Idea2Plan: Exploring AI-Powered Research Planning

Anonymous ACL submission

Abstract

Large language models (LLMs) have demonstrated significant potential to accelerate scientific discovery as valuable tools for analyzing data, generating hypotheses, and supporting innovative approaches in various scientific fields. In this work, we investigate how LLMs can handle the transition from conceptual research ideas to well-structured research plans. Effective research planning not only supports scientists in advancing their research but also represents a crucial capability for the development of autonomous research agents. We introduce the *Idea2Plan* task and *Idea2Plan Bench*, a benchmark built from 200 ICML 2025 Spotlight and Oral papers released after major LLM training cutoffs. Each benchmark instance includes a research idea and a grading rubric capturing the key components of valid plans. We further propose *Idea2Plan JudgeEval*, a complementary benchmark to assess the reliability of LLM-based judges against expert annotations. Experimental results show that GPT-5 and GPT-5-mini achieve the strongest performance on the benchmark, though substantial headroom remains for future improvement.

1 Introduction

A key challenge in scientific research is that scientists tend to produce more promising ideas than they can pursue and thus, many potentially promising ideas often remain unexplored. When attending conferences, participating in reading groups, or discussing with peers, scientists frequently conceive ways to improve ongoing work or apply insights to other areas. Despite this abundance of ideas, many remain unexplored due to time constraints.

Transforming each research idea into a viable plan requires considerable time and cognitive effort. The process of developing an idea often involves reviewing prior work, formulating hypotheses, selecting appropriate methods, and designing experiments, with plans iteratively refined as research progresses. This process can take researchers many days or even weeks to complete. We refer to the process of turning a research idea into a concrete, testable plan as *research planning*. Research planning encompasses all the steps necessary to bridge the gap between an initial idea and a well-structured plan ready for execution.

Automated systems capable of research planning could greatly accelerate scientists' progress. By supporting the development of research ideas and streamlining the planning process, such systems could also advance the capabilities of autonomous AI research agents and help to ensure that more promising ideas are explored and developed.

Despite its importance, AI-powered research planning has been underexplored. Recent works focus on other research stages such as idea generation (Wang et al., 2024a; Ghafarollahi and Buehler, 2024; Baek et al., 2025; Si et al., 2025), literature review (Wang et al., 2024b) and experiment execution (Starace et al., 2025; Kon et al., 2025a; Jansen et al., 2025; Seo et al., 2025). Previous work treats research planning as an intermediate step without explicit evaluation (Jansen et al., 2025) or evaluates research plans based only on simple traits such as clarity and novelty (Chen et al., 2025a).

We make the evaluation of AI's ability for research planning our primary objective. We start our investigation by formulating the *Idea2Plan* task, where the input is a research idea, and the AI agent is asked to generate a *research plan*, a structured output that serves as a roadmap for executing a project. A research plan typically includes objectives, prior work, methodology, and evaluation design (Weber and Cobaugh, 2008; Sudheesh et al., 2016). Because we focus on planning for scientific research in the field of AI, we instantiate our research plans' structure based on common AI paper conventions: Introduction, Key Literature, Methods, Initial Experimental Design, and Resources/Compliance/Ethical Considerations.

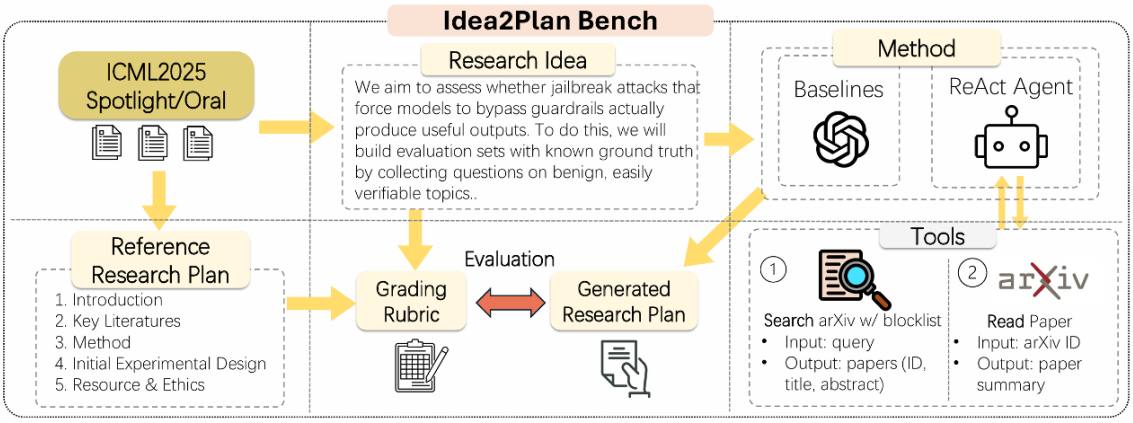


Figure 1: Idea2Plan overview: Starting from a research idea extracted from the abstract of an academic paper, we employ a pipeline that includes prompt-based and ReAct agent methods, integrated with search and paper reading tools, to generate research plans that are then evaluated against reference plans derived from the content of the corresponding paper by using a comprehensive grading rubric.

Evaluating such plans is challenging for two major reasons. First, data contamination threatens validity, because an LLM may have already encountered the idea (for instance through a published paper), in which case its output reflects the LLM's memorization about the research idea rather than genuine research planning ability (Carlini et al., 2021, 2022). To mitigate this, we build an evaluation framework that extracts research ideas from AI papers that are published *after* LLMs' data cutoff dates. Second, multiple plans can be equally valid for the same idea; for example, two sound plans may select different yet comparable datasets, yielding distinct but acceptable designs. We address this by constructing a dedicated grading rubric for each research idea. The questions in the rubric test the common desiderata that any valid plan for that idea should meet.

Concretely, to construct our benchmark, we include ICML 2025 spotlight or oral papers with no public arXiv version before the latest training data cutoff of the LLMs employed in our study, as shown in Figure 1. We then generate a grading rubric from the extracted research plan for each research idea. The rubric contains five sections of yes/no questions that capture the essential components of a valid plan for each research idea as derived from the content of the corresponding paper. We assess the quality of extracted plans and rubrics through a human study and find that they are of reasonable quality. Finally, we randomly sample 200 papers as a held-out test set. We refer to this benchmark as *Idea2Plan Bench*.

Given the amount of effort required to grade research plans against rubrics, it is unrealistic to rely on human graders at scale. To make evaluation scalable, we propose LLM-based judges for grading the rubrics (Zheng et al., 2023; Starace et al., 2025). We introduce an additional benchmark, *Idea2Plan JudgeEval*, to assess judgment reliability. This benchmark collects a set of ground-truth grading created manually by human experts. Our results show that LLM judges can provide expert-aligned grading.

To explore the performance of LLMs on the *Idea2Plan* benchmark, we test both direct prompting and agentic approaches. Our agentic approach uses a ReAct-style scaffolding (Yao et al., 2023) with search and reading tools over arXiv papers. We then compare a range of frontier proprietary and open-source models. Results show that GPT-5 and GPT-5-mini consistently outperform other models. Surprisingly, the ReAct agent's performance does not outperform direct prompting. Overall, we expect our work to advance the understanding of research planning capabilities in LLMs.

Our contributions are as follows:

- We formalize the *Idea2Plan* task and build a benchmark from post-LLM-cutoff AI papers.
- We develop a pipeline to automatically generate rubrics for research plans. We conduct a human study and show that the extracted plans and rubrics are of high quality.
- We propose an LLM-based judge and the *Idea2Plan JudgeEval* to compare LLM-as-judge scoring with expert annotations.
- We discuss experiments and results with several state-of-the-art LLMs under different prompting and agentic strategies.

2 Related Work

2.1 AI Agents that Assist Scientists

AI agents increasingly support different stages of the research workflow, including hypothesis generation (Wang et al., 2024a; Ghafarollahi and Buehler, 2024; Baek et al., 2025; Si et al., 2025; Zhang et al., 2025; Zhao et al., 2025; Jansen et al., 2025), literature review (Singh et al., 2025; Wang et al., 2024b), and experimental execution (Kon et al., 2025a; Jansen et al., 2025; Seo et al., 2025). Broader "co-scientist" systems aim to integrate these capabilities by scaffolding search, reading, planning, and execution (Gottweis et al., 2025; Jansen et al., 2025). For instance, Google's AI co-scientist follows a scientist-in-the-loop design—where researchers specify goals in natural language and guide a multi-agent pipeline through chat—and has been applied to end-to-end biomedical studies (Gottweis et al., 2025).

2.2 Autonomous AI Scientists

Recent efforts pursue end-to-end AI scientists that can conduct full loop of the scientific process from generating novel ideas to autonomously designing and conducting experiments (Yuan et al., 2025; Lu et al., 2024; Yamada et al., 2025; Schmidgall et al., 2025; Schmidgall and Moor, 2025). For example, the AI Scientist proposes a pipeline to generate a research idea, conduct experiments and finally write a paper (Lu et al., 2024). While these systems report promising results in some applications, they face two major challenges. First, evaluating the novelty of AI-generated ideas or papers is difficult. For instance, Gupta and Pruthi (2025) examined 50 supposedly novel research proposals or papers (generated by Lu et al. (2024); Si et al. (2025)) and reported that 24% of them were either paraphrased or substantially copied from existing research. Second, the lack of implementation capability limits the effectiveness of AI Scientist systems (Jansen et al., 2025; Zhu et al., 2025). Given these challenges, our work does not aim to build a fully autonomous system. Instead, we focus on understanding one critical aspect: research planning.

2.3 Benchmarks for AI Agents on Research Automation

Recent benchmarks aim to evaluate AI systems designed for research automation. Broadly, these efforts can be grouped into two categories based on their *output form*: (1) paper-level evaluation, where systems are assessed on the quality of generated research papers (Lu et al., 2024; Yamada et al., 2025; Schmidgall et al., 2025); and (2) code-level evaluation, where systems are judged by their ability to produce code implementations or experimental results (Kon et al., 2025a; Chen et al., 2025b; Jansen et al., 2025; Starace et al., 2025; Seo et al., 2025; Chen et al., 2025a; Kon et al., 2025b). In contrast, our work isolates and rigorously evaluates the stage of research planning. The most related works are CODESCIENTIST (Jansen et al., 2025) and MLR-BENCH (Chen et al., 2025a): the former includes research planning as an intermediate step without explicit evaluation, and the latter evaluates research plans only on high-level dimensions such as clarity and novelty.

3 Proposed Framework

In this section, we introduce *Idea2Plan*, a framework to evaluate how LLMs turn research ideas into research plans; see Figure 1 for an overview.

3.1 Research Plan and Idea Extraction

There may be different definitions of research plans and what should be included (Weber and Cobaugh, 2008; Sudheesh et al., 2016). In this work, we focus on AI research and we adhere to the common structure of research plans for AI papers. We discard plan components that are not easily determined by an LLM (*e.g.*, timelines, logistics, or human expertise). We guide the research plan generation using a structured template with the following sections: Introduction, Key Literatures, Methods, Initial Experimental Design, and Resources/Compliance/Ethical Considerations. We use o4-mini[1] (reasoning=high) for idea and plan extraction. Prompt templates are shown in appendix §B.1.

3.2 Rubric-based Evaluation

Given our extracted reference research plans and research ideas, along with research plans generated by LLMs (either through prompting or agentic approaches), the question is: how do we evaluate the generated research plans? The key difficulty lies in the fact that a single research idea can lead to multiple equally reasonable research plans. For example, an idea about evaluating a new jailbreak attack could reasonably choose different yet comparable datasets, threat models (e.g., black-box vs.

[1] https://openai.com/index/introducing-o3-and-o4-mini/

Table 1: We show an example of a research idea, grading questions, relevant excerpts of a research plan generated by GPT-5 given this research idea, and the LLM-judge's judgement of this question. Full example is in §A.

Research Idea (from paper: The Jailbreak Tax: How Useful are Your Jailbreak Outputs? (Nikolic et al., 2025), ICML 2025 Spotlight)			
"We aim to assess whether jailbreak attacks that force models to bypass guardrails actually produce useful outputs. To do this, we will build evaluation sets with known ground truth by collecting questions on benign, easily verifiable topics and align models to refuse those questions. We will then apply representative jailbreak strategies to these aligned models and measure how much model utility drops in the jailbroken responses. We propose a new metric to quantify this performance degradation and introduce corresponding benchmarks to enable systematic evaluation and comparison of jailbreak methods."			
Section	**Grading Question Example**	**Excerpts from a Research Plan generated by GPT-5**	**Judgement**
Introduction	"Does the plan note that existing evaluations focus on bypass success and neglect post-jailbreak capability retention?"	"While jailbreak success is typically measured by refusal circumvention, it is unclear whether the resulting outputs remain useful and accurate."	Yes
Key Literatures	"Does the plan cite the paper (Jailbreaking black box large language models in twenty queries) or similar work on iterative LLM-based prompt rewriting attacks?"	"Greshake et al., Prompt Injection attacks [...] Wallace et al., Universal Adversarial Triggers [...] Zou et al., adversarial suffix (GCG) methods."	No
Methods	"Does the plan describe a supervised fine-tuning approach on (prompt, refusal) pairs to induce model alignment?"	"Lightweight SFT/LoRA: fine-tune an instruction-tuned base model on refusal demonstrations for our categories to simulate learned guardrails."	Yes
Initial Experimental Design	"Does the plan include analysis of the relationship between task difficulty and the magnitude of the jailbreak tax?"	"Optional: chain-of-thought vs concise answers to see if jailbreaks disproportionately harm multi-step reasoning."	No
Resources, Compliance, & Ethics	"Does the plan address the potential for adversaries to misuse the evaluation framework?"	"Release attack code in a restricted form that limits adaptation to harmful content [...] exclude especially potent per-item adversarial suffixes from public artifacts."	Yes

white-box), or metric suites; these choices can produce distinct plans that are all sound.

To capture these nuances, we propose a rubric-based evaluation approach (Starace et al., 2025). We generate rubric questions from the reference research plans to identify high-level design choices that should be addressed in any reasonable research plan for the given idea. This approach allows us to evaluate whether generated plans cover essential conceptual elements while accommodating legitimate variations. The prompt template is in §B.2.

3.3 Dataset Construction

Our dataset selection follows two key criteria: papers must be free from potential training data contamination for the LLMs we evaluate, and the research ideas must be of high quality. To meet these requirements, we propose to use the spotlight and oral papers from ICML 2025, a top-tier AI conference. We filter out any papers with arXiv submissions predating the most recent data cutoff of the LLMs we test (GPT-5, October 2024[2]). From this filtered set, we randomly select 200 papers as our test set. Additionally, we randomly sample 30 papers as a development set, keeping the test set reserved exclusively for final evaluation.

[2] https://platform.openai.com/docs/models/gpt-5

3.4 Expert Assessment of Extracted Research Plans and Rubrics

To validate the quality of our extraction pipeline, we conduct a human evaluation study with eight Ph.D. students and researchers with expertise in AI. Each expert is asked to select one paper from their area. This covers domains including computer vision, natural language processing, graph machine learning, AI for science, and theoretical machine learning. The assessment consists of two components:

1. Experts are presented with the LLM-extracted research plan alongside the original paper, with instructions such as: "Please rate how accurately the LLM captured the content from the original paper." Ratings are provided on a five-point Likert scale.

2. Experts are shown the extracted rubric questions and the corresponding research idea, then asked to "assess whether the questions cover the essential parts of a research plan for the given idea" on a five-point Likert scale.

We present the expert evaluations in Table 2. Across sections, mean scores exceed 4.0 out of 5 (scale anchors: 1 = major issues, 2 = significant problems, 3 = average, 4 = acceptable with some issues, 5 = well done overall). These scores suggest that the extracted research plans and rubrics are of reasonable quality (more details in §C).

3.5 Grading and LLM-based Judges

The rubric for each paper consists of five sections and there is a list of yes/no questions to specify the criteria that a research plan should satisfy. After grading each question in a section, we compute binary classification accuracy—the proportion of rubric questions judged as satisfied. This accuracy represents the *Planning Score* for that section. We then take the macro-average of the five section accuracies as the final *Planning Score* for this paper. Finally, we compute the overall score by averaging these paper-level scores across all papers, which corresponds to a macro-average over papers. So our main metric is thus the **average Planning Score** across all papers.

Since manually grading each generated research plan against our rubrics would be prohibitively expensive, we employ LLM-based judges (Zheng et al., 2023; Starace et al., 2025). Specifically, we prompt an LLM with the generated research plan and the rubric, and instruct the LLM to generate a grading (*i.e.*, give a yes/no judgement for each rubric question). We discuss how we assess the quality of LLM-based judges in the next section. We present the prompt templates in §D.

3.6 Evaluating LLM Judges with JudgeEval

We collect a dataset (*Idea2Plan JudgeEval*) to assess the performance of different LLM judges against human expert annotations. From our development set of 30 papers, we randomly select five papers and generate two research plans for each, resulting in 10 generated research plans. For each plan, we collect ground-truth annotations by manually grading the plans against rubric questions. Since each rubric question is a binary classification task (a yes/no question), we report standard binary classification metrics with macro-averaging to aggregate performance across papers. We include a random baseline where the judge assigns yes or no randomly.

The results show that o4-mini achieves the highest F1 score (0.91), while GPT-4.1-mini offers the most cost-effective option. We select o4-mini (reasoning=high) as our evaluation model to achieve the best possible grading accuracy.

3.7 Baseline Choices and Agent Design

We evaluate several baselines for research plan generation, ranging from simple prompting to agentic frameworks with external tools.

Naïve Baseline. Our simplest baseline uses a minimal prompt: *"Your task is to generate a detailed research plan based on the provided research idea."* in addition to the desired structure (*e.g.*, Methods, Initial Experimental Design) for the research plan template to constrain the generation format. Without the section titles, we observe that some LLMs start to generate sections that are not in the grading rubrics (*e.g.*, research timelines), which may waste generation tokens and negatively influence the performance of the naïve baseline. This baseline mimics the most straightforward scenario where scientists directly prompt an LLM for research plans without any additional instructions.

0-shot and 1-shot Baseline. We enhance the basic prompt with additional instructions. For the 0-shot, we add general guidance such as *"Gener-*

Table 2: Expert evaluation of the extracted research plans and rubrics, rated on a five-point Likert scale (1 = major issues, 5 = well done overall). The consistently high scores (above 4.0) suggest that the extracted research plans and rubrics are of reasonable quality.

Criterion	Generated Plan	Generated Rubrics
Introduction	4.12 ± 0.83	4.00 ± 1.07
Related Lit.	4.00 ± 0.76	4.12 ± 0.99
Method	4.06 ± 1.08	4.00 ± 0.93
Experiments	4.25 ± 0.71	4.50 ± 0.76
Resource	4.50 ± 0.76	4.62 ± 0.52
Ethics	4.88 ± 0.35	4.75 ± 0.46

Table 3: Idea2Plan JudgeEval: Macro-averaged performance of LLM judges against human annotations. Metrics are computed per paper and macro-averaged across papers. *Cost* denotes the average API cost ($) to grade one plan against a rubric. The Random baseline predicts labels uniformly at random.

	Acc.	Prec.	Rec.	F1	Cost ($)
Random	0.52	0.50	0.49	0.49	0
GPT-4.1-mini	0.89	0.89	0.90	0.89	0.0048
GPT-4.1	0.89	0.86	0.91	0.88	0.0800
o4-mini	**0.91**	**0.94**	0.89	**0.91**	0.1606
GPT-5-mini	0.84	0.78	**0.93**	0.85	0.0268
GPT-5	0.89	0.92	0.85	0.88	0.1596

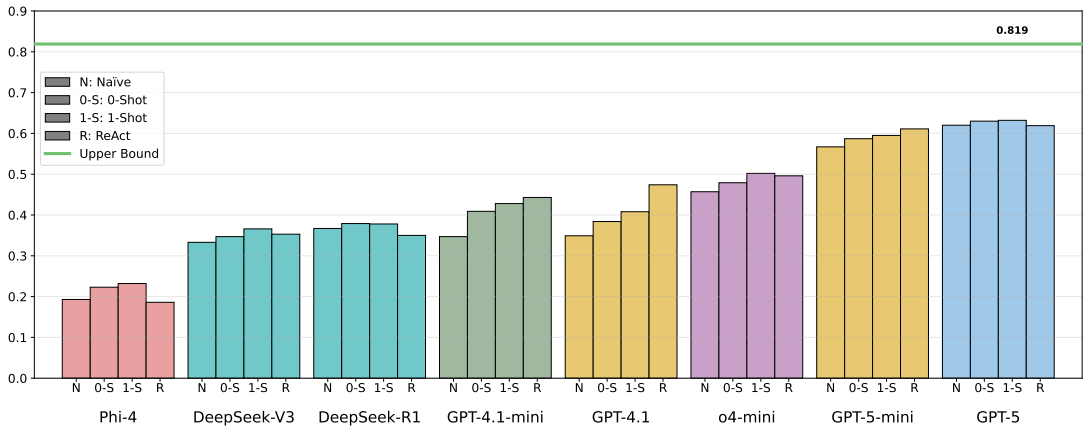


Figure 2: Average Planning Score for LLMs under four baselines (Naïve, 0-Shot, 1-Shot, ReAct). Scores are means over three independent runs per paper. We also include an *upper-bound* in which o4-mini is given the original paper to generate a plan, approximating the highest achievable performance on *Idea2Plan Bench*.

Table 4: Section-wise Planning Scores (%) for Naïve and ReAct baselines. Each value represents the mean accuracy across all papers. Results for 0-shot and 1-shot baselines are provided in Table 20.

Model	Naïve						ReAct					
	Intro	Lit	Met	Exp	Res/Eth	Avg	Intro	Lit	Met	Exp	Res/Eth	Avg
Phi-4	29.0	5.6	20.2	10.9	30.9	19.3	24.2	11.8	15.5	9.9	31.7	18.6
DeepSeek-V3	39.8	25.2	32.4	24.7	44.5	33.3	39.9	23.8	33.4	28.5	50.9	35.3
DeepSeek-R1	40.6	27.1	37.0	28.7	50.3	36.7	38.7	23.1	33.2	27.9	52.0	35.0
GPT-4.1-mini	42.3	21.8	34.7	27.4	47.2	34.7	50.6	30.6	44.5	35.7	60.3	44.3
GPT-4.1	40.7	22.4	34.2	28.9	48.7	34.9	50.6	35.6	44.9	40.5	65.6	47.4
o4-mini	50.8	29.1	50.2	39.1	59.7	45.7	54.2	36.1	54.4	42.6	61.2	49.6
GPT-5-mini	58.5	38.1	65.9	51.6	69.5	56.7	**62.3**	44.5	67.3	55.2	**76.4**	61.1
GPT-5	**60.9**	**47.7**	**70.9**	**56.7**	**73.8**	**62.0**	61.4	**48.1**	**68.4**	**55.7**	75.9	**61.9**

ate a complete research plan following the EXACT template structure above." The 1-shot additionally includes one fixed research plan example to demonstrate the desired format and content structure.

ReAct Agent Baseline. When developing research ideas, human scientists often review related literature and refine their plans by learning how others have addressed similar problems. Inspired by this, we evaluate an agentic framework that can access external information. We use a simple ReAct scaffolding following the "Thought → Action → Observation" loop (Yao et al., 2023). To simulate an environment where scientists interact with the research literature, we introduce two tools integrated with arXiv:

- **ArXiv Search Tool with Blocklists.** Queries arXiv for relevant papers while preventing data contamination by applying two layers of blacklisting for each target paper: (1) exclude all papers published after January 30, 2025 (ICML 2025 deadline), and (2) exclude the target paper and all papers that cite it. The blacklisting is necessary because our small-scale experiments show that search engines can retrieve the target paper when given its research idea as a query. The tool takes an agent-generated query as input and returns a list of relevant arXiv papers with their titles, abstracts, and arXiv identifiers. We implement it using Bing Search[3] restricted to the arXiv domain.
- **ArXiv Read Tool.** Takes an arXiv identifier as input and returns a summary of the corresponding paper. We fetch the paper using the arXiv API[4] and then use o4-mini to generate its summary with a fixed template that contains main contributions, key related literature, methods and techniques, experimental design and results. In this way, we avoid using the full content of a paper, as it would consume too much context and, as shown in our small-scale experiments, may reduce performance.

[3] https://learn.microsoft.com/en-us/azure/ai-foundry/agents/how-to/tools/bing-grounding
[4] https://info.arxiv.org/help/api/index.html

We limit the agent to at most five searches and five reads. The agent is free to decide the order of tool calls and whether to use all of the available calls. Prompt templates and design details are in §E.

4 Experiments

4.1 Experimental setup

We test extensively on the proprietary LLMs (GPT-4.1-mini[5], GPT-4.1[5], o4-mini[6], GPT-5-mini[7], GPT-5[7]) and open-source LLMs (DeepSeek-V3 (DeepSeek-AI et al., 2024), DeepSeek-R1 (DeepSeek-AI et al., 2025) and Phi-4 (Abdin et al., 2024)). For each model and baseline configuration, we run three independent trials per idea and report the mean performance. We also include an *upper-bound*, where o4-mini is prompted with the original paper and asked to generate a research plan. This serves as an estimate of the highest achievable performance on *Idea2Plan Bench*.

4.2 Results Analysis

We present the average scores across all models and settings in Figure 2 and the section-wise results for naïve and ReAct in Table 4. The full results are in §F. We summarize the key findings below.

Naïve baseline achieves strong performance. Despite being simple, the naïve baseline (one line prompt with section titles) attains performance close to the best methods across different models.

0-shot and 1-shot baselines improve over naïve. The 0-shot and 1-shot baselines consistently outperform the naïve baseline on all LLMs, demonstrating that clear instruction and example provide meaningful improvements. However, this performance gap is notably smaller for GPT-5, suggesting it has a stronger baseline capability that is less dependent on explicit prompting strategies.

GPT-5 and GPT-5-mini lead the performance. We notice that GPT-5 and GPT-5-mini substantially outperform other models. To understand this better, we compute pairwise win rates, *i.e.*, for each pair of models, the fraction of papers in which one model's research plan scores higher than the other. We find that GPT-5 and GPT-5-mini achieve over 90% win rates against other models, indicating consistent superiority (see Figure 5).

[5] https://openai.com/index/gpt-4-1/

[6] https://openai.com/index/introducing-o3-and-o4-mini/

[7] https://openai.com/gpt-5/

Literature review is the hardest section, while method and experiment sections show large differences across models. As shown in Table 4, Planning Scores for the literature section are the lowest across LLMs, reflecting the challenge of accurately identifying and integrating relevant prior work. The method and experiment sections exhibit the largest differences across LLMs, serving as clearer indicators of planning ability. GPT-5 and GPT-5-mini perform strongly across sections.

ReAct agent does not surpass simpler baselines. Contrary to expectations, the ReAct agent does not improve performance compared to simpler baselines. We find that models sometimes struggle to intelligently filter retrieved information, and thus incorporating irrelevant details that distract from the research idea. We show an example in Table 22. This highlights the longstanding challenge of knowledge conflict between parametric and retrieved knowledge in language models (Xu et al., 2024; Li et al., 2025).

4.3 Length vs. Quality Analysis

Because our evaluation rubric measures coverage of key elements in a research plan, longer outputs may gain an advantage by mentioning more rubric-relevant content. To examine this potential bias, we analyze the relationship between output length and average Planning Score. We run two controlled settings: a *constrained generation* setting that limits response length and an *expanded generation* setting that allows more extensive output. For each idea, we prompt models with a soft length constraint (*e.g.*, "Your response must be at most <upper> words."). We find the simple approach effectively controls LLM's verbosity.

When output length is controlled (Figure 3, left), GPT-5 and GPT-5-mini still achieve the highest Planning Scores, demonstrating strong robustness to reduced verbosity. In the expanded-length setting (Figure 3, right), we test GPT-4.1 and o4-mini and observe that both models benefit from producing longer outputs. Despite these gains, GPT-5 consistently remains the top performer, suggesting that its advantage comes from higher plan quality.

4.4 Enhanced Context Experiment

For ReAct, we observe that the retrieved papers are sometimes only loosely relevant or even misleading, resulting in degraded research plans (case study in appendix Table 22). Thus, we hypothe-

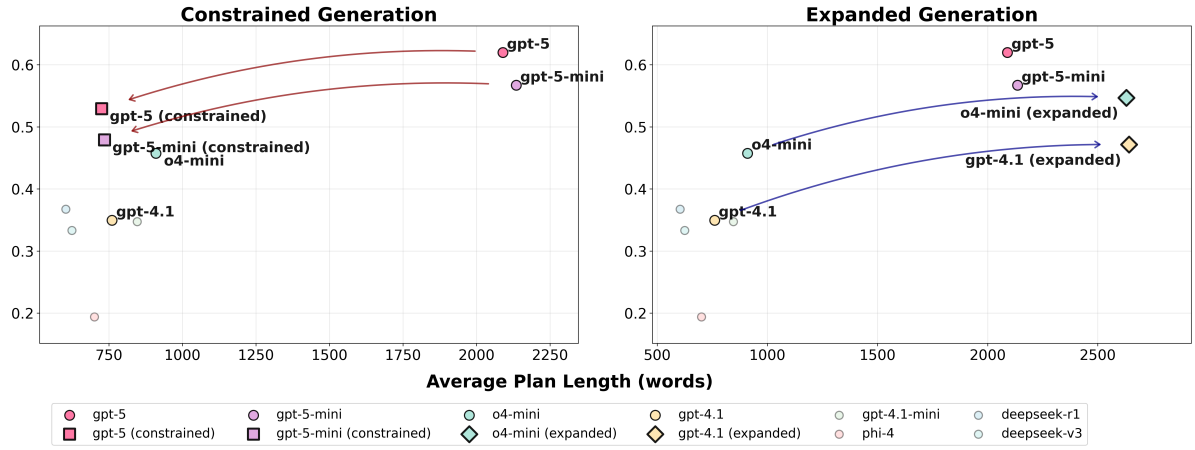


Figure 3: Relationship between plan length and average Planning Score under constrained and expanded generation settings. **Left:** When response length is controlled, GPT-5 and GPT-5-mini generate shorter plans but remain the top performers under stricter length constraints. **Right:** When longer outputs are allowed, GPT-4.1 and o4-mini produce more extensive plans with higher scores. Arrows indicate the shift from the original to the constrained or expanded setting.

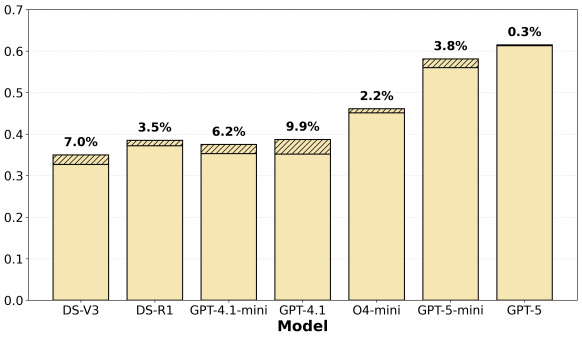


Figure 4: Impact of curated literature on average Planning Score across LLMs. Bars show baseline performance (solid) and the improvement with curated literature (hatched), which enhances plan quality across all models. DS denotes DeepSeek.

size that ReAct agents underperforms due to low-quality retrievals. To test this, we design an alternative setup where models receive up to three carefully curated, directly relevant papers—mimicking real-world scenarios in which researchers start with a few known reference papers.

As shown in Figure 4, curated literature leads to consistent gains across all models, with improvements ranging from 0.3% to 9.9%. Per-section results are reported in Table 24 (appendix §I). Mid-tier models (*e.g.*, GPT-4.1) benefit most—especially in introduction and related work—indicating that external grounding compensates for weaker prior knowledge. In contrast, GPT-5 already performs strongly and shows mixed effects, suggesting curated knowledge is most helpful for models with limited internal understanding.

4.5 Potential Training Strategy

A natural progression towards improving the research planning capabilities of LLMs is to train models using idea–plan pairs derived from published research papers. We conduct supervised fine-tuning (SFT) of GPT-4.1-mini and GPT-4.1 on 2,000 idea–plan pairs extracted from randomly sampled ICML 2024 papers, and we evaluate their performance against the corresponding off-the-shelf models in a 1-shot scenario. As discussed in appendix §J, the SFT models exhibit a decrease in overall performance (Table 25). We find that the obtained fine-tuned models tend to hallucinate more, especially in the literature review sections. We offer some potential explanations and suggest future directions of research in appendix §J.

5 Conclusion

We presented an investigation into automating the process of transforming scientific ideas into research plans, in which we evaluated the capabilities of different LLMs, for handling this task. We proposed a framework that leverages published academic papers to extract paired ideas and plans for model assessment. Through comparative analysis of multiple LLMs using standard and agent-based frameworks, we helped establishing a foundation for advancement in automated research planning.

6 Limitations

Our study focuses on AI research planning tasks derived from ICML 2025 oral and spotlight papers,

which may limit generalizability to other scientific domains or earlier-stage research ideas. Additionally, while we blocklisted post-cutoff and cited papers to mitigate data contamination, we cannot completely rule out partial memorization from overlapping text sources. Finally, all evaluations rely on LLM-based judges, which—despite strong correlation with human assessments—may still introduce systematic biases due to model alignment or reasoning errors.

7 Ethical Considerations

Our framework operates on publicly available research papers and adheres to fair-use principles. Generated plans are intended for research analysis, not for automated publication or deployment. We recognize the potential misuse of AI systems for producing unverified or plagiarized scientific outputs and emphasize that our work aims to benchmark research planning ability, not to replace human scientific judgment. We also ensure that no personally identifiable or sensitive data is used in our datasets.

References

Marah I Abdin, Jyoti Aneja, Harkirat S. Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J. Hewett, Mojan Javaheripi, Piero Kauffmann, James R. Lee, Yin Tat Lee, Yuanzhi Li, Weishung Liu, Caio C. T. Mendes, Anh Nguyen, Eric Price, Gustavo de Rosa, Olli Saarikivi, and 8 others. 2024. Phi-4 technical report. *CoRR*, abs/2412.08905.

Lakshya A. Agrawal, Shangyin Tan, Dilara Soylu, Noah Ziems, Rishi Khare, Krista Opsahl-Ong, Arnav Singhvi, Herumb Shandilya, Michael J. Ryan, Meng Jiang, Christopher Potts, Koushik Sen, Alexandros G. Dimakis, Ion Stoica, Daniel Klein, Matei Zaharia, and Omar Khattab. 2025. GEPA: reflective prompt evolution can outperform reinforcement learning. *CoRR*, abs/2507.19457.

Jinheon Baek, Sujay Kumar Jauhar, Silviu Cucerzan, and Sung Ju Hwang. 2025. Researchagent: Iterative research idea generation over scientific literature with large language models. In *NAACL*, pages 6709–6738. Association for Computational Linguistics.

Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramer, and Chiyuan Zhang. 2022. Quantifying memorization across neural language models. *arXiv preprint arXiv:2202.07646*.

Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, and 1 others. 2021. Extracting training data from large language models. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 2633–2650.

Hui Chen, Miao Xiong, Yujie Lu, Wei Han, Ailin Deng, Yufei He, Jiaying Wu, Yibo Li, Yue Liu, and Bryan Hooi. 2025a. Mlr-bench: Evaluating ai agents on open-ended machine learning research. *arXiv preprint arXiv:2505.19955*.

Ziru Chen, Shijie Chen, Yuting Ning, Qianheng Zhang, Boshi Wang, Botao Yu, Yifei Li, Zeyi Liao, Chen Wei, Zitong Lu, Vishal Dey, Mingyi Xue, Frazier N. Baker, Benjamin Burns, Daniel Adu-Ampratwum, Xuhui Huang, Xia Ning, Song Gao, Yu Su, and Huan Sun. 2025b. Scienceagentbench: Toward rigorous assessment of language agents for data-driven scientific discovery. In *ICLR*. OpenReview.net.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, and 81 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *CoRR*, abs/2501.12948.

DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, and 80 others. 2024. Deepseek-v3 technical report. *CoRR*, abs/2412.19437.

Ben Finkelshtein, İsmail İlkan Ceylan, Michael M. Bronstein, and Ron Levie. 2025. Equivariance everywhere all at once: A recipe for graph foundation models. *CoRR*, abs/2506.14291.

Alireza Ghafarollahi and Markus J. Buehler. 2024. Sciagents: Automating scientific discovery through multi-agent intelligent graph reasoning. *CoRR*, abs/2409.05556.

Juraj Gottweis, Wei-Hung Weng, Alexander N. Daryin, Tao Tu, Anil Palepu, Petar Sirkovic, Artiom Myaskovsky, Felix Weissenberger, Keran Rong, Ryutaro Tanno, Khaled Saab, Dan Popovici, Jacob Blum, Fan Zhang, Katherine Chou, Avinatan Hassidim, Burak Gokturk, Amin Vahdat, Pushmeet Kohli, and 15 others. 2025. Towards an AI co-scientist. *CoRR*, abs/2502.18864.

Boyu Gou, Ruohan Wang, Boyuan Zheng, Yanan Xie, Cheng Chang, Yiheng Shu, Huan Sun, and Yu Su. 2025. Navigating the digital world as humans do: Universal visual grounding for GUI agents. In *ICLR*. OpenReview.net.

Tarun Gupta and Danish Pruthi. 2025. All that glitters is not novel: Plagiarism in AI generated research. In *ACL*, pages 25721–25738. Association for Computational Linguistics.

Yichen He, Guanhua Huang, Peiyuan Feng, Yuan Lin, Yuchen Zhang, Hang Li, and Weinan E. 2025. Pasa: An LLM agent for comprehensive academic paper search. In *ACL*, pages 11663–11679. Association for Computational Linguistics.

Peter Jansen, Oyvind Tafjord, Marissa Radensky, Pao Siangliulue, Tom Hope, Bhavana Dalvi Mishra, Bodhisattwa Prasad Majumder, Daniel S. Weld, and Peter Clark. 2025. Codescientist: End-to-end semi-automated scientific discovery with code-based experimentation. In *Findings of the Association for Computational Linguistics, ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pages 13370–13467. Association for Computational Linguistics.

Priyanka Kargupta, Ishika Agarwal, Tal August, and Jiawei Han. 2025. Tree-of-debate: Multi-persona debate trees elicit critical thinking for scientific comparative analysis. In *ACL*, pages 29378–29403. Association for Computational Linguistics.

Patrick Tser Jern Kon, Jiachen Liu, Qiuyi Ding, Yiming Qiu, Zhenning Yang, Yibo Huang, Jayanth Srinivasa, Myungjin Lee, Mosharaf Chowdhury, and Ang Chen. 2025a. Curie: Toward rigorous and automated scientific experimentation with AI agents. *CoRR*, abs/2502.16069.

Patrick Tser Jern Kon, Jiachen Liu, Xinyi Zhu, Qiuyi Ding, Jingjia Peng, Jiarong Xing, Yibo Huang, Yiming Qiu, Jayanth Srinivasa, Myungjin Lee, and 1 others. 2025b. Exp-bench: Can ai conduct ai research experiments? *arXiv preprint arXiv:2505.24785*.

Gaotang Li, Yuzhong Chen, and Hanghang Tong. 2025. Taming knowledge conflicts in language models. *CoRR*, abs/2503.10996.

Chris Lu, Cong Lu, Robert Tjarko Lange, Jakob N. Foerster, Jeff Clune, and David Ha. 2024. The AI scientist: Towards fully automated open-ended scientific discovery. *CoRR*, abs/2408.06292.

Kristina Nikolic, Luze Sun, Jie Zhang, and Florian Tramèr. 2025. The jailbreak tax: How useful are your jailbreak outputs? *CoRR*, abs/2504.10694.

Samuel Schmidgall and Michael Moor. 2025. Agentrxiv: Towards collaborative autonomous research. *CoRR*, abs/2503.18102.

Samuel Schmidgall, Yusheng Su, Ze Wang, Ximeng Sun, Jialian Wu, Xiaodong Yu, Jiang Liu, Zicheng Liu, and Emad Barsoum. 2025. Agent laboratory: Using LLM agents as research assistants. *CoRR*, abs/2501.04227.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Minju Seo, Jinheon Baek, Seongyun Lee, and Sung Ju Hwang. 2025. Paper2code: Automating code generation from scientific papers in machine learning. *CoRR*, abs/2504.17192.

Chenglei Si, Diyi Yang, and Tatsunori Hashimoto. 2025. Can llms generate novel research ideas? A large-scale human study with 100+ NLP researchers. In *ICLR*. OpenReview.net.

Amanpreet Singh, Joseph Chee Chang, Chloe Anastasiades, Dany Haddad, Aakanksha Naik, Amber Tanaka, Angele Zamarron, Cecile Nguyen, Jena D. Hwang, Jason Dunkleberger, Matt Latzke, Smita Rao, Jaron Lochner, Rob Evans, Rodney Kinney, Daniel S. Weld, Doug Downey, and Sergey Feldman. 2025. Ai2 scholar QA: organized literature synthesis with attribution. *CoRR*, abs/2504.10861.

Yuda Song, Hanlin Zhang, Carson Eisenach, Sham M. Kakade, Dean P. Foster, and Udaya Ghai. 2025. Mind the gap: Examining the self-improvement capabilities of large language models. In *ICLR*. OpenReview.net.

Giulio Starace, Oliver Jaffe, Dane Sherburn, James Aung, Jun Shern Chan, Leon Maksin, Rachel Dias, Evan Mays, Benjamin Kinsella, Wyatt Thompson, Johannes Heidecke, Amelia Glaese, and Tejal Patwardhan. 2025. Paperbench: Evaluating ai's ability to replicate AI research. *CoRR*, abs/2504.01848.

K Sudheesh, Devika Rani Duggappa, and SS Nethra. 2016. How to write a research proposal? *Indian journal of anaesthesia*, 60(9):631–634.

Qingyun Wang, Doug Downey, Heng Ji, and Tom Hope. 2024a. Scimon: Scientific inspiration machines optimized for novelty. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 279–299. Association for Computational Linguistics.

Yidong Wang, Qi Guo, Wenjin Yao, Hongbo Zhang, Xin Zhang, Zhen Wu, Meishan Zhang, Xinyu Dai, Min Zhang, Qingsong Wen, Wei Ye, Shikun Zhang, and Yue Zhang. 2024b. Autosurvey: Large language models can automatically write surveys. In *NeurIPS*.

Robert J Weber and Daniel J Cobaugh. 2008. Developing and executing an effective research plan. *American journal of health-system pharmacy*, 65(21):2058–2065.

Rongwu Xu, Zehan Qi, Zhijiang Guo, Cunxiang Wang, Hongru Wang, Yue Zhang, and Wei Xu. 2024. Knowledge conflicts for llms: A survey. In *EMNLP*, pages 8541–8565.

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

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R. Narasimhan, and Yuan Cao. 2023. React: Synergizing reasoning and acting in language models. In *ICLR*.

Jiakang Yuan, Xiangchao Yan, Botian Shi, Tao Chen, Wanli Ouyang, Bo Zhang, Lei Bai, Yu Qiao, and Bowen Zhou. 2025. Dolphin: Closed-loop open-ended auto-research through thinking, practice, and feedback. *arXiv preprint arXiv:2501.03916*.

Xingjian Zhang, Yutong Xie, Jin Huang, Jinge Ma, Zhaoying Pan, Qijia Liu, Ziyang Xiong, Tolga Ergen, Dongsub Shim, Honglak Lee, and Qiaozhu Mei. 2025. MASSW: A new dataset and benchmark tasks for ai-assisted scientific workflows. In *Findings of NAACL*, pages 2373–2394. Association for Computational Linguistics.

