SciDQA: A Deep Reading Comprehension Dataset over Scientific Papers

Shruti Singh[1] **Nandan Sarkar**[2] **Arman Cohan**[2,3]
[1]IIT Gandhinagar [2]Yale University [3]Allen Institute for AI

Abstract

Scientific literature is typically dense, requiring significant background knowledge and deep comprehension for effective engagement. We introduce SciDQA, a new dataset for reading comprehension that challenges LLMs for a deep understanding of scientific articles, consisting of 2,937 QA pairs. Unlike other scientific QA datasets, SciDQA sources questions from peer reviews by domain experts and answers by paper authors, ensuring a thorough examination of the literature. We enhance the dataset's quality through a process that carefully filters out lower quality questions, decontextualizes the content, tracks the source document across different versions, and incorporates a bibliography for multi-document question-answering. Questions in SciDQA necessitate reasoning across figures, tables, equations, appendices, and supplementary materials, and require multi-document reasoning. We evaluate several open-source and proprietary LLMs across various configurations to explore their capabilities in generating relevant and factual responses. Our comprehensive evaluation, based on metrics for surface-level similarity and LLM judgements, highlights notