic peer-review iteration process.

By decoupling process guidance from outcome evaluation, our design enforces a strict separation of responsibilities: the comment model `Comment` instructs how to improve while the reward model `Reward` solely judges whether it has genuinely improved. This separation removes one direct pathway for comment imitation, but does not eliminate all judge-specific biases; therefore, we interpret the mechanism as a mitigation of reward-hacking risk rather than a complete solution. Since the reward signal is entirely independent of the comment model `Comment`, the trained research agent cannot please the reward mechanism by mimicking the phrasing of comments, inserting high-scoring keywords, or padding responses with verbose yet vacuous content, as the comment model `Comment`, which provides feedback to the agent, is completely excluded from the reward computation. Consequently, the agent is compelled to focus on substantively improving the quality of the idea itself, rather than fabricating an illusion of progress.

5 Experiments

5.1 Experimental Settings

Datasets. Using the data collection process described in Sec. 4.1, we constructed the training dataset for DeepInnovator. It comprises target papers extracted from four domains on arXiv, along with all their references and the original ideas of the target papers. We only collected papers published after March 2025 to mitigate data leakage issues. We curated a training set containing 1,012 research ideas and a validation set containing 113 research objectives, covering Computer Science, Mathematics, Finance, and Statistics.

Model. We initialize the policy model using Qwen-2.5-14B-Instruct (Bai et al., 2023). Our RL framework is implemented with the VeRL (Sheng et al., 2024) library.

Reward. The reward is computed by Qwen-plus acting as the scorer. Moreover, the scorer is required to explicitly enumerate its reasoning before assigning a score, a step that significantly enhances scoring rigor. Additionally, we observe that the length of the generated idea influences the scorer. To mitigate this behavior, we enforce strict length constraints: ideas with fewer than 3,000 or more than 5,000 characters are penalized.

Evaluation Protocols and Metrics. To validate DeepInnovator, we conducted both automated evaluation and expert evaluation. The automated evaluation was performed on our constructed validation set, which includes computer science, finance, statistics, and mathematics. The expert evaluation

encompassed three new domains: law, education, and biotechnology.

- **Baseline.** We place our DeepInnovator alongside the Qwen-14B-Instruct and five leading LLMs, including GPT-4o, gemini-2.5-pro (Google, 2023), Qwen3-max (Yang et al., 2025), Deepseek-r1 (Guo et al., 2025), Grok-4.1, and Minimax-M2.1 within the same workflow as shown in Fig. 1. All models receive the contextual input and perform idea generation, with up to three rounds of iterative refinement permitted. The final idea from each model is used for comparison.
- **Automated Evaluation.** We begin with automated evaluation. To enhance the reliability of validation, we employ two evaluation approaches: rubrics-based assessment and winrate analysis. **Rubrics:** We adopt the general rubrics from Goel et al. (2025) to evaluate whether the generated idea meet the fundamental rubrics of a scientific research idea. These rubrics are designed based on recurring failure patterns in research ideas generated by language models, as documented in earlier studies (Si et al., 2024). For each rubric, we determine whether a generated idea satisfies it based on majority voting among the five leading LLMs: GPT-4o, Kimi-K2 (Team et al., 2025a), gemini-2.5-pro, Deepseek-r1, and Qwen3-max. **Winrate:** We compute the average score assigned by the same five LLMs through pairwise comparisons of all generated ideas. We adopt the comparison methodology and prompt template introduced in SGI-bench (Xu et al., 2025). SGI-Bench evaluates ideas along four dimensions: *Effectiveness* (whether the problem can be solved), *Novelty* (whether the approach is innovative), *Detailedness* (whether the proposal is complete and specific), and *Feasibility* (whether it can be practically implemented), as these four dimensions collectively span the entire pipeline of scientific research.
- **Expert Evaluation.** To systematically evaluate DeepInnovator's adaptability in unseen scenarios and to obtain expert-based validation beyond synthetic benchmarks, we further conducted expert evaluations in three entirely new domains: law, education, and biotechnology. In each domain, we recruited three domain experts. Each evaluator was assigned 10 pairs of research ideas, where one idea in each pair was generated by DeepInnovator and the other by either GPT-4o or Qwen-14B-Instruct. Evaluators were allocated 60 minutes per annotation task and, following the prompt guidelines provided by SGI-Bench, selected a winner for each idea pair across four dimensions: effectiveness, novelty, detailedness, and feasibility.

For more details of datasets, baselines, and implementation details, please refer to Appendix A.

5.2 Rubrics Evaluation Result

In the rubrics evaluation, we comprehensively evaluate the solutions generated by models across six core rubrics: (1) *Detailed, Specific Solution*, which examines whether the output exhibits actionable steps and depth of detail; (2) *No Overlooked Flaws or Weaknesses*, which assesses the model's ability to identify potential issues; (3) *Well-Justified Rationale*, which measures the logical coherence and strength of supporting reasoning; (4) *Cost and Effort Efficient*, which evaluates the reasonableness of resource utilization in the proposed solution; (5) *No Ethical Issues*, which ensures that the recommendations adhere to ethical standards; and (6) *Consistent with Overall Plan*, which verifies whether the solution aligns with the overarching contextual objectives. The details can be found in Sec. A.3.

As shown in Fig. 2, compared to the baseline Qwen-14B-Instruct, DeepInnovator demonstrates clear performance improvements across multiple dimensions, with gains ranging from 1.05% to 8.43%. Furthermore, it approaches the performance (top-5) of top-tier LLMs on 3/6 rubrics. Notably, DeepInnovator (82.3%) surpasses GPT-4o (77.88%) in *Well-Justified Rationale* (Rubric 3), which emphasizes reasoning detail and argumentation. Additionally, its *No Ethical Issues* (Rubric 5) exceeds that of GPT-4o, reflecting strong safety alignment capabilities.

Moreover, we observe that *No Overlooked Flaws or Weaknesses* (Rubric 2) and *Cost and Effort Efficient* (Rubric 4) present common challenges for current models, reflecting persistent limitations in systematic risk anticipation and resource-optimization modeling. DeepInnovator's relatively weak cost-and-effort performance suggests that process-oriented refinement can encourage more complete and technically detailed proposals, but may also increase implementation complexity.

5.3 Winrate Evaluation Result

Table. 1 reports the win rates of DeepInnovator on SGI-bench across four key dimensions: *Novelty*,

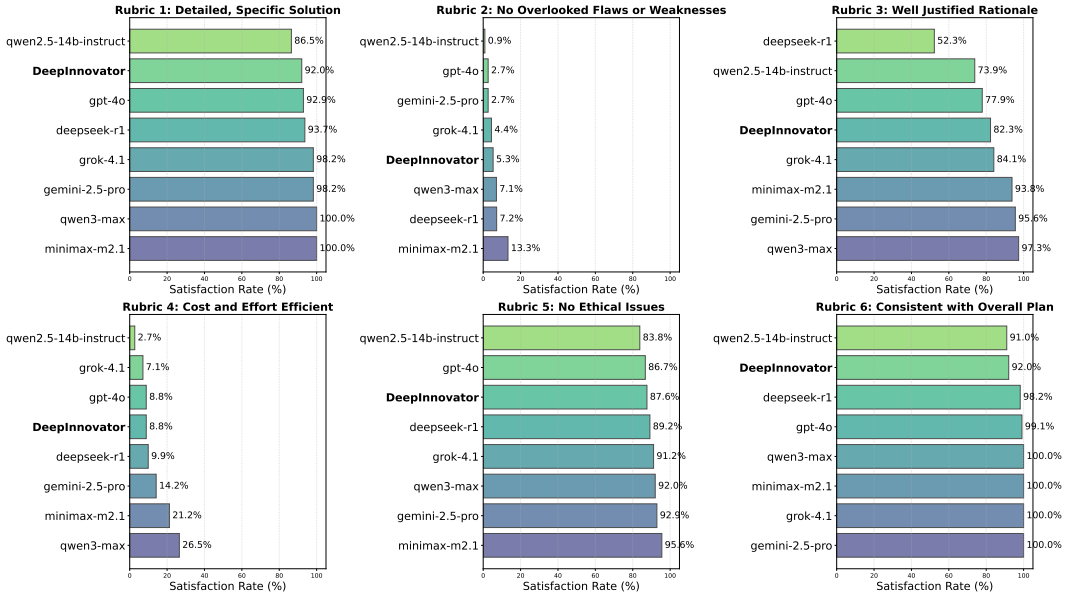

Figure 2: Evaluation results of ideas generated by eight models across six rubrics. DeepInnovator outperforms Qwen-14B-Instruct on all dimensions and demonstrates competitive capability against leading LLMs.

Table 1: Win rate results of DeepInnovator vs Other Models. *Novelty*: whether the approach is innovative; *Effectiveness*: whether the problem can be solved; *Feasibility*: whether it can be practically implemented; *Detailedness*: whether the proposal is complete and specific. **Bold** indicate that DeepInnovator wins (i.e., win rate > 50%).

Metric	Qwen-14B-Instruct	Deepseek-r1	Gemini-2.5-pro	GPT-4o	Minimax-M2.1	Qwen3-max	Grok-4.1
Novelty	**80.53%**	47.79%	15.93%	49.56%	**59.29%**	32.74%	37.17%
Feasibility	**87.61%**	23.01%	18.58%	11.50%	10.62%	19.47%	10.62%
Effectiveness	**92.92%**	45.13%	43.36%	**76.11%**	**50.44%**	**66.37%**	43.36%
Detailedness	**93.81%**	**59.29%**	**68.14%**	44.25%	**61.06%**	**91.15%**	45.13%

Effectiveness, *Feasibility*, and *Detailedness*. All idea pairs were evaluated through majority voting among five independent LLMs to ensure objectivity and consistency in the assessment.

The proposed training methodology improves DeepInnovator's idea-generation ability. DeepInnovator outperforms Qwen-14B-Instruct on all dimensions, especially *Effectiveness* and *Detailedness*, where win rates exceed 90%. It also beats 7/12 larger leading LLM comparisons, suggesting that training improves both originality (*Novelty*) and practical viability (*Feasibility*) while producing detailed solutions.

Parameter scale significantly influences innovative capability. Although DeepInnovator outperforms Qwen-14B-Instruct, it defeats only 8/24 other LLMs, primarily excelling in the dimensions of *Effectiveness* and *Detailedness*. In contrast, it struggles to compete with larger-parameter LLMs in *Novelty* and *Feasibility*. Notably, in the feasibility dimension, despite achieving a decisive advantage over Qwen-14B-Instruct (87.61%), its highest win rate against other models is merely 23.01%. This highlights that smaller-parameter models face inherent limitations in simultaneously ensuring *Effectiveness*, *Detailedness*, and *Feasibility* during innovation.

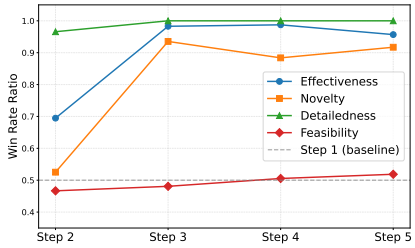

Figure 3: Comparison of win rates between ideas generated by DeepInnovator at each step and the initial idea (step 1). Our training method effectively enhances DeepInnovator's ability to refine research ideas.

The refinement process contributes to the improvement of research ideas. Fig. 3 shows the win rates of ideas generated at each step compared to the initial idea (Step 1) across multiple dimensions. Although these evaluation dimensions were not explicitly specified in the training task, the win rates of the newly generated "next idea" consistently improve across these dimensions as the refinement process progresses (Steps 2–3), and stabilize during Steps 4–5. This indicates that DeepInnovator can effectively enhance the overall quality of ideas during the refinement process.

Table 2: Winrate results annotated by human experts. "Much better" or "Worse" indicates that the expert thinks that DeepInnovator is much better/worse than Qwen-14B-Instruct/GPT-4o. The value before the slash "/" denotes DeepInnovator vs Qwen-14B-Instruct, and the value after the slash denotes DeepInnovator vs GPT-4o. The values represent the sum of annotators' judgments. **Bold** indicate that DeepInnovator wins (i.e., win rate > 50%).

Domain	Metric	Much Better	Better	Worse	Much Worse	Both Bad	Avg. Winrate
Law	*Novelty*	0 / 0	9 / 7	3 / 6	1 / 0	0 / 0	**69.2% / 53.8%**
	Feasibility	1 / 0	5 / 4	3 / 7	1 / 1	3 / 1	**60.0%** / 33.3%
	Effectiveness	2 / 1	5 / 4	2 / 3	1 / 3	3 / 2	**70.0%** / 45.5%
	Detailedness	7 / 1	3 / 4	2 / 5	1 / 0	0 / 3	**76.9%** / 50.0%
Education	*Novelty*	3 / 0	9 / 9	2 / 5	1 / 1	0 / 0	**80.0% / 60.0%**
	Feasibility	3 / 0	5 / 0	5 / 7	1 / 3	1 / 5	**57.1%** / 0.0%
	Effectiveness	2 / 0	6 / 0	4 / 5	3 / 4	0 / 6	**53.3%** / 0.0%
	Detailedness	1 / 0	8 / 4	3 / 5	0 / 2	3 / 4	**75.0%** / 36.4%
Biotech	*Novelty*	3 / 2	8 / 6	1 / 5	0 / 0	2 / 2	**91.7% / 61.5%**
	Feasibility	2 / 0	6 / 5	0 / 3	0 / 5	6 / 2	**100.0%** / 38.5%
	Effectiveness	1 / 2	6 / 5	4 / 2	1 / 4	2 / 2	**58.3% / 53.8%**
	Detailedness	6 / 3	4 / 2	1 / 4	0 / 5	3 / 1	**90.9%** / 35.7%

5.4 Expert Evaluation

Table. 2 reports the win rates of DeepInnovator in three specialized domains outside the training set (law, education, and biotechnology).

Our training approach effectively enhances the novelty of DeepInnovator. A consistent pattern across all three domains shows that *Novelty* is the only dimension where DeepInnovator maintains a win rate above 50% against both Qwen-14B-Instruct and GPT-4o. This consistent advantage suggests that the training methodology of DeepInnovator is particularly effective at fostering original and innovative thinking, regardless of the parameter scale of the baseline models. Even when compared to large models such as GPT-4o, DeepInnovator remains competitive in terms of *Novelty*.

The generalizability of innovative capabilities is challenged. The performance of DeepInnovator still exhibits domain dependence. Although DeepInnovator achieves consistently superior results against Qwen-14B-Instruct across all domains, its performance in the education domain is relatively weaker—particularly when compared to GPT-4o, where win rates for both *Feasibility* and *Effectiveness* drop to 0.0%. In the legal domain, DeepInnovator demonstrates balanced performance: all dimensions exceed 50% win rates against Qwen-14B-Instruct, yet only *Novelty* maintains an advantage over GPT-4o (53.8%). This domain-specific variation suggests that the training data or methodology of DeepInnovator is better aligned with STEM-oriented fields such as biotechnology, while offering insufficient advantages in humanities-related domains.

The feasibility dimension exhibits the largest performance gap in baseline comparisons. In the biotechnology domain, DeepInnovator received no judgments of "Worse" or "Much Worse" when compared against Qwen-14B-Instruct on *Feasibility*. However, comparisons with GPT-4o reveal significant challenges: win rates drop to 33.3% (law), 0.0% (education), and 38.5% (biotechnology). Notably, in the education domain, DeepInnovator failed to win a single feasibility comparison (0.0% win rate). This substantial gap suggests that larger models possess stronger capabilities in evaluating and generating practically feasible solutions, indicating that improving feasibility may require broader world knowledge and more advanced reasoning abilities.

6 Conclusion

We study how to stimulate LLMs' innovative capability in generating and refining research ideas. The DeepInnovator framework (1) extracts and synthesizes structured knowledge from scientific literature as *"shoulders of giants"* context, and (2) uses "Next Idea Prediction" with process rewards and decoupled reward-comment feedback to model the *"conjectures and refutations"* cycle of scientific inquiry. Experiments demonstrate that DeepInnovator-14B generates high-quality, novel research ideas and exhibits exceptional cross-domain generalization. Even in domains absent from its training data (law, education, and biotechnology), DeepInnovator-14B consistently outperforms Qwen-14B-Instruct and rivals leading LLMs. This result indicates that a carefully designed training framework can effectively stimulate the innovative capability of LLMs. Our work explores important training methodologies for scientific discovery AI, effectively triggers the innovative capabilities of research agents.

Limitations

This work has several limitations. First, the training data relies on citation links and automatic extraction from scientific papers, so it may be affected by incomplete literature coverage and extraction errors. Second, although the reward-comment decoupling design mitigates reward-hacking pressure, the evaluation still relies on LLM judges and therefore cannot fully remove model-specific preferences or biases. Third, our experiments cover a limited set of scientific domains and model scales; future work should further validate the generality of this approach across broader disciplines, task types, and model architectures.

Ethical considerations

This work aims to support, rather than replace, human scientific reasoning. Research ideas generated by LLM-based agents may be plausible but incorrect, misleading, or difficult to validate, and should therefore be treated only as preliminary hypotheses requiring expert scrutiny and empirical verification.

References

Anthropic. 2025. Natural emergent misalignment from reward hacking. Anthropic Research Blog.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, and 29 others. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Chaomei Chen. 2006. Citespace ii: Detecting and visualizing emerging trends and transient patterns in scientific literature. *Journal of the American Society for information Science and Technology*, 57(3):359–377.

Lishui Fan, Yu Zhang, Mouxiang Chen, and Zhongxin Liu. 2025a. Posterior-grpo: Rewarding reasoning processes in code generation. *arXiv preprint arXiv:2508.05170*.

Tianyu Fan, Xinyao Niu, Yuxiang Zheng, Fengji Zhang, Chengen Huang, Bei Chen, Junyang Lin, and Chao Huang. 2025b. Understanding deepresearch via reports. *arXiv preprint arXiv:2510.07861*.

Long Gao, John Schulman, and Jacob Hilton. 2023. Scaling laws for reward model overoptimization. *Proceedings of the 40th International Conference on Machine Learning (ICML)*, 202:10666–10682.

Dedre Gentner. 2011. Analogy in scientific discovery: The case of johannes kepler. In *Model-based reasoning: Science, technology, values*, pages 21–39. Springer.

Shashwat Goel, Rishi Hazra, Dulhan Jayalath, Timon Willi, Parag Jain, William F. Shen, Ilias Leontiadis, Francesco Barbieri, Yoram Bachrach, Jonas Geiping, and Chenxi Whitehouse. 2025. Training ai co-scientists using rubric rewards. *Preprint*, arXiv:2512.23707.

Google. 2023. Gemini: A family of highly capable multimodal models. https://goo.gle/GeminiPaper.

Anisha Gunjal, Anthony Wang, Elaine Lau, Vaskar Nath, Yunzhong He, Bing Liu, and Sean Hendryx. 2025. Rubrics as rewards: Reinforcement learning beyond verifiable domains. *arXiv preprint arXiv:2507.17746*.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, and 175 others. 2025. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 645(8081):633–638.

Sarah Harvey. 2014. Creative synthesis: Exploring the process of extraordinary group creativity. *Academy of management review*, 39(3):324–343.

Yutong Hu, Shuai Ouyang, Qinyuan Li, Hao Yi, Guodong Chen, Fan Zhang, and Xiaofei Li. 2025. Bias fitting to mitigate length bias of reward model in rlhf. *arXiv preprint arXiv:2505.12843*.

Zenan Huang, Yihong Zhuang, Guoshan Lu, Zeyu Qin, Haokai Xu, Tianyu Zhao, Ru Peng, Jiaqi Hu, Zhanming Shen, Xiaomeng Hu, and 1 others. 2025. Reinforcement learning with rubric anchors. *arXiv preprint arXiv:2508.12790*.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani, and Jiawei Han. 2025. Search-r1: Training llms to reason and leverage search engines with reinforcement learning. *arXiv preprint arXiv:2503.09516*.

John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, and 1 others. 2021. Highly accurate protein structure prediction with alphafold. *nature*, 596(7873):583–589.

Hyeonsu B Kang, Sheshera Mysore, Kevin Huang, Haw-Shiuan Chang, Thorben Prein, Andrew McCallum, Aniket Kittur, and Elsa Olivetti. 2022. Augmenting scientific creativity with retrieval across knowledge domains. *arXiv preprint arXiv:2206.01328*.

Ian G Kennedy, Dean Whitehead, and Debra Ferdinand-James. 2022. Serendipity: A way of stimulating researchers' creativity. *Journal of Creativity*, 32(1):100014.

Harrison Lee, Samrat Phatale, Hassan Mansoor, Thomas Mesnard, Johan Ferret, Kellie Lu, Colton Bishop, Ethan Hall, Victor Carbune, Abhinav Rastogi, and 1 others. 2024. Rlaif vs. rlhf: scaling reinforcement learning from human feedback with ai feedback. In *Proceedings of the 41st International Conference on Machine Learning*, pages 26874–26901.

Yecheng Jason Ma, William Liang, Guanzhi Wang, De-An Huang, Osbert Bastani, Dinesh Jayaraman, Yuke Zhu, Linxi Fan, and Anima Anandkumar. 2024. Eureka: Human-level reward design via coding large language models. In *The Twelfth International Conference on Learning Representations*.

Amil Merchant, Simon Batzner, Samuel S Schoenholz, Muratahan Aykol, Gowoon Cheon, and Ekin Dogus Cubuk. 2023. Scaling deep learning for materials discovery. *Nature*, 624(7990):80–85.

Chao Min, Yi Bu, Ding Wu, Ying Ding, and Yi Zhang. 2021. Identifying citation patterns of scientific breakthroughs: A perspective of dynamic citation process. *Information Processing & Management*, 58(1):102428.

William H Newell, Jay Wentworth, and David Sebberson. 2001. A theory of interdisciplinary studies. *Issues in Interdisciplinary Studies*.

Isaac Newton. 1959. *The Correspondence of Isaac Newton*. Cambridge University Press.

Alexander Novikov, Ngân Vũ, Marvin Eisenberger, Emilien Dupont, Po-Sen Huang, Adam Zsolt Wagner, Sergey Shirobokov, Borislav Kozlovskii, Francisco JR Ruiz, Abbas Mehrabian, and 1 others. 2025. Alphaevolve: A coding agent for scientific and algorithmic discovery. *arXiv preprint arXiv:2506.13131*.

OpenAI. 2025. Introducing deep research.

Karl Popper. 2014. *Conjectures and refutations: The growth of scientific knowledge*. routledge.

ByteDance Seed, Jiaze Chen, Tiantian Fan, Xin Liu, Lingjun Liu, Zhiqi Lin, Mingxuan Wang, Chengyi Wang, Xiangpeng Wei, Wenyuan Xu, and 1 others. 2025. Seed1. 5-thinking: Advancing superb reasoning models with reinforcement learning. *arXiv preprint arXiv:2504.13914*.

Rulin Shao, Akari Asai, Shannon Zejiang Shen, Hamish Ivison, Varsha Kishore, Jingming Zhuo, Xinran Zhao, Molly Park, Samuel G Finlayson, David Sontag, and 1 others. 2025. Dr tulu: Reinforcement learning with evolving rubrics for deep research. *arXiv preprint arXiv:2511.19399*.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, and 1 others. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.

Archit Sharma, Sedrick Scott Keh, Eric Mitchell, Chelsea Finn, Kushal Arora, and Thomas Kollar. 2024. A critical evaluation of ai feedback for aligning large language models. *Advances in Neural Information Processing Systems*, 37:29166–29190.

Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. 2024. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv: 2409.19256*.

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

Jiabin Tang, Lianghao Xia, Zhonghang Li, and Chao Huang. 2025. Ai-researcher: Autonomous scientific innovation. *arXiv preprint arXiv:2505.18705*.

Kimi Team, Yifan Bai, Yiping Bao, Guanduo Chen, Jiahao Chen, Ningxin Chen, Ruijue Chen, Yanru Chen, Yuankun Chen, Yutian Chen, and 1 others. 2025a. Kimi k2: Open agentic intelligence. *arXiv preprint arXiv:2507.20534*.

Tongyi DeepResearch Team, Baixuan Li, Bo Zhang, Dingchu Zhang, Fei Huang, Guangyu Li, Guoxin Chen, Huifeng Yin, Jialong Wu, Jingren Zhou, and 1 others. 2025b. Tongyi deepresearch technical report. *arXiv preprint arXiv:2510.24701*.

Vijay Viswanathan, Yanchao Sun, Shuang Ma, Xiang Kong, Meng Cao, Graham Neubig, and Tongshuang Wu. 2025. Checklists are better than reward models for aligning language models. *arXiv preprint arXiv:2507.18624*.

Jiawei Wang, Jiacai Liu, Yuqian Fu, Yingru Li, Xintao Wang, Yuan Lin, Yu Yue, Lin Zhang, Yang Wang, and Ke Wang. 2025. Harnessing uncertainty: Entropy-modulated policy gradients for long-horizon llm agents. *arXiv preprint arXiv:2509.09265*.

Yixuan Weng, Minjun Zhu, Qiujie Xie, Qiyao Sun, Zhen Lin, Sifan Liu, and Yue Zhang. 2025. Deepscientist: Advancing frontier-pushing scientific findings progressively. *arXiv preprint arXiv:2509.26603*.

Shirley Wu, Michel Galley, Baolin Peng, Hao Cheng, Gavin Li, Yao Dou, Weixin Cai, James Zou, Jure Leskovec, and Jianfeng Gao. 2025. Collabllm: From passive responders to active collaborators. In *International Conference on Machine Learning (ICML)*.

Renjun Xu and Jingwen Peng. 2025. A comprehensive survey of deep research: Systems, methodologies, and applications. *arXiv preprint arXiv:2506.12594*.

Wanghan Xu, Yuhao Zhou, Yifan Zhou, Qinglong Cao, Shuo Li, Jia Bu, Bo Liu, Yixin Chen, Xuming He, Xiangyu Zhao, and 1 others. 2025. Probing scientific general intelligence of llms with scientist-aligned workflows. *arXiv preprint arXiv:2512.16969*.

Yutaro Yamada, Robert Tjarko Lange, Cong Lu, Shengran Hu, Chris Lu, Jakob Foerster, Jeff Clune, and David Ha. 2025. The ai scientist-v2: Workshop-level automated scientific discovery via agentic tree search. *arXiv preprint arXiv:2504.08066*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Wenlin Zhang, Xiaopeng Li, Yingyi Zhang, Pengyue Jia, Yichao Wang, Huifeng Guo, Yong Liu, and Xiangyu Zhao. 2025. Deep research: A survey of autonomous research agents. *arXiv preprint arXiv:2508.12752*.

Yulai Zhao, Haolin Liu, Dian Yu, Sunyuan Kung, Meijia Chen, Haitao Mi, and Dong Yu. 2025. One token to fool llm-as-a-judge. *arXiv preprint arXiv:2507.08794*.

A Experiment Details.

A.1 Datasets.

We employ the official arXiv API[1] to crawl papers. We use Qwen-OCR[2] for PDF parsing and Qwen3-max (Yang et al., 2025) to extract references from the parsed paper text. Our training set comprises 1,012 samples, distributed as follows: cs: 433, math: 194, q-fin: 181, stat: 204. Our validation set contains 113 samples, with the following distribution: cs: 51, math: 22, q-fin: 20, stat: 20.

A.2 Implementation Details.

The training process utilized multi-turn dialogue data and was optimized with a custom dialogue-level reward function. Both our training and validation datasets were partitioned into four disciplinary categories: Computer Science (CS), Mathematics (math), Quantitative Finance (q-fin), and Statistics (stat). The training set comprises 1,012 samples, with CS being the largest category (433 samples), followed by stat (204), math (194), and q-fin (181). The validation set contains 113 samples, with a distribution closely mirroring that of the training set: CS (51), stat (20), math (22), and q-fin (20). Regarding the training script, we referred to the implementation of CollabLLM (Wu et al., 2025).

The key hyperparameter settings are listed in the following table:

Table A1: Hyperparameters used in DeepInnovator.

Category	Parameter	Value
Dataset Statistics		
	Training set size (by category)	CS: 433, math: 194, q-fin: 181, stat: 204
	Validation set size (by category)	CS: 51, math: 22, q-fin: 20, stat: 20
Training Data		
	Train batch size	16
	Max prompt length	8192 tokens
	Max response length	2048 tokens
	Filter overlong prompts	True
PPO Algorithm		
	PPO mini-batch size	8
	Advantage estimator	GRPO
	KL loss coefficient	0.001
	KL loss type	low_var_kl
Optimizer		
	Actor learning rate	5×10^{-7}
	Critic warmup epochs	0
Rollout Configuration		
	Number of rollouts	8
	Temperature	1.0
	Max user turns	5
	Max assistant turns	6
	Repeat rollouts per trajectory	3

[1] https://info.arxiv.org/help/api/index.html

[2] https://www.alibabacloud.com/help/en/model-studio/qwen-vl-ocr

Reward. The reward is computed by Qwen-plus acting as the scorer. Moreover, the scorer is required to explicitly enumerate its reasoning before assigning a score, a step that significantly enhances scoring rigor. Additionally, we observe that the length of the generated idea influences the scorer. Specifically, since we perform comparative scoring, in the early stages of training, extremely short initial ideas tend to yield more pronounced improvements. In later stages, however, the trained research agent tends to generate increasingly longer content to continually secure improvement rewards. To mitigate this behavior, we enforce strict length constraints: ideas with fewer than 3,000 or more than 5,000 characters are penalized.

A.3 Rubrics Evaluation Details

We employ 6 basic rubrics from Goel et al. (2025) for evaluation. The specific details of these assessments are presented in Table A2.

Table A2: Evaluation Rubrics

Rubric	Description
Detailed, specific solution	Does the part of the plan relevant to satisfying this rubric item include fully specified details on how to implement it? There should be no claims of handling something without actually doing so, no vague terms, ambiguity, or lack of clarity. The description should use simple, easy-to-understand language.
No overlooked flaws or weaknesses	Are there any important overlooked flaws or weaknesses in the part of the plan addressing this rubric item that would invalidate its claimed satisfaction of the criterion?
Well-justified rationale	Is the part of the plan relevant to this rubric item well-motivated and justified? For example, does it provide convincing arguments that the chosen approach is better than simpler alternatives or competing hypotheses?
Cost and effort efficient	Does the plan address this rubric item in a cost- and effort-efficient manner, avoiding unnecessary complexity? Consider whether a less resource-intensive or less labor-demanding solution could achieve comparable effectiveness.
No ethical issues	Does this part of the plan pose any potential for negative societal impact or raise ethical concerns (e.g., bias, privacy violations, manipulation, or unfair outcomes)?
Consistent with overall plan	Is this component of the plan logically consistent with the rest of the proposed strategy? Ensure it does not contradict other stated assumptions, methods, or objectives elsewhere in the plan.

A.4 Winrate Evaluation Details

SGI-Bench (Xu et al., 2025) evaluates ideas along four dimensions: **effectiveness** (whether the problem can be solved), **novelty** (whether the approach is innovative), **detailedness** (whether the proposal is complete and specific), and **feasibility** (whether it can be practically implemented), as these four dimensions collectively span the entire pipeline of scientific research.

In the original idea evaluation framework of SGI-Bench, the final score for each dimension is the average of the Score component and the Win Rate component.

Regarding the Score component:

1. Effectiveness is measured by keyword hit rate.
2. Novelty is assessed by the embedding dissimilarity between the generated idea and existing literature.
3. Detailedness is evaluated via content completeness checks, with penalties for redundancy.
4. Feasibility is determined by the similarity between the implementation graph generated by the model and an expert-provided template graph.

For the Win Rate component: For each dimension, a LLM serves as the judge in pairwise comparisons, pitting the model-generated idea against a reference answer. Independent votes are cast for each dimension, and the win rate of the model's idea is computed accordingly.

Since the Score component imposes strict formatting requirements on data and ideas (e.g., implementation details must be represented as a directed graph), which do not reflect general scenarios, we adopt only the Win Rate component for evaluation.

A.5 Expert Evaluation Details.

A.5.1 Details of Expert Evaluation.

We assigned 10 judged idea pairs to each expert and report the results in Table. 2. In Table. 2, the total number of evaluations for Law is 26 and for Biotech is 29. The following are the reasons why these five pairs were not successfully evaluated:

Table A3: Issue Table

ID	Reason	Model Generating the Issue
Law1	Incomplete idea generation	Qwen-14B-Instruct
Law2	Incomplete idea generation	Qwen-14B-Instruct
Law3	Incomplete idea generation	Qwen-14B-Instruct
Law4	Generated idea is a hybrid of law and computer science	DeepInnovator
Biotech1	Incomplete idea generation	Qwen-14B-Instruct

A.5.2 Breakdown of Participant Positions.

We show the detailed position breakdown of our idea judge participants in Table. A4. We show the detailed institutions breakdown of our idea

judge participants in Table. A5. We recruited 3 experts in each of the fields of Law, Biotechnology, and Education. Experts are presented with an idea pair generated from the same reference, read idea pair, determine the winner, and provide a justification. In each idea pair, one idea originates from DeepInnovator and the other from either Qwen-14B-Instruct or gpt-4o. During evaluation, the experts are unaware of the specific model that generated each idea.

Table A4: Positions of the idea judge participants.

Position	**Count**
Postdoc	1
PhD	6
Mphil	2

Table A5: Institutions of the idea judge participants.

Institution	**Count**
The University of Hong Kong (HKU)	3
Zhejiang University (ZJU)	3
Broad Institute of MIT & Harvard (Broad Institute)	1
Swiss Federal Technology Institute of Lausanne (EPFL)	1
The California Institute of Technology (Caltech)	1

B Stability of the reward model

As stated in our experimental Sec. 5.1, we employ the Qwen-Plus model as the reward model. The reward model is required to observe both the real idea and the model-generated idea and assign scores accordingly. In Table A6, we present an experiment demonstrating that Qwen-Plus can effectively distinguish between real ideas and generated ideas. Here, accuracy denotes the probability that Qwen-Plus correctly identifies the type of idea—assigning a score of 1 to the real idea and a score of 0 to the fake idea.

It can be seen that Qwen-Plus achieves high accuracy in identifying both real and fake ideas, demonstrating its suitability as a reward model.

Table A6: Discrimination Accuracy of Qwen-plus on arXiv ideas (Real) vs. Qwen-14B-Instruct Generated Ideas (Fake)

	Real Ideas	Fake Ideas
Accuracy (%)	80.77	75.12

C All prompts used in DeepInnovator

C.1 Comment Model

```
Name: Idea Authenticity Checker
Description: Determine whether a given research
idea comes from real, published research work or
is a fictional/hypothetical research idea.
# Task Overview
1. Carefully analyze the provided research idea.
2. Make a judgment: 1 = real research work, 0 =
fictional research idea.
3. Provide confidence level and detailed
reasoning for the judgment.
# Important: Do NOT Rely on Format Features
- DO NOT use citation formats (e.g., arXiv
citations, BibTeX format) to judge authenticity
- DO NOT use DOI numbers, paper IDs, or
publication identifiers as indicators
- DO NOT rely on formatting styles (e.g., LaTeX
formatting, citation styles) to make judgments
- Focus ONLY on the CONTENT QUALITY and SUBSTANCE
of the research idea itself
- Format features can be easily fabricated and
are not reliable indicators of authenticity
- Base your judgment solely on the technical
depth, problem clarity, and limitations
discussion in the content
- Be mindful to avoid excessive focus on details,
as this is a research idea rather than a complete
research proposal.
# Analysis Steps
## Step 1: Extract Key Components
- Identify the core problem being addressed
- Extract the main technical approach or
methodology
- Note any specific methods, algorithms, or
techniques mentioned
- Identify novelty claims and contributions
- Note any limitations or challenges discussed
- CRITICAL: Check the technical_approach field
(if present) for:
* Whether steps are truly executable or just
superficially detailed
* Presence of concrete implementation details
(data sources, parameters, algorithms)
* Discussion of computational complexity or
resource requirements
* Missing critical details that would be needed
for actual implementation
## Step 2: Deep Technical Analysis
- CRITICAL: Evaluate Technical Approach
Executability
* If technical_approach is present, check if
steps are truly executable:
- Do steps include concrete data sources,
parameters, or algorithms?
- Are there missing critical details needed for
implementation?
- Can a researcher actually follow these steps to
reproduce results?
- CRITICAL: Check Technical Integration Depth
* Verify if multiple techniques are meaningfully
integrated or just superficially combined:
- Are integration challenges acknowledged?
- Is there explanation of how incompatible
assumptions are resolved?
- Are synergistic principles explained, or just
stated?
* Red Flag: Multiple complex techniques combined
without explaining fundamental incompatibilities
or theoretical soundness
## Step 3: Make Judgment
- Consider all indicators together
- IMPORTANT: Base judgment ONLY on content
quality, NOT on format features (citations, DOI,
formatting styles)
- CRITICAL Red Flags for Fictional Ideas:
* Superficial Technical Depth: Technical terms
are mentioned but not meaningfully integrated;
steps look detailed but lack executable specifics
* Missing Practical Challenges: No discussion of
computational complexity, data availability,
measurement error, endogeneity, cross-country
data harmonization, or other real-world
implementation difficulties
* Unrealistic Integration: Multiple complex
techniques are combined without explaining how
they overcome fundamental incompatibilities or
why the integration is theoretically sound
* Abstract Limitations Only: Limitations are
described in abstract terms without engaging with
specific empirical challenges or methodological
tensions
* Overly Ambitious Scope: Simultaneously
addresses too many disparate problems without
acknowledging the complexity or potential
conflicts
- Real research work typically shows:
* High technical specificity with meaningful
integration of concepts
* Well-defined problem context with clear
boundaries
* Detailed limitations discussion including
practical challenges (computational, data,
```

methodological)
* Technical steps that are executable with concrete implementation details
* Acknowledgment of potential failures, edge cases, or methodological tensions
- Fictional ideas typically show:
* Low to medium technical specificity OR high specificity but superficial integration
* Vague problem context OR overly broad scope without clear boundaries
* Minimal limitations discussion OR abstract limitations without practical challenges
* Technical steps that look detailed but lack executable specifics
* Missing discussion of computational complexity, data requirements, or implementation challenges
- Make the authenticity judgment: 1 (real) or 0 (fictional)
- Your knowledge maybe outdated, so do not use your knowledge to judge if some methods are present or absent.
Step 4: Assess Confidence
- High confidence (0.8-1.0): Clear indicators strongly support the judgment
- Medium confidence (0.5-0.8): Most indicators support the judgment, but some ambiguity exists
- Low confidence (0.0-0.5): Mixed indicators or insufficient information
Step 5: Generate Reasoning
- List 3-5 key reasons supporting the judgment
- Reference specific aspects of the idea that led to the conclusion
- Be specific and concrete in the reasoning
Output Requirements
Authenticity Field
- Must be exactly 1 (real research work) or 0 (fictional research idea)
- Based on comprehensive evaluation of all indicators
Confidence Field
- Number between 0.0 and 1.0
- Reflects how certain you are about the judgment
- Consider the strength and consistency of indicators
Reasoning Field
- Array of 3-5 strings
- Each string should be a clear, specific reason
- Reference concrete aspects of the idea
- Explain why the judgment was made
Here is a typical true idea:
{ground_truth}
Input Data
Research Idea
<idea>
{idea}
</idea>
Notes:
the result should be a valid JSON object, wrapped in <Judgment>...</Judgment>
Output Format:
<Judgment>
{
"authenticity": 0,
"confidence": 0.95,
"reasoning": ["The idea is not clearly from real research work", "The idea is not well-defined and has a clear technical approach", "The idea is fake and not executable with concrete implementation details"]
}
</Judgment>

C.2 Reward Model

You will be given two research ideas proposed by two PhD students attempting to formulate a research hypothesis.
Additionally, you will be given a ground truth idea that has been peer-reviewed and validated.
Task
Evaluate how close each of the two ideas is to the ground truth idea. Provide scores on a scale of 0-1, where higher scores indicate better alignment with the ground truth.
- IMPORTANT: Base judgment ONLY on content quality, NOT on format features (citations, DOI, formatting styles)
- **CRITICAL Red Flags for Fictional Ideas**:
* **Superficial Technical Depth**: Technical terms are mentioned but not meaningfully integrated; steps look detailed but lack executable specifics (e.g., missing computational complexity, data requirements, implementation challenges)
* **Missing Practical Challenges**: No discussion of computational complexity, data availability, measurement error, endogeneity, or other real-world implementation difficulties
* **Unrealistic Integration**: Multiple complex techniques are combined without explaining how they overcome fundamental incompatibilities or why the integration is theoretically sound
* **Abstract Limitations Only**: Limitations are described in abstract terms without engaging with specific empirical challenges or methodological tensions
* **Overly Ambitious Scope**: Simultaneously addresses too many disparate problems without acknowledging the complexity or potential conflicts
Output Format
You must return a JSON object with the following structure:
{{
"idea1_improve_score": <float between 0 and 1>, // Higher score indicates better alignment with ground truth
"idea2_improve_score": <float between 0 and 1>, // Higher score indicates better alignment with ground truth
"reason": "<explanation of your evaluation>"
}}
Example
{{
"idea1_improve_score": 0.8,
"idea2_improve_score": 0.6,
"reason": "Idea1 shows better alignment with the ground truth idea because it more closely matches the structure and content of the validated research approach."
}}
Input
idea1: {idea1}

idea2: {idea2}
ground_truth: {ground_truth}

C.3 System Prompt of DeepInnovator

Name: Idea Quality Improver
Description: Improve a research idea, making refinements to address weaknesses while preserving the core concept.
Task Overview
1. You will be given some critics from users, you should update the idea based on those critics.
2. Make improvements to address the weaknesses.
3. Enhance technical_approach with detailed step-by-step workflow if missing or insufficient.