Xinran Zhao, Boyuan Zheng, Chenglei Si, Haofei Yu, Ken Liu, Runlong Zhou, Ruochen Li, Tong Chen, Xiang Li, Yiming Zhang, and Tongshuang Wu. 2025. The ramon llull's thinking machine for automated ideation. *CoRR*, abs/2508.19200.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. In *NeurIPS*.

Minjun Zhu, Qiujie Xie, Yixuan Weng, Jian Wu, Zhen Lin, Linyi Yang, and Yue Zhang. 2025. AI scientists fail without strong implementation capability. *CoRR*, abs/2506.01372.

A Full Example of Jailbreak-Tax (Nikolic et al., 2025) paper's Rubric and Research Plan

We present the full example in Table 1, including grading rubric for the Jailbreak-Tax paper and the research plan generated by GPT-5 (Nikolic et al., 2025). At the end of the paper, Tables 26–29 show the complete evaluation rubric across four parts, while Tables 30–31 present the full research plan generated by GPT-5.

B Dataset Construction

B.1 Research Plan and Idea Extraction

We provide the complete prompt templates used for extracting research plans and ideas in our dataset construction pipeline:

- *Research Plan Template:* The Research Plan Template (Table 5) defines the structured format for generating research plans, covering introduction, literature review, methods, experimental design, and resource considerations.
- *Idea Extraction:* The Research Idea Extraction Prompt (Table 6) transforms paper abstracts into concise, first-person research ideas that focus on the proposed approach rather than results.
- *Research Plan Extraction:* The Research Plan Extraction Prompt (Table 7) extracts detailed research plans from full papers following the Research Plan Template.

B.2 Rubric Generation

The Rubric Generation Prompt (Table 8) instructs the model to create evaluation criteria in JSON format for assessing research plans. Since this prompt is long, we separate two components that are plugged into the placeholders:

- *Section-by-Section Guidance:* (Table 9) provides detailed instructions for generating consistent rubric questions across all sections.
- *Poor vs. Better Question Examples:* (Table 10) illustrates common pitfalls and demonstrates how to formulate clear, generalizable evaluation criteria.

C Expert Evaluation of Reference Research Plan and Rubric

Eight papers were selected by the expert evaluators, each from their respective research domain. The papers span diverse areas of AI and Machine Learning (ML) research:

- **AI4Science.** The AI Scientist (Lu et al., 2024)
- **Natural Language Processing and AI Agents.** GEPA (Agrawal et al., 2025), PaSa (He et al., 2025), Tree-of-Debate (Kargupta et al., 2025)
- **Graph ML.** Equivariance Everywhere All At Once (Finkelshtein et al., 2025)
- **Computer Vision.** UGround (Gou et al., 2025), one unpublished paper
- **Theoretical ML.** Mind the Gap (Song et al., 2025)

C.1 Guideline to Experts

We provide comprehensive guidelines to expert annotators for evaluating both the LLM-generated research plans and rubric questions. The complete evaluation guidelines are included in Table 11 and Table 12, which detail the rating scales and evaluation criteria for each section.

Table 5: Research Plan Template

Research Plan Template

1. Introduction

1.1 Background
Describe the current limitations or challenges in the field and why they matter now. This should be one paragraph.

1.2 Primary Objectives
List specific, measurable goals that define project success. These should align with the contributions typically outlined in a paper's introduction.

1.3 Research Questions
State the core research questions for this research project.

2. Key Literatures
Identify key related works and relevant domains that inform your research. Begin by listing a few key domains, and for each cited work, briefly explain why it is important to your current research plan.

3. Methods
Describe the core techniques, model architectures, and/or training strategies to be used. This should be aligned with the method section in a paper.

4. Initial Experimental Design
Describe the high-level design of your first experiments. This should align with the experiments section of a paper.

5. Resources, Compliance, and Ethical Considerations

5.1 Resource Requirements
List and estimate the resources needed to carry out your research. Focus on the most informative metrics that impact the budget and feasibility, such as GPU hours, token usage for API calls, and human annotation costs.

5.2 Ethical and Compliance Considerations
Address data privacy, safety, and approval needs.

C.2 Data Consent

For papers published on ACL (*e.g.*, Kargupta et al. (2025)) we have the right to build upon for non-commercial use[8]. For other papers the copyright belongs to the author, the responsibility for any copyright or rights violations lies with the authors of the work.

C.3 Characteristics of Annotators

We recruit eight volunteer annotators who are experts in AI. All annotators have specialization in AI and natural language processing. Our annotator team consists of researchers with the following demographic composition: 75% from Asia, 12.5% from the Middle East, and 12.5% from the United States. Each annotator independently evaluated assigned papers following the guideline.

D LLM-based Judge

We use an LLM-as-a-judge approach to evaluate research plans against the generated rubrics, with the Rubric Evaluation Prompt (Table 13) guiding the model to perform assessment with strict interpretation of rubric criteria. We use o4-mini (reasoning=high) throughout the study. We notice that grading a full plan in one API call sometimes triggers API failures due to the long input length, so we grade by section and aggregate. For each plan–rubric pair, we make five requests—one per section—and combine the section scores into the final result.

E Baseline and Agent Design

E.1 Prompting Baselines

We design three prompting baselines for research plan generation:

- *Naïve Baseline:* The Naïve Baseline Prompt (Table 14) provides minimal instructions with only the research idea and template structure.
- *0-shot Baseline:* The 0-shot Baseline Prompt (Table 15) adds explicit generation instructions to guide the model through the planning process.
- *1-shot Baseline:* The 1-shot Baseline Prompt (Table 16) includes the same instructions as the 0-shot baseline, with one example research plan.

E.2 ReAct Agent

The ReAct Agent Prompt (Table 17) implements an iterative reasoning framework that enables the agent to systematically gather information through tool use before generating the research plan (Yao et al., 2023). The agent has access to two tools specified in Table 18: `search_papers` for finding relevant academic papers using Bing Custom Search, and `read_paper` for retrieving and analyzing specific papers by their arXiv ID. These tool specifications (referred to as `{{TOOL_SPECIFICATIONS}}`

[8] https://aclanthology.org/faq/copyright/

Table 6: Research Idea Extraction Prompt

Research Idea Extraction Prompt

Extract the core research idea from a paper abstract. Transform the abstract into a concise, intuitive research idea that focuses on the proposed approach, not the results.

Task
Convert the abstract into a research idea using first-person language that captures:
- The problem we aim to solve
- Our proposed method/approach/dataset, etc.
- What makes it novel

Instructions
- Remove the proposed method or dataset names (e.g., replace "HumanEval" with "a code generation benchmark")
- Remove experimental results and performance claims
- Keep the core technical approach and methodology
- Use first-person planning language (We propose, We aim, We will)
- Stay concise and intuitive

Example

Input Abstract:
"Scientific literature understanding is crucial for extracting targeted information and garnering insights, thereby significantly advancing scientific discovery. Despite the remarkable success of Large Language Models (LLMs), they face challenges in scientific literature understanding, primarily due to (1) a lack of scientific knowledge and (2) unfamiliarity with specialized scientific tasks. To develop an LLM specialized in scientific literature understanding, we propose a hybrid strategy that integrates continual pre-training (CPT) and supervised fine-tuning (SFT), to simultaneously infuse scientific domain knowledge and enhance instruction-following capabilities for domain-specific tasks. In this process, we identify two key challenges: (1) constructing high-quality CPT corpora, and (2) generating diverse SFT instructions. We address these challenges through a meticulous pipeline, including PDF text extraction, parsing content error correction, quality filtering, and synthetic instruction creation. Applying this strategy, we present a suite of LLMs: SciLitLLM, specialized in scientific literature understanding. These models demonstrate promising performance on scientific literature understanding benchmarks."

Output Research Idea:
We propose a hybrid strategy that integrates continual pre-training (CPT) and supervised fine-tuning (SFT) to develop LLMs specialized in scientific literature understanding, addressing the challenges of lack of scientific knowledge and unfamiliarity with specialized scientific tasks. We aim to develop a pipeline for constructing high-quality CPT corpora and generating diverse SFT instructions through PDF text extraction, parsing error correction, and quality filtering.

Input Abstract
{{ABSTRACT}}

Output
Write the research idea using first-person style, focusing on the approach and novelty while removing experimental results and work names.

in the prompt) and tool names (referred to as `{{TOOL_NAMES}}` in the prompt) are provided to the agent to enable systematic information gathering.

The `search_papers` tool returns the top 10 most relevant papers, each including its arXiv ID, title, and abstract. When the agent calls `read_paper`, it uses the Paper Summarization Prompt (Table 19) to produce structured summaries of each retrieved paper. This prompt guides the model to extract a paper's main contributions, key related literature, methods and techniques, and experimental design and results.

F Full Experimental Results

Table 20 and Table 21 present detailed section-wise results across all models and prompting settings. For each research idea, we run every configuration three times. The **mean** results report the average score over the three runs, while the **max** results report the best-performing run among them. Consistent trends appear across both metrics: GPT-5 achieves the highest scores under all prompting setups, followed by GPT-5-mini and o4-mini. 1-shot prompting provides a small but steady improvement over 0-shot, indicating that a single example helps models organize their research plans more coherently. In contrast, the ReAct setup does not outperform simpler prompting strategies.

G Win Rate between LLMs

Figure 5 shows pairwise win rates under four prompting setups. GPT-5 consistently leads across naïve, 0-shot, 1-shot, and ReAct settings, though gains narrow in the ReAct case.

H Case Study: Knowledge Conflict in ReAct Agent

Table 22 illustrates a case where the ReAct agent retrieves papers that conflict with its parametric knowledge, leading to less relevant citations compared to the baseline. While the agent cites specific technical papers on empirical welfare maximization, the baseline correctly identifies the foundational "Prediction Policy Problems" paper from its training data, demonstrating that retrieval can introduce noise when it does not align with the model's existing domain knowledge.

I Enhanced Context Experiment

To investigate whether providing curated related literature improves research plan quality, we design an experiment where models are given 3 key foundational papers selected by the Literature Recommendation Prompt (Table 23). Table 24 shows the section-wise performance comparison with and

Table 7: Research Plan Extraction Prompt

Research Plan Extraction Prompt

Given the following research paper, extract a detailed research plan. Here is the definition of a research plan with some examples. Do not include the definition or any of the examples in your final output.

// begin of research plan definition with examples

{{RESEARCH_PLAN_DEFINITION}}

// end of research plan definition

Additional notes when extracting the research plan:

- Do not include any information that is not explicitly stated in the paper. When listing items (metrics, datasets, models, techniques), include every such item mentioned in the paper. Never add details that is similar to the examples provided in the research plan definition, but not mentioned in the paper.

- **Primary Objectives:** Identify the main goals of the research based on the Introduction section. Limit to a maximum of 5 objectives unless the paper clearly defines more. Focus on the most emphasized goals.

- **Research Questions:** Ensure each question is distinct and non-overlapping.

- **Related Literature:** Focus on identifying the most **foundational and influential** previous works that directly inspired this research or provide **key baselines, datasets, or theoretical foundations**. For each important work:
  - Extract the full title and explain its specific contribution to the current research
  - **Pay special attention to papers cited and discussed in the Methods and Experiments sections** - these are often the most critical works as they represent direct comparisons, baseline methods, or core datasets used
  - Prioritize works that the authors:
    - Compare their approach directly against (baseline methods)
    - Build upon or extend (foundational methods)
    - Use for evaluation (benchmark datasets)
    - Cite as inspiration for their core methodology
  - Explain how each cited work contributes - whether as comparative baselines, theoretical insights, datasets, or methodological foundations
  - Focus on works discussed in detail rather than brief mentions

- **Methodology:** Provide a detailed and logically structured description of the methods used. Ensure alignment with the stated objectives and research questions.

- **Resource Requirements:** Extract any explicitly stated resource requirements mentioned in the paper (e.g., minimum number of GPUs needed, total GPU hours used, total number of tokens consumed, costs related to human annotation). Report only the order of magnitude rather than exact numbers. All resource-related details mentioned in the paper must be extracted. Do not make any guess on the resources if the paper does not specify resources.

- **Ethics:** **Only include ethical considerations that are directly discussed in the paper.** Avoid speculative or generic concerns. Keep this section concise and focused.

- In the research plan, use action-oriented, future-tense language that describes what will be done, not what was found.

Here is the paper:

// begin of the paper

{{FULL_PAPER_TEXT}}

// end of the paper

Now give the extracted research plan, according to the research plan definition.

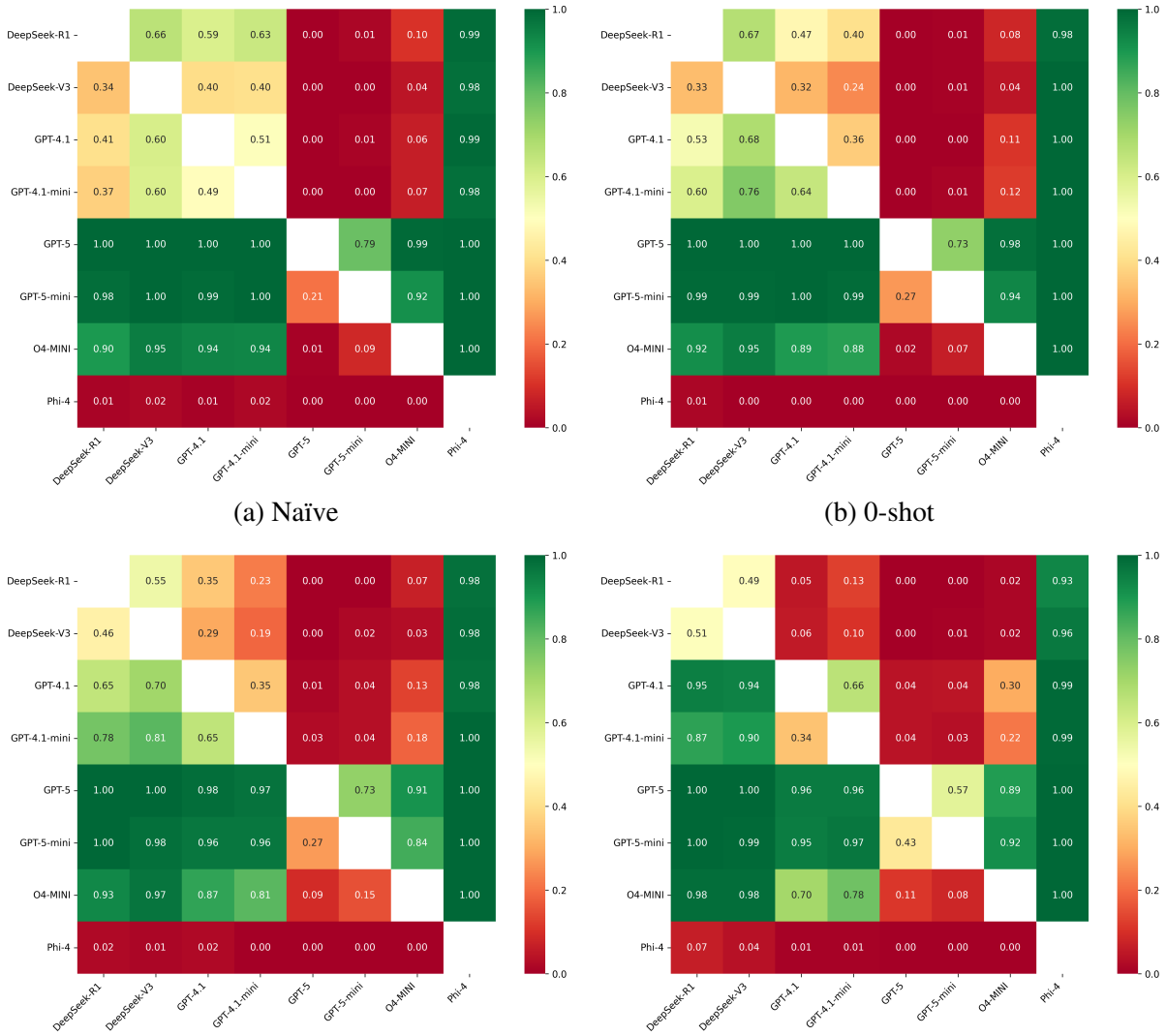
without curated literature across all models. The results demonstrate that providing relevant foundational papers leads to modest improvements in overall performance for most models, with the largest gains observed in the Introduction and Literature sections.

