AAAR-1.0: Assessing AI's Potential to Assist Research

Anonymous NAACL submission

Abstract

Numerous studies have assessed the proficiency of AI systems, particularly large language models (LLMs), in facilitating everyday tasks such as email writing, question answering, and creative content generation. However, researchers face unique challenges and opportunities in leveraging LLMs for their own work, such as brainstorming research ideas, designing experiments, and writing or reviewing papers. In this study, we introduce AAAR-1.0, a benchmark dataset designed to evaluate LLM performance in three fundamental, expertise-intensive research tasks: (i) EQUATIONINFERENCE, assessing the correctness of equations based on the contextual information in paper submissions; (ii) EXPERIMENTDESIGN, designing experiments to validate research ideas and solutions; and (iii) PAPERWEAKNESS, identifying weaknesses in paper submissions. AAAR-1.0 differs from prior benchmarks in two key ways: first, it is explicitly research-oriented, with tasks requiring deep domain expertise; second, it is researcher-oriented, mirroring the primary activities that researchers engage in on a daily basis. An evaluation of both open-source and proprietary LLMs reveals their potential as well as limitations in conducting sophisticated research tasks. We will release the AAAR-1.0 and keep iterating it to new versions.

1 Introduction

Although AI has brought transformative changes to various aspects of life, its impact on researchers unfolds in a nuanced manner. On the one hand, AI assists in various research disciplines, such as Social Science, Finance, Medicine, GeoScience, Math, etc.(Yue et al., 2023; Li et al., 2023b), significantly expediting academic processes. However, many of these applications are superficial, often limited to data-driven clustering or classification. On the flip side, the AI era poses challenges for researchers. Despite its ability to streamline some activities,

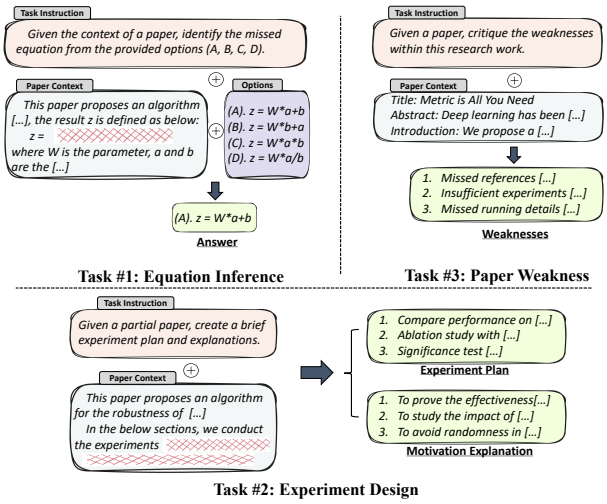


Figure 1: The input-output illustration of three tasks in the proposed AAAR-1.0 benchmark.

researchers still face demanding, cognitively intensive tasks such as staying current through extensive paper reading, rapidly generating ideas in response to fast-paced advancements, conducting rigorous experiments to substantiate claims, and managing an increasing volume of peer reviews. Then a question looms: How effectively can AI assist researchers in tasks that are domain-specific, expertise-demanding, and knowledge-intensive?

Existing works proved the promising potential for using LLMs in assisting AI research. For example, Si et al. (2024) conducted a large-scale human study and found that LLMs can generate creative research ideas. Lu et al. (2024) proposed an autonomous agent to handle complicated research workflow and write a whole research paper. However, most of these works focus on addressing highly professional and subjective problems, making evaluation laborious and hard to reproduce. This underscores the need for a comprehensive benchmark that rigorously assesses LLMs' capabilities in expertise-intensive research activities

To this end, in this work, we introduce AAAR-1.0, a novel benchmark that aims to comprehen-

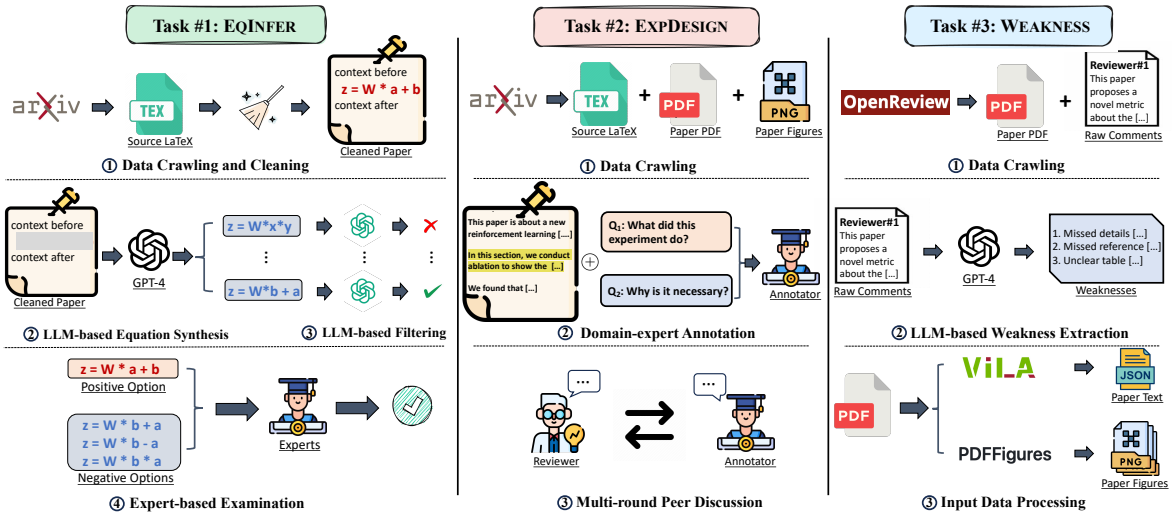


Figure 2: Data construction workflows of the three tasks in AAAR-1.0.

sively assess the LLMs' capacity on expert-level research tasks. As illustrated in Figure 1, AAAR-1.0 decomposes three distinct expert-level AI research tasks from the researcher's daily activities, including i) EQUATIONINFERENCE, investigating whether the LLMs can infer the equation correctness based on the paper context; ii) EXPERIMENTDESIGN, validating LLMs' ability on designing reliable experiments for a research idea; iii) PAPERWEAKNESS, testing the quality of the weaknesses criticism written by the LLMs. To ensure data quality, senior AI researchers with extensive domain expertise perform data annotation for AAAR-1.0, followed by rigorous multi-round data examination and filtering. All three tasks require models to possess strong domain knowledge covering various cutting-edge research findings, as well as expert-level research experience, to the extent that even humans need substantial research accumulation to tackle the tasks we designed. Crucially, tasks here are singular, stand-alone challenges (with clear input and output expectations) rather than a complicated task chain (Li et al., 2024; Lu et al., 2024), providing a more transparent assessment of the model's intermediate output.

Benefiting from the proposed automatic metrics, we conduct extensive experiments across numerous mainstream LLMs, where we find that:

- Closed-source LLMs generally outperform open-source LLMs on AAAR-1.0, likely due to their richer scientific knowledge stemming from a larger model size.
- Contrary to human behaviour, neither extending the input modality (i.e., leveraging text and figures) nor enlarging the input context guarantees enhanced performance. This underlines most current LLMs' limitations in processing diverse, extensive information coming from scientific documents.
- LLM-designed experiments are innovative and more diverse than those by humans; however, many are trivial, lack feasibility, and stray from the original research objectives.
- LLM-generated weaknesses often lack ample domain knowledge, especially on cutting-edge research topics, leading to the vague weaknesses applicable to various papers.

2 AAAR-1.0

Figure 2 provides an overview of constructing AAAR-1.0. In the following sections, we elaborate on the data collection details of the aforementioned three tasks, including § 2.1 EQUATIONINFERENCE (EQINFER), § 2.2 EXPERIMENTDESIGN (EXPDESIGN), and § 2.3 PAPERWEAKNESS (WEAKNESS).

2.1 EQUATIONINFERENCE

Writing a correct scientific equation is challenging because it involves an in-depth understanding towards an algorithm or the relations between the massive notations. However, directly asking LLMs to generate equations is over-challenging. For this reason, in this task, we adopt the conventional multi-choice classification paradigm for building EQINFER, as shown in Figure 1.

① **Data crawling and cleaning.** For the data source, we adopt the pre-compilation LaTeX code for two reasons: i) existing PDF parsing tools, such as PyMuPDF and PaperMage (Lo et al., 2023), can

introduce considerable noise to the parsed equation text; ii) considering most of exiting LLMs are capable with processing LaTeX code, using LaTeX source instead of parsed text can be more accurate and provide LLMs with richer information. Meanwhile, to avoid using any low-quality human-written equations, we only crawl those peer-reviewed papers accepted by top-tier conferences. Accordingly, we first obtain the accepted paper list from ACL Anthology, from year `2019` to `2023`. Next, we search each paper on arXiv to crawl its LaTeX source (if it exists). Finally, we get a total of 1,762 papers' source LaTeX packages.

We then clean the LaTeX sources by deleting all the comments and combining multiple cross-referred `.tex` files into a main file. Afterwards, we use regex to randomly extract (at most) 3 equations' code snippets per paper, finally resulting in 3,877 human-written equations are extracted.

② **LLM-based equation synthesis.** As we formulate this task as classification, for each human-written positive equation, we have to craft at least three counterpart negative equations. To this end, we prompt GPT-4 to synthesize more equations based on the paper context. For each positive equation, we repeat this prompting (with a high decoding temperature) until three different negative equations are synthesized.

③ **LLM-based filtering.** However, the LLM-synthetic equations can sometimes be context-unaligned, i.e., some synthesized equations contain notations that are never defined in the paper context, which is a superficial shortcut for the classification tasks (Geirhos et al., 2020). To improve the data quality, we prompt GPT-4 to identify those context-unaligned negative equations. We then discard those instances where all three negative equations are identified as contextually unaligned. This filtering leads to a final of 1,449 classification instances (62.3% instances are filtered).

④ **Expert-based examination.** Furthermore, it's also possible that synthesized negative equations are actually correct (i.e., false negative options) — even if the negative and positive equations are written differently, the final compiled results might be the same. To filter out the false negative equations and to have a final check on the classification instances, we then employ human experts to conduct a further data review.

We asked 5 senior PhD students who are experienced in AI research to manually check all the instances. For each classification instance, we ask human experts to consider the following criteria: i) **are all four equations (both positive and negative) grammatically correct?** ii) **after compilation, is there only one correct answer?** We ask every human expert to use external LaTeX compilation tools (e.g., TeXlive), and identify the instances that cannot meet the criteria. Each instance is examined by at least two experts, and we only keep instances that all experts decide to keep. After this strict examination, a total of 1,049 instances are eventually kept (27.6% instances are filtered)

Final data. We finally shuffle the four equations for each classification instance and randomly assign letters (`A`, `B`, `C`, and `D`) to the equations. We show the data statistics of the final EQINFER in Table 4 and the sample data cases in Appendix E.5.

2.2 EXPERIMENTDESIGN

Given a research topic, such as a novel ML algorithm, a qualified researcher can design a solid experiment plan for it, and clarify underlying motivation to ensure the reliability of the designed experiment. Unlike the concurrent works that focus on the experiment implementation (Lu et al., 2024; Huang et al., 2024), we emphasize the importance of assessing the high-level experiment design of LLMs before the subsequent implementation to avoid any expensive execution iteration. Therefore, as shown in Figure 1, we formulate EXPDESIGN as a text-generation task that takes pre-experiment paper context as input, and then generates the experiment and explanation list.

① **Data crawling.** As for the data source, we first collect $\geq$ 10k papers' data from arXiv, including LaTeX sources and PDFs, which cover broad AI categories, including `cs.AI`, `cs.CL`, and `cs.CV`, from year `2018` to `2023`. Similarly, to ensure the source data quality, we only use papers that have appeared at well-known conferences.

② **Domain-expert annotation.** Making a reliable and executable experiment plan requires solid foundation knowledge of a specific research area. Consequently, we set a high standard for choosing annotators: i) be a senior PhD student with at least one peer-reviewed publication; ii) have more than 4 years of AI research experience; iii) frequently serve as conference reviewers. Finally, we invite a total of 10 qualified experts to participate in our

data collection procedure. Given the 10k crawled papers, we first ask every annotator to bid on the papers that they are interested in. After bidding, each of them is assigned 10 papers by us, i.e., a total of 100 papers to be annotated. During annotation, we post each paper PDF on online Google Drive and ask the annotator to first carefully read the whole paper. Then, we ask them to identify and locate the key experiments in each paper (i.e., highlighting the relevant paragraphs of each experiment). We don't consider some trivial experiments, such as those supplemental analyses in the appendix section. For each identified experiment, the annotator has to concisely answer two questions: i) **what did this experiment do?** ii) **why did the paper authors conduct this experiment?** In other words, we ask the annotator to summarize all the key experiments in this paper and explain the underlying motivations based on their rich domain experience.

③ **Multi-round peer discussion.** Intuitively, different experts might have different opinions on the same research topic. Particularly, when explaining the underlying motivation of an experiment, adopting only a single expert's opinion might introduce bias to our annotation. Hence, we conduct a further multi-round peer discussion. For each online paper PDF, where all the key experiments are identified, summarized, and explained, we ask a different expert (reviewer) to review the annotation by considering the following three criteria: i) **are the identified experiments all the key experiments?** ii) **does each experiment summarization covers all key information?** iii) **does each explanation sound reasonable and reliable?** Each reviewer has to leave comments to the online PDF regarding the above criteria, and then the annotator has to respond to each comment — either accept the suggestion and revise the previous annotation, or provide a "rebuttal" to the reviewer to uphold the annotation. This discussion iterates until both opinions align with each other. Eventually, for each paper, we collect two lists: i) the experiment list, summarizing each experiment step of the paper; ii) the explanation list, the underlying motivations that are one-one corresponding to the experiment.

Final data. After annotation, we use the pre-experiment context of each paper (according to the first-experiment location identified by the annotator) as the input. Furthermore, we use GPT-4 to delete any sentence that potentially leaks the experiment from the input.[1] Similar to the EQINFER, we utilize the source LaTeX as the input text to avoid PDF paring noise. As for the image input, we collect those figures within each paper's source LaTeX package and only keep figures that are used in the pre-experiment context. Overall, a total of 100 instances are collected. As shown in Figure 1, the input of each instance is the pre-experiment context (including the figures), and the ground-truth output is the expert-annotated experiment plan and the explanations. Table 5 shows data statistics.

2.3 PAPERWEAKNESS

Another critical research task is paper review. Previous works have demonstrated the usefulness of the LLM-based review feedback (Gao et al., 2024; Jin et al., 2024; Lu et al., 2024). However, as indicated by Du et al. (2024); Liang et al. (2024), LLMs only excel at summarizing the research strengths while falling significantly short on weakness criticism. Hence, we build WEAKNESS for particularly investigating the LLM-generated weaknesses.

① **Data crawling.** We first crawl a total of 3,779 anonymous submissions of *ICLR 2023* from OpenReview,[2] including PDF and other meta information (e.g., scores, decisions, and tracks). As the *ICLR 2023* has 13 distinct tracks while the paper distribution across different tracks is highly biased, we then uniformly sample papers from different research tracks to improve the domain diversity. Meanwhile, during sampling, we also keep the accept/reject papers distributed equally to avoid data bias. In a word, we finally collect a total of 1,000 papers (500 accepted; 500 rejected), uniformly covering all 13 tracks. Please refer to Figure 3 for the track and score distribution of the 1,000 papers.

② **LLM-based weakness extraction.** Since the raw comments crawled from *ICLR 2023* are mixed with both strengths and weaknesses, we further employ GPT-4 to extract all the weaknesses from each reviewer's comments and compose multiple weaknesses into a list. Notably, we force GPT-4 to keep the original text of the reviewer, i.e., all weaknesses in our dataset are those original sentences written by the reviewer without any modifications.[3]

[1] About 9.8% sentences are deleted.

[2] We adopt ICLR because it releases full submissions, while some other conferences only release accepted papers.

[3] We manually checked GPT-4's extraction results of 200 cases — GPT-4 only missed $\leq$1% of reviewer-written weaknesses and maintained almost all the original text.

What's more, sometimes one reviewer might repeatedly mention the same weakness throughout the comment. In this case, we simply keep all the repeated weaknesses because, if one weakness is repeatedly mentioned by the reviewer, it's intuitively an important weakness that the reviewer wants to emphasise; accordingly, keeping the repeat items can penalize LLMs more on missing this weakness.

All in all, for each paper, we can finally get multiple weakness lists (one weakness list per reviewer, one paper can have multiple reviewers). We further delete a few papers without any weaknesses found in the raw comments, resulting in a total of 993 instances, i.e., 993 {paper, weakness lists} pairs.

③ **Input data processing.** As we mentioned before, we crawl papers from OpenReview instead of arXiv because the under-review paper draft is required for this task. However, not every paper from OpenReview can be found on arXiv, i.e., the source LaTeX code and figures of most under-review papers are unavailable. Therefore, we utilize VILA (Lin et al., 2023) to parse text data out from the PDF; we also employ PDFFigures-2.0 (Clark and Divvala, 2016) to extract all the figures and tables (in image) from the paper, as Vila is not good at processing the table data.

Final data. Our final data is composed of 993 instances, each input is paper text along with figure/table images, and each output is peer reviewers' weakness lists. Table 6 shows data statistics.

3 Evaluation Criteria

For EQINFER, we adopt accuracy as the classification criterion. For EXPDESIGN and WEAKNESS, since both tasks have natural language outputs, semantic-based metrics are necessary. Hence, in addition to the conventional ROUGE (Lin, 2004), we also develop several novel similarity-based metrics for each specific task, including:

- **S-F**$_1$ (equation 1 and 2): similarity-based F_1 for assessing the experiment design quality. It measures how well each model-generated experiment aligns with the human experiments.
- **S-Match** (equation 3): "soft" match score for evaluating the explanation. It calculates the similarity between human and model-generated explanations.
- **SN-F**$_1$ (equation 4 and 5): updated version of S-F_1 to deal with the "nested" review weaknesses.
- **ITF-IDF** (equation 6): inspired by the classic TF-IDF; measures the inter- and intra-paper diversity of model-generated weaknesses.

We sincerely recommend referring to Appendix B for the formal equation definitions of the above metrics.

4 Experiments and Analyses

In this section, we conduct extensive experiments on AAAR-1.0, across various popular LLMs, to quantify the current LLMs' capacity to tackle high-level research tasks. Specifically, § 4.1 for EQINFER, § 4.2 for EXPDESIGN, and § 4.3 for WEAKNESS. Please refer to Appendix D.2 for running details of the LLMs.

4.1 EQUATIONINFERENCE

Settings. As different LLMs have distinct context windows, to ensure a fair comparison, we fix the maximum input length for all models. According to the data statistics of Table 4, we empirically use 1,000 words for both contexts before and after equations, i.e., 2,000 surrounded words.

Main results. Table 1 shows the main results. Firstly, the open-source LLMs, especially the Falcon and Gemma, perform unexpectedly disappointing (even worse than random guesses). These weird scores are mainly due to the poor long-context instruction following ability, where we find some open-source LLMs are confused with the massive input and often copy the LaTeX code from the input. In contrast, closed-source LLMs generally achieve superior accuracy, probably owing to the richer scientific knowledge from the larger model parameters. However, considering the conventional multi-choice QA formulation of EQINFER, the recently-released GPT4-o solely gets 43.18, implying the unique challenge of EQINFER compared with other scientific QA benchmarks (Song et al., 2023). Notably, with the help of internal CoT, o1 gains stronger performances than GPT-4/GPT-4o, indicating the potential benefits of adopting reasoning for this task.

$\mathcal{Q}$: do more contexts boost performance? Table 1 unifies the input context lengths to 1,000 words for various LLMs. In this paragraph, we experiment with long-context LLMs to investigate the impact of the input context lengths. Particularly, we scale the input length (per side) from 100 to 1,500 words. As shown in Figure 4, for the open-source

Methods	Accuracy (%)
Random Guess	25.00
Open-source LLMs	
OLMo-7B (Groeneveld et al., 2024)	19.00
Falcon-40B (Almazrouei et al., 2023)	4.39
Gemma 2-27B (Gemma Team, 2024)	3.24
Mistral-7B (Jiang et al., 2023)	22.21
Mixtral-8x22B-MoE (Jiang et al., 2024)	37.08
Llama 3.1-70B (MetaAI, 2024)	38.13
Qwen 2.5-72B (Qwen Team, 2024)	35.93
Closed-source LLMs	
Gemini 1.5 Pro (Anil et al., 2023)	34.31
Claude 3.5 sonnet (Anthropic, 2024)	**61.10**
GPT-4 (OpenAI et al., 2023)	49.85
GPT-4o (OpenAI, 2024a)	43.18
o1-preview (OpenAI, 2024b)	59.49

Table 1: Various LLMs' performances on the 1,049 instances of EQINFER task.

LLMs (Llama and Qwen), after 300 words length, increasing the input context doesn't help the performance and even significantly drops Qwen's scores. While for the closed-source GPT-4-Turbo and GPT-4o, scaling up input length gradually boosts the performances at the first 1,000 words, but stabilizes afterwards. This is in line with human intuition, i.e., surrounding context is required for the equation inference, as the adjacent context usually provides important information, such as the target algorithm description or the notation definition. However, after exceeding a specific threshold, more context information is not beneficial anymore and even confuses those LLMs with poor long-context handling capacity (Wang et al., 2024; Liu et al., 2024).

4.2 EXPERIMENTDESIGN

Settings. Similarly, we unify the input context length of different LLMs to ensure a fair comparison. We empirically set 2,000 and 3,000 input words for open- and closed-source LLMs, respectively. In the meantime, as motivation explanation is the subsequent task of experiment design, using model-generated experiments can propagate errors in explanation, leading to inferior results for most LLMs. To this end, we provide LLMs with the oracle experiments when generating explanations.

Main results. Table 2 shows the main results. For the experiment design, the closed-source LLMs generally outperform open-source LLMs, and both closed-/open-source LLMs are superior to the "Copy Input" baseline (except the Falcon). Despite the higher S-Precision, the open-source LLMs are seriously deficient in S-Recall compared with closed-source LLMs ($\sim$10%$\downarrow$). We find that closed-source LLMs are more creative in experiment design and tend to generate more experiment ideas than open-source LLMs (though most of the experiment ideas are trivial), leading to excellent S-Recall. As for the motivation explanation, the S-Match scores of closed-source LLMs still surpass the open-source LLMs, while the score difference is not significant. Furthermore, we find the negative correlation between S-Match and the ROUGE, where the ROUGE scores of closed-source LLMs are broadly inferior. We find that the open-source LLMs often try to copy the terms or phrases from the given experiment, or even simply paraphrase the experiment instead of explaining, which results in a high superficial overlap with the ground-truth explanation. This observation highlights the importance of adopting the proposed S-Match to avoid evaluation bias of traditional generation metrics.

$\mathcal{Q}_1$: can self-contained experiments enhance the explanation of motivation? When generating the explanation in Table 2, we provide LLMs with each individual experiment and let them explain one by one, because we find that, when providing the whole experiment list, those open-source models only explain partial experiments because of their poor instruction-following capacity. However, there are intuitively some semantic or logical relations between different experiments, e.g., some experiments are prerequisites to others. Therefore, this one-by-one prompting might break the self-containment of an experiment plan. Consequently, we test with the "whole-list" prompting, where the LLMs are given the complete experiment list and are asked to explain all experiment steps together.

As shown in Table 8, unlike the open-source LLMs, the explanation performances of those closed-source LLMs are generally improved after adopting whole-list prompting. According to further manual checking, after maintaining the self-containment of the experiments, the LLMs can refer to other experiments and better grasp the underlying motivation of the current experiment.

$\mathcal{Q}_2$: do human evaluation results align with automatic metrics for explanation? As the explanation can be open-ended, in this paragraph, we provide the human evaluation results on different LLMs' motivation explanation outputs. In detail, we randomly select 20 out of 100 papers and ask 5 annotators to read the experiments along with each model's explanations; we then let the annotator decide whether each model's explanation is acceptable (see Appendix E.3 for more details). Table 9

Methods	Experiment Design			Motivation Explanation		
	S-F$_1$	S-Precision	S-Recall	S-Match	ROUGE-L	ROUGE-1
Copy Input	21.13	17.94	26.76	40.32	22.06	25.28
Open-source LLMs						
OLMo-7B (Groeneveld et al., 2024)	33.94	37.25	31.79	45.78	26.30	30.38
Falcon-40B (Almazrouei et al., 2023)	17.87	21.78	15.35	17.03	12.10	12.72
Gemma 2-27B (Gemma Team, 2024)	34.33	39.71	30.51	42.77	26.20	29.63
Mistral-7B (Jiang et al., 2023)	37.62	43.09	34.19	50.18	**30.20**	34.69
Mixtral-8x22B-MoE (Jiang et al., 2024)	42.21	50.13	36.82	49.07	29.96	34.53
Llama 3.1-70B (MetaAI, 2024)	40.57	48.43	35.43	50.05	29.33	34.11
Qwen 2.5-72B (Qwen Team, 2024)	43.24	**51.73**	37.55	51.12	29.46	34.68
Closed-source LLMs						
Gemini 1.5 Pro (Anil et al., 2023)	51.87	50.77	53.37	52.87	28.52	33.80
Claude 3.5 sonnet (Anthropic, 2024)	48.74	46.49	51.53	53.03	18.75	26.15
GPT-4 (OpenAI et al., 2023)	43.89	42.34	45.82	55.03	22.82	30.01
GPT-4o (OpenAI, 2024a)	**53.00**	51.24	**55.12**	54.79	27.54	34.31
o1-preview (OpenAI, 2024b)	46.67	45.04	48.70	**58.55**	29.11	**36.70**

Table 2: Various LLMs' performances on the 100 instances of EXPDESIGN. The motivation explanation is based on the oracle experiments to prevent error propagation. "Copy Input" is a random baseline: for experiment design, randomly select 5 sentences from the input paper; for motivation explanation, directly copy each experiment idea.

illustrates the results, where the score variance is higher than Table 2. However, the performance ranking of both tables is perfectly correlated with each other (Pearson coefficient = 1), demonstrating the effectiveness of S-Match.

$\mathcal{Q}_3$: do more contexts boost performance? We also investigate the impact of input context length for EXPDESIGN. As shown in Figure 5, we scale up the input pre-experiment context length from 0.1k to 10k words (10k words is the maximum paper context length in the dataset). For the experiment planning, more input context does improve the performance of different LLMs, while this benefit stops after exceeding 5k words, which is similar to EQINFER's scaling results — after the necessary information has been covered, scaling more up doesn't boost the performance. Meanwhile, the results of the motivation explanation demonstrate that explaining motivations almost doesn't require any paper context, i.e., the LLMs solely rely on the given experiments. However, we do not expect this because we hope LLMs can explain the motivation based on a thorough understanding of the paper, just like how human experts do. Hence, there is still a considerable gap between the LLMs and humans in terms of grasping research motivations.

$\mathcal{Q}_4$: does multi-modal input boost performance? Intuitively, besides the text, when designing experiments for a given research topic, the figures can provide rich supplementary information, such as an algorithm illustration that can help better understand this research topic and underlying motivations. Hence, we test different MLLMs' performances, including GPT4-o, GPT-4, and InternVL2 (Chen et al., 2024a). Table 10 shows the ablation results on the figure data. To our surprise, the figure data doesn't improve the MLLMs' results in this task, even harming the performances. This might be due to the low informativeness of the figures, as figures usually consume more input tokens but act only as supplementary information to the text, indicating future work on developing MLLMs that can effectively leverage the scientific figures.

4.3 PAPERWEAKNESS

Settings. Intuitively, the full paper context is necessary for conducting a review. Therefore, instead of setting a maximum input length, in WEAKNESS, we try to feed all the paper context into the LLMs. As the input length of WEAKNESS is extremely long (see Table 6), we adopt a "split-combine" method — we first split the whole paper into several smaller pieces and let LLMs predict the weaknesses of each piece separately; after that, we combine all pieces' weaknesses as a final complete prediction. In practice, for the length of each small piece, we set 2,000 and 3,000 words for open- and closed-source LLMs, respectively. Additionally, in this task, we also examine the performance of a recent agent framework, i,e., AI-SCI (Lu et al., 2024), which enhances GPT-4o's paper review ability by leveraging advanced prompting techniques, e.g., self-reflection (Shinn et al., 2024) and response ensembling (Wang et al., 2023).[4]

[4]We don't run AI-SCI on EXPDESIGN, as it takes model-generated ideas as the inputs, which are incompatible with our task setting.

Methods	SN-F$_1$ (%)	SN-Precision (%)	SN-Recall (%)	Review Diversity ITF-IDF (↑)
Peer Review	—	—	—	7.69
Open-source LLMs				
OLMo-7B (Groeneveld et al., 2024)	43.25	40.38	47.04	2.45
Falcon-40B (Almazrouei et al., 2023)	27.34	25.13	30.88	1.06
Gemma 2-27B (Gemma Team, 2024)	35.85	34.68	37.91	1.43
Mistral-7B (Jiang et al., 2023)	42.03	43.80	40.77	1.17
Mixtral-8x22B-MoE (Jiang et al., 2024)	43.23	**44.59**	42.23	0.98
Llama 3.1-70B (MetaAI, 2024)	42.78	43.19	42.70	2.60
Qwen 2.5-72B (Qwen Team, 2024)	42.74	43.80	42.05	1.21
Closed-source LLMs				
Gemini 1.5 Pro (Anil et al., 2023)	**48.75**	43.97	55.08	5.88
Claude 3.5 sonnet (Anthropic, 2024)	47.85	41.97	56.00	3.91
GPT-4 (OpenAI et al., 2023)	47.66	42.15	55.19	5.31
GPT-4o (OpenAI, 2024a)	47.73	42.09	55.48	**5.95**
o1-preview (OpenAI, 2024b)	48.62	42.54	**57.08**	5.63
LLM Agent Framework				
AI-SCI (GPT-4o) (Lu et al., 2024)	45.05	40.02	51.91	2.23

Table 3: Various LLMs' performances on the 993 instances of WEAKNESS.

Main results. Table 3 shows the main results, where the closed-source LLMs' overall performances are generally superior to the results of open-source LLMs. Similarly, closed-source LLMs are particularly excellent in SN-Recall because of more generated weaknesses. However, there is still a considerable gap in the weakness diversity between the LLMs and human experts.[5] Compared with human review, most LLM-generated weaknesses are vague and lack the necessary knowledge about some frontier research works. Surprisingly, AI-SCI performs worse than backbone GPT-4o, especially on ITF-IDF, which might indicate the challenge of WEAKNESS, i.e., simply adopting popular prompting techniques cannot well address this task.

$\mathcal{Q}_1$: is the split-combine effective? Ideally, if a LLM has a sufficient context window size, its might not be that necessary to split the input papers for separate processing. Consequently, in this paragraph, we utilize the LLMs accepting long context input to compare "split-combine" with "no-split", i.e., letting LLMs write weaknesses by given the full paper. In practice, we set the maximum number of input words to 20k, which ensures $\geq$95% papers in the WEAKNESS can be fully processed. As shown in Table 7, compared with giving the full paper contexts, split-combine generally brings about superior performances. During manual checking, we find that, when full paper is available, LLMs frequently neglect some important sections and omit weaknesses accordingly, while split-combine ensures that the LLMs can carefully brainstorm weaknesses within each smaller piece. Surprisingly, the LLMs' performances with full paper context can even worse than just remaining the first 3,000 words. This might imply that even the current powerful long-context LLMs still fall short when processing long scientific documents.

$\mathcal{Q}_2$: does multi-modal input boost performance? Our dataset covers both tables and figure illustrations extracted from the paper PDF as inputs. Intuitively, when reviewing a paper, both figures and tables are critical, not only for a better understanding, but also because some weaknesses are related to tables/figures.[6] Therefore, in Table 11, we adopt two MLLMs to investigate the effectiveness of image inputs. Overall, images information, including both figures and tables, doesn't bring significant performance improvement, i.e., only InternVL2 gains a performance boost after incorporating figures; while tables slightly drop both models' results. This is probably because the MLLMs cannot reason well over the information-intensive images, especially the table images (Deng et al., 2024).

5 Conclusion

In this work, we propose AAAR-1.0, a novel benchmark targeting a comprehensive evaluation for the current LLMs' AI research capacity. We devise three distinct expertise-intensive tasks along with the curated evaluation metrics, and collect high-quality data by employing senior AI researchers. Extensive experiments highlight the challenges and values of AAAR-1.0.

[5] Note that the human's ITF-IDF score in Table 3 can be slightly underestimated. This is because we keep the repeated weaknesses in the human review, which affects the human review's informativeness (lower ITF) but is useful when calculating the SN-Recall for LLMs.

[6] We find that there is approximately one human-written weakness related to figures or tables in each paper.

Limitations

We shed light on two limitations of this work: i) since we collect data from open-sourced platforms like arXiv and Openreview, it's possible that the current or future LLMs might be trained on the source data that we are using in our benchmark, thus affecting the conclusions drawn from this paper (this is a common limitation of most existing LLM-oriented benchmarks). Although we acknowledge this potential data leakage, we still believe this work can provides positive inspiration, at least for now, for the design of current LLMs and agents and help the community better understand the drawbacks and strengths of using LLMs for scientific research; ii) meanwhile, due to the high consumption of employing senior researchers in conducting data annotation, the data size for some tasks, such as EXPDESIGN, is relatively small. This might introduce variability in LLM performance, even with repeated runs. As this work marks the beginning of a series, we plan to release larger-scale datasets that will cover more diverse research tasks in AAAR-2.0.

Ethics Statement

Our study highlights that LLMs can be used to assist humans in AI research. This doesn't mean we encourage the use of AI to replace human researchers. In contrast, we emphasize that the main body of conducting scientific research shall always be humans, while the LLMs are only tools for making human research more effective. To this end, we commit to careful distribution of the data collected in our research, ensuring it serves strictly for research purposes. Our goal is to mitigate the risks while maximizing the benefits offered by LLMs.

References

Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Merouane Debbah, Etienne Goffinet, Daniel Heslow, Julien Launay, Quentin Malartic, Badreddine Noune, Baptiste Pannier, and Guilherme Penedo. 2023. Falcon-40B: an open large language model with state-of-the-art performance.

Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Gemini Team, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Anthropic. 2024. Introducing claude 3.5 sonnet. https://www.anthropic.com/news/claude-3-5-sonnet.

Eric Chamoun, Michael Schlichktrull, and Andreas Vlachos. 2024. Automated focused feedback generation for scientific writing assistance. *arXiv preprint arXiv:2405.20477*.

Zhe Chen, Weiyun Wang, Hao Tian, Shenglong Ye, Zhangwei Gao, Erfei Cui, Wenwen Tong, Kongzhi Hu, Jiapeng Luo, Zheng Ma, et al. 2024a. How far are we to gpt-4v? closing the gap to commercial multimodal models with open-source suites. *arXiv preprint arXiv:2404.16821*.

Ziru Chen, Shijie Chen, Yuting Ning, Qianheng Zhang, Boshi Wang, Botao Yu, Yifei Li, Zeyi Liao, Chen Wei, Zitong Lu, et al. 2024b. Scienceagentbench: Toward rigorous assessment of language agents for data-driven scientific discovery. *arXiv preprint arXiv:2410.05080*.

Christopher Clark and Santosh Divvala. 2016. Pdffigures 2.0: Mining figures from research papers. In *Proceedings of the 16th ACM/IEEE-CS on Joint Conference on Digital Libraries*, pages 143–152.

Naihao Deng, Zhenjie Sun, Ruiqi He, Aman Sikka, Yulong Chen, Lin Ma, Yue Zhang, and Rada Mihalcea. 2024. Tables as texts or images: Evaluating the table reasoning ability of LLMs and MLLMs. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 407–426, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.

Jiangshu Du, Yibo Wang, Wenting Zhao, Zhongfen Deng, Shuaiqi Liu, Renze Lou, Henry Peng Zou, Pranav Narayanan Venkit, Nan Zhang, Mukund Srinath, et al. 2024. Llms assist nlp researchers: Critique paper (meta-) reviewing. *arXiv preprint arXiv:2406.16253*.

Zhaolin Gao, Kianté Brantley, and Thorsten Joachims. 2024. Reviewer2: Optimizing review generation through prompt generation. *arXiv preprint arXiv:2402.10886*.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. 2020. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673.

Dirk Groeneveld, Iz Beltagy, Pete Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Harsh Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, Shane Arora, David Atkinson, Russell Authur, Khyathi Chandu, Arman Cohan, Jennifer Dumas, Yanai Elazar, Yuling Gu, Jack Hessel, Tushar Khot, William Merrill, Jacob Morrison, Niklas Muennighoff, Aakanksha Naik, Crystal Nam, Matthew E. Peters, Valentina Pyatkin, Abhilasha Ravichander, Dustin Schwenk, Saurabh Shah, Will Smith, Nishant Subramani, Mitchell Wortsman, Pradeep Dasigi, Nathan Lambert, Kyle Richardson, Jesse Dodge,

Kyle Lo, Luca Soldaini, Noah A. Smith, and Hannaneh Hajishirzi. 2024. Olmo: Accelerating the science of language models. *Preprint*.

Qian Huang, Jian Vora, Percy Liang, and Jure Leskovec. 2024. Mlagentbench: Evaluating language agents on machine learning experimentation. In *Forty-first International Conference on Machine Learning*.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. 2024. Mixtral of experts. *arXiv preprint arXiv:2401.04088*.

Yiqiao Jin, Qinlin Zhao, Yiyang Wang, Hao Chen, Kaijie Zhu, Yijia Xiao, and Jindong Wang. 2024. Agentreview: Exploring peer review dynamics with llm agents. *arXiv preprint arXiv:2406.12708*.

Sandeep Kumar, Tirthankar Ghosal, Vinayak Goyal, and Asif Ekbal. 2024. Can large language models unlock novel scientific research ideas? *arXiv preprint arXiv:2409.06185*.

Yanis Labrak, Adrien Bazoge, Emmanuel Morin, Pierre-Antoine Gourraud, Mickael Rouvier, and Richard Dufour. 2024. Biomistral: A collection of open-source pretrained large language models for medical domains. *arXiv preprint arXiv:2402.10373*.

Haoyuan Li, Hao Jiang, Tianke Zhang, Zhelun Yu, Aoxiong Yin, Hao Cheng, Siming Fu, Yuhao Zhang, and Wanggui He. 2023a. Traineragent: Customizable and efficient model training through llm-powered multi-agent system. *arXiv preprint arXiv:2311.06622*.

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

Zekun Li, Wenxuan Zhou, Yao-Yi Chiang, and Muhao Chen. 2023b. Geolm: Empowering language models for geospatially grounded language understanding. *arXiv preprint arXiv:2310.14478*.

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

Chin-Yew Lin. 2004. Rouge: A Package for Automatic Evaluation of Summaries. In *Text summarization branches out*, pages 74–81.

Ji Lin, Hongxu Yin, Wei Ping, Yao Lu, Pavlo Molchanov, Andrew Tao, Huizi Mao, Jan Kautz, Mohammad Shoeybi, and Song Han. 2023. Vila: On pre-training for visual language models. *Preprint*, arXiv:2312.07533.

Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173.

Kyle Lo, Zejiang Shen, Benjamin Newman, Joseph Z Chang, Russell Authur, Erin Bransom, Stefan Candra, Yoganand Chandrasekhar, Regan Huff, Bailey Kuehl, et al. 2023. Papermage: A unified toolkit for processing, representing, and manipulating visually-rich scientific documents. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 495–507.

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

MetaAI. 2024. Introducing llama 3.1: Our most capable models to date. https://ai.meta.com/blog/meta-llama-3-1/.

OpenAI. 2024a. Hello gpt-4o. https://openai.com/index/hello-gpt-4o/.

OpenAI. 2024b. Introducing openai o1. https://openai.com/index/introducing-openai-o1-preview/.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

N Reimers. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2024. Reflexion: Language agents with verbal reinforcement learning. *Advances in Neural Information Processing Systems*, 36.

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*.

Linxin Song, Jieyu Zhang, Lechao Cheng, Pengyuan Zhou, Tianyi Zhou, and Irene Li. 2023. Nlpbench: Evaluating large language models on solving nlp problems. *arXiv preprint arXiv:2309.15630*.

Xiangru Tang, Yuliang Liu, Zefan Cai, Yanjun Shao, Junjie Lu, Yichi Zhang, Zexuan Deng, Helan Hu, Kaikai An, Ruijun Huang, et al. 2023. Ml-bench: Evaluating large language models and agents for machine learning tasks on repository-level code. *arXiv e-prints*, pages arXiv–2311.

Gemma Team. 2024a. Google launches gemma 2, its next generation of open models. https://blog.google/technology/developers/google-gemma-2/.

Qwen Team. 2024b. Qwen2.5: A party of foundation models.

Minzheng Wang, Longze Chen, Cheng Fu, Shengyi Liao, Xinghua Zhang, Bingli Wu, Haiyang Yu, Nan Xu, Lei Zhang, Run Luo, et al. 2024. Leave no document behind: Benchmarking long-context llms with extended multi-doc qa. *arXiv preprint arXiv:2406.17419*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Botao Yu, Frazier N Baker, Ziqi Chen, Xia Ning, and Huan Sun. 2024. Llasmol: Advancing large language models for chemistry with a large-scale, comprehensive, high-quality instruction tuning dataset. *arXiv preprint arXiv:2402.09391*.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al. 2023. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. *arXiv preprint arXiv:2311.16502*.

Appendices

Within this supplementary material, we elaborate on the following aspects:

- Appendix A: Related Work
- Appendix B: Formal Definition of Evaluation Criteria
- Appendix C: Data Statistics and Diversity
- Appendix D: Implementation Details
- Appendix E: More Experiment Results
- Appendix F: Prompt Templates

A Related Work

LLMs for AI Research. With the rapid evolution of pertaining techniques, LLMs are found to be useful in assisting various research disciplines (Yu et al., 2024; Labrak et al., 2024), particularly in AI research, such as generating novel research ideas (Kumar et al., 2024), reviewing research draft (Gao et al., 2024; Du et al., 2024; Liang et al., 2024), and writing scientific papers (Chamoun et al., 2024; Lu et al., 2024). For example, Si et al. (2024) conducted a large-scale human investigation on LLM-generated research ideas and found that LLMs can generate novel ideas compared with humans while lacking feasibility. Du et al. (2024) found that while LLMs are effective at summarizing papers, they tend to overly trust the authors' claimed strengths and struggle to identify weaknesses specific to the paper. Furthermore, some works try to employ LLMs to solve more complicated research tasks that are composed of multiple steps (Li et al., 2024, 2023a; Tang et al., 2023). Notably, Lu et al. (2024) proposed AI-SCIENTIST, an autonomous agent framework that can handle a series of challenging research tasks consecutively, including generating research ideas, coming up with the corresponding experiments along with the implementations, and then writing the final research paper — exactly how human conduct a whole research pipeline. However, there is still a lack of systematic evaluations and quantitative analyses on the LLMs' (intermediate) output of each single-step research task. Our work focuses on building a benchmark that has individual research steps with clear input-output expectations, thus making it suitable for comprehensive LLMs evaluation.

Benchmarks for AI Research Tasks. Existing "LLM assists research" benchmarks mainly focus on the implementation and execution part of the research pipeline (Lu et al., 2024; Chen et al., 2024b; Li et al., 2024). For instance, Huang et al. (2024) proposed MLAgentBench to test the LLMs' capacity for writing project code and training the ML models, where the evaluation metric is the test performance of the models trained by LLMs. However, real-world AI research activities are diverse and some of them are hard to assess for quality, such as generating research ideas, which requires intensive manual assessment (Si et al., 2024; Liang et al., 2024), or LLM-based estimation (Lu et al., 2024). Our work mainly focuses on high-level experience-based research tasks, and we try to build curated

task-specific metrics for every single task for a more efficient and accurate LLMs appraisal.

B Formal Definition of Evaluation Criteria

For the experiment design of EXPDESIGN, we hope the LLMs can mention as many similar experiment steps as the expert's plan. Nevertheless, we also don't expect LLMs to generate too many irrelevant or redundant steps in the plan. This intuition covers both the "recall" and "precision" aspects. Therefore, we develop semantic similarity-based F_1 score, denoted as S-F_1, which is the harmonic mean of S-Precision and S-Recall:

$$\text{S-Precision} = \frac{1}{m}\sum_{i=1}^{m}\max_{j}\text{sim}(p_i, g_j) \tag{1}$$

$$\text{S-Recall} = \frac{1}{n}\sum_{j=1}^{n}\max_{i}\text{sim}(g_j, p_i) \tag{2}$$

where the p and g represent the LLM's prediction plan and the ground-truth plan, respectively. The m and n are the list length of p and g (e.g., m experiment steps in p). We use Sentence-BERT (Reimers, 2019) to measure the semantic similarity between the p_i step and the g_j step.

Meanwhile, S-F_1 omits the item order difference of two lists, but when giving same-length lists (items have one-one correspondence), we can utilize the following similarity-based matching score:

$$\text{S-Match} = \frac{1}{m}\sum_{i=1}^{m}\text{sim}(p_i, g_i) \tag{3}$$

Unlike EXPDESIGN, the output of WEAKNESS is multiple reviewers' weakness lists, which means we have to measure LLM's single prediction list with a "nested" list. Hence, we rewrite S-Precision , S-Recall to SN-Precision , SN-Recall:

$$\text{SN-Precision} = \frac{1}{m}\sum_{i=1}^{m}\left(\frac{1}{r}\sum_{k=1}^{r}\max_{j}\text{sim}(p_i, g_j^k)\right) \tag{4}$$

$$\text{SN-Recall} = \frac{1}{r}\sum_{k=1}^{r}\left(\frac{1}{n_k}\sum_{j=1}^{n_k}\max_{i}\text{sim}(g_j^k, p_i)\right) \tag{5}$$

where r is the number of reviewers of the given paper, n_k means the length of k-th reviewer's weakness list, and g_j^k indicates the j-th item in k-th reviewer's weakness list.

Additionally, in the real world, we would think a review weakness is reliable if it is specific to a paper. Meanwhile, we also hope the review is informative, i.e., no excessive similar weaknesses in one review. Inspired by the classic TF-IDF, we propose a novel review diversity metric:

$$\text{ITF-IDF} = \frac{1}{w}\sum_{j=1}^{w}\left(\frac{1}{m_j}\sum_{i=1}^{m_j}\log\left(\frac{m_j}{O_i^j}\right)\times\log\left(\frac{w}{R_i^j}\right)\right) \tag{6}$$

$$O_i^j = \sum_{k=1}^{m_j}\text{sim}(p_i^j, p_k^j) \tag{7}$$

$$R_i^j = \sum_{l=1}^{w}\max_{s}\text{sim}(p_i^j, p_s^l) \tag{8}$$

where the w is the total number of papers in the dataset, p^j is j-th paper's prediction weakness list, p_i^j is the i-th weakness in p^j. Moreover, O_i^j calculates the intra-paper occurrence frequency of p_i^j; R_i^j is the "soft" number of papers that also contain the p_i^j, which is computed by summing the maximum similarity scores between p_i^j and other paper's weaknesses. In a word, O_i^j measures informativeness, and R_i^j measures specificity. The complete ITF-IDF consider both aspects and reflects the overall weakness diversity.

C Data Statistics and Diversity

We provide the detailed data statistics of three datasets in our benchmark, as shown in Table 4, 5, and 6. We use the NLTK package[7] to tokenize words and count the length. When calculating the length of equations, we use the pylatexenc tool[8] to simplify the equations first.

Meanwhile, for the WEAKNESS, we also plot the review scores distribution of the papers used in the dataset, as well as the track distribution. As can be found in Figure 3, our dataset has a decent distribution, where the papers are uniformly distributed across 13 tracks, and most papers' scores ranged from 5 to 8 (i.e., most papers are weakly rejected or accepted).

D Implementation Details

D.1 Metric Details

When calculating the metrics, specifically for the similarity-based scores, we utilize Sentence-BERT (Reimers, 2019) to encode each segment

[7] https://www.nltk.org/

[8] https://github.com/phfaist/pylatexenc

# of classification instances	1,049
# of source papers	869
ave. "`left`" input context length (in words)	4,377
ave. "`right`" input context length (in words)	6,362
max "`left`" input context length (in words)	24,849
max "`right`" input context length (in words)	32,948
min "`left`" input context length (in words)	711
min "`right`" input context length (in words)	8
ave. "`pos.`" output equation length (in character)	55
ave. "`neg.`" output equation length (in character)	48
max "`pos.`" output equation length (in character)	1,039
max "`neg.`" output equation length (in character)	306
min "`pos.`" output equation length (in character)	6
min "`neg.`" output equation length (in character)	4

Table 4: The statistics of EQINFER. Here, the "`left`" and "`right`" input context indicates the paper contexts before and after the missed equation; "`pos.`" means the ground-truth equations (written by the source paper authors), while "`neg.`" is the GPT4-synthetic wrong equations.

# of instances	100
# of source papers	100
ave. input context length (in words)	4,288
max input context length (in words)	9,799
min input context length (in words)	698
ave. # of input figures	2.6
max # of input figures	16.0
min # of input figures	0.0
ave. length of Experiment&Explanation list	5.7
ave. length per experiment (in words)	34.3
ave. length per explanation (in words)	27.1
max length of Experiment&Explanation list	13
max length per experiment (in words)	135
max length per explanation (in words)	89
min length of Experiment&Explanation list	2
min length per experiment (in words)	9
min length per explanation (in words)	9

Table 5: The statistics of EXPDESIGN.

into a dense vector.[9], which takes about 1GB of memory when running on a single A100 GPU.

D.2 LLMs Running Details

In our experiments, we utilize various LLMs, including both closed and open-sourced. We list the model weight sources for the open-source LLMs:

- OLMo-7B: https://huggingface.co/allenai/OLMo-7B
- Falcon-40B: https://huggingface.co/tiiuae/falcon-40b

[9] https://huggingface.co/sentence-transformers/all-mpnet-base-v2

# of instances	993
# of source papers	993
ave. input context length (in words)	9,811
max input context length (in words)	49,195
min input context length (in words)	24
ave. # of input figures	7.0
max # of input figures	37.0
min # of input figures	0.0
ave. # of input tables	4.3
max # of input tables	53.0
min # of input tables	0.0
ave. # of reviewers per paper	3.8
max # of reviewers per paper	9.0
min # of reviewers per paper	3.0
ave. # of weaknesses per reviewer	4.8
max # of weaknesses per reviewer	39.0
min # of weaknesses per reviewer	1.0
ave. length of weakness (in words)	39.1
max length of weakness (in words)	371.0
min length of weakness (in words)	2.0

Table 6: The statistics of WEAKNESS.

- Gemma 2-27B: https://huggingface.co/google/gemma-2-27b
- Mistral-7B: https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.3
- Mixtral-8x22B-MoE: https://huggingface.co/mistralai/Mixtral-8x22B-Instruct-v0.1
- Llama 3.1-70B: https://huggingface.co/meta-llama/Llama-3.1-70B
- Qwen 2.5-72B: https://huggingface.co/Qwen/Qwen2.5-72B

We use VLLM to unify the inference endpoints of all the above models.[10] We use Pytorch 2.4.0 with CUDA 12.1, and use 8 A100 GPUs for the LLMs inference.

Meanwhile, we use the gpt-4o-2024-08-06, gpt-4-1106-preview, o1-preview-2024-09-12, gemini-1.5-pro-002, and claude-3-5-sonnet-20240620 for the closed-source LLMs. We use LiteLLM to unify the API calling for all these LLMs.[11]

Models	Input Context Processing	Window Size (in words)	SN-F1	SN-Precision	SN-Recall	ITF-IDF
GPT-4-Turbo	split-combine	3,000	**47.66**	42.15	**55.19**	5.31
	no-split	3,000	45.80	**43.66**	48.39	**5.58**
	no-split	20,000	44.99	42.64	47.82	**5.58**
GPT-4o	split-combine	3,000	**47.73**	42.09	**55.48**	5.95
	no-split	3,000	45.74	**43.45**	48.54	5.92
	no-split	20,000	45.47	42.97	48.51	**6.02**
AI-SCI	split-combine	3,000	**45.05**	40.02	**51.91**	2.23
	no-split	3,000	42.56	**40.90**	44.65	2.53
	no-split	20,000	42.53	40.75	44.78	**2.58**

Table 7: The performance comparison of different input processing methods. We use GPT-4o and GPT-4-Turbo because both accept a maximum of 128k tokens input. We also put the results of AI-SCIin the table for reference. Here, "split-combine" splits the input paper into several pieces, where each piece's length is denoted as "window size"; "no-split" means the conventional input cutting, for example, if the window size is 3,000, then only the first 3,000 words in the paper are used. According to the data statistics, 20,000 words can covers more than 95% papers' maximum lengths in our dataset.

Models	One-by-One	Whole-List
Llama 3.1-70B	50.05	49.36 (↓ 0.7)
Qwen 2.5-72B	51.12	48.56 (↓ 2.6)
Gemini 1.5 Pro	52.87	57.48 (↑ 4.6)
Claude 3.5 sonnet	53.03	59.11 (↑ 6.1)
GPT-4	55.03	56.95 (↑ 1.9)
GPT-4o	54.79	58.54 (↑ 3.8)
o1	58.55	61.58 (↑ 3.0)

Table 8: The impact on S-Match scores of maintaining the experiment's self-containment.

Models	Acc. ratio
Llama 3.1-70B	22.93
Gemini 1.5 Pro	55.07
Claude 3.5 sonnet	61.46
GPT-4o	69.72
o1	**76.14**

Table 9: The human evaluation results on LLMs' output explanations. "Acc. ratio" means how many model outputs are accepted by the annotator.

E More Experiment Results

E.1 Input Context Scaling Investigation

E.2 Two Different Explanation Generation Methods for LLMs

E.3 Human Evaluation on LLM-Generated Explanation

put some human evaluation procedure details here.

E.4 Multi-Modal Input Ablation

E.5 Data cases and Annotation Platform Illustration

We put the data cases of the tasks in this anonymous repository: https://anonymous.4open.science/r/LLM4NLP-07D8/

F Prompt Templates

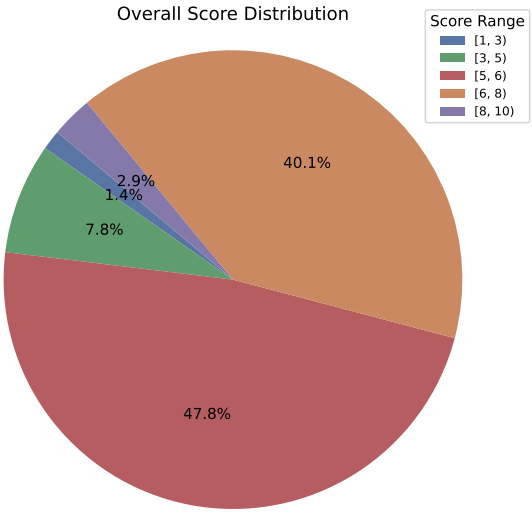
[10] https://github.com/vllm-project/vllm
[11] https://github.com/BerriAI/litellm

Models	Experiment Planning			Motivation Explanation		
	S-F1	**S-Precision**	**S-Recall**	**S-Match**	**ROUGE-L**	**ROUGE-1**
GPT-4o	**53.00**	**51.24**	**55.12**	**58.54**	**29.25**	**35.50**
w/ figures	50.11	48.94	51.59	58.53	27.87	34.30
GPT-4	**43.89**	42.34	**45.82**	**56.95**	**25.98**	**33.37**
w/ figures	43.54	**42.56**	44.85	55.03	22.82	30.01
InternVL2-26B	**40.52**	**48.95**	**35.20**	50.03	29.13	**34.26**
w/ figures	38.83	46.91	33.70	**50.29**	**29.29**	34.06

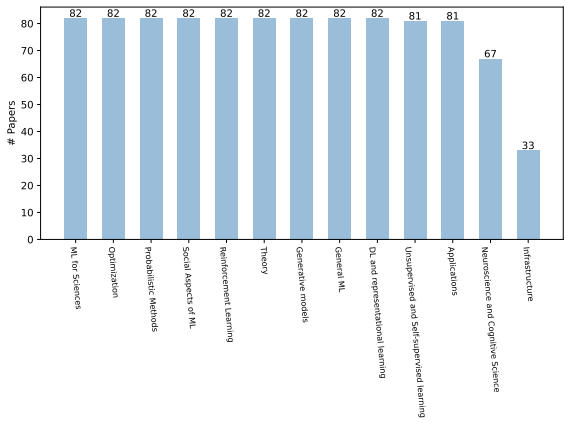
Table 10: The figure inputs ablation of EXPDESIGN. For the maximum text input length, same as the setting in Table 2, we use 2,000 and 3,000 words for open- and closed-source models, respectively. For the closed-source GPT-4o and GPT-4, as they have long context window, we use all the figures of each paper. While for InternVL2, we randomly select two figures per input paper.

Models	SN-F1	SN-Precision	SN-Recall	ITF-IDF
GPT-4o	**47.73**	**42.09**	**55.48**	**5.95**
w/ tables	46.76	41.32	54.17	5.53
w/ figures	46.62	41.20	54.04	5.48
w/ tables & figures	46.58	41.17	53.98	5.36
InternVL2-26B	41.91	41.02	43.28	**1.48**
w/ tables	40.55	40.37	42.91	1.46
w/ figures	**42.88**	**42.10**	**43.76**	1.46
w/ tables & figures	42.44	42.00	43.31	1.44

Table 11: The ablation study about the paper tables and figures of WEAKNESS. Based on the conclusion in Table 7, we use the "split-combine" to process the text input here (2,000 and 3,000 words context window size for open- and closed-source models). For GPT-4o, we use all the table/figure images; while for InternVL2, we randomly select two images per paper, i.e., two random figures, two random tables, or one random figure + table.



(a) The review score distribution of the papers used in WEAKNESS.



(b) The track distribution of the papers used in WEAKNESS.

Figure 3: The data diversity illustration of WEAKNESS.

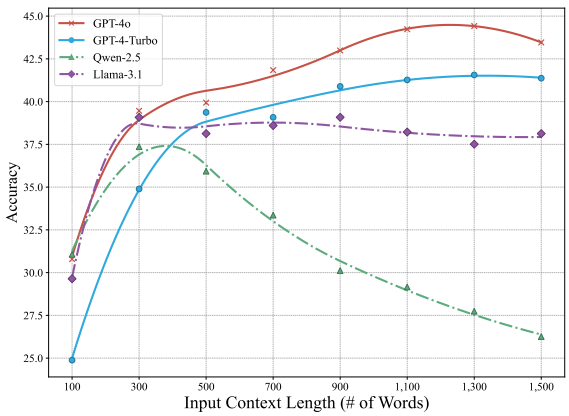


Figure 4: The input context length scaling trend on the EQINFER task.

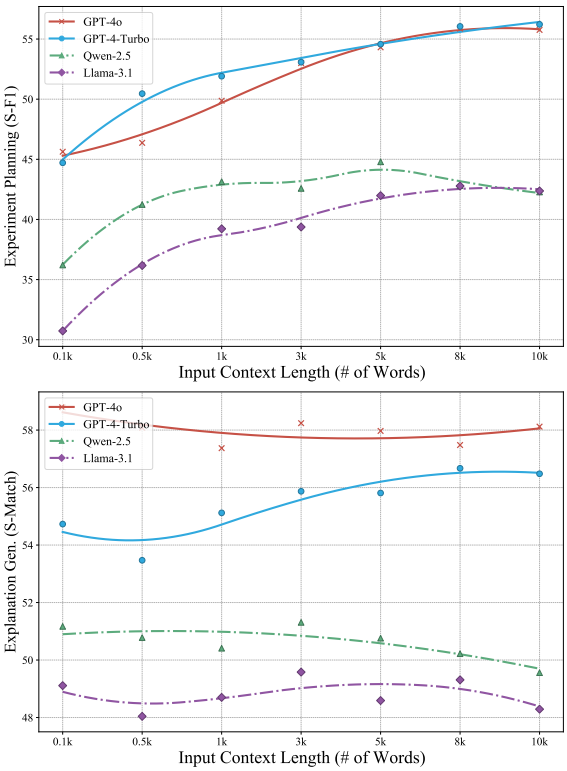


Figure 5: The input context length scaling trend of different LLMs on EXPDESIGN.