All That Glitters is Not Novel: Plagiarism in AI Generated Research

Anonymous ACL submission

Abstract

Automating scientific research is considered the final frontier of science. Recently, several papers claim autonomous research agents can generate novel research ideas. Amidst the prevailing optimism, we document a critical concern: a considerable fraction of such research documents are smartly plagiarized. Unlike past efforts where experts evaluate the novelty and feasibility of research ideas, we request 13 experts to operate under a different situational logic: to identify similarities between LLM-generated research documents and existing work. Concerningly, the experts identify 24% of the 50 evaluated research documents to be either directly copied (with one-to-one methodological mapping), or significantly borrowed from existing work. These reported instances are cross-verified by authors of the source papers. Problematically, these LLM-generated research documents do not acknowledge original sources, and bypass inbuilt plagiarism detectors. Lastly, through controlled experiments we show that automated plagiarism detectors are inadequate at catching deliberately plagiarized ideas from an LLM. We recommend a careful assessment of LLM-generated research, and discuss the implications of our findings on research and academic publishing.[1]

1 Introduction

Automating research and discovering new knowledge has been a longstanding aspiration. The first step of scientific research is coming up with a bold hypothesis or a conjecture (Popper, 2014). Thus, automating this step is arguably the most critical and challenging aspect of automating scientific research. Recent research presents a positive case of LLMs' ability to generate novel scientific contributions—be they hypotheses, or proposals or papers (Li et al., 2024a; Lu et al., 2024a; Baek et al., 2024; Li et al., 2024c; Wang et al., 2023; Yang et al., 2023; Li et al., 2024b; Weng et al., 2024).

Understandably, evaluating the novelty LLM-generated ideas is challenging, especially given their subjective nature of scientific innovation. Previous studies evaluate novelty either through automated LLM-based judges (Lu et al., 2024a), or rely on small set of experts (Li et al., 2024a; Baek et al., 2024; Li et al., 2024c; Wang et al., 2023; Yang et al., 2023; Li et al., 2024b; Weng et al., 2024). Notably, the most rigorous evaluation to date engaged experts to evaluate 81 LLM-generated research proposals, implementing strict controls for confounding factors (Si et al., 2024). Their study leads to an important finding: human experts find LLM-generated research proposals to be *more novel* than human-written ones. The study also publicly releases four exemplar LLM-generated proposals, holding back others to be used for future work.

We conduct an expert-led evaluation where participants are instead instructed to presume plagiarism and actively search for it in LLM-generated research documents. This situational logic (Popper, 2013; Hoover, 2016) contrasts with prior assessment of LLM generated ideas, where experts evaluate a shuffled set of LLM and human-generated documents, presuming no deliberate plagiarism and scoring them on novelty and other factors, such as excitement and feasibility. In total, experts in our study evaluate 50 LLM-generated research documents, including 10 exemplar papers generated by the "The AI Scientist" (Lu et al., 2024a), four public research proposals from Si et al. (2024), and fresh 36 proposals.

We request the experts in our study to identify a topic of their expertise, and share with them 5 research proposals on their topics of expertise. We generate these proposals through the code made available by Si et al. (2024). The experts are then

[1] Our code, along with expert-provided scores and explanations for each proposal, is available at: `https://anonymous.4open.science/r/AI-Papers-Plagiarism-ECCA`

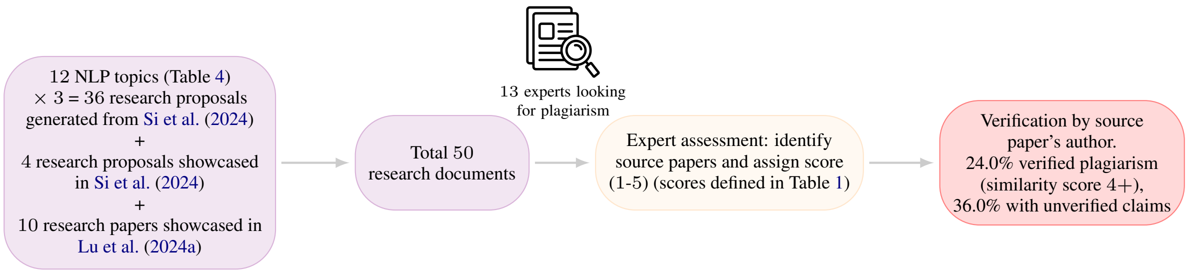

Figure 1: Overview of our expert-led evaluation for detecting plagiarism in LLM-generated research proposals. Unlike prior work, participants in our study are instructed to actively search for potential sources of plagiarism.

requested to score any 3 of the 5 research proposals on a scale of 1-5, where 5 indicates direct copying—that is, there exists a one-to-one mapping between the LLM proposed methodology and existing methods in one or two closely related prior papers. A score of 4 denotes that a significant portion of the LLM proposed method was borrowed from 2-3 prior works without credit. On the other hand, a score of 1 is reserved for cases when experts find the proposal to be completely original (see Table 1 for the rubric).

Our expert-led analysis reveals that 14.0% of LLM-generated research documents score 5 on our similarity evaluation scale (see Table 1), while another 10.0% receive a score of 4. We consider both scores 4 and 5 as instances of plagiarism, totaling 24.0% of proposals with noticeable plagiarism. These scores are verified by emailing the authors of referenced papers. When including claims where source paper authors are unreachable, these numbers increase—18.0% of proposals receive a score of 5 and 18.0% receive a score of 4, totalling to 36.0%. Notably, we also find several research documents previously showcased as exemplars of LLM generated research in prior studies (Si et al., 2024; Lu et al., 2024a) to be plagiarized. Our experimental setup and key findings are illustrated in Figure 1.

It is important to note that the examined documents do not acknowledge the original sources and the high degree of similarity with past work is not caught by inbuilt plagiarism detection systems. Typically, these inbuilt plagiarism detection checks use large language models, which has access to the Semantic Scholar API to retrieve similar papers (Si et al., 2024; Lu et al., 2024a; Li et al., 2024a).[2] To systematically evaluate automated detection methods, we create a synthetic dataset of research proposals intentionally plagiarized from existing papers. Using this controlled dataset, we evaluate common automated detection methods, including SSAG, OpenScholar (embedding-based search) (Asai et al., 2024), and a commercial service (Turnitin, LLC, 2025), and find them inadequate for detecting plagiarism in LLM-generated research proposals.

Our results reveal a concerning pattern wherein a significant portion of LLM-generated research ideas appear novel on the surface but are actually skillfully plagiarized in ways that make their originality difficult to verify. The case study presented in §5.2 supports this thesis, examining a research proposal (an exemplar in Si et al. (2024)) that appears to be skillfully plagiarized from an existing paper. Our analysis in §6 also finds that LLM-generated research content is less diverse and follows more predictable patterns. Our preliminary investigation suggests these patterns might be detectable through basic classification methods, potentially helping flag content for additional review, though more research would be needed to develop robust detection approaches.

While we do not recommend wholesale dismissal of LLM-generated research, our findings suggest that they may not be as novel as previously thought, and additional scrutiny is warranted. The sophisticated nature of the plagiarism we uncover suggests that widespread adoption of these tools could significantly impact the peer review process, requiring (already overwhelmed) reviewers to spend additional time searching for potential content misappropriation.

2 Related Work

LLM Generated Research And Its Novelty Evaluation. Evaluation approaches in this domain typically follow two paths: automated evaluation using LLMs themselves (Lu et al., 2024a) or human expert review (Li et al., 2024a; Baek et al., 2024; Li

[2]We refer to this approach as Semantic Scholar Augmented Generation (SSAG) throughout the paper.

Score	Description
5	**Direct Copy:** One-to-one mapping between the LLM proposed methodology and existing methods in one or two closely related prior papers.
4	**Combined Borrowing:** A significant portion of LLM proposed method is a mix-and-match from two-to-three prior works.
3	**Partial Overlap:** The LLM proposed method bears decent similarity with some existing methods, but there's no exact correspondence with a limited set of papers.
2	**Minor Similarity:** The LLM proposal bears very slight resemblance with some existing papers. Mostly novel.
1	**Original:** The LLM proposal is completely novel.

Table 1: Scoring rubric shared with experts to evaluate similarity of LLM-generated research proposals with prior work.

et al., 2024c; Wang et al., 2023; Yang et al., 2023; Li et al., 2024b; Weng et al., 2024), often close collaborators of the authors, which limits their findings as the judgments are done by a small, insular group. The scope and detail of generated outputs varies across studies, with some work focusing on concise proposals (Wang et al., 2023; Yang et al., 2023), while other approaches generate more detailed proposals or complete research papers (Si et al., 2024; Lu et al., 2024a). A recent large-scale study improves how we evaluate LLMs' ability to generate research proposals, with experts from multiple institutions reviewing 81 LLM-generated proposals (Si et al., 2024). This study reveals several key findings. First, LLMs demonstrate limited ability in evaluating research ideas. Second, through a carefully controlled methodology that generates and ranks multiple candidates using significant computational resources, the study finds a surprising result: human participants judge LLM-generated proposals as more novel than human-written ones.

Automated Plagiarism Detection Tools. Several studies integrate LLMs with academic search engines like Semantic Scholar to detect and filter out potential plagiarism in LLM-generated research ideas (Si et al., 2024; Lu et al., 2024a; Li et al., 2024a). The typical methodology involves extracting keywords from titles and abstracts using LLMs, querying these through Semantic Scholar's API, and performing one-to-one comparisons between retrieved papers and the generated ideas. While not specifically designed for plagiarism detection, OpenScholar (Asai et al., 2024) is a specialized retrieval-augmented LM that leverages a database of 45 million open-access papers with 237 million passage embeddings. Its retrieval mechanism combines nearest neighbor search over passage embeddings, keyword-based search through Semantic Scholar API, and academic web search results. Given this sophisticated multi-source retrieval system and its vast database of passage embeddings, OpenScholar could potentially serve as a powerful tool for embedding-based plagiarism detection. Traditional text-matching tools like Turnitin are also commonly used.

LLMs in Research Tasks. Recent research has explored LLMs' capabilities in predicting experimental outcomes (Luo et al., 2024), conducting research experiments (Huang et al., 2023; Tian et al., 2024), paper reviewing (Weng et al., 2024; Zeng et al., 2023), and related work generation (Hu et al., 2024a). We refer the reader to Luo et al. (2025) for a survey on this topic.

Creativity in AI. Our work connects to studies on AI creativity (Ismayilzada et al., 2024). For instance, studies on LLM poetry generation reveal that while humans struggle to distinguish between AI-generated and human-written poems (Porter and Machery, 2024), the outputs contain substantial verbatim matches with web text (Lu et al., 2024b), suggesting limited original creativity.

3 Background: Generating Proposals

Here, we elaborate on the methodology used to generate research proposals (Si et al., 2024), with particular attention to the plagiarism detection module, as understanding it is central to our discussion.

The research proposal generation process consists of six sequential steps, with Claude 3.5 Sonnet (Anthropic, 2024) being the backbone LLM. For a given topic, the first step uses a retrieval-augmented generation (RAG) system to retrieve and rank relevant papers using the Semantic Scholar API. The second step generates initial seed ideas using these retrieved papers, while the third step involves deduplication using text embeddings, retaining about

5% of the original ideas. The fourth step expands these seed ideas into detailed project proposals, and the fifth step implements a Swiss tournament system where proposals compete in pairwise comparisons over five rounds to identify the strongest candidates. We refer readers to Si et al. (2024) for additional details on these steps.

The final step, most relevant to our work, attempts to detect potential plagiarism through Semantic Scholar Augmented Generation (SSAG). First, an LLM iteratively generates queries for the Semantic Scholar API to find papers similar to the proposal's content, using results from previous iterations as context to inform each new query. This search process continues either until they collect 100 papers or reach 10 iterations, whichever comes first. The retrieved papers are then scored by an LLM based on their relevance, narrowing down to 10 most similar papers. Further, Claude 3.5 Sonnet performs pairwise comparisons between the proposal and each of these top 10 papers. The LLM is prompted to determine if the research proposal and the retrieved paper are substantially similar, discarding the proposal if they are. This process removes *only* about 1% of the generated proposals.

Similar detection approaches have been implemented in other research agents (Li et al., 2024a; Lu et al., 2024a), with minor variations in prompting strategies and parameters—Li et al. (2024a) and Lu et al. (2024a) query Semantic Scholar 3 and 10 times to search for similar papers, respectively.

For our study's implementation, Si et al. (2024)'s method is slightly modified to optimize computational costs while maintaining effectiveness. Instead of generating 4000 ideas per research topic, we generate 500 ideas per topic. This reduces our initial pool from ~ 200 unique ideas per topic (as in the original paper (Si et al., 2024)) to ~ 138, resulting in significant computational cost savings while only reducing unique ideas by 31%. All other components of the pipeline, including the plagiarism detection and ranking systems, are exactly the same as Si et al. (2024)'s original implementation.

The choice to use this specific methodology (Si et al., 2024) for generating research proposals is motivated by several factors. First, it is representative of fundamental prompt engineering techniques used across various research idea generation systems (Li et al., 2024a; Lu et al., 2024a; Baek et al., 2024; Li et al., 2024c; Wang et al., 2023; Yang et al., 2023; Weng et al., 2024). Second, as discussed in §2, this study conducts the most comprehensive evaluation to date, and finds that human participants judge LLM-generated research more novel than those by humans. Third, this method uses minimal prompt engineering. Other approaches like Li et al. (2024a) use sophisticated prompt engineering that function as "natural language programs," potentially incorporating human creativity. By keeping prompts simple, we can better assess the raw capabilities of LLMs themselves.

4 Experimental Design

We present our expert evaluation design for detecting plagiarism in LLM-generated research documents and our evaluation methodology for automated plagiarism detection tools.

4.1 Expert Evaluation Design

Our experimental setup involves generating five research proposals for each of the twelve topics listed in Table 4. These NLP topics are determined by asking participants to describe their areas of expertise, ensuring evaluators have in-depth familiarity with the current literature for detecting potential plagiarism. Each participant evaluates three out of five proposals that aligns with their domain knowledge, resulting in $3 \times 12 = 36$ total proposals.

In addition to these 36 proposals, we include fourteen previously showcased exemplars—four research proposals from Si et al. (2024) and ten research papers from Lu et al. (2024a), bringing the total to 50 research documents. These fourteen exemplars were evaluated by two experts. We use convenience sampling to recruit participants who are actively conducting research in NLP through our professional networks. Our participant pool comprises experts from 5 universities and 2 industrial labs, with 69% being Ph.D. students or recent graduates and 31% being associate researchers in industrial or academic labs. Unlike prior studies where participants assess LLM-generated research documents without suspecting plagiarism, our participants are instructed to actively search for potential overlaps with existing work.

Participants are asked to only consider papers available online on or before April 2024, corresponding to Claude 3.5 Sonnet's training cutoff date. Using a consistent evaluation rubric (defined in Table 1), participants assign similarity scores from 1-5 to each proposal. We focus on documents scoring 4 or 5, as these represent clear cases of content misappropriation—score 5 indicates direct copying with one-to-one mapping to existing methods, while score 4 indicates significant borrowing from prior work without attribution. These scores reflect methodological overlap considered plagiarism in academic publishing, as they represent either wholesale adoption of existing methods (score 5) or substantial uncredited incorporation of others' technical contributions (score 4). Lower scores represent more ambiguous cases of potential similarity. For all documents with scores 4 and 5, we email source paper authors for verification and adjust scores based on their feedback. Since some authors were unreachable, we report both verified claims and total claims separately.

While conventional human studies aim for objectivity through unbiased instructions and experimental design, our expert evaluation design derives objectivity primarily from author verifications and the inherently verifiable nature of plagiarism claims—readers can independently examine both the source and generated works through our open-sourced results. The instructions shared with participants are shown in Table 5. We discuss some limitations of our expert evaluation design in §8.

4.2 Evaluating Plagiarism Detectors

To systematically evaluate automated plagiarism detection tools, we require a dataset of plagiarized research documents paired with their original source papers. While our expert evaluation study uncovers instances of plagiarism, the sample size is too small for comprehensive testing of automated tools. We therefore create a synthetic test set by generating plagiarized research proposals from papers retrieved during the literature review step of the baseline method (Si et al., 2024).

For the twelve research topics listed in Table 4, we select 40 papers per topic, creating a test set of 480 papers. We then use GPT-4o (OpenAI et al., 2024) to generate plagiarized versions of these papers by skillfully paraphrasing the paper's details to avoid detection. We choose GPT-4o for this task because Claude 3.5 Sonnet abstains from plagiarism requests. The specific prompt used for this process is detailed in Table 6.

This synthetic testing approach poses a significantly easier challenge than detecting plagiarism in proposals generated by sophisticated research agents. Since we explicitly instruct GPT-4o to plagiarize from a single paper, novel elements in these deliberately plagiarized proposals are likely more limited than those produced by systems designed to generate novel research. Therefore, the performance of automated detection systems on our test set very likely overestimates their ability to detect more subtle plagiarism in LLM systems designed to generate novel research content.

Successful plagiarism detection requires two steps: first retrieving potentially plagiarized source papers, and then determining whether the retrieved papers are substantially similar to the proposal in question. We design experiments to evaluate these components both separately and together across three approaches. The first approach uses two LLMs (GPT-4o and Claude 3.5 Sonnet) in three distinct scenarios: (a) oracle access, wherein we provide an LLM with both the proposal and its source paper and evaluate its ability to detect plagiarism, (b) parametric knowledge testing, which examines LLMs' ability to both retrieve and determine similarity using only their training data without external tools, and (c) Semantic Scholar Augmented Generation (SSAG), described in Section 3, which explicitly separates the steps by first retrieving papers through Semantic Scholar and then determining similarity.[3]

The other two approaches, described in §2, are OpenScholar (Asai et al., 2024) (prompt in Table 7), an academic search system with a database of 45 million papers and sophisticated retrieval mechanisms, and Turnitin (Turnitin, LLC, 2025), a widely-used commercial plagiarism detection service. Due to the manual effort required for submission and analysis, we limit our testing with these tools to 3 papers per research topic, totalling 36.

5 Results

Our code and detailed findings, including all evaluated proposals and author verifications, are available at `https://anonymous.4open.science/r/AI-Papers-Plagiarism-ECCA`.

5.1 Expert Evaluation Results

Our expert evaluation reveals traces of plagiarism in LLM-generated research documents, as shown in Table 2. The distribution of similarity scores across verified claims indicates substantial content misappropriation at various levels of severity. Since our evaluation is limited by our participants' time constraints and the laborious manual effort required for thorough plagiarism checks, our results likely

[3]For SSAG, we use the following hyperparameters: maximum 50 papers and 5 iterations.

Score	Total Claims (%)	Verified (%)
5	**18.0% (9/50)**	**14.0% (7/50)**
4	**18.0% (9/50)**	**10.0% (5/50)**
3	32.0% (16/50)	8.0% (4/50)
2	28.0% (14/50)	4.0% (2/50)
1	4.0% (2/50)	0.0% (0/50)

Table 2: Distribution of similarity scores across evaluated LLM research documents. Higher scores indicate greater similarity to existing papers (scores defined in Table 1). Total Claims column shows summation of verified and unverified claims. Considering scores 4 and 5 as instances of plagiarism, 24.0% (36.0% if including unverified claims) have noticeable plagiarism.

Method		Accuracy
Claude 3.5 Sonnet	Oracle access	88.8%
	Parameteric Knowledge	1.3%
	SSAG	51.3%
GPT-4o	Oracle access	89.0%
	Parameteric Knowledge	32.7%
	SSAG	68.5%
OpenScholar		0%
Turnitin		0%

Table 3: Comparison of automated plagiarism detection methods: LLMs in three scenarios (oracle access, parametric knowledge, and SSAG), OpenScholar, and Turnitin.

represent a lower bound on actual plagiarism rates. Interestingly, of the four exemplars presented in Si et al. (2024), one received a similarity score of 5 and another received a score of 4, while among the ten exemplars in Lu et al. (2024a), two received scores of 5 and one received score 4—all of these ratings are cross-verified by the source papers' authors. It is crucial to note the documents that receive similarity scores of 5 in our expert-led evaluation passed through SSAG plagiarism detection checks, indicating serious limitations of such systems (discussed further in §5.3).

We also evaluate OpenScholar and Turnitin on research documents with verified scores of 4 or above. Using the prompt in Table 7 with the document's title, problem statement, motivation, and method, OpenScholar's suggested related works (typically 4-5 papers) included the correct source paper only in one case. Turnitin did not identify the original paper in any case. We further analyze these findings through a detailed case study in §5.2.

5.2 A Case Study

To illustrate the nature of plagiarism in LLM-generated research documents, we examine a proposal (an exemplar in (Si et al., 2024)) titled "Semantic Resonance Uncertainty Quantification" (readers can find this document in their paper). This document receives a similarity score of 5 in our evaluation, with direct correspondence to an existing published paper, "Generating with Confidence: Uncertainty Quantification for Black-box Large Language Models" (Lin et al., 2023). The authors of the original paper confirm our plagiarism assessment after reviewing the proposal in question.

As shown in Figure 2, the proposed methodology exhibits a clear one-to-one mapping with the original paper. Each component of the LLM-generated proposal corresponds to specific sections in the source paper, albeit with skillfully reworded descriptions. The proposal proposes the same technical approach while using different terminology (e.g., "resonance graph" instead of "weighted adjacency matrix") and restructuring the presentation. This may be interpreted as adversarial behavior, where the LLM has learned to disguise existing work as novel research through careful rewording. Notably, expert reviewers in Si et al. (2024)'s study do not identify this plagiarism, likely because their evaluation focuses on assessing novelty and feasibility rather than actively searching for potential sources of plagiarism. In contrast, our study's participants, operating under different situational logic (Popper, 2013; Hoover, 2016) that presumes potential plagiarism, are able to identify the original paper. A similar analysis of plagiarism in a showcased paper from Lu et al. (2024a) is provided in Appendix A.

This case-study, along with results of expert evaluation presented in §5.1, showcases the flaws of previous studies on human evaluation of LLM generated research ideas—without a skeptical eye for plagiarism, human experts may be fooled into considering LLM generated research ideas as novel.

5.3 Performance of Plagiarism Detectors

To evaluate automated plagiarism detection methods, we test LLMs in three scenarios (oracle access, parametric knowledge, and SSAG), OpenScholar, and Turnitin. Table 3 presents their performance metrics across these settings. Even in our simplified test scenario with deliberately plagiarized

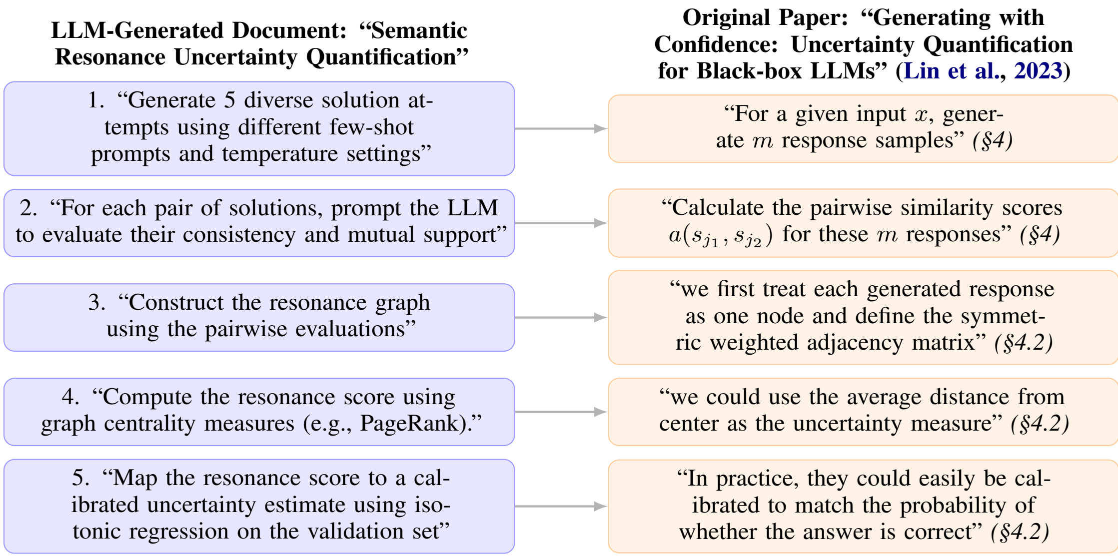

Figure 2: Visual mapping between an LLM-generated research document (an exemplar in Si et al. (2024)) and its corresponding source paper, showing direct correspondence between methodology components. Each element of the proposed method has a corresponding match in the source paper, suggesting sophisticated rewording rather than novel contribution. This pair receives a similarity score of 5 in our expert evaluation, which is verified by the authors of Lin et al. (2023).

proposals, detection accuracy remains remarkably low across all methods.

The oracle access setting, where models are given both the ground truth paper and plagiarized proposal, yields the highest accuracy (88.8% for Claude 3.5 Sonnet and 89.0% for GPT-4o). However, this represents an idealized scenario rarely available in practice, where the source of potential plagiarism is unknown. When testing models' parametric knowledge without access to external tools, GPT-4o achieves notably higher accuracy (32.7%) compared to Claude 3.5 Sonnet (1.3%), likely because GPT-4o is used to generate these plagiarized proposals in the first place.

The SSAG approach, a common plagiarism detection method in research document generation systems (Si et al., 2024; Lu et al., 2024a; Li et al., 2024a), attains moderate performance (51.3% for Claude, 68.5% for GPT-4o). The considerable gap between SSAG performance and oracle access indicates that retrieving relevant papers, not determining similarity, is the bottleneck.

The inadequate performance of SSAG is particularly concerning for research agent systems that rely on it (Si et al., 2024; Lu et al., 2024a; Li et al., 2024a), especially given our test scenario represents a much easier challenge—our proposals are deliberately plagiarized from single papers, while research agent systems are specifically designed to produce allegedly novel content. (An illustrative example of successful plagiarism detection using the SSAG from our synthetic dataset is provided in Appendix B.)

Both OpenScholar and Turnitin, tested on a smaller subset of proposals (3 per topic), fail to detect any instances of plagiarism. These results, combined with significant plagiarism found in our expert evaluation (§5.1), substantiate that current automated plagiarism detection methods are inadequate for identifying content misappropriation in LLM-generated research documents.

6 Discussion

Implications for Academic Publishing. Presence of considerable plagiarism in LLM-generated research documents suggests that widespread adoption of these tools could lead to an increase in publications with improper citations or inadvertent plagiarism. Furthermore, if researchers submit LLM-generated work to conferences and journals, the sophisticated nature of this plagiarism would require conference and journal reviewers to spend considerably more time searching for potential content misappropriation, adding to an already heavy reviewer workload.

Citing Relevant Work Is Challenging. One might argue that requesting LLMs to provide citations while generating research documents could mitigate plagiarism concerns. In our preliminary experiments, we manually examine several cases where we ask LLMs to generate proposals with cita-

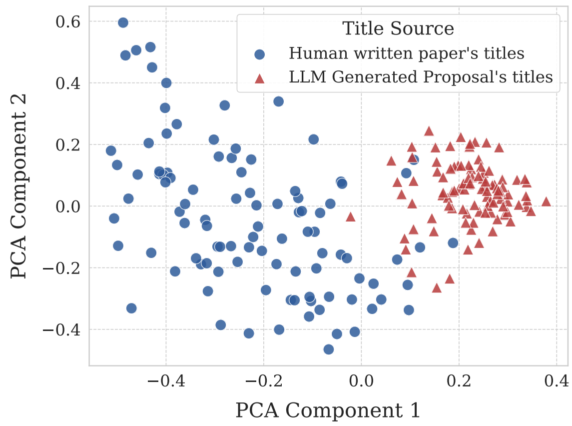

Figure 3: PCA projection of title embeddings for human-written papers and LLM-generated proposals on the topic "Novel AI-assisted formal proof generation methods". LLM-generated proposals occupy a more confined region, indicating less diversity in outputs.

tions. However, the LLMs typically reference a few well-known papers, raising concerns about citation accuracy. A comprehensive evaluation of citation quality would require expert verification of each citation and, more importantly, identification of relevant citations that were omitted, making large-scale testing expensive. This limitation aligns with broader findings in the field—Asai et al. (2024) find that 78-90% of papers cited by non-retrieval augmented LLMs are hallucinated. While recent works (Asai et al., 2024; Gao et al., 2023; Qian et al., 2024) attempt to improve citation generation, their findings suggest that LLMs struggle with accurate citations, leading researchers to rely on RAG and other external tools for citation support.

Limited Diversity in Proposals. Prior research notes limited diversity in LLM-generated research proposals (Si et al., 2024), which our analysis confirms. For each research topic in Table 4, we analyze titles of 100 human-written papers and 100 LLM-generated proposals.[4] Using the all-MiniLM-L6-v2 sentence transformer (Reimers and Gurevych, 2020), we generate embeddings for these titles. Calculating cluster spread via mean squared distance from centroids, we find that despite aggressive deduplication strategies (Si et al., 2024), LLM-generated titles are more tightly clustered—the ratio of LLM to human cluster spreads averages 0.76 across topics. Figure 3 illustrates this pattern, showing that LLM proposals occupy a more confined region in the embedding space. While this analysis is limited since titles only reflect broad research directions rather than specific ideas, the clear clustering pattern suggests current LLM systems explore a narrower range of research directions compared to humans, undermining their utility for scientific innovation.

Proposal Titles Are Easily Distinguishable. Beyond being more tightly clustered, LLM-generated proposals also appear to follow distinctive patterns in their research directions. In a preliminary analysis using 100 human-written papers and 100 LLM-generated proposals per topic from Table 4, a simple logistic regression model with bag-of-words features on a 60:40 train-test split achieves 94.2% accuracy in classifying titles.[4] While this basic classification experiment has limitations as titles only capture broad research directions, it provides initial evidence that LLM-generated research follows predictable patterns. However, developing reliable detection methods will require significant further research and is beyond the scope of this work.

Future Research Directions. Developing ways to identify candidate source papers is one of the most pressing future research direction, as current automated detection methods are inadequate and manual evaluation by domain experts is both time-consuming and laborious. Future work could explore post-training strategies that could help reduce plagiarism in LLM-generated research content. Lastly, future studies could examine whether LLM-generated content directly copy, or significantly borrow, content from copyrighted materials, possibly through the lens of fair-use doctrine (Patterson, 1992; Balaji, 2024).

7 Conclusion

We conducted the first systematic study of plagiarism in LLM-generated research documents. Through an expert-led evaluation, we discovered significant levels of plagiarism in these documents. We demonstrated the inadequacy of current automated plagiarism detection methods, while our case studies revealed sophisticated forms of content misappropriation that passed through multiple layers of filtering and expert review. Our findings raise important concerns about potential wide-spread use of LLMs for research ideation and highlights the need for better plagiarism detection methods.

[4]Due to deduplication and filtering in Si et al. (2024), some topics had fewer than 100 proposals, resulting in 2370 total samples rather than the maximum of 2400 (200 samples × 12 topics)

8 Limitations

Automation Challenges. A key limitation identified in our study stems from the challenge of automating the detection of original papers that may have been plagiarized. Currently, no reliable automated systems exist for this task, necessitating reliance on human expertise. This manual process is time-intensive, making it a critical bottleneck.

Constraints Related to Expert Evaluation. Our expert evaluation design has some limitations. First, we ask our human participants to presume adversarial plagiarism and actively search for it. This may introduce confirmation bias, leading them to give higher similarity scores in order to complete their task. We indeed find 4 instances where authors of source-papers over-turned the score downwards with difference of 2 or more. However, our (author-verified) scores already account for these adjustments, and our findings remain independently verifiable through our open-sourced results. Second, we provide our human-participants with 5 research proposals and ask them to choose any 3. While this gives flexibility to our participants, since even in a single topic there can be sub-topics that participants may not be familiar with, it may introduce sampling bias.

Computational Parameter Reductions. We reduce the quantity of certain hyperparameters to optimize computational costs. First, as detailed and motivated in §3, we decrease the number of candidate proposals generated per topic, resulting in approximately 31% fewer unique ideas compared to the baseline method (Si et al., 2024). Although this reduction might marginally affect the quality of research proposals relative to Si et al. (2024), the validity of our findings remains, particularly given that we identify plagiarism even in the LLM-generated proposals showcased in their original work. Second, in our automated plagiarism detection experiments (§4.2), we limit Semantic Scholar queries to a maximum of 5 iterations, fewer than the 10 iterations employed in Si et al. (2024); Lu et al. (2024a), but more than the 3 iterations in Li et al. (2024a). However, as elaborated in §4.2, our synthetic dataset presents a simpler challenge than detecting plagiarism in actual research proposal generation pipelines, possibly counterbalancing the reduced computational parameters (note that for generating proposals in our expert evaluation study, we maintain the same number of Semantic Scholar queries as Si et al. (2024) in the plagiarism filtering step).

Title-only Analysis. Our clustering and classification experiments (§6) rely solely on paper titles as proxies for broad research directions and topics, rather than using full text. While this approach may seem limited, we deliberately avoid using complete documents because LLM-generated proposals and human-written papers follow different organizational structures. These structural differences would confound any full-text analysis, making titles the most comparable unit for exploring patterns in research directions. These two experiments should be viewed as preliminary explorations rather than methodological recommendations.

Despite these limitations, we believe that our study highlights a critical concern, and adds nuance to the ongoing discourse about the role of LLMs in scientific research.

References

Anthropic. 2024. Introducing claude 3.5 sonnet.

Akari Asai, Jacqueline He, Rulin Shao, Weijia Shi, Amanpreet Singh, Joseph Chee Chang, Kyle Lo, Luca Soldaini, Sergey Feldman, Mike D'arcy, et al. 2024. Openscholar: Synthesizing scientific literature with retrieval-augmented lms. *arXiv preprint arXiv:2411.14199*.

Jinheon Baek, Sujay Kumar Jauhar, Silviu Cucerzan, and Sung Ju Hwang. 2024. Researchagent: Iterative research idea generation over scientific literature with large language models. *arXiv preprint arXiv:2404.07738*.

Suchir Balaji. 2024. When does generative AI qualify for fair use? `https://suchir.net/fair_use.html`.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. 2023. Enabling large language models to generate text with citations. In *Conference on Empirical Methods in Natural Language Processing*.

Jiatao Gu, Shuangfei Zhai, Yizhe Zhang, Joshua M Susskind, and Navdeep Jaitly. 2023. Matryoshka diffusion models. In *The Twelfth International Conference on Learning Representations*.

Kevin D Hoover. 2016. Situational analysis. In *The Routledge companion to philosophy of social science*, pages 202–210. Routledge.

Yuntong Hu, Zhuofeng Li, Zheng Zhang, Chen Ling, Raasikh Kanjiani, Boxin Zhao, and Liang Zhao. 2024a. Hireview: Hierarchical taxonomy-driven automatic literature review generation. *arXiv preprint arXiv:2410.03761*.

Zhiyuan Hu, Yuliang Liu, Jinman Zhao, Suyuchen Wang, Yan Wang, Wei Shen, Qing Gu, Anh Tuan Luu, See-Kiong Ng, Zhiwei Jiang, et al. 2024b. Longrecipe: Recipe for efficient long context generalization in large language models. *arXiv preprint arXiv:2409.00509*.

Qian Huang, Jian Vora, Percy Liang, and Jure Leskovec. 2023. Benchmarking large language models as ai research agents. In *NeurIPS 2023 Foundation Models for Decision Making Workshop*.

Mete Ismayilzada, Debjit Paul, Antoine Bosselut, and Lonneke van der Plas. 2024. Creativity in ai: Progresses and challenges. *arXiv preprint arXiv:2410.17218*.

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

Ruochen Li, Liqiang Jing, Chi Han, Jiawei Zhou, and Xinya Du. 2024b. Learning to generate research idea with dynamic control. *arXiv preprint arXiv:2412.14626*.

Ruochen Li, Teerth Patel, Qingyun Wang, and Xinya Du. 2024c. Mlr-copilot: Autonomous machine learning research based on large language models agents. *arXiv preprint arXiv:2408.14033*.

Zhen Lin, Shubhendu Trivedi, and Jimeng Sun. 2023. Generating with confidence: Uncertainty quantification for black-box large language models. *arXiv preprint arXiv:2305.19187*.

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

Ximing Lu, Melanie Sclar, Skyler Hallinan, Niloofar Mireshghallah, Jiacheng Liu, Seungju Han, Allyson Ettinger, Liwei Jiang, Khyathi Chandu, Nouha Dziri, et al. 2024b. Ai as humanity's salieri: Quantifying linguistic creativity of language models via systematic attribution of machine text against web text. *arXiv preprint arXiv:2410.04265*.

Xiaoliang Luo, Akilles Rechardt, Guangzhi Sun, Kevin K Nejad, Felipe Yáñez, Bati Yilmaz, Kangjoo Lee, Alexandra O Cohen, Valentina Borghesani, Anton Pashkov, et al. 2024. Large language models surpass human experts in predicting neuroscience results. *Nature human behaviour*, pages 1–11.

Ziming Luo, Zonglin Yang, Zexin Xu, Wei Yang, and Xinya Du. 2025. Llm4sr: A survey on large language models for scientific research. *arXiv preprint arXiv:2501.04306*.

OpenAI, :, Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Mądry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, Alex Nichol, Alex Paino, Alex Renzin, Alex Tachard Passos, Alexander Kirillov, Alexi Christakis, Alexis Conneau, Ali Kamali, Allan Jabri, Allison Moyer, Allison Tam, Amadou Crookes, Amin Tootoochian, Amin Tootoonchian, Ananya Kumar, Andrea Vallone, Andrej Karpathy, Andrew Braunstein, Andrew Cann, Andrew Codispoti, Andrew Galu, Andrew Kondrich, Andrew Tulloch, Andrey Mishchenko, Angela Baek, Angela Jiang, Antoine Pelisse, Antonia Woodford, Anuj Gosalia, Arka Dhar, Ashley Pantuliano, Avi Nayak, Avital Oliver, Barret Zoph, Behrooz Ghorbani, Ben Leimberger, Ben Rossen, Ben Sokolowsky, Ben Wang, Benjamin Zweig, Beth Hoover, Blake Samic, Bob McGrew, Bobby Spero, Bogo Giertler, Bowen Cheng, Brad Lightcap, Brandon Walkin, Brendan Quinn, Brian Guarraci, Brian Hsu, Bright Kellogg, Brydon Eastman, Camillo Lugaresi, Carroll Wainwright, Cary Bassin, Cary Hudson, Casey Chu, Chad Nelson, Chak Li, Chan Jun Shern, Channing Conger, Charlotte Barette, Chelsea Voss, Chen Ding, Cheng Lu, Chong Zhang, Chris Beaumont, Chris Hallacy, Chris Koch, Christian Gibson, Christina Kim, Christine Choi, Christine McLeavey, Christopher Hesse, Claudia Fischer, Clemens Winter, Coley Czarnecki, Colin Jarvis, Colin Wei, Constantin Koumouzelis, Dane Sherburn, Daniel Kappler, Daniel Levin, Daniel Levy, David Carr, David Farhi, David Mely, David Robinson, David Sasaki, Denny Jin, Dev Valladares, Dimitris Tsipras, Doug Li, Duc Phong Nguyen, Duncan Findlay, Edede Oiwoh, Edmund Wong, Ehsan Asdar, Elizabeth Proehl, Elizabeth Yang, Eric Antonow, Eric Kramer, Eric Peterson, Eric Sigler, Eric Wallace, Eugene Brevdo, Evan Mays, Farzad Khorasani, Felipe Petroski Such, Filippo Raso, Francis Zhang, Fred von Lohmann, Freddie Sulit, Gabriel Goh, Gene Oden, Geoff Salmon, Giulio Starace, Greg Brockman, Hadi Salman, Haiming Bao, Haitang Hu, Hannah Wong, Haoyu Wang, Heather Schmidt, Heather Whitney, Heewoo Jun, Hendrik Kirchner, Henrique Ponde de Oliveira Pinto, Hongyu Ren, Huiwen Chang, Hyung Won Chung, Ian Kivlichan, Ian O'Connell, Ian O'Connell, Ian Osband, Ian Silber, Ian Sohl, Ibrahim Okuyucu, Ikai Lan, Ilya Kostrikov, Ilya Sutskever, Ingmar Kanitscheider, Ishaan Gulrajani, Jacob Coxon, Jacob Menick, Jakub Pachocki, James Aung, James Betker, James Crooks, James Lennon, Jamie Kiros, Jan Leike, Jane Park, Jason Kwon, Jason Phang, Jason Teplitz, Jason Wei, Jason Wolfe, Jay Chen, Jeff Harris, Jenia Varavva, Jessica Gan Lee, Jessica Shieh, Ji Lin, Jiahui Yu, Jiayi Weng, Jie Tang, Jieqi Yu, Joanne Jang, Joaquin Quinonero Candela, Joe Beutler, Joe Landers, Joel Parish, Johannes Heidecke, John Schulman, Jonathan Lachman, Jonathan McKay, Jonathan Uesato, Jonathan Ward, Jong Wook Kim, Joost Huizinga, Jordan Sitkin, Jos Kraaijeveld, Josh Gross, Josh Kaplan, Josh Snyder, Joshua Achiam, Joy Jiao, Joyce Lee, Juntang Zhuang, Justyn Harriman, Kai

Fricke, Kai Hayashi, Karan Singhal, Katy Shi, Kavin Karthik, Kayla Wood, Kendra Rimbach, Kenny Hsu, Kenny Nguyen, Keren Gu-Lemberg, Kevin Button, Kevin Liu, Kiel Howe, Krithika Muthukumar, Kyle Luther, Lama Ahmad, Larry Kai, Lauren Itow, Lauren Workman, Leher Pathak, Leo Chen, Li Jing, Lia Guy, Liam Fedus, Liang Zhou, Lien Mamitsuka, Lilian Weng, Lindsay McCallum, Lindsey Held, Long Ouyang, Louis Feuvrier, Lu Zhang, Lukas Kondraciuk, Lukasz Kaiser, Luke Hewitt, Luke Metz, Lyric Doshi, Mada Aflak, Maddie Simens, Madelaine Boyd, Madeleine Thompson, Marat Dukhan, Mark Chen, Mark Gray, Mark Hudnall, Marvin Zhang, Marwan Aljubeh, Mateusz Litwin, Matthew Zeng, Max Johnson, Maya Shetty, Mayank Gupta, Meghan Shah, Mehmet Yatbaz, Meng Jia Yang, Mengchao Zhong, Mia Glaese, Mianna Chen, Michael Janner, Michael Lampe, Michael Petrov, Michael Wu, Michele Wang, Michelle Fradin, Michelle Pokrass, Miguel Castro, Miguel Oom Temudo de Castro, Mikhail Pavlov, Miles Brundage, Miles Wang, Minal Khan, Mira Murati, Mo Bavarian, Molly Lin, Murat Yesildal, Nacho Soto, Natalia Gimelshein, Natalie Cone, Natalie Staudacher, Natalie Summers, Natan LaFontaine, Neil Chowdhury, Nick Ryder, Nick Stathas, Nick Turley, Nik Tezak, Niko Felix, Nithanth Kudige, Nitish Keskar, Noah Deutsch, Noel Bundick, Nora Puckett, Ofir Nachum, Ola Okelola, Oleg Boiko, Oleg Murk, Oliver Jaffe, Olivia Watkins, Olivier Godement, Owen Campbell-Moore, Patrick Chao, Paul McMillan, Pavel Belov, Peng Su, Peter Bak, Peter Bakkum, Peter Deng, Peter Dolan, Peter Hoeschele, Peter Welinder, Phil Tillet, Philip Pronin, Philippe Tillet, Prafulla Dhariwal, Qiming Yuan, Rachel Dias, Rachel Lim, Rahul Arora, Rajan Troll, Randall Lin, Rapha Gontijo Lopes, Raul Puri, Reah Miyara, Reimar Leike, Renaud Gaubert, Reza Zamani, Ricky Wang, Rob Donnelly, Rob Honsby, Rocky Smith, Rohan Sahai, Rohit Ramchandani, Romain Huet, Rory Carmichael, Rowan Zellers, Roy Chen, Ruby Chen, Ruslan Nigmatullin, Ryan Cheu, Saachi Jain, Sam Altman, Sam Schoenholz, Sam Toizer, Samuel Miserendino, Sandhini Agarwal, Sara Culver, Scott Ethersmith, Scott Gray, Sean Grove, Sean Metzger, Shamez Hermani, Shantanu Jain, Shengjia Zhao, Sherwin Wu, Shino Jomoto, Shirong Wu, Shuaiqi, Xia, Sonia Phene, Spencer Papay, Srinivas Narayanan, Steve Coffey, Steve Lee, Stewart Hall, Suchir Balaji, Tal Broda, Tal Stramer, Tao Xu, Tarun Gogineni, Taya Christianson, Ted Sanders, Tejal Patwardhan, Thomas Cunninghman, Thomas Degry, Thomas Dimson, Thomas Raoux, Thomas Shadwell, Tianhao Zheng, Todd Underwood, Todor Markov, Toki Sherbakov, Tom Rubin, Tom Stasi, Tomer Kaftan, Tristan Heywood, Troy Peterson, Tyce Walters, Tyna Eloundou, Valerie Qi, Veit Moeller, Vinnie Monaco, Vishal Kuo, Vlad Fomenko, Wayne Chang, Weiyi Zheng, Wenda Zhou, Wesam Manassra, Will Sheu, Wojciech Zaremba, Yash Patil, Yilei Qian, Yongjik Kim, Youlong Cheng, Yu Zhang, Yuchen He, Yuchen Zhang, Yujia Jin, Yunxing Dai, and Yury Malkov. 2024. Gpt-4o system card. *Preprint*, arXiv:2410.21276.

Byeongjun Park, Hyojun Go, Jin-Young Kim, Sangmin Woo, Seokil Ham, and Changick Kim. 2024. Switch diffusion transformer: Synergizing denoising tasks with sparse mixture-of-experts. *Preprint*, arXiv:2403.09176.

L Ray Patterson. 1992. Understanding fair use. *Law and contemporary problems*, 55(2):249–266.

Karl Popper. 2013. *The poverty of historicism*. Routledge.

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

Brian Porter and Edouard Machery. 2024. Ai-generated poetry is indistinguishable from human-written poetry and is rated more favorably. *Scientific Reports*, 14(1):26133.

Haosheng Qian, Yixing Fan, Ruqing Zhang, and Jiafeng Guo. 2024. On the capacity of citation generation by large language models. *ArXiv*, abs/2410.11217.

Nils Reimers and Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation. *arXiv preprint arXiv:2004.09813*.

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

Minyang Tian, Luyu Gao, Shizhuo Dylan Zhang, Xinan Chen, Cunwei Fan, Xuefei Guo, Roland Haas, Pan Ji, Kittithat Krongchon, Yao Li, et al. 2024. Scicode: A research coding benchmark curated by scientists. *arXiv preprint arXiv:2407.13168*.

Turnitin, LLC. 2025. Turnitin. Accessed: 2025-02-01.

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

Yixuan Weng, Minjun Zhu, Guangsheng Bao, Hongbo Zhang, Jindong Wang, Yue Zhang, and Linyi Yang. 2024. Cycleresearcher: Improving automated research via automated review. *arXiv preprint arXiv:2411.00816*.

Zonglin Yang, Xinya Du, Junxian Li, Jie Zheng, Soujanya Poria, and Erik Cambria. 2023. Large language models for automated open-domain scientific hypotheses discovery. *arXiv preprint arXiv:2309.02726*.

Qi Zeng, Mankeerat Sidhu, Hou Pong Chan, Lu Wang, and Heng Ji. 2023. Scientific opinion summarization: Meta-review generation with checklist-guided iterative introspection. *arXiv preprint arXiv:2305.14647*.

Research Topics
Improving long context capabilities of large language models.
Evaluating abstention capabilities and techniques for language models.
Evaluating geographical and cultural bias in large language models.
Novel methods to improve trustworthiness and reduce hallucinations of large language models.
Novel methods for mechanistic interpretability of large language models.
Novel methods to add speech and audio processing capabilities into large language models.
Novel AI-assisted formal proof generation methods.
Human-centric evaluation of large language models, and development of language technologies from a social perspective.
Novel techniques and metrics for evaluating machine translation systems.
Novel methods to understand neural scaling laws for large language model training.
Novel methods to improve inference performance of large language models.
Novel methods for developing and evaluating large language model based personas.

Table 4: Research topics in natural language processing used for generating LLM proposals and matching with expert evaluators' domain expertise.

A Additional Case Study

To provide another example of sophisticated plagiarism in LLM-generated research, we examine a paper titled "DualDiff: Enhancing Mode Capture in Low-dimensional Diffusion Models via Dual-expert Denoising" (an exemplar in Lu et al. (2024a)). This paper receives a similarity score of 5 in our expert evaluation study, with direct correspondence to an existing paper, "Switch Diffusion Transformer: Synergizing Denoising Tasks with Sparse Mixture-of-Experts" (Park et al., 2024). The authors of the original paper confirm our plagiarism assessment after reviewing both works.

As shown in Figure 4, the proposed methodology exhibits clear similarities with the original paper. The LLM-generated paper proposes combining outputs from two diffusion models for lower and higher resolution data using learned weights, concepts previously explored in different contexts—combining multiple diffusion paths in Park et al. (2024) and jointly training diffusion models at multiple resolutions in Gu et al. (2023). Each major component of the proposed method has a corresponding match in Park et al. (2024): the gating mechanism is identical, and the "diversity loss" is closely analogous to "diffusion prior loss" from the original work, both utilizing pair-wise distance functions.

The methodology for generating these research papers involves 20 rounds of iterative search using LLM and Semantic Scholar to identify relevant citations (Lu et al., 2024a). However, this process fails to identify and cite the original work (Park et al., 2024). This oversight demonstrates important limitations in using current methods to find closely related work in LLM-generated research documents, which in turn reveals their inadequacy for automated plagiarism detection. The case also illustrates how generated content can reformulate existing technical approaches using different terminology while maintaining the same underlying methodology.

This case, along with our main example 5.2, reveals a pattern where generated content appears to systematically reformulate existing research while maintaining core methodological similarities that are difficult to detect through conventional review processes.

B Example Case from Synthetic Dataset

The SSAG method successfully detects plagiarism in one of our synthetic test cases, where a research proposal is deliberately plagiarized from "LongRecipe: Recipe for Efficient Long Context Generalization in Large Language Models" (Hu et al., 2024b). The plagiarized version is shown in Table 8. When processed through the LLM Semantic Scholar RAG pipeline, the system successfully retrieves the original paper and conducted a one-to-one comparison. Based on its analysis of similari-

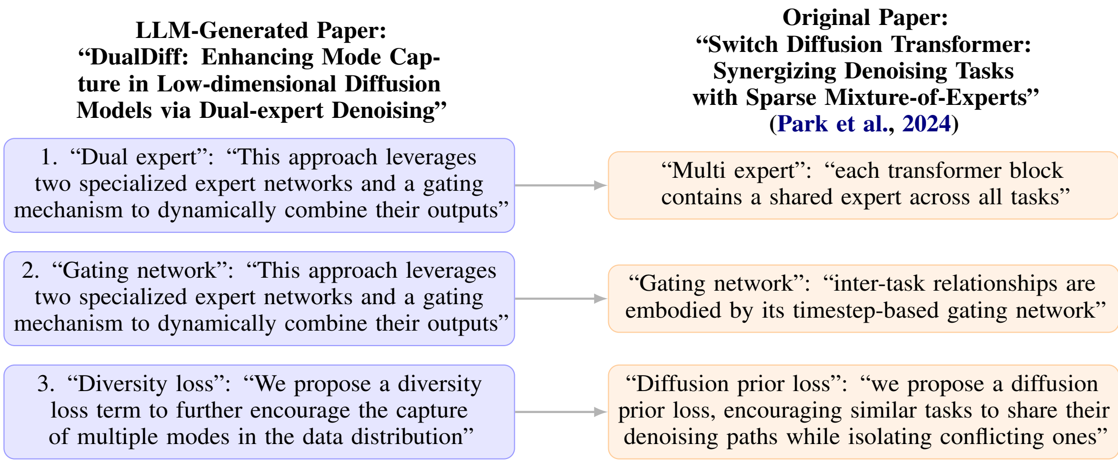

Figure 4: Visual mapping between an LLM-generated research proposal (an exemplar in Lu et al. (2024a)) and its corresponding source paper, showing direct correspondence between analogous methodology components. This pair received a similarity score of 5 in our expert evaluation, which is verified by the authors of Park et al. (2024).

ties and differences between the works, presented in Table 9, the LLM correctly concludes that the proposal is plagiarized.

Expert Evaluation Instructions

Plagiarism definition:

Presenting work or ideas from another source as your own, with or without consent of the original author, by incorporating it into your work without full acknowledgement.

Scoring guidelines:

Score 5: One-to-one mapping between the LLM proposed methodology and existing methods in one or two closely related prior papers.

Score 4: A significant portion of LLM proposed method is a mix-and-match from two-to-three prior works.

Score 3: The LLM proposed method bears decent similarity with some existing methods, but there's no exact correspondence with a limited set of papers.

Score 2: The LLM proposal bears very slight resemblance with some existing papers. Mostly novel.

Score 1: The LLM proposal is completely novel.

Note: Two detailed example evaluations were provided to participants, including LLM proposals, corresponding source papers, plagiarism scores, and detailed explanations. These examples demonstrated how to thoroughly document and justify plagiarism assessments.

Practical tips on finding plagiarism quickly

1. You (usually) only need to read first 4 sections of the proposal (Title, Problem Statement, Motivation, Proposed method). The proposed method section is most relevant in identifying plagiarism. Any other sections apart from these 4 are usually irrelevant.

2. https://openscholar.allen.ai/ is sometimes quite useful in identifying plagiarism. Use the template: "Check for prior work: {summary of 'Proposed Method' section of the LLM proposal}."

Important notes:

1. You only need to find plagiarism in any and only 3 LLM proposals. We have provided 5 LLM proposals so that you can pick any 3 you like.