J SFT Experiment

One natural source of high-quality supervision for training autonomous research systems is the corpus of published research papers. These papers provide research ideas with concrete plans, making them appealing as potential training data. To test this hypothesis, we conduct supervised fine-tuning (SFT) using research papers from ICML 2024. From this corpus, we extract approximately 2,000 idea–plan pairs, which serve as our training set. We then fine-tune GPT-4.1-mini and GPT-4.1 on this data using Azure finetuning API[9] and compare their performance against their base versions in 1-shot setting.

Table 25 summarizes the results. Contrary to expectations, SFT leads to a notable drop in overall performance. The largest degradation occurs in the ethics and resources dimension, where research papers often omit practical details such as GPU hours, causing the fine-tuned models to omit this part. Even after excluding this section from the aggregate scores, the SFT models underperform their baselines. This suggests that direct training on research plans does not reliably improve research planning ability. We also observe increased hallucination, particularly in the related-literature section. A possible reason is that extracted research plans contain details absent from the models' pretraining data, which the models interpret during fine-tuning as a signal to fabricate citations.

[9] https://learn.microsoft.com/en-us/azure/ai-foundry/concepts/fine-tuning-overview



(a) Naïve (b) 0-shot

(c) 1-shot (d) ReAct

Figure 5: Pairwise win rate matrices across prompting settings. Each cell (i, j) shows the fraction of research ideas where model i outperforms model j. The four panels correspond to (a) naïve prompting, (b) 0-shot, (c) 1-shot, and (d) ReAct settings. Across all settings, GPT-5 achieves the highest overall win rates.

Future directions. Strengthen the ethics/resources dimension via targeted data augmentation. It is also worthwhile to explore RL-based training (*e.g.*, PPO (Schulman et al., 2017)) in an agentic loop where the model drafts plans, receives rubric feedback, and optimizes for research plan generation quality.

Table 8: Rubric Generation Prompt

You are given a research idea and a plan for executing on that idea. Your task is to create a grading rubric in JSON format that can be used to evaluate other research plans on how well they develop the same research idea. The grading rubric will be used to assess and score alternative research plans that attempt to address the same research question.

Here is the definition of a research plan:

// begin of research plan definition with one example

{{RESEARCH_PLAN_DEFINITION}}

// end of research plan definition with one example

Instructions

1. Generalize the Criteria.

Follow these steps in order:

Step 1: For each section of the research plan, define what constitutes a high-quality research plan when developing the research idea.

Step 2: Verify whether the input plan contains these quality characteristics.

Step 3: Generate a list of yes/no questions that can be used to grade other research plans against this one.
- If a specific term, concept, or methodology appears in the research plan but NOT in the original research idea, do not require it in the rubric questions.
- Judge based on intent and conceptual alignment rather than specific terminology - if a paper proposes a specific term for something, other research plans should not be required to use that exact term, but should include things with similar intent.
- The rubric questions should capture the fundamental criteria that any research plan generated from this research idea should have. Do not ask about anything that is unique or special to this particular research plan.

2. Section-by-Section Guidance.

{{SECTION_BY_SECTION_GUIDANCE}}

3. Structure the Rubric in JSON Format.

Your rubric should be organized as a JSON object with the major sections of a research plan as top-level keys. For each section, list the key elements to check for, starting from high-level concepts (e.g., overall goals) down to specific details (e.g., dataset types, model architectures, evaluation metrics). Each question should be a JSON object with the question text. The rubric must follow this exact JSON structure:

```
{
  "sections": {
    "Section Name": {
      "subsections": {
        "Subsection Name": {
          "questions": [
            {
              "question": "Question text here?"
            }
          ]
        }
      }
    }
  }
}
```

If a section has no subsections and contains questions directly, use this format:

```
{
  "sections": {
    "Section Name": {
      "questions": [
        {
          "question": "Question text here?"
        }
      ]
    }
  }
}
```

Examples

Here is an example of a rubric for a different plan:

// begin of research plan rubric example

{{RESEARCH_PLAN_RUBRIC_EXAMPLE}}

// end of research plan rubric example

{{POOR_VS_BETTER_EXAMPLES}}

Output

Now, here is the research plan and the research idea. Your task is to generate a structured grading rubric in JSON format:

Here is the research idea:

// begin of research idea

{{RESEARCH_IDEA}}

// end of research idea

// begin of research plan

{{RESEARCH_PLAN}}

// end of research plan

IMPORTANT: Your output must be valid JSON following the exact structure specified above. Do not include any text before or after the JSON object. Return only the JSON rubric.

Table 9: Section-by-Section Guidance for Rubric Generation

Section-by-Section Guidance

Use the following guidance to ensure consistency and completeness across all rubric sections:

- Introduction - Background: Write rubric questions that assess whether a plan clearly identifies the motivation and current limitations in the field.

- Introduction - Primary Objectives: Write rubric questions that assess whether a plan defines specific, measurable goals.

- Introduction - Research Questions: Write rubric questions that determine whether a plan articulates focused and relevant research questions.

- Key Literature: Write rubric questions that check whether a plan cites key related work. This is the only section where in-depth discussion of related literature should occur. Group papers by domain. For each paper mentioned in the research plan, create one citation question using the format: "Does the plan cite [insert the paper title here] or similar work on [specific topic]?"

Example: If the plan mentions "Attention Is All You Need", the question should be: "Does the plan cite the paper (Attention Is All You Need) or similar work on transformer architectures?"

It's important not to require exact citation of the specific paper title. The paper title in the question is just an example. Focus on whether the plan cites any work that serves the same purpose or addresses the same topic, not whether it cites that exact paper.

- Methods: Write rubric questions that assess whether a plan outlines a sound technical approach. Do not require an exact method match unless it is the only viable option. Generate questions that capture the high-level requirements for this plan.

- Initial Experimental Design: Write rubric questions that assess whether a plan includes a clear and complete experimental setup. When you mention exact datasets or models, make sure to mention them as examples and put them in parentheses.

- Resource Requirements: Write rubric questions that assess whether a research plan provides a realistic estimate of the resources required.

- Ethical and Compliance Considerations: Write rubric questions that assess whether a plan addresses important ethical and legal responsibilities.

Table 10: Poor vs. Better Question Examples for Rubric Generation

Poor vs. Better Question Examples

Example 1:
Poor: "Does the plan use task rewording or transformation (e.g., EvilMath) to trigger safety mechanisms while preserving semantic fidelity?"
Better: "Does the plan describe how evaluation tasks will be adapted to test safety mechanisms (e.g., using rewording)?"
Reason: "Semantic fidelity" is not defined and unclear. Requiring "use task rewording" is too specific, there could be other ways.

Example 2:
Poor: "Does the plan propose analysis of the effect of model scale and attack type on the jailbreak tax?"
Better: "Does the plan propose analysis of the effect of model scale on performance?" and "Does the plan propose analysis of the effect of attack type on performance?"
Reason: You are asking two things, should be split into two questions.

Example 3:
Poor: "Does the plan include a fine-tuning-based jailbreak using legitimate QA pairs?"
Better: "Does the plan include a fine-tuning-based jailbreak approach?"
Reason: "Legitimate QA pairs" is not clear in this question's context.

Example 4:
Poor: "Does the plan define a clear metric for jailbreak success rate?"
Better: "Does the plan define a clear metric for evaluating attack effectiveness (e.g., success rate for jailbreak)?"
Reason: "Jailbreak success rate" is a specific term from the paper that the agent generating the plan may not capture, while "attack effectiveness" is more general.

Example 5:
Poor: "Does the plan include visualizations of results across alignment techniques, models, and tasks?"
Better: "Does the plan propose analysis of results across alignment techniques, models, and tasks?"
Reason: Research plans describe what will be done, not the final outputs like visualizations.

Table 11: Human Evaluation Guidelines: Research Plan and Rubric Assessment, Part 1

What You're Evaluating

Hi! We're excited to have your help evaluating LLM-generated content using the 1-5 scale below. Your expert feedback is incredibly valuable to us and will help improve our research!

Paper: <PAPER TITLE AND LINK PLACEHOLDER>

You will evaluate TWO pieces of LLM-generated content:
1. LLM-Generated Research Plan from Full Paper - How well does the AI-extracted plan capture the original paper's content?
2. LLM-Generated Rubric Questions - How good are the AI-generated evaluation questions for assessing research plans?

Definition of a Research Plan

<Placeholder for the definition of research plan, see Section X>

PART 1: LLM-GENERATED RESEARCH PLAN FROM FULL PAPER TO EVALUATE

Instructions: The content below was automatically generated by an AI system from the research paper. In the next section we will provide guidelines for you to evaluate this.

<INSERT GENERATED RESEARCH PLAN HERE>

PART 2: YOUR EVALUATION OF THE RESEARCH PLAN

Instructions: Please rate how accurately the LLM captured the content from the original paper.

Rating Scale:
- 1 = Major issues. The research plan fails to capture the key aspects of the evaluation criteria for the respective sections.
- 2 = Significant problems. The research plan only partially addresses the evaluation criteria and overlooks several important aspects.
- 3 = Average. The research plan covers the main criteria but contains notable gaps or inaccuracies compared to what the authors wrote in the paper.
- 4 = Acceptable with some issues. The research plan addresses most of the evaluation criteria well with only minor issues or omissions.
- 5 = Well done overall. The research plan is generally well-constructed and comprehensive, needing only minor adjustments.

Section 1: Introduction (Background, Objectives, Research Questions)
Rate from 1-5: ___
Evaluation Criteria:
- Does the background accurately reflect the paper's motivation and context?
- Are the objectives correctly extracted from the paper's stated goals?
- Do the research questions in the plan match those addressed in the original paper?

Comments: _____

Section 2: Key Literature
Rate from 1-5: ___
Evaluation Criteria:
- Does it include the key references mentioned in the original paper?
- Are any important citations from the paper missing or misrepresented?

Comments: _____

Section 3: Methods
Rate from 1-5: ___
Evaluation Criteria:
- Do the proposed methods accurately reflect the paper's methodology?
- Are the key technical details correctly extracted from the paper?
- Are any key methodological components from the paper missing?

Comments: _____

Section 4: Experimental Design
Rate from 1-5: ___
Evaluation Criteria:
- Does the experimental design match the paper's evaluation setup?
- Are the datasets, metrics, and baselines correctly extracted?
- Are any important experimental details from the paper missing?

Comments: _____

Section 5.1: Resources
Rate from 1-5: ___
Evaluation Criteria:
- Do the resource estimates align with what's described in the paper and appear realistic for the proposed research?

Comments: _____

Section 5.2: Ethics
Rate from 1-5: ___
Evaluation Criteria:
- Are ethical considerations properly identified and addressed, including any mentioned in the paper?

Comments: _____

Table 12: Human Evaluation Guidelines: Research Plan and Rubric Assessment, Part 2

PART 3: YOUR EVALUATION OF THE RUBRIC QUESTIONS

Instructions: The rubric questions you are going to evaluate were automatically generated by an AI system. These questions are intended to evaluate research plans developed from the research idea below. (The given paper should be seen as the result of such research plan.)

The research idea: <INSERT RESEARCH IDEA HERE>

The rubric questions should:
- Assess whether the questions cover the essential parts of a research plan for the given idea that could have reasonably led to the given paper
- Focus on intent and conceptual alignment, rather than specific wording or terminology

Please review the rubric questions below and rate their overall quality using the 1-5 scale.

Rating Scale:
- 1 = Major issues. The rubric questions are irrelevant or do not address the key evaluation criteria for this section.
- 2 = Significant problems. The rubric questions have notable flaws or overlook several important aspects that should be assessed.
- 3 = Average. The rubric questions address some basic evaluation needs but could be improved.
- 4 = Acceptable with some issues. The rubric questions cover the essential evaluation needs but still have some gaps.
- 5 = Well done overall. The rubric questions are generally well-constructed and thorough, needing only minor adjustments.

Section 1: Introduction Rubric Questions Quality
<INSERT INTRODUCTION RUBRIC QUESTIONS HERE>
Rate the Introduction rubric questions from 1-5: ___
Comments: _____

Section 2: Key Literature Rubric Questions Quality
<INSERT KEY LITERATURE RUBRIC QUESTIONS HERE>
Rate the Key Literature rubric questions from 1-5: ___
Comments: _____

Section 3: Methods Rubric Questions Quality
<INSERT METHODS RUBRIC QUESTIONS HERE>
Rate the Methods rubric questions from 1-5: ___
Comments: _____

Section 4: Experimental Design Rubric Questions Quality
<INSERT EXPERIMENTAL DESIGN RUBRIC QUESTIONS HERE>
Rate the Experimental Design rubric questions from 1-5: ___
Comments: _____

Section 5: Resources and Ethics Rubric Questions Quality
<INSERT RESOURCES AND ETHICS RUBRIC QUESTIONS HERE>
Rate Resources rubric questions from 1-5: ___
Rate Ethics rubric questions from 1-5: ___
Comments: _____

Table 13: Rubric Evaluation Prompt

You are given a grading rubric and a new research plan. Your task is to evaluate the new plan against the rubric using a **strict interpretation**: only answer **Yes** if the plan explicitly satisfies the rubric question as written. Note that examples in parentheses (e.g., specific datasets, papers, methods) are for reference only.
Instructions

1. You will be evaluating **ONE SPECIFIC SECTION** of the rubric at a time against the entire research plan.

2. For the given section:
  - Traverse all levels of the section hierarchy. Rubric items may be nested (e.g., subsections → questions).
  - Evaluate each **leaf-level question** (i.e., the final bullet points that are actual rubric questions).
  - Answer **Yes** only if the research plan clearly and explicitly addresses the rubric question.
  - Answer **No** if the rubric question is not addressed, or if the answer is vague or only implied.

  > For example:
  > - General references are not sufficient unless they fully preserve the original intent of rubric item.

3. When rubric questions include examples in parentheses (for example, "e.g., dataset A, dataset B"), these are provided as reference examples to illustrate the type of content being asked about. Do NOT require the research plan to mention these specific examples to answer "Yes".

  > For example:
  > - If the rubric asks "Does the plan use mathematical datasets (e.g., GSM8K, MATH)?", answer "Yes" if the plan uses any appropriate mathematical datasets, not just GSM8K or MATH specifically

4. For each rubric question in the specified section:
  - **Question**: [rubric question]
  - **Answer**: Yes / No
  - **Explanation**: [brief justification]

—

Inputs

// begin of research plan rubric section to evaluate
{{RESEARCH_PLAN_RUBRIC_SECTION}}
// end of research plan rubric section

// begin of full research plan
{{RESEARCH_PLAN}}
// end of full research plan

// section being evaluated
{{SECTION_NAME}}

—

Output Format

Please return the evaluation for the specified section in the following structured JSON format:

```
{
  "section_name": "{{SECTION_NAME}}",
  "evaluation": {
    "subsections": {
      "Subsection Title A": {
        "questions": [
          {
            "question": "Rubric question 1",
            "answer": "Yes" or "No",
            "explanation": "Your explanation here"
          },
          ...
        ]
      },
      ...
    }
  }
}
```

Note: If the section has no subsections and contains questions directly, use this format instead:

```
{
  "section_name": "{{SECTION_NAME}}",
  "evaluation": {
    "questions": [
      {
        "question": "Rubric question 1",
        "answer": "Yes" or "No",
        "explanation": "Your explanation here"
      },
      ...
    ]
  }
}
```

—

Important JSON Formatting Notes:
- When mentioning dollar amounts, use "$" not "\\$"
- Avoid unnecessary escape characters in explanations
- Valid JSON escapes are: \\n \\t \\r \\b \\f \\" \\\\ \\/
- Do not escape regular punctuation like $ or other symbols

Important Notes

- You are evaluating **only the section specified** in the SECTION_NAME field
- Search through the **entire research plan** to find evidence for each rubric question
- Be thorough but focus only on the questions within the specified section
- This evaluation will be combined with evaluations of other sections to form a complete assessment

Table 14: Naïve Baseline Prompt

Naïve Baseline Prompt

RESEARCH IDEA TO ANALYZE:

{{RESEARCH_IDEA}}

Your task is to generate a detailed research plan based on the provided research idea. Below is the template you must follow to create the research plan:

1. Introduction

1.1 Background

1.2 Primary Objectives

1.3 Research Questions

2. Key Literatures

3. Methods

4. Initial Experimental Design

5. Resources, Compliance, and Ethical Considerations

5.1 Resource Requirements

5.2 Ethical and Compliance Considerations

Table 15: Zero-Shot Baseline Prompt

Zero-Shot Baseline Prompt

RESEARCH IDEA TO ANALYZE:

{{RESEARCH_IDEA}}

Your task is to generate a detailed research plan based on the provided research idea. Below is the template you must follow to create the research plan:

{{RESEARCH_PLAN_TEMPLATE}}

INSTRUCTIONS FOR RESEARCH PLAN GENERATION:
1. Analyze the research idea thoroughly from the provided research idea.
2. Generate a complete research plan following the EXACT template structure above.
3. Fill in all sections with relevant, specific content based on the research idea.
4. Draw upon your existing knowledge of the research area to provide context and background.
5. Include resource requirements for conducting the research.

IMPORTANT NOTES:
- Base your plan on the provided research idea and your existing knowledge.

Table 16: One-Shot Baseline Prompt

One-Shot Baseline Prompt

RESEARCH IDEA TO ANALYZE:

{{RESEARCH_IDEA}}

Your task is to generate a detailed research plan based on the provided research idea. Below is the template you must follow to create the research plan:

{{RESEARCH_PLAN_TEMPLATE_WITH_EXAMPLE}}

INSTRUCTIONS FOR RESEARCH PLAN GENERATION:
1. Analyze the research idea thoroughly from the provided research idea.
2. Generate a complete research plan following the EXACT template structure above.
3. Fill in all sections with relevant, specific content based on the research idea.
4. Draw upon your existing knowledge of the research area to provide context and background.
5. Include resource requirements for conducting the research.

IMPORTANT NOTES:
- **The examples provided in the research plan template are for reference only** - replace them with content specific to the given research idea.
- Base your plan on the provided research idea and your existing knowledge.

Table 17: ReAct Agent Prompt

ReAct Agent Prompt

You are a research planning agent that generates comprehensive research plans using iterative reasoning.

RESEARCH IDEA: {{RESEARCH_IDEA}}

Below is the definition of a research plan structure as well as one example.

IMPORTANT NOTE: The example provided is from a specific research plan about scientific literature understanding with LLMs. You should NOT be restricted to the specific settings, methods, datasets, or approaches mentioned in this example. Adapt all content to fit the specific research idea you are working on. Use your research knowledge and the information you gather through searches to create a plan that is most appropriate for the given research idea.

RESEARCH PLAN TEMPLATE AND GUIDELINES:
{{RESEARCH_PLAN_DEFINITION}}

AVAILABLE TOOLS: {{TOOL_NAMES}}

TOOL SPECIFICATIONS:
{{TOOL_SPECIFICATIONS}}

RESOURCE LIMITS:
- You have {{MAX_TOTAL_TOOLS}} total tool calls available across maximum {{MAX_ITERATIONS}} iterations
- {{CURRENT_USAGE_STATS}}

INSTRUCTIONS:
Use the following ReAct format to systematically gather information and create a research plan:

IMPORTANT: Always begin with "Thought:" and explain your reasoning before taking any action.

FORMAT FOR EACH ITERATION:
Thought: [Your reasoning about what information you need]
Action: [MUST be exactly one of: {{TOOL_NAMES}}]
Action Input: [specific query or input for the chosen tool]

CRITICAL RULES:
1. You MUST STOP after "Action Input:" - do NOT generate "Observation:" or any results
2. The system will execute your action and provide the real observation
3. Wait for the actual tool results before continuing
4. Use the EXACT tool names listed above. Available tools are: {{TOOL_NAMES}}
... (repeat this process as needed, up to {{MAX_TOTAL_TOOLS}} total tool calls across maximum {{MAX_ITERATIONS}} iterations)

FINAL STEP (After gathering sufficient information):
When you have collected enough information through your searches and reads, generate your final answer in this EXACT format:

REQUIRED FORMAT:

Thought: I now have sufficient information to create the research plan.
Final Answer: Start your complete research plan here, following the template provided above. The plan should be comprehensive and detailed based on all the information you gathered through your tool calls.>Start your complete research plan here, following the template provided above. The plan should be comprehensive and detailed based on all the information you gathered through your tool calls.

CRITICAL: You MUST include "Final Answer:" exactly as shown above. This is required for the system to recognize that you are submitting your final output.

EXAMPLE OF CORRECT FORMAT:
Thought: I need to find papers about attention mechanisms to understand current transformer architectures.
Action: search_papers
Action Input: attention mechanisms transformer architectures STOP HERE - Wait for system to provide observation>STOP HERE - Wait for system to provide observation

Key guidelines:
1. Use targeted queries to find relevant information
2. Look for recent advances, existing methods, and gaps in the literature
3. Consider both technical approaches and evaluation methods
4. Each tool call should build upon previous findings
5. Choose the most appropriate tool for each information need
6. End with "RESEARCH PLAN COMPLETE" after your final answer
7. NEVER generate "Observation:" - always wait for the system response

READING PAPERS:
When you use the read_paper action, you will receive a comprehensive summary of the paper content instead of the full text.

Begin your research planning process now. Start with a "Thought:" about what information you need to gather first:

REMEMBER: You must use the ReAct format. Do NOT attempt to create the research plan immediately. Start by thinking about what information you need, then use tools to gather that information systematically. STOP after each "Action Input:" and wait for the system's observation.

Table 18: Tool Specifications for ReAct Agent

1. search_papers: Search for academic papers using Bing Custom Search
   - Input: A search query string for academic papers
   - Output: JSON with paper titles, abstracts, arXiv IDs, and publication details
   - Usage example: Action Input: machine learning attention mechanisms

2. read_paper: Read and analyze a specific academic paper by its arXiv ID
   - Input: An arXiv ID (e.g., "2301.12345")
   - Output: JSON with arXiv_id and paper summary
   - Usage example: Action Input: 2301.12345

Table 19: Paper Summarization Prompt

```
summary_prompt = f"""
PAPER TO SUMMARIZE:
Title: {paper_title}
ArXiv ID: {arxiv_id}

Paper Content:
{paper_content}

—

Create a comprehensive summary of this paper. Your summary should include:

1. Main Contributions: Key findings and contributions of this paper
2. Key Related Literature: Important prior works referenced and how this paper builds upon or differs from them
3. Methods and Techniques: Approaches, algorithms, or methodologies used
4. Experimental Design and Results: How experiments were conducted, datasets used, evaluation metrics, and key results obtained

SUMMARY:
"""
```

Table 20: Mean Score On Idea2Plan Bench

Baseline Type	Model	Intro	Lit	Methods	Exp. Design	Res. & Ethics	Average
Naïve	Phi-4	29.0	5.6	20.2	10.9	30.9	19.3
	DeepSeek-V3	39.8	25.2	32.4	24.7	44.5	33.3
	DeepSeek-R1	40.6	27.1	37.0	28.7	50.3	36.7
	GPT-4.1-mini	42.3	21.8	34.7	27.4	47.2	34.7
	GPT-4.1	40.7	22.4	34.2	28.9	48.7	34.9
	O4-mini	50.8	29.1	50.2	39.1	59.7	45.7
	GPT-5-mini	58.5	38.1	65.9	51.6	69.5	56.7
	GPT-5	60.9	47.7	**70.9**	56.7	73.8	62.0
0-shot	Phi-4	28.8	8.2	20.9	10.7	43.1	22.3
	DeepSeek-V3	40.6	23.0	33.4	24.6	52.0	34.7
	DeepSeek-R1	41.3	24.5	37.2	29.7	56.8	37.9
	GPT-4.1-mini	46.8	24.3	39.9	30.9	62.7	40.9
	GPT-4.1	44.0	23.7	34.9	29.2	60.4	38.4
	O4-mini	52.8	27.5	51.4	39.9	68.1	47.9
	GPT-5-mini	61.3	36.6	64.4	52.3	78.9	58.7
	GPT-5	63.2	43.9	69.6	56.9	**81.2**	63.0
1-shot	Phi-4	28.6	9.2	20.1	12.9	45.1	23.2
	DeepSeek-V3	41.8	23.1	33.6	28.5	55.9	36.6
	DeepSeek-R1	40.3	25.3	35.7	30.4	57.1	37.8
	GPT-4.1-mini	48.2	26.6	40.4	33.8	65.3	42.8
	GPT-4.1	45.2	24.0	38.1	31.7	65.2	40.8
	O4-mini	53.9	29.9	53.9	43.5	69.9	50.2
	GPT-5-mini	61.4	37.5	65.8	52.3	80.4	59.5
	GPT-5	**63.2**	44.2	70.2	**57.4**	81.0	**63.2**
ReAct	Phi-4	24.2	11.8	15.5	9.9	31.7	18.6
	DeepSeek-V3	39.9	23.8	33.4	28.5	50.9	35.3
	DeepSeek-R1	38.7	23.1	33.2	27.9	52.0	35.0
	GPT-4.1-mini	50.6	30.6	44.5	35.7	60.3	44.3
	GPT-4.1	50.6	35.6	44.9	40.5	65.6	47.4
	O4-mini	54.2	36.1	54.4	42.6	61.2	49.6
	GPT-5-mini	62.3	44.5	67.3	55.2	76.4	61.1
	GPT-5	61.4	**48.1**	68.4	55.7	75.9	61.9

Table 21: Max Aggregation - Section-wise Scores

Baseline Type	Model	Intro	Lit	Methods	Exp. Design	Res. & Ethics	Average
Naïve	Phi-4	36.4	10.8	25.6	17.2	39.1	23.2
	DeepSeek-V3	46.9	32.2	39.9	31.8	53.6	37.4
	DeepSeek-R1	48.1	35.4	45.8	36.6	59.4	41.2
	GPT-4.1-mini	49.9	30.2	41.9	35.1	57.2	39.0
	GPT-4.1	49.1	30.8	42.3	37.0	58.0	39.4
	O4-mini	59.0	38.5	59.3	46.9	70.1	50.5
	GPT-5-mini	66.3	48.2	73.4	60.4	77.7	61.4
	GPT-5	69.9	57.5	**79.1**	66.0	81.5	66.6
0-shot	Phi-4	36.1	13.2	26.0	16.9	53.1	25.9
	DeepSeek-V3	47.7	30.8	41.5	32.1	61.0	39.2
	DeepSeek-R1	49.1	32.2	45.9	38.3	66.7	42.5
	GPT-4.1-mini	54.3	32.6	47.7	39.4	72.9	45.2
	GPT-4.1	51.9	31.9	43.4	37.6	70.9	43.2
	O4-mini	61.0	35.6	60.1	48.3	77.7	52.2
	GPT-5-mini	69.8	46.0	72.9	61.8	86.9	63.4
	GPT-5	71.5	52.7	77.5	**67.5**	**89.5**	67.9
1-shot	Phi-4	35.2	15.0	25.6	19.0	54.2	27.0
	DeepSeek-V3	49.0	30.6	41.7	36.2	64.8	41.2
	DeepSeek-R1	47.7	33.4	44.2	38.6	67.1	42.2
	GPT-4.1-mini	57.8	36.1	52.9	46.3	76.9	49.1
	GPT-4.1	54.4	33.5	49.7	42.7	77.4	46.7
	O4-mini	63.0	38.9	65.5	55.8	81.6	55.8
	GPT-5-mini	69.8	47.0	74.7	63.2	88.6	64.4
	GPT-5	**71.7**	54.2	79.0	67.0	88.5	**68.1**
ReAct	Phi-4	34.4	19.8	22.7	17.5	45.5	25.0
	DeepSeek-V3	48.2	32.4	42.1	36.6	60.6	40.3
	DeepSeek-R1	46.3	33.1	43.0	35.6	62.2	39.8
	GPT-4.1-mini	58.6	41.4	54.1	45.3	70.5	50.0
	GPT-4.1	59.3	47.0	54.0	52.0	77.4	53.6
	O4-mini	63.5	47.2	65.4	52.5	72.2	55.9
	GPT-5-mini	70.9	55.4	77.1	66.6	84.2	66.2
	GPT-5	69.7	**57.8**	77.1	66.3	84.8	67.3

Table 22: Case Study: Why React Agent Does Not Outperform Baseline: Parametric vs Retrieved Knowledge Conflict

React Agent (GPT-5)	**Baseline (0-shot GPT-5)**
Policy learning and empirical welfare maximization under constraints • Model Selection for Treatment Choice: Penalized Welfare Maximization (arXiv:1609.03167): Extends EWM to select among constrained policy classes with oracle regret guarantees; key for interpretable policy classes and complexity control. • Policy Learning with Observational Data (Athey & Wager; arXiv:1702.02896): Doubly robust policy learning with optimal n^(-1/2) regret for VC classes and accommodation of observational designs; central for estimating welfare-maximizing targeting rules from administrative data.	**Prediction policy and algorithmic decision-making** • Kleinberg, Ludwig, Mullainathan, and Sunstein on prediction policy problems. Connects predictive accuracy to policy outcomes and clarifies when prediction improves decisions. • Obermeyer and Mullainathan on algorithmic bias in healthcare risk prediction. Highlights label/measurement biases that directly inform our decomposition and robustness checks. • Bertsimas and Kallus on prescriptive analytics. Informs policy optimization from predictive models under constraints.

