Predicting Empirical AI Research Outcomes with Language Models

Jiaxin Wen[1], **Chenglei Si**[2], **Chen Yueh-Han**[3], **He He**[3], **Shi Feng**[4]

[1]UC Berkeley [2]Stanford [3]New York University [4]George Washington University

Abstract

Many promising-looking ideas in AI research fail to deliver, but their validation takes substantial human labor and compute. Predicting an idea's chance of success is thus crucial for accelerating empirical AI research, a skill that even expert researchers can only acquire through substantial experience. We build the first benchmark for this task and compare LMs with human experts. Concretely, given two research ideas (e.g., two jailbreaking methods), we aim to predict which will perform better on a set of benchmarks. We scrape ideas and experimental results from conference papers, yielding 1,444 human-verified idea pairs *published after our base model's cut-off date* for testing, and 6,000 pairs for training. We then develop a system that combines a fine-tuned GPT-4.1 with a paper retrieval agent, and we recruit 25 human experts to compare with. In the NLP domain, our system beats human experts by a large margin (64.4% v.s. 48.9%). On the full test set, our system achieves 77% accuracy, while off-the-shelf frontier LMs like o3 perform no better than random guessing, even with the same retrieval augmentation. We verify that our system does not exploit superficial features like idea complexity through extensive human-written and LM-designed robustness tests. Finally, we evaluate our system on unpublished novel ideas, including ideas generated by an AI ideation agent. Our system achieves 63.6% accuracy, demonstrating its potential as a reward model for improving idea generation models. Altogether, our results outline a promising new direction for LMs to accelerate empirical AI research.
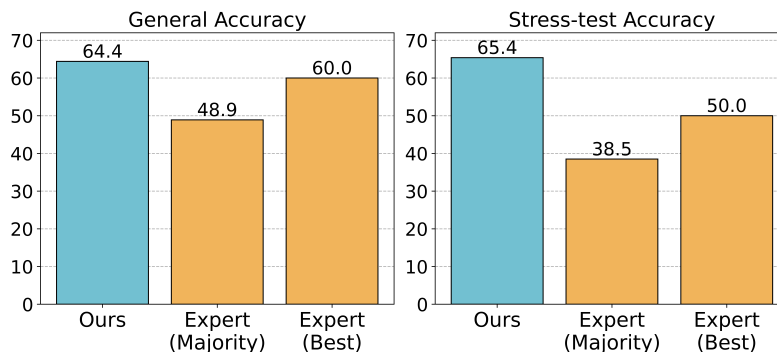
Figure 1: **Our system is more accurate than human NLP experts**. *Majority* aggregates predictions from all annotators, while *Best* keeps only the best-performing annotator per research topic. For the stress test, we select a subset of idea pairs where the mathematically complex one is actually ineffective. While humans often get misled by this feature, our model does not.

1   Introduction

Many promising-looking AI research ideas turn out to be ineffective when executed. Unfortunately, the only way to find out is to actually implement them. The failed ideas can add up to a significant cost in both human labor and computational resources. Better prioritization—implementing the more promising ideas first—can thus significantly improve research efficiency. But doing so requires predicting the outcomes of experiments without actually running them, a seemingly impossible task.

We hypothesize that language models (LMs) can do this task better than human experts. Humans develop such research intuition through experience, but LMs can acquire it more efficiently by consuming countless research papers, analyzing experimental results, and potentially discovering subtle patterns that are difficult for humans to identify.

Concretely, given the description of a pair of research ideas, our goal is to predict which one works better on a set of benchmarks. For example, given two jailbreaking methods, we aim to predict which one achieves higher attack success rates. This is an objective task: we can obtain the groundtruth by actually implementing both ideas. But making a prediction without implementing the ideas is difficult, and any non-trivial accuracy could be valuable as it informs resource allocation, prioritization of experiments, and iterative refinement of ideas. Crucially, this task provides more actionable information than evaluating subjective aspects of ideas such as novelty or excitement [14, 9, 4], which existing work focuses on.

We construct a benchmark to facilitate the study of this task by scraping both ideas and results from existing conference papers. Each example consists of a research goal specified by a set of benchmarks, two competing ideas, and a binary outcome label indicating which idea performs better across these benchmarks. After four rounds of human verification, we obtain 1,444 verified test examples. To avoid data contamination, we ensure that each test example contains at least one idea published after July 1st, 2024, the knowledge cut-off date of our base model GPT-4.1.

We then develop a system that combines a specialized GPT-4.1 fine-tuned on 6,000 historical idea pairs from the training set and a paper retrieval agent. This system achieves a promising 77% accuracy on the test set. Off-the-shelf frontier models (e.g., o3, Claude 3.5 Sonnet)—even when augmented by the same paper retrieval agent—perform no better than random guessing, demonstrating the importance of proper capability elicitation. We then identify a challenging test subset consisting of 45 NLP idea pairs, and recruit 25 NLP experts to establish a human baseline. Specifically, each human prediction is made by an ensemble of 5 experts who collectively spent over 45 minutes. Our system beats this strong expert baseline by a large margin (64.4% v.s. 48.9%).

To test the system's generalizability, we design stress tests to measure its sensitivity to superficial features like idea recency and complexity. On three human-designed stress tests and hundreds tests proposed by LMs, our system demonstrates robust behavior.

Lastly, we test our system on brand new, unpublished research projects from a recent study on using AIs to generate ideas [14]. This is a set of 35 ideas with experiments implemented by NLP researchers (recruited by the study's authors). These projects have never been released publicly anywhere on the internet, thus avoiding any possible contamination. Our system achieves an accuracy of 63.6%, indicating its generalizability and its potential to improve existing automated research systems [14, 9] as an idea ranker or a reward model.

2   Benchmark

We define our task as predicting the outcomes of two competing research ideas for a given research goal. We focus on pairwise evaluation, where the binary outcome label is aggregated over multiple benchmarks to avoid the ambiguity and noise when evaluating individual ideas [14, 9, 4]. Each example involves the following components (Figure 2):

- **Idea pair**, each defined by a detailed description following a standard format.
- **Empirical research goal**, defined by a set of benchmarks, each with a quantitative metric.
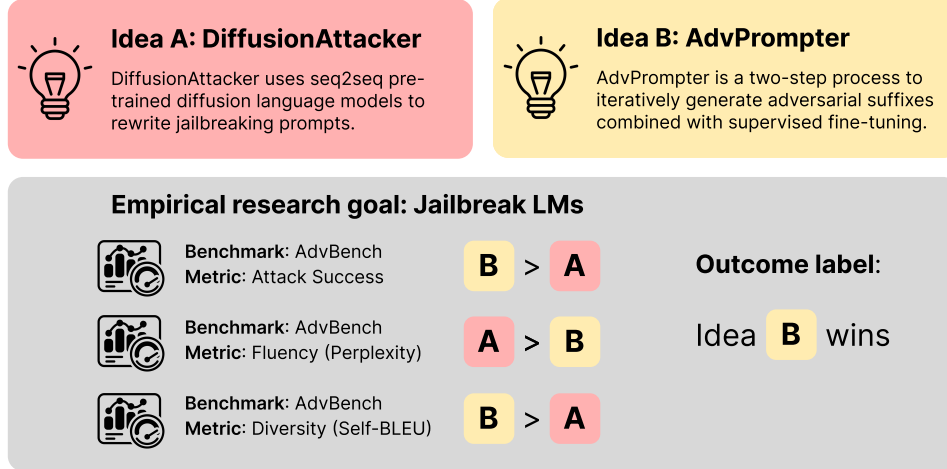- **Binary outcome label**, indicating which idea wins on more benchmarks.

Figure 2: Comparing two ideas for jailbreaking methods on three benchmarks. Each idea is defined by a detailed summary. The goal is to predict the outcome, whose label is determined by actually evaluating the ideas on these benchmarks and seeing who wins more.

2.1 Benchmark Construction

We develop an automatic pipeline to extract ideas and their corresponding empirical results from existing papers. The pipeline operates in four steps using Claude 3.5 Sonnet as the main LM.

Step 1: Paper Collection. We start by collecting papers from top AI conferences in various domains. Table 1 outlines our data sources. To collect test examples published after frontier LMs' knowledge cut-off, we mainly collect papers published in the past two years. After scraping, we use LMs to filter out non-empirical papers, such as those focus on "rethinking", and "understanding" topics.

Step 2: Data Extraction. For each paper PDF, we prompt LMs to extract the research goal, idea names and their outcomes, mainly from the result tables. Next, we generate summaries for each idea based on the PDF. While the main idea is explained in detail, baseline ideas are often described only briefly and potentially in a biased way. To address this, we extract the reference for each baseline idea and download the corresponding paper PDF. If we fail to find references (e.g., the reference is missing or the extracted reference is invalid), we discard these ideas. To avoid information leakage, we explicitly instruct LMs not to mention any actual empirical results in summaries.

Step 3: Idea Pairing. We then group ideas into comparison pairs. Pairs are created only within the same paper, since cross-paper comparisons might suffer from confounding factors (e.g., differences in hyperparameters or data preprocessing). For example, comparing the results of two parameter-efficient tuning methods reported in different papers (e.g., LoRA and prompt tuning) could be invalid if they use different numbers of trainable parameters.

Step 4: Label Aggregation. To determine the final comparison label for each idea pair, we do majority voting over all benchmarks. For example, if Idea A outperforms Idea B on 4 out of 5 benchmarks, we label the pair as "Idea A is better". This aggregation strategy mitigates label noise in individual benchmark comparisons. To further improve label quality, we remove idea pairs that result in a tie or are evaluated on fewer than three benchmarks.

Train/Test Splitting. We then split our collected data into train and test sets using our main base model GPT-4.1's cut-off date, July 1st, 2024. In particualr, we use the date that the paper first appeared on arxiv (which we extract from its arxiv ID), not the date that it is published at the conference. This ensures no leakage from the fine-tuning process. So the model must predict ideas proposed in the "future work". Data statistics are shown in Table 2.

2.2 Human Verification of Extracted Examples

Since our benchmark is automatically constructed by parsing paper PDFs with LMs, parsing errors are inevitable. To ensure the quality and reliability of our test set, we recruit human annotators to

Table 1: Data source.	
Domain	**Conferences**
NLP	ACL, EMNLP, NAACL, COLM
ML	NeurIPS, ICLR
CV	CVPR, ECCV, ACMMM
Robotics	CoRL

Table 2: Data statistics.		
Split	**Train**	**Test**
Size	6,000	1,444
# Benchmark	3.2	3.4
After Cut-off	No	Yes
Human-Verified	No	Yes

carefully go through our test set, rewriting or removing problematic examples. Note that this set of human annotators are tasked only to verify the extracted examples; the annotators for our expert baseline have much higher expertise in their respctive domains.

Recruiting Human Annotators. We select 16 college students majoring in Computer Science or Electronic Engineering out of 30 pre-screened candidates. All of them are experienced in reading AI papers, and 25% of them have published papers at AI conferences. We train annotators to do the verification task with 50 warmup examples and verify their skills through their evaluation error rate.

Annotation Process. Annotators verify and correct each example by consulting the original paper PDF. Table 7 in the Appendix showcases typical errors found by our annotators, sorted by frequency. The most common error is "Incorrect Win Condition", e.g., mistakenly marking the perplexity metric as higher is better. Additionally, the win conditions for certain metrics are context-dependent. For example, Attack Success Rate (ASR) should be maximized in adversarial attack papers but minimized in defense papers. Annotators fix these errors by rewriting examples or removing them when the results are fabricated or incomparable due to unfair experiment settings.

To incentivize our annotators to provide high-quality annotations, we design the following bonus schemes. Each annotation initially earns $5. However, if an annotation is found incorrect during cross-validation by another annotator, the original annotator gets a penalty of $3, while the annotator who catches the error gets an additional $3 reward. We conduct four rounds of cross-validation. In each round, we randomly sample 400 examples and send them to review by new annotators. From the first to the last round, the rate of identified mislabels decreased from 11% to 2.5%.

3 Building a System for Research Outcome Prediction

3.1 Retrieval

When predicting outcomes of new ideas, human researchers often draw inspiration from existing literature. While the exact same ideas do not exist, prior studies can offer indirect insights, transferable knowledge, or analysis of similar sub-components. Therefore, we develop an agentic retrieval module that iteratively performs the following four steps: query generation, paper retrieval, paper summarization, relevance checking and filtering. The final retrieval results will be used as parts of inputs for prompting or fine-tuning.

Step 1: Query Generation. At iteration t, given a research goal, two research ideas, and previous queries and retrieval results, our system first determines whether sufficient information has been collected, thus allowing for early exit. If further research is required, the system prompts an LM to generate a new query distinct from previous queries. Importantly, since at least one idea in the given comparison pair is entirely novel, the system would not try to directly search for identical idea comparisons. Instead, the query generation employs two strategies: 1) retrieving ideas that offer indirect or transferable insights, and 2) decomposing the novel idea into sub-components and retrieving related literature for these sub-components.

Step 2: Paper Retrieval. We use `https://exa.ai` as the paper search engine. Unlike conventional keyword search, they support neural search that can directly take natural language queries as inputs, e.g., "effectiveness of confidence calibration and iterative refinement in question answering". EXA also supports automatic query optimization [12] for better retrieval performance. For each query, we retrieve

Table 3: Impacts of paper summarization methods: reuse abstracts or summarize the full paper. Evaluated on 600 test examples.

Method	**Accuracy**
GPT-4.1	42.0
w/ RAG (abstract-only)	38.8
w/ RAG (whole paper)	53.0

the top-15 relevant papers from arxiv.org. To prevent
information leakage from retrieval, we only retrieve papers published before July 1st, 2024.

Step 3: Paper Summarization. Prior work mainly resues paper abstracts as summaries [14, 9]. However, abstracts are often too brief to cover rich details (e.g., ablation studies or comparison with specific baselines). Therefore, we download each paper's PDF, and prompt LMs to summarize the paper with respect to the current query. As shown in Table 3, this substantially boosts the accuracy from 38.8% to 53.0%. We find that using abstracts alone biases the model towards favoring old ideas, while summarizing the whole paper alleviates such biases.

Step 4: Relevance Checking and Filtering. The initially retrieved 15 papers per query ensure coverage, but may also introduce irrelevant papers that may negatively mislead LMs' predictions. To alleviate this, we prompt LMs to judge the relevance of each paper summary (binary classification: relevant or irrelevant), and subsequently filter out those irrelevant ones.

3.2 Fine-tuning

Next, we fine-tune LMs to reason over the research goal, two research ideas, and all retrieval results, to make the final prediction. A straightforward setting is fine-tuning LMs with golden outcome labels.

In addition, we also explore fine-tuning LMs to generate chain-of-thought (CoT) reasoning [15] before the final prediction. High-quality CoTs are not available for this task. Thus, following recent practices [17], we use the LM to augment the training data with its own generated CoTs. Specifically, we sample multiple CoTs for each query, filter out CoTs that lead to incorrect predictions, and try several strategies to select one CoT from all candidates (e.g., random or LM-based selection).

4 Evaluation on Human-written Ideas

4.1 Experiment Setup

We use GPT-4.1 as the main base model. Additionally, we evaluate multiple frontier LMs such as o3 and Claude 3.5 Sonnet. The knowledge cut-off of all these tested models is before July 1st, 2024[1].

Prior work shows that LMs often yield inconsistent predictions when swapping orders of inputs in pair-wise comparison tasks [3]. To mitigate potential order bias, we obtain two predictions for each test example by swapping the order of two ideas. We only consider the predictions valid when both predictions are consistent; otherwise, inconsistent predictions are considered incorrect by default.
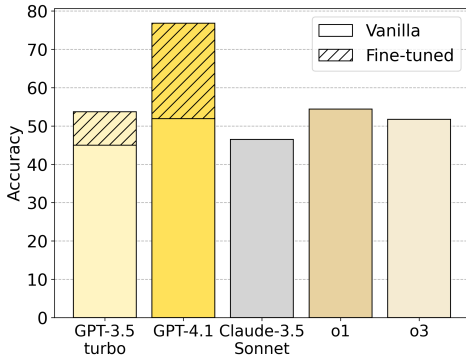
4.2 Automatic Evaluation

Figure 3: Automatic evaluation results.

Table 4: Fine-tuning GPT-4.1 with raw outcome labels or self-augmented CoTs.

Model	Accuracy
GPT-4.1	51.4
+ FT on Raw Label	77.0
+ FT on Random CoT	48.0
+ FT on LM-selected CoT	47.8

Finding 1: LMs are not naturally good at the task. We start with evaluating the zero-shot performance as a baseline. We select the best-performing zero-shot prompt among 10 candidates. As shown in Figure 3, even with the same retrieval augmentation, none of the frontier models are

---

[1]We didn't evaluate Claude 3.7 Sonnet since its knowledge cut-off is November 1st, 2024.

naturally good at this task. For example, GPT-4.1 only achieves an accuracy of 51.9%; reasoning models like o1 and o3 also just achieve similar or slightly better performance.

Finding 2: LMs can learn to predict idea outcome via fine-tuning on past ideas. Fine-tuning GPT-4.1 on past ideas substantially boosts its accuracy in predicting "future" ideas, from 51.9% to 77%. This demonstrates the importance of capability elicitation on this task.

However, fine-tuning GPT-3.5-turbo on the same data only yields a moderate improvement (45.0% to 53.7%). Given that GPT-3.5-turbo has a much earlier cut-off (September 1st, 2021), we suspect that GPT-4.1's performance arises from both its stronger general reasoning abilities and the updated knowledge of recent AI research — a crucial advantage given the rapid pace of AI in recent years.

In addition, the limited gains with GPT-3.5-turbo, a weaker but still capable model, reveal that our dataset does not have some trivial shortcuts that can be easily fit.

Finding 3: Self-augmented CoTs do not improve performance. Prior work [17] shows that fine-tuning LMs on their own generated CoTs can improve performance, e.g., in mathematical reasoning. On our benchmark, however, this approach yields no gain. As Table 4 shows, neither randomly sampled nor LM-selected CoTs improves upon the baseline. We suspect that because of GPT-4.1's near-chance performance, its generated CoTs are often not sound. Consequently, these low-quality CoTs provide little useful signals for fine-tuning.

4.3   Comparison with Expert Researchers

Data Preparation. Since our recruited annotators are primarily NLP researchers, we select five popular NLP topics: long-context, alignment, reasoning, agent, and efficient LM. For each topic, we sample 6 to 12 idea pairs from our test set, resulting in a total of 45 NLP idea pairs.

Human Annotation. For each topic, we recruit 5 researchers with relevant backgrounds. Among them, 68% have published papers on their assigned topics, while the remaining 32% also have extensive paper reading experience. Table 5 summarizes their profiles.

Given a test example, annotators first indicate whether they have read any of these two ideas. Then, they provide a binary prediction and a natural language rationale. The average time cost per annotation is 9.1 minutes. They are paid $50 for completing examples under their assigned topics. In total, we collect 225 labels; three are excluded since the annotators have read both ideas before. The final data thus has 222 valid annotations for the 45 idea pairs.

Table 5: Statistics of research profile and the time cost (minutes) per annotation of our recruited AI researchers.

Metric	Mean	Min	Max
Paper	18.6	2	90
Citation	1273.8	14	14,378
H-index	8.9	2	41
i10-index	9.3	1	56
Time Cost	9.1	4.5	15

Results. As shown in Figure 1 (left), even expert NLP researchers struggle to predict research outcomes in their own research domains. Specifically, doing majority voting over five experts' predictions on each example achieves only 48.9% accuracy. We further measure inter-annotator agreement, i.e., how often does each annotator's label agree with the majority voting label. The agreement score is 75.1%, which is similar to the inter-reviewer agreement on conference paper decisions, e.g., 75.8% in NeurIPS 2021 [11].

We also establish a ceiling human baseline by aggregating the best-performing researchers per research topic, where the best-performing researcher is selected based on the exact same test set. This ceiling human baseline yields a 60.0% accuracy. Overall, these results show that predicting an idea's chance of success is a difficult task that even domain experts struggle with. In comparison, our system achieves an accuracy of 64.4%, substantially surpassing human experts.

5   Stress-testing Our System's Robustness

We run extensive stress-testing to ensure that our system does not exploit ideas' superficial features, such as complexity and stylistic cues. Each test is based on one potential superficial feature that the model can potentially rely on. We divide our test set based on whether the label is correlated with the feature, e.g., examples where the more complex method performs better vs. examples
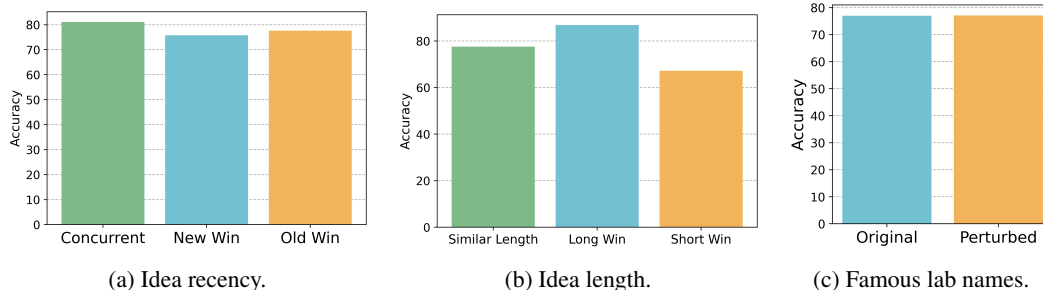
(a) Idea recency.   (b) Idea length.   (c) Famous lab names.

Figure 4: Stress-testing our system's sensitivity to three biases that humans might be prone to. The low accuracy variance across subsets of the test data indicates that our system is robust.
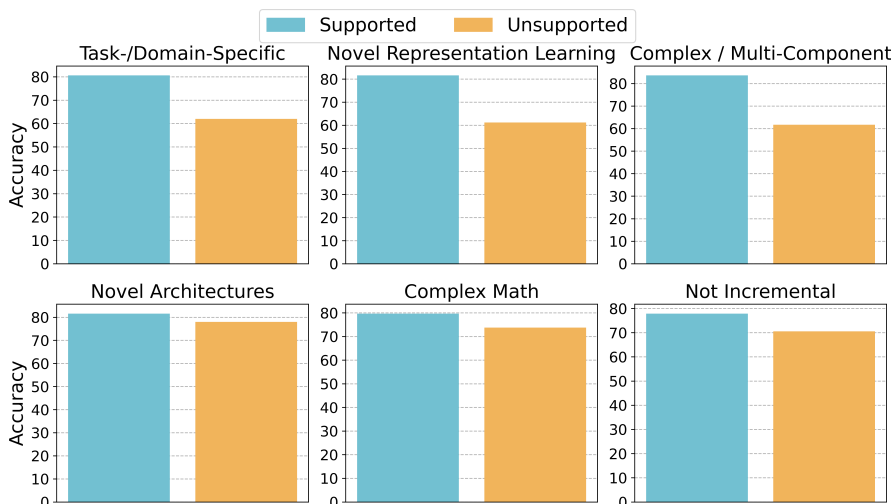

Figure 5: Our system is robust based on LM-designed stress tests. The *Supported* subset includes examples where the feature is noticeably more prominent in the winning idea, e.g., includes more complex math; the *Unsupported* subset is the opposite. Our system shows a slight (potentially justifiable) bias, but is generally robust.

where the simpler one is better. We then compare the model's accuracy on these subsets: a small accuracy difference indidates a robust model that's not overly sensitive to superficial features. We hand-designed three tests, then use an LM to propose many more tests.

5.1 Human-Designed Tests

We design three tests based on biases that human experts might be prone to when evaluating ideas.

Recency, i.e., favoring new ideas. We divide our test set into three sets based on whether the newer idea is better, or if the two ideas are concurrent (published within three months). Although we strip date information from idea descriptions, LMs might be able to infer that information based on prior knowledge and indirect cues such as references. As shown in Figure 4a, our system's accuracy is not sensitive to idea recency.

Length, i.e., favoring long and complicated ideas. We divide the test set into three based on whether the longer idea is better, and ties. Results shown in Figure 4b indicate that our system has a slight bias but can still identify complicated yet ineffective ideas.

Famous Lab Names, i.e., favoring ideas from "big names". To study this, we append the names of 10 famous labs (e.g., Anthropic, OpenAI, Google Deepmind) to the summary of the *losing* ideas. If our system exploits the shortcut, accuracy should notably drop after this perturbation. Results shown in Figure 4c indicate our system's resilience.

7

5.2 LM-Designed Tests

To improve the coverage of our stress-testing, we follow [18]and use LMs to automatically propose robustness tests at scale.

Method. Our pipeline consists of three steps: 1) proposing hypotheses about differences between two idea groups (model-preferred v.s. model-dispreferred), 2) verifying hypotheses across the test set, and 3) splitting the test set into supported and unsupported subsets according to each hypothesis.

In step 1, due to the limited model context length, each time we randomly sample 5 idea pairs and group them into model-preferred or model-dispreferred sets. We then query the LM: "how is the idea in Group A different from the idea in Group B". We repeat this 500 times to obtain 500 hypotheses. An example of LM-proposed hypothesis is: "Each research idea in Group A is more focused on utilizing advanced probabilistic and mathematical techniques for optimization and learning in machine learning models compared to Group B."

Since hypotheses are derived from only 5 examples, we further verify them across the whole test set. Specifically, for each example, we query the LM: "Compared to {losing idea}, is {winning idea} {Hypothesis}?". If so, this example is classified as supported under that hypothesis, because simply exploiting this shortcut can get perfect accuracy on such examples. Each hypothesis thus divides our test set into supported and unsupported subsets. For each hypothesis, we calculate a validity score, defined as the proportion of supported examples. We then discard hypotheses with scores outside the 25%-75% range, leaving us with 289 hypotheses.

For each test, the decrease in accuracy from the supported to the unsupported subset indicates sensitivity to the corresponding feature.

Results. Hypotheses are classified as *Flagged* if the accuracy on the corresponding unsupported subset drops below 62% (given the average accuracy of our system over the entire test set is 77%), otherwise they are marked as *Cleared*. Across all 289 verified hypotheses, our system achieves an average unsupported accuracy of 68.5%, and notably, over 88% (255 out of 289) are *Cleared*, indicating robustness against most hypothesized superficial biases.

Table 6: Results on LM-proposed hypothetical biases. Hypotheses are *Flagged* if the unsupported subset's accuracy falls below 62%, otherwise *Cleared*.

Metric	All	Flagged	Cleared
Size	289	34	255
Acc (Supported)	81.3	82.1	81.1
Acc (Unsupported)	68.5	61.3	69.4

Even among the 34 *Flagged* hypotheses, our model maintains a considerable average unsupported accuracy of 61.3%. The first row in Figure 5 shows the evaluation results of three flagged cases. In particular, we find two representative hypotheses: model-preferred ideas are often more complex or more specialized (i.e., task- or domain-specific). However, predicting more complicated or specialized yet ineffective ideas is inherently more challenging. Therefore, under such tests, our system's $> 61\%$ accuracy (Figure 5) is non-trivial.

Further, examining the 255 *Cleared* hypotheses reveals potential shortcuts that our model does not rely on. In the second row of Figure 5, we present the valuation results of three cleared cases, which stree-tests whether our system always prefers favor ideas that are 1) focused on novel architectures, 2) mathematically complex, or 3) entirely novel instead of doing incremental changes to existing methods. As we can see, our system remains robust under these features.

6 Evaluation on Novel Unpublished Ideas

So far, our evaluation is based on published ideas with control measures for contamination. The real test, however, is to make predictions about unpublished ideas. The bottleneck to such an evaluation is to obtain groundtruth by actually implementing the ideas.

We conduct a small-scale experiment with unpublished ideas with actual groundtruths. We start with a dataset of 35 ideas from an AI ideation study [14], half generated by their LM agent, and the other half generated by their recruited NLP experts on the spot. None of these ideas have been released anywhere on the internet. The research topics focus on novel prompting methods to improve various capabilities of LMs. The outcomes are obtained by expert researchers implementing the ideas, costing 103.4 hours per idea on average. For each idea, the experiments and results are documented as a

paper, from which we can extract the idea into our standard format for our benchmarking. We pair the proposed novel idea with its baseline and derive the golden label based on empirical outcomes. After excluding ties, we finally collect 33 labeled idea pairs.

Our system achieves an accuracy of 63.6%, demonstrating the potential to use our system as a reward model in a complete automated research pipeline, either to guide sampling at inference time, or optimizing the idea generation model using reinforcement learning.