2. The original papers would be published on or before April 2024 (cut-off date for Claude).

3. If you find some paper which is a very close match but published after April 2024, please put it in "Any other comments" section.

4. If some proposal seems absurd or completely-infeasible you can mark it as "absurd", but provide a brief paragraph to explain the same.

5. The correct mindset while looking for plagiarism would be: "I am suspicious of LLMs being able to generate novel research. Hence, its likely some rephrased version of existing research. Let me find the original paper to prove this point".

6. The hypothesis of our study is that LLM-generated research proposals contain significant traces of plagiarism. To support this thesis, please focus on identifying instances of strong plagiarism (score 4 or above), as scores below 4 are considered ambiguous. All proposals receiving scores of 4 or above will be verified with the original papers' authors.

Table 5: Instructions provided to expert participants for evaluating potential plagiarism in LLM-generated research proposals. These instructions deliberately encourage a skeptical approach, which may introduce confirmation bias. However, the objectivity is maintained through author verification and our open-sourced results that allow independent examination of source and generated works. See §4.1 and §8 for detailed discussion.

Plagiarized Research Proposal Generation Prompt

Your task is to craft a sophisticated research proposal that captures the essence of the following paper, while being artfully disguised to avoid direct detection. The research proposal should be most skillfully plagiarized from the original paper such that even on direct comparison an expert shouldn't be able to find the plagiarism.

Original Paper Details:

Title: `{paper title}`

Abstract: `{paper abstract}`

Research Proposal Guidelines:

1. Develop a novel-sounding title that subtly reflects the original paper's core contribution
2. Reframe the research problem using different terminology, keywords, perspective and motivation
3. Reformulate the methodology using alternative technical language and approach. You may use superfluous language to make the methodology appear original and compelling
4. Skillfully paraphrase the fundamental scientific insight and contribution of the original work
5. Ensure the proposal sounds like an original, innovative research concept
6. Avoid direct textual overlap or obvious copying
7. Use academic language that sounds fresh, compelling and confident

Specifically, structure your proposal with:

- A provocative, original-sounding title
- Clear problem statement
- Unique motivation that echoes the original paper's core motivation
- Innovative methodology that fundamentally matches the original paper's approach
- Expected scientific or practical contributions

Your goal is to create a proposal that would be extremely difficult to trace back to the original paper, yet captures its scientific essence and methodology completely. Now provide the research proposal without any additional unrelated text.

Table 6: The prompt used to generate plagiarized research proposals from existing papers.

OpenScholar Search Prompt

Check for prior research work closest to the following:

`{research proposal}`

Table 7: The OpenScholar search prompt used to find similar existing research.

Sample Generated Plagiarized Research Proposal
Title: Contextual Horizons: A Scalable Framework for Expanding Long-Sequence Comprehension in Advanced Language Models

Problem Statement: Contemporary large-scale language models (LLMs) exhibit remarkable proficiency in a wide range of natural language processing tasks. However, their ability to process and generalize over extended sequences remains constrained by the limited effective context length available during pretraining. This limitation hampers their performance on tasks requiring long-range dependency understanding, such as document summarization, multi-turn dialogue, and code generation. While extending the context window post-pretraining is a potential solution, it demands substantial computational resources, making it impractical for many research and deployment scenarios. Thus, there is a pressing need for an efficient and scalable approach to enhance the long-context capabilities of LLMs without compromising their general-purpose performance or incurring prohibitive computational costs.

Motivation: The ability to process long sequences is critical for advancing the utility of LLMs in real-world applications, where tasks often involve extended textual inputs. Current methods for increasing context length typically rely on full-sequence training, which is computationally expensive and resource-intensive. This inefficiency creates a significant barrier to democratizing access to high-performing LLMs with extended context capabilities. A novel, resource-efficient training paradigm is required to bridge this gap, enabling LLMs to effectively handle long-context tasks while maintaining their versatility and accessibility. By addressing this challenge, we can unlock new possibilities for LLMs in domains such as legal document analysis, scientific literature review, and large-scale conversational AI.

Methodology: We propose a novel training framework, termed **Contextual Horizons**, designed to extend the effective context window of LLMs in a computationally efficient manner. The framework is built on three core innovations:

1. **Dynamic Sequence Simulation:** Instead of directly training on full-length sequences, we introduce a dynamic sequence simulation mechanism that emulates long-context scenarios using strategically sampled token subsets. This approach ensures that the model is exposed to long-range dependencies without the need for full-sequence processing, significantly reducing computational overhead.

2. **Adaptive Positional Encoding Transformation:** To enable the model to generalize over extended contexts, we develop a positional encoding transformation strategy that reconfigures the positional indices of tokens during training. This transformation allows the model to reinterpret positional relationships in a way that aligns with longer context windows, enhancing its ability to capture dependencies across distant tokens.

3. **Resource-Aware Optimization Protocols:** We implement a suite of optimization techniques tailored to minimize memory and computational requirements during training. These include gradient checkpointing, selective parameter updates, and efficient batching strategies, which collectively reduce the resource demands of the training process while preserving model performance.

The proposed framework is designed to be compatible with a wide range of LLM architectures and can be applied to both open-source and proprietary models.

Expected Contributions: The **Contextual Horizons** framework is anticipated to deliver the following key contributions:

1. **Extended Contextual Understanding:** By leveraging dynamic sequence simulation and adaptive positional encoding, the framework enables LLMs to process and generalize over sequences up to 128k tokens in length, a significant improvement over the typical 8k-token limit.

2. **Resource Efficiency:** The proposed approach achieves long-context training with only 30% of the target context window size and reduces computational resource requirements by over 85% compared to conventional full-sequence training methods.

3. **Preservation of General Capabilities:** Unlike many specialized training techniques, the framework ensures that the enhanced long-context capabilities do not come at the expense of the model's performance on general NLP tasks.

4. **Practical Accessibility:** The framework is designed to be implementable on modest hardware setups, enabling researchers and practitioners to extend the context capabilities of LLMs using a single GPU with 80GB memory in as little as one day of training.

By addressing the critical challenge of long-context generalization in LLMs, **Contextual Horizons** has the potential to significantly advance the state of the art in natural language processing, making high-performing, long-context LLMs more accessible and practical for a wide range of applications.

Table 8: A research proposal generated by GPT-4o via the prompt in Table 6, plagiarizing Hu et al. (2024b).

Plagiarism Analysis
Let me analyze both works: **1. Proposal Summary:** The "Contextual Horizons" proposal addresses LLM context length limitations through a framework comprising dynamic sequence simulation, adaptive positional encoding transformation, and resource-aware optimization protocols. It aims to extend context windows efficiently while maintaining general capabilities and reducing computational costs. **2. Related Paper Summary:** LongRecipe presents a training strategy for extending LLM context windows through impactful token analysis, position index transformation, and training optimization strategies. It achieves extended context processing while using reduced computational resources. **3. Key Similarities and Differences:** **Similarities:** • Both target the same core problem of extending LLM context windows efficiently • Both use position encoding transformation techniques • Both claim similar performance metrics (30% context window size, 85% resource reduction) • Both advertise identical specific capabilities (8k to 128k tokens, single 80GB GPU, one day training) **The similarities are striking and extend beyond coincidence:** • The exact same numerical claims (30%, 85%, 8k to 128k) • Identical hardware specifications • Same training duration claims • Nearly identical methodology structure, just with renamed components **The proposal appears to be a direct reformulation of LongRecipe, with mainly cosmetic changes in terminology:** • "Dynamic sequence simulation" vs "Impactful token analysis" • "Adaptive positional encoding" vs "Position index transformation" • "Resource-aware optimization" vs "Training optimization strategies" The level of overlap in both technical approach and specific implementation details strongly suggests this is not an independent development but rather a repackaging of LongRecipe's work.

Table 9: Analysis by Claude 3.5 Sonnet comparing the deliberately plagiarized proposal in Table 8 with the original paper (Hu et al., 2024b), concluding plagiarism based on methodological similarities.