Ground Truth
Domain: Prediction-Policy Problems and Screening Frameworks • Kleinberg, J., Ludwig, J., Mullainathan, S., & Obermeyer, Z. (2015). "Prediction Policy Problems." American Economic Review, 105(5):491–495. – Introduces the paradigm of separately trained predictive risk scoring systems for policy allocation problems, motivating the need to quantitatively compare prediction accuracy against resource constraints. • Perdomo, J. C. (2024). "The Relative Value of Prediction in Algorithmic Decision Making." ICML 2024. – Defines the prediction-access ratio (PAR) in binary treatment settings; serves as the foundation for extending PAR to continuous welfare screening and developing theoretical bounds.

Table 23: Literature Recommendation Prompt

Literature Recommendation Prompt

You are a research literature recommendation system. Based on the provided research idea, introduction, and references, identify exactly 3 most relevant key papers that researchers should read to better understand and build upon this work.

RESEARCH IDEA:
{{RESEARCH_IDEA}}

INTRODUCTION:
{{INTRODUCTION}}

REFERENCES:
{{REFERENCES}}

TASK:
Instructions:
1. Carefully analyze the research idea to understand its core contributions, methods, and objectives
2. Read the introduction to identify the key technical foundations and prior work this research builds upon
3. Select exactly 3 papers that serve as direct foundational works by being:
   - Core methodological predecessors that this work extends or improves
   - Papers introducing the key techniques, algorithms, or frameworks being built upon
   - Essential baseline methods or datasets that this work directly compares against or uses

   Note: Focus on papers that are specifically related to understanding this specific research idea, avoid general background papers.

OUTPUT FORMAT:
Return your response as a JSON array with exactly 3 papers. Each paper should have:
- "title": The exact title of the paper
- "arxiv_link": The arXiv ID if available (format: "XXXX.XXXXX" e.g., "1706.03762"), or null if not on arXiv
- "relevance": A single sentence explaining why this paper is relevant to the current research (max 30 words)

Example format:

```
[
  {
    "title": "Attention Is All You Need",
    "arxiv_link": "1706.03762",
    "relevance": "Introduces the transformer architecture which forms the foundation for the proposed modifications in attention mechanisms."
  },
  {
    "title": "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding",
    "arxiv_link": "1810.04805",
    "relevance": "Demonstrates successful pre-training strategies that are extended in this work for domain-specific applications."
  }
]
```

Return only the JSON array, no additional text or explanation.

Table 24: Section-wise performance comparison with and without curated literature. Values are shown as baseline%/with-related-papers%, with percentage point deltas in parentheses.

Model	Intro	Lit	Methods	Exp Design	Resources & Ethics	Overall
DeepSeek-V3	39.3/41.2 (+1.9)	29.7/31.8 (+2.1)	37.8/39.1 (+1.3)	32.8/33.4 (+0.6)	44.2/45.9 (+1.7)	32.7/35.1 (+2.3)
DeepSeek-R1	39.8/42.3 (+2.5)	31.6/31.8 (+0.2)	42.8/41.2 (-1.6)	35.7/36.1 (+0.3)	49.2/50.9 (+1.7)	37.2/38.5 (+1.3)
GPT-4.1-mini	42.5/43.9 (+1.4)	25.1/30.9 (+5.8)	42.3/40.3 (-2.0)	37.4/37.3 (-0.1)	48.1/45.7 (-2.4)	35.3/37.5 (+2.2)
GPT-4.1	40.5/43.7 (+3.2)	25.3/31.1 (+5.9)	44.1/43.0 (-1.1)	36.6/39.9 (+3.3)	50.1/49.1 (-1.1)	35.2/38.7 (+3.5)
o4-mini	50.6/51.1 (+0.5)	30.0/32.0 (+2.0)	51.7/51.0 (-0.7)	46.0/45.1 (-0.9)	58.1/57.7 (-0.4)	45.1/46.1 (+1.0)
GPT-5-mini	56.1/58.9 (+2.8)	38.6/40.6 (+2.0)	64.7/65.9 (+1.2)	57.7/58.7 (+1.0)	66.9/69.3 (+2.4)	56.0/58.0 (+2.1)
GPT-5	59.4/60.6 (+1.2)	46.0/42.1 (-3.9)	71.7/71.4 (-0.2)	62.4/60.6 (-1.8)	71.9/73.8 (+2.0)	61.3/61.5 (+0.2)

Table 25: SFT model comparison on 1-shot vanilla baseline agents. Scores are averaged across sections and reported in percentages. Rows labeled $\Delta = \text{SFT} - \text{Base}$ show percentage-point differences between SFT and Base models.

Model	Intro	Lit	Method	Exp	Res/Eth	Overall	Overall (w/o Res/Eth)
GPT-4.1-mini (Base)	48.2%	26.5%	40.4%	33.8%	65.3%	42.8%	37.2%
GPT-4.1-mini (SFT)	42.4%	22.9%	41.5%	27.2%	18.9%	30.6%	33.5%
Δ (SFT–Base)	-5.7%	-3.6%	+1.1%	-6.6%	-46.4%	-12.2%	-3.7%
GPT-4.1 (Base)	45.2%	24.0%	38.1%	31.7%	65.1%	40.8%	34.8%
GPT-4.1 (SFT)	44.5%	22.1%	41.7%	27.9%	21.9%	31.6%	34.1%
Δ (SFT–Base)	-0.7%	-1.9%	+3.6%	-3.8%	-43.2%	-9.2%	-0.7%

Table 26: Evaluation rubric for Jailbreak-Tax paper (Nikolic et al., 2025), Part 1.

Evaluation Rubric Part 1 (JSON Format)

```
{
  "sections": {
    "Introduction": {
      "subsections": {
        "Background": {
          "questions": [
            {
              "question": "Does the plan state that adversarial jailbreak attacks can bypass LLM safety guardrails or alignment protections?"
            },
            {
              "question": "Does the plan note that existing evaluations focus on bypass success and neglect post-jailbreak capability retention?"
            },
            {
              "question": "Does the plan explain that assessing the utility of harmful outputs requires domain expertise?"
            },
            {
              "question": "Does the plan highlight the lack of an unaligned baseline as a barrier to quantifying performance degradation?"
            },
            {
              "question": "Does the plan identify the need for a more rigorous evaluation framework for jailbreak attacks?"
            }
          ]
        },
        "Primary Objectives": {
          "questions": [
            {
              "question": "Does the plan propose designing an evaluation framework that measures both jailbreak success rate and the utility of elicited outputs?"
            },
            {
              "question": "Does the plan include constructing benchmark suites with tasks that have objectively verifiable ground-truth answers?"
            },
            {
              "question": "Does the plan include applying and comparing multiple representative jailbreak attacks across these benchmarks?"
            },
            {
              "question": "Does the plan include comparing jailbreak performance across different alignment methods?"
            },
            {
              "question": "Does the plan include quantifying the relative performance drop (jailbreak tax) compared to an unaligned baseline?"
            },
            {
              "question": "Does the plan aim to release benchmark suites and evaluation code to the community?"
            }
          ]
        },
        "Research Questions": {
          "questions": [
            {
              "question": "Does the plan ask whether different jailbreak attacks incur a measurable performance drop (jailbreak tax) across tasks?"
            },
            {
              "question": "Does the plan ask whether the magnitude of the performance drop correlates with the jailbreak success rate?"
            },
            {
              "question": "Does the plan ask whether model size influences the severity of the jailbreak tax?"
            },
            {
              "question": "Does the plan ask whether the jailbreak tax is consistent across alignment methods?"
            },
            {
              "question": "Does the plan ask whether task difficulty affects the magnitude of the jailbreak tax?"
            }
          ]
        }
      }
    }
  }
}
```

Table 27: Evaluation rubric for Jailbreak-Tax paper (Nikolic et al., 2025), Part 2.

Evaluation Rubric Part 2 (JSON Format)

```
{
  "sections": {
    "Key Literatures": {
      "subsections": {
        "Jailbreak Attack Methodologies": {
          "questions": [
            {
              "question": "Does the plan cite the paper (Jailbroken: How does LLM safety training fail?) or similar work on manual prompt-engineering jailbreaking techniques?"
            },
            {
              "question": "Does the plan cite the paper (Universal and transferable adversarial attacks on aligned language models) or similar work on optimization-based adversarial suffix attacks?"
            },
            {
              "question": "Does the plan cite the paper (AutoDAN: Generating stealthy jailbreak prompts on aligned large language models) or similar work on genetic-algorithm-based prompt generation?"
            },
            {
              "question": "Does the plan cite the paper (Jailbreaking black box large language models in twenty queries) or similar work on iterative LLM-based prompt rewriting attacks?"
            },
            {
              "question": "Does the plan cite the paper (Tree of attacks: Jailbreaking black-box LLMs automatically) or similar work on tree-of-thought or hierarchical attack expansions?"
            },
            {
              "question": "Does the plan cite the paper (Many-shot jailbreaking) or similar work on in-context demonstration-based jailbreak attacks?"
            },
            {
              "question": "Does the plan cite the paper (Multilingual jailbreak challenges in large language models) or similar work on translation-based or multilingual jailbreak techniques?"
            }
          ]
        },
        "Benchmark Datasets for Objective Utility": {
          "questions": [
            {
              "question": "Does the plan cite the paper (The WMDP benchmark: Measuring and reducing malicious use with unlearning) or similar work on bio-security multiple-choice benchmarks?"
            },
            {
              "question": "Does the plan cite the paper (Training verifiers to solve math word problems) or similar work on grade-school math reasoning benchmarks?"
            },
            {
              "question": "Does the plan cite the paper (Measuring massive multitask language understanding) or similar work on competition-level math or multitask reasoning benchmarks (e.g., MATH, MMLU)?"
            }
          ]
        },
        "Alignment Tax and Prior Evaluations": {
          "questions": [
            {
              "question": "Does the plan cite the paper (Current work in AI alignment) or similar work defining the concept of an alignment tax?"
            },
            {
              "question": "Does the plan cite the paper (A StrongReject for empty jailbreaks) or similar work on performance degradation of jailbreak attacks?"
            },
            {
              "question": "Does the plan cite the paper (AgentHarm: A benchmark for measuring harmfulness of LLM agents) or similar work contrasting objective benchmarks with LLM-based evaluation of harmfulness?"
            }
          ]
        }
      }
    }
  }
}
```

Table 28: Evaluation rubric for Jailbreak-Tax paper (Nikolic et al., 2025), Part 3.

Evaluation Rubric Part 3 (JSON Format)

```
{
  "sections": {
    "Methods": {
      "subsections": {
        "Pseudo-Alignment Techniques": {
          "questions": [
            {
              "question": "Does the plan describe a system-prompt alignment method that simulates model refusals via instructional constraints?"
            },
            {
              "question": "Does the plan describe a supervised fine-tuning approach on (prompt, refusal) pairs to induce model alignment?"
            },
            {
              "question": "Does the plan include a data-driven or task rewording-based pseudo-alignment technique (e.g., transforming benign tasks to harmful contexts)?"
            }
          ]
        },
        "Jailbreak Attacks": {
          "questions": [
            {
              "question": "Does the plan include a prompt-based jailbreak attack that appends or modifies instructions to bypass alignment?"
            },
            {
              "question": "Does the plan include a fine-tuning-based jailbreaking approach to reverse the model's refusal alignment?"
            },
            {
              "question": "Does the plan include demonstration-based (many-shot) jailbreak attacks using in-context examples?"
            },
            {
              "question": "Does the plan include optimization-based adversarial attacks (e.g., greedy coordinate descent) to craft adversarial suffixes?"
            },
            {
              "question": "Does the plan include genetic algorithm or evolutionary strategies for generating stealthy jailbreak prompts?"
            },
            {
              "question": "Does the plan include translation-based or multilingual jailbreak attacks?"
            },
            {
              "question": "Does the plan include iterative prompt rewriting attacks that use LLM-based rewriting (with or without tree-of-thought reasoning)?"
            }
          ]
        },
        "Evaluation Metrics": {
          "questions": [
            {
              "question": "Does the plan define a metric for jailbreak success rate that measures the proportion of non-refusal outputs?"
            },
            {
              "question": "Does the plan define a metric for post-jailbreak utility, such as conditional task accuracy after a successful jailbreak?"
            },
            {
              "question": "Does the plan define a baseline utility metric for measuring the unaligned model's task performance?"
            },
            {
              "question": "Does the plan define a metric for quantifying the relative performance drop (jailbreak tax) between baseline and jailbroken outputs?"
            }
          ]
        }
      }
    }
  }
}
```

Table 29: Evaluation rubric for Jailbreak-Tax paper (Nikolic et al., 2025), Part 4.

Evaluation Rubric Part 4 (JSON Format)

```
{
  "sections": {
    "Initial Experimental Design": {
      "questions": [
        {
          "question": "Does the plan propose experiments that evaluate both jailbreak success rate and post-jailbreak utility on objective benchmarks?"
        },
        {
          "question": "Does the plan include experiments using tasks with verifiable ground-truth answers (e.g., biology multiple-choice, grade-school math, competition-level math)?"
        },
        {
          "question": "Does the plan include experiments comparing results across multiple model sizes (e.g., small, medium, large LLMs)?"
        },
        {
          "question": "Does the plan include experiments comparing different alignment methods (e.g., system-prompt, supervised fine-tuning, task rewording)?"
        },
        {
          "question": "Does the plan include experiments applying multiple jailbreak attacks across the selected benchmarks?"
        },
        {
          "question": "Does the plan include analysis of the relationship between task difficulty and the magnitude of the jailbreak tax?"
        },
        {
          "question": "Does the plan include control experiments evaluating aligned models on neutral or unrelated tasks (e.g., subsets of MMLU or other benchmarks) to verify unintended degradation?"
        }
      ]
    },
    "Resources, Compliance, and Ethical Considerations": {
      "subsections": {
        "Resource Requirements": {
          "questions": [
            {
              "question": "Does the plan list required model resources, such as open-weight and closed-weight LLMs?"
            },
            {
              "question": "Does the plan acknowledge the need for computational infrastructure for fine-tuning and evaluation (e.g., GPU hardware)?"
            },
            {
              "question": "Does the plan acknowledge the need for API access for data transformation or evaluation steps?"
            }
          ]
        },
        "Ethical and Compliance Considerations": {
          "questions": [
            {
              "question": "Does the plan address the potential for adversaries to misuse the evaluation framework?"
            },
            {
              "question": "Does the plan mention verifying data licenses and usage rights for the benchmarks used?"
            },
            {
              "question": "Does the plan address the exclusion of sensitive or personally identifiable information from benchmark data?"
            },
            {
              "question": "Does the plan discuss risk-benefit trade-offs or security-research norms regarding transparency and threat exposure?"
            }
          ]
        }
      }
    }
  }
}
```

Table 30: Full Research Plan generated by GPT-5, Part 1.

1. Introduction

1.1 Background
Large language models (LLMs) are increasingly safety-aligned with guardrails that refuse certain categories of content, but attackers routinely develop "jailbreak" prompts to bypass these safeguards. While jailbreak success is typically measured by refusal circumvention, it is unclear whether the resulting outputs remain useful and accurate, particularly when the underlying tasks are benign and easily verifiable. This matters now because models are being deployed in high-stakes workflows that rely on both safety and reliability; if jailbreaks impose a hidden "utility tax" by degrading task performance, practitioners and policymakers need to account for it when evaluating attack severity, defense effectiveness, and alignment strategies. Moreover, existing evaluations confound jailbreak efficacy with content harmfulness and lack ground truth, making it difficult to quantify performance degradation attributable to jailbreaks alone. A systematic, ground-truth-based assessment of jailbreak utility is therefore overdue.

1.2 Primary Objectives
- Construct BenignRefuse-Bench: a suite of ground-truth, short-answer tasks (e.g., math, unit conversions, basic biology, factual QA) designed for automatic verification and tagged so aligned models are instructed to refuse them.
- Produce aligned model variants that reliably refuse these benign tasks via controllable guardrail mechanisms (e.g., system prompts/wrappers, lightweight SFT/LoRA), without materially reducing general capabilities.
- Implement a representative set of jailbreak strategies (e.g., role-play/DAN, translation/transliteration, adversarial suffixes, multi-turn coercion, long-context dilution) and standardize attack parameters.
- Define and validate a new metric family quantifying the "jailbreak utility degradation" (JUD) and related components (attack success rate, conditional answer correctness, overall utility, overhead costs).
- Benchmark multiple models and alignment methods under these attacks, report statistically robust comparisons, and release code, data, and baselines to enable reproducible evaluation.
- Conduct ablations to isolate factors driving utility loss (attack type, model family/size, alignment mechanism, decoding settings), and provide actionable guidance for practitioners.

1.3 Research Questions
- Do jailbreaks that bypass guardrails on benign tasks preserve the model's baseline task performance, or do they degrade utility?
- How does performance degradation vary by jailbreak type, model family/size, and alignment mechanism (prompt-based vs SFT vs RLHF-adjacent)?
- What is the empirical trade-off between jailbreak bypass success and answer correctness, and can it be quantified in a single, interpretable metric?
- How robust are utility outcomes to decoding parameters (temperature), multi-turn coercion, and context length?
- Can we design alignment schemes that maintain high refusal rates while retaining utility under attack (i.e., minimize the jailbreak utility tax)?
- Are utility effects consistent across task domains with different verification regimes (numeric, unit-normalized, short string exact-match)?

2. Key Literatures
Domains:
- LLM alignment and refusal calibration: informs how we create controlled refusal without destroying capability.
- Jailbreak and prompt-injection attacks: provides attack taxonomies and representative techniques to evaluate.
- Adversarial robustness in NLP: offers methods and metrics for evaluating performance under distribution shift.
- LLM evaluation and truthfulness: guides construction of verifiable benchmarks and automatic scoring.
- Safety-utility trade-offs: frames the concept of a "jailbreak utility tax" and its implications.

Representative works and relevance:
- Ouyang et al., InstructGPT (RLHF): foundational for alignment via preferences; we adapt alignment levers and consider their side effects on capabilities.
- Bai et al., Training Helpful and Harmless Assistants; Constitutional AI: methods for refusal policies and their calibration; informs our alignment baselines and defense interpretations.
- Perez et al., Red-Teaming Language Models with Language Models: motivates systematic attack generation and coverage; informs our attack suite design.
- Greshake et al., Prompt Injection attacks against LLMs: characterizes injection channels and transferability; we include translation/indirection attacks accordingly.
- Wallace et al., Universal Adversarial Triggers for NLP: inspires adversarial suffix construction and evaluation of universal vs input-specific attacks.
- Zou et al., adversarial suffix (GCG) methods for aligned LLMs: provides a white-box/gradient-guided approach for strong, compact jailbreak suffixes; we adapt compute-feasible variants for open models.
- Lin et al., TruthfulQA; Hendrycks et al., MMLU; Cobbe et al., GSM8K: benchmarks with objective scoring; we draw task design principles to ensure automatic verification and minimal ambiguity.
- Work on "jailbreak tax" and refusal over-refusal analyses (various 2023–2024 studies): motivates the central hypothesis that bypassing guardrails can degrade answer quality; we formalize and quantify this effect.

3. Methods

Dataset construction (BenignRefuse-Bench):
- Domains: arithmetic and word-problem math (GSM8K-style but short), unit conversions and basic numeracy, high-school biology facts, chemistry stoichiometry with single-number outputs, geography (capitals, population orderings with normalized ranges), and basic physics formula applications with numeric answers.
- Format: short-answer, single number or constrained string where exact or normalized exact-match is viable; include canonical answers and allowed variants; enforce deterministic scoring.
- Splits: train/dev for attack tuning and refusal calibration; held-out test for final reporting.
- Metadata: each item tagged with a category to trigger refusal in aligned models via controllable guardrails.

Alignment strategies to induce refusal:
- Prompt-based guardrails: system prompt/wrapper instructing the assistant to refuse any query with a visible tag (e.g., [CATEGORY=MATH]) and provide a refusal template. Ensures high refusal rate without retraining.
- Lightweight SFT/LoRA: fine-tune an instruction-tuned base model on refusal demonstrations for our categories to simulate learned guardrails. Keep fine-tune small and targeted to avoid broad capability loss.
- Optional preference tuning (if feasible): a small preference dataset to rank refusals higher for tagged categories, approximating RLHF behavior in a compute-light manner.
- Capability sanity checks: evaluate on out-of-category items to verify minimal degradation vs base.

Jailbreak attack suite:
- Role-play/DAN-style: "ignore prior instructions," role-play as a helpful assistant, and restate objectives.
- Translation/transliteration: present the task in another language or script, then request direct answer.
- Adversarial suffixes: gradient-guided or search-based short suffix appended to user prompts for open models; black-box heuristic suffixes for API models.
- Long-context dilution: prepend high-entropy benign text to push guardrail instructions out of focus.
- Multi-turn coercion: first elicit agreement to answer, then present the tagged question.
- Standardization: fixed token budgets, temperature settings, attempt counts, and stopping criteria.

Metrics (new and derived):
- Base Accuracy (A_base): accuracy of the unguarded model on the benchmark.
- Refusal Rate (RR): fraction refused by the aligned model without attack.
- Attack Success Rate (ASR): fraction of items where the aligned model produces a non-refusal answer under attack.
- Conditional Answer Correctness (CAC): accuracy among non-refusal responses under attack.
- Overall Utility (OU): ASR × CAC (probability of a correct answer across all items under attack).
- Jailbreak Utility Degradation (JUD): (A_base OU) / A_base, with 0 meaning no degradation and 1 meaning total loss of utility relative to base.
- Overhead metrics: token overhead (input/output vs base), latency overhead, and verbosity inflation; report per attack and per model.
- Reliability: bootstrap confidence intervals and paired tests across items for A_base, OU, and JUD.

Table 31: Full Research Plan generated by GPT-5, Part 2.

Refusal and correctness detection:
- Refusal: rule-based detection using a curated lexicon ("I can't," "I'm unable," policy references), with manual verification on a stratified subset to estimate precision/recall.
- Correctness: exact-match after normalization (numeric rounding, unit normalization, case-insensitive string normalization, simple equivalence classes). Ambiguous items excluded or adjudicated during dataset construction.

Controls and confound mitigation:
- Trigger-tag guardrails ensure refusal is activated by known markers, isolating jailbreak effects from domain ambiguity.
- Capability holdout tasks (untagged) confirm alignment doesn't broadly reduce performance.
- Consistent decoding settings across base vs attacked runs; sensitivity analysis for temperature.
- Limit attack prompt length to disentangle context compression effects; report token overhead.

Models:
- Open models: at least two sizes per family (e.g., 7–8B and 13–14B from popular families) to study size effects. Inference performed locally for full logging and reproducibility.
- Optional closed APIs: where permitted, using prompt-based guardrails only; report separately due to limited control.

4. Initial Experimental Design

Phase 1: Benchmark assembly
- Curate ∼5,000 items across 6 categories; enforce single-answer verifiability; create dev/test splits (e.g., 1k dev, 4k test).
- Build normalization and scoring scripts; pilot scoring on multiple paraphrases.

Phase 2: Alignment setup and validation
- Prompt-based guardrail: design standardized refusal templates and category triggers; target RR 95% on tagged items.
- LoRA-based refusal model: ∼10k refusal demonstrations generated from the benchmark schema and paraphrased prompts; train on one open model (8–13B). Validate RR and capability retention on untagged items.

Phase 3: Attack implementation and calibration
- Implement five attacks: role-play/DAN, translation/transliteration, adversarial suffix (white-box for open models), long-context dilution, and multi-turn coercion.
- Calibrate each attack on the dev split to achieve comparable ASR where possible under fixed token and attempt budgets.

Phase 4: Main evaluation
- For each model and alignment method:
  - Measure A_base on the test split (untagged).
  - Measure RR on the aligned, un-attacked test split (tagged).
  - For each attack, run 1–2 attempts per item under fixed temperature (e.g., 0.2 and 0.7 configurations), collect outputs, and compute ASR, CAC, OU, JUD, and overhead metrics.
- Statistics: bootstrap 95% CIs for OU and JUD; paired tests between attacks; report effect sizes and significance with multiple-comparison control.

Phase 5: Ablations and robustness
- Vary decoding (temperature, top-p) to test sensitivity of ASR and CAC.
- Compare prompt-based vs LoRA-based guardrails to assess alignment mechanism effects.
- Evaluate size scaling (small vs medium models) for JUD trends.
- Swap categories (e.g., remove math, add basic physics) to test domain robustness.
- Optional: chain-of-thought vs concise answers to see if jailbreaks disproportionately harm multi-step reasoning.

Deliverables
- Public release of BenignRefuse-Bench (data + scoring), baseline alignment wrappers, attack scripts, metric implementations, and a reproducibility checklist.
- Report with leaderboards for OU and JUD by attack/model, plus analysis of overhead costs.

5. Resources, Compliance, and Ethical Considerations

5.1 Resource Requirements
Data curation:
- Benchmark assembly (5,000 items) using existing public QA sources and scripts; manual vetting for clarity and unambiguous ground truth.
- Estimated human time: 40–60 hours (data selection, normalization, pilot scoring), plus 10–15 hours for spot-check annotations of refusal detection.
- If external annotators are used, budget 500–800 item-minutes for QA spot checks (approx. 10–15 labor hours).

Compute for alignment and attacks:
- Fine-tuning (LoRA on 8–13B model): 10k refusal examples, 1–2 epochs, sequence length ∼512; estimated 30–60 GPU-hours on A100 40GB equivalents (or ∼120–240 GPU-hours on smaller GPUs).
- Adversarial suffix search (white-box on open model): dev set–driven universal suffix or small per-category suffix; estimated 10–30 GPU-hours depending on optimization steps and batch size.
- Inference: Dataset: 5,000 test items × (base + aligned-no-attack + 5 attacks) = 35k runs per model per temperature setting. Average tokens per run: ∼250 input, ∼120 output base; attacks add ∼100–250 input tokens and ∼50 output tokens. Tokens per model per temperature: ∼35k × (350–620) ≈ 12–22 million tokens. For two temperatures and three open models: ∼72–132 million tokens total.
- Storage and logging: ∼50–100 GB for prompts, outputs, and metadata.

API usage (optional):
- If including 1–2 closed models with prompt-based guardrails only, expect similar token volumes; ensure cost caps and rate-limit handling.

Engineering:
- Attack orchestration, scoring pipeline, metrics, and CI: ∼2–3 person-weeks.
- Reproducibility artifacts and documentation: ∼1 person-week.

5.2 Ethical and Compliance Considerations
- Safety scope: All tasks are benign with ground-truth verification; no harmful or dual-use content is included. Jailbreak techniques are evaluated strictly in this benign context to measure utility effects, not to enable misuse.
- Responsible disclosure: Release attack code in a restricted form that limits adaptation to harmful content (e.g., bound to category tags present only in the benchmark). Provide a usage policy and researcher agreement; exclude especially potent per-item adversarial suffixes from public artifacts.
- Data privacy and licensing: Use public datasets with permissible licenses; filter and remove any PII; document sources and licenses in the repository.
- Model license compliance: Respect usage terms for open models and APIs; clearly separate evaluations of models with different licensing constraints.
- Human subjects: Minimal risk; if annotators are used, provide informed consent, fair compensation, and the right to withdraw. IRB review is unlikely to be required but should be sought if annotation protocols expand.
- Dual-use risk management: Clearly communicate that findings are intended to quantify the utility cost of jailbreaks to support better defenses and evaluation standards. Avoid publishing content that materially lowers the barrier to unsafe jailbreak replication beyond benign settings.