7 Related Work

Accelerating Research with AI. With the overarching goal of using AI to build AI, recent work has explored integrating AI into the entire research workflow, including literature review [2], ideation [14, 4], experimental validation [6, 9], paper writing [9], and review [8]. Notably, one fully AI-generated paper passed the peer-review process of an ICLR workshop [16]. However, discovering such a workshop-level idea is non-trivial: Starting from 256 candidate papers, the authors manually select the top-3 papers since LM judges are often unreliable, while only one paper gets accepted. Importantly, executing 256 AI research ideas into whole papers is expensive and slow because of the significant GPU or API costs. To accelerate AI research, our paper aims to answer the question: "can we predict the outcomes of ideas before actually implementing them?"

Research Evaluation. Prior work mainly focuses on peer-review style evaluation; the goal is to predict scores that align with human reviewers based on fully written papers. However, such evaluations still require implementing ideas. Moreover, peer-review scores are often subjective. Critiques of novelty, excitement, and paper presentation are often independent of the ideas' actual empirical effectiveness. Consequently, humans can get misled to prefer "fancy" (e.g., mathematically complex) yet ineffective ideas. Instead, this papers focuses on objective empirical effectiveness, for each we can reliably establish ground truth labels.

Research Outcome Prediction. Prior work also attempts to use AI for predicting empirical research outcomes. The most relevant work is [10], which uses AIs to predict neuroscience results. However, their actual experiment setup is entirely different from ours. Specifically, given a published paper abstract, they use LMs to alter descriptions about results while keeping the method and background unchanged. The goal is then to distinguish the original abstract from its altered counterpart. This task is easy, even a fine-tuned Llama-2-7B model achieves an 80%+ accuracy. In contrast, we study a more realistic and challenging scenario: given two independently written idea summaries that exclude any empirical results, predict which is more effective.

Forecasting. Our task is a type of forecasting, i.e., forecasting the outcomes of ideas before running actual experiments to get the outcome. Prior attempts on neural forecasting methods demonstrate the gains from retrieval and scale (e.g., model and data scale) [19, 13, 1, 7]. For example, [5] builds a human-level forecasting system by retrieval augmentation and fine-tuning on historical data. Inspired by these works, our work demonstrates the potential of building specialized LMs to surpass human experts in forecasting outcomes of AI research ideas.

8 Discussion

Limitations. Our current system functions as a black-box and is not conducive to human-AI cooperation beyond prioritizing research ideas based on model predictions. And despite our best attempts to verify its robustness, we cannot rule out the possible reliance on spurious features.

Future Work. Our fine-tuning method is straightforward but works well. Future work can explore advanced modeling methods, e.g., inference-time simulation of experiments. Our system can also be used as a reward model to improve automated ideation systems.

Conclusion. A crucial skill in empirical research is being able to predict which ideas are more likely to work out. Human experts develop this skill by reading papers and observing experiments, we show that LMs can do this more efficiently; our LM system outperforms human experts at making direct predictions about the outcome of AI research without running experiments. In addition to higher accuracy, our system is robust to biases like recency that humans are prone to. The system also generalizes to completely novel, unpublished ideas, including AI-generated ones, demonstrating the potential of further accelerations of AI research.

9 Acknowledgements

References

[1] Mahdi Abolghasemi, Odkhishig Ganbold, and Kristian Rotaru. Humans vs large language models: Judgmental forecasting in an era of advanced ai. *arXiv preprint arXiv:2312.06941*, 2024.

[2] Shubham Agarwal, Issam H. Laradji, Laurent Charlin, and Christopher Pal. Litllm: A toolkit for scientific literature review. *arXiv preprint arXiv:2402.01788*, 2024.

[3] Yann Dubois, Chen Xuechen Li, Rohan Taori, Tianyi Zhang, Ishaan Gulrajani, Jimmy Ba, Carlos Guestrin, Percy S Liang, and Tatsunori B Hashimoto. Alpacafarm: A simulation framework for methods that learn from human feedback. *Advances in Neural Information Processing Systems*, 36:30039–30069, 2023.

[4] Juraj Gottweis, Wei-Hung Weng, Alexander 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, Yossi Matias, Andrew Carroll, Kavita Kulkarni, Nenad Tomasev, Vikram Dhillon, Eeshit Dhaval Vaishnav, Byron Lee, Tiago R D Costa, José R Penadés, Gary Peltz, Yunhan Xu, Annalisa Pawlosky, Alan Karthikesalingam, and Vivek Natarajan. Towards an ai co-scientist. *arXiv preprint arXiv:2502.18864*, 2025. CC BY-NC-ND 4.0.

[5] Danny Halawi, Fred Zhang, Yueh-Han Chen, and Jacob Steinhardt. Approaching human-level forecasting with language models. *arXiv preprint arXiv:2402.18563*, 2024.

[6] Qian Huang, Jian Vora, Percy Liang, and Jure Leskovec. Mlagentbench: Evaluating language agents on machine learning experimentation. *arXiv preprint arXiv:2310.03302*, 2023.

[7] Ezra Karger, Houtan Bastani, Yueh-Han Chen, Zachary Jacobs, Danny Halawi, Fred Zhang, and Philip E. Tetlock. Forecastbench: A dynamic benchmark of ai forecasting capabilities. *arXiv preprint arXiv:2409.19839*, 2025.

[8] Weixin Liang, Yuhui Zhang, Hancheng Cao, Binglu Wang, Daisy Yi Ding, Xinyu Yang, Kailas Vodrahalli, Siyu He, Daniel Scott Smith, Yian Yin, et al. Can large language models provide useful feedback on research papers? a large-scale empirical analysis. *NEJM AI*, 1(8):AIoa2400196, 2024.

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

[10] Xiaoliang Luo, Akilles Rechardt, Guangzhi Sun, Kevin K Nejad, Felipe Yáñez, Bati Yilmaz, Kangjoo Lee, Alexandra O Cohen, Valentina Borghesani, Anton Pashkov, et al. Large language models surpass human experts in predicting neuroscience results. *Nature human behaviour*, 9(2):305–315, 2025.

[11] None. Neurips 2021 summary, 2021.

[12] Reid Pryzant, Dan Iter, Jerry Li, Yin Tat Lee, Chenguang Zhu, and Michael Zeng. Automatic prompt optimization with" gradient descent" and beam search. *arXiv preprint arXiv:2305.03495*, 2023.

[13] Philipp Schoenegger and Peter S. Park. Large language model prediction capabilities: Evidence from a real-world forecasting tournament. *arXiv preprint arXiv:2310.13014*, 2023.

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

Table 7: Typical errors found in the human verification process sorted by frequency. Human annotators will rewrite or directly remove the incorrect data.

Error Type	Example	Solution
Incorrect Win Condition	The metric (e.g., perplexity) is the lower the better, but is extracted as the higher the better.	Rewrite
Incorrect Empirical Results	The numerical results are incorrect or hallucinated (i.e., do not exist in the paper).	Rewrite & Remove
Confusing Benchmark & Metric Description	The metric name is mistakenly extracted as the benchmark name.	Rewrite
Incomparable Ideas	One of the idea adopts an unfair experiment setup to serve as a ceiling or flooring baseline[2].	Remove

[15] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

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

[17] Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah D Goodman. Star: Self-taught reasoner bootstrapping reasoning with reasoning. In *Proc. the 36th International Conference on Neural Information Processing Systems*, volume 1126, 2024.

[18] Ruiqi Zhong, Peter Zhang, Steve Li, Jinwoo Ahn, Dan Klein, and Jacob Steinhardt. Goal driven discovery of distributional differences via language descriptions. *Advances in Neural Information Processing Systems*, 36:40204–40237, 2023.

[19] Andy Zou, Tristan Xiao, Ryan Jia, Joe Kwon, Mantas Mazeika, Richard Li, Dawn Song, Jacob Steinhardt, Owain Evans, and Dan Hendrycks. Forecasting future world events with neural networks. *arXiv preprint arXiv:2206.15474*, 2022.

Appendix

A Human Verification of Test Examples

Table 7 presents the typical errors found by our human annotators during the verification process.

B LM-Designed Tests

Table 8: Examples of LM-proposed hypotheses.

Hypothesis	Acc (Pos.)	Acc (Neg.)
Each research idea in Group A is more focused on specific domains or task improvements with detailed mechanisms and methodologies tailored for unique problem sets.	80.6	62.0
Group A ideas focus more on novel conceptual frameworks and mechanisms for learning representations and transferring knowledge.	81.6	61.2
Group A ideas focus more on novel machine learning and AI model architectures while Group B ideas concentrate on practical methodologies and algorithmic improvements in specific domains.	81.2	73.5
Each research idea in Group A is more focused on utilizing advanced probabilistic and mathematical techniques for optimization and learning in machine learning models compared to Group B	79.7	73.8
Each research idea in Group A is more focused on specific application areas or tasks compared to Group B's broader methodological innovations.	79.2	73.0
Group A provides more detailed descriptions and technical depth about model architectures and implementations whereas Group B offers more concise summaries and emphasizes key features and differences from existing methods.	77.9	70.6
Each research idea in Group A is more focused on innovative techniques for enhancing AI model efficiency and effectiveness without extensive model training or fine-tuning.	80.2	71.0
Each research idea in Group A is more focused on innovative architectural and training strategies tailored to specific types of data or tasks compared to Group B's emphasis on optimizing existing models and frameworks for efficiency and performance.	83.2	64.6

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: See experiments in Sec4, Sec5 and Sec6.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: See Sec8 Discussion.

   Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: The paper does not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: See Sec4, Sec5 and Sec6.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.

- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We will open source code and publicly available data we used in our experiments.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: See Sec2, Sec3 and Sec4.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: See Sec5.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: See Sec3 and Sec4.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: See Sec3 and Sec4.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: See Sec7 Conclusion.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.

- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: See references.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: see Sec2.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [Yes]

Justification: See Sec 2, sec 4 and Appendix.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.

- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [No]

    Justification: The paper does not collect demographic information from annotators.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.