Critical Principles
Make Improvement
- Enhance clarity, specificity, and depth rather than changing direction
- Do NOT change the fundamental nature of the idea
Analysis Steps
- Address weaknesses found by users
- **CRITICAL: Enhance or add technical_approach**
* If technical_approach is missing or insufficient, add or enhance it with detailed step-by-step workflow
* Include: data preparation, model/algorithm design, key technical components, implementation details, evaluation methodology
* Present as a clear, sequential workflow that enables implementation and validation
- Maintain coherence between different parts of the idea
- Verify that improvements don't contradict the core concept
Output Requirements
Improved Idea Object
- Output an improved_idea object with the same structure as the original
- All required fields must be present:
* current_limitations: Refined but preserving core problem.
* idea_summary: Enhanced sentences maintaining core concept. Focus solely on describing this idea itself.
* technical_approach: Enhanced step-by-step description of the methodological workflow.
* novelty_statement: Enhanced novelty claim
* confidence: Confidence level
Improvement Summary Object
- Output an improvement_summary object explaining what was improved:
* key_improvements: Array of specific improvements made
* improvement_rationale: Brief explanation of why improvements were made
# Output Format
- Output must be valid JSON with no extra fields or commentary, wrapped in <Idea>...</Idea>
- Output must be in the following format:
<Idea>
{{
"improved_idea": {{
"current_limitations": "...",
"idea_summary": "...",
"technical_approach": "...",
"novelty_statement": "...",
"confidence": "...",
"improvement_summary": {{
"key_improvements": ["...", "..."],
"improvement_rationale": "..."
}}
}}
}}
</Idea>

C.4 Win Rate Evaluator

You are assisting researchers tasked with comparing TWO research hypotheses (Hypothesis A and Hypothesis B).
Your job is to evaluate both hypotheses across five separate dimensions defined below, and to choose a winner (either Hypothesis A or Hypothesis B) for each dimension. Ties are NOT allowed — you MUST pick one winner per dimension. Base your judgments on scientific principles and the provided context only.
##Background context:
{context_text}
##Hypothesis A:
{hypothesis_A}
##Hypothesis B:
{hypothesis_B}
##Definition of each dimension:
###1) Effectiveness
Which hypothesis is more likely to produce a successful experimental or empirical outcome in service of the stated research objective? Evaluate the likelihood that, if implemented using standard practices in the relevant discipline, the hypothesis will achieve the intended measurable result. Focus on mechanistic plausibility, causal logic, and whether the hypothesis addresses the core problem directly.
###2)Novelty
Novelty: Which hypothesis presents more innovative or original approaches? Compare the similarity between the idea and the related work and existing solutions in the background to assess its novelty. A lower similarity to the core idea indicates greater novelty.
###3) Detailedness (Level of Specification)
Which hypothesis provides clearer, more actionable, and more complete specification of mechanisms, assumptions, experimental steps, required variables, and dependencies? Detailedness rewards clarity that would enable a competent researcher to design an experiment or implementation with minimal ambiguity.
###4) Feasibility
Which hypothesis presents a more realistic and implementable solution given current technological constraints?
###5) Overall
Considering the overall aspects together but emphasizing conceptual coherence and scientific grounding, which hypothesis is superior overall? This is a synthesis judgment: prefer the hypothesis that is logically consistent, grounded in accepted principles, avoids critical unstated assumptions or contradictions, and is most defensible as a scientific proposition.

Unified constraints:
- Use only the provided context and widely accepted scientific principles in the relevant discipline. Do NOT invent facts external to the context unless they are broadly standard domain knowledge.
- When a dimension explicitly says to ignore other factors (e.g., Novelty should ignore feasibility), strictly follow that guidance for that dimension. When evaluating a certain dimension, it should focus on this dimension itself and ignore the influence of other dimensions.
- Be concise but specific: for each dimension provide a short judgment line (exact format below) plus 1-3 sentences of succinct reasoning grounded in the definitions above.
- Format must match exactly (case-insensitive for "Win A/Win B") and include a reason after "because".
##Output format (MUST FOLLOW EXACTLY)
Format your response exactly as follows:
Effectiveness: [Win A/Win B] because ...
Novelty: [Win A/Win B] because ...
Detailedness: [Win A/Win B] because ...
Feasibility: [Win A/Win B] because ...
Overall: [Win A/Win B] because ...

C.5 Rubrics Evaluator

You are a strict and unbiased evaluator.
Your goal is to analyze whether a proposed plan violates any desiderata with respect to a given rubric item.
==========
DESIDERATA DEFINITIONS
==========
1. DETAILED, SPECIFIC SOLUTION
- Does the part of the plan relevant to this rubric item give concrete details on HOW to implement it?
- There should be no vague claims like "we will handle X" without explaining how.
- Avoid ambiguity and unclear language; the explanation should be easy to understand.
2. NO OVERLOOKED FLAWS OR WEAKNESSES
- Are there important flaws or weaknesses in how the plan addresses this rubric item that would invalidate its satisfaction?
3. WELL JUSTIFIED RATIONALE
- Is the reasoning behind the plan for this rubric item well-motivated and justified?
- Are there convincing arguments that this approach is better than simpler solutions or alternative hypotheses?
4. COST AND EFFORT EFFICIENT
- Does the plan handle this rubric item in a cost- and effort-efficient way, without unnecessary complexity?
- Would a simpler solution with less human or resource cost be equally effective?
5. NO ETHICAL ISSUES
- Does this part of the plan avoid negative consequences or ethically problematic behavior?
6. CONSISTENT WITH OVERALL PLAN
- Is this part of the plan consistent with the rest of the plan, without contradictions?
==========
EVALUATION INSTRUCTIONS
==========
- Be skeptical, careful, and provide valid criticisms.
- Be as strict as possible while remaining unbiased and reasonable.
- Do NOT accept the plan merely because it claims to satisfy the desiderata.
- For each desideratum (1-6), decide only whether it is satisfied (‘true‘) or violated (‘false‘), focusing on parts relevant to the given rubric item.
==========
OUTPUT FORMAT
==========
You MUST respond in valid JSON only, with the following structure:
{{
"per_desideratum_satisfied": {
"1": bool,
"2": bool,
"3": bool,
"4": bool,
"5": bool,
"6": bool
},
"violated_desiderata": [int],
"global_reason": string
}}
ADDITIONAL RULES
- "violated_desiderata" must be the list of desiderata numbers (1-6) that are violated, i.e., those with ‘satisfied = false‘.
- If NO part of the plan addresses this rubric item, then:
- set all values in "per_desideratum_satisfied" to false,
- set "violated_desiderata" to [1,2,3,4,5,6],
- explain in "global_reason" that the plan does not address the rubric item at all.
- Do NOT include any text outside of the JSON object.
Below is an EXAMPLE of a valid JSON response format (values are illustrative; you must adapt them to the actual input):
{{
"per_desideratum_satisfied": {
"1": true,
"2": false,
"3": true,
"4": false,
"5": true,
"6": true
},
"violated_desiderata": [2,4],
"global_reason": "The plan is detailed and mostly well-justified, but it omits some required stages, overlooks key risks, and is more resource-intensive than necessary for the rubric item."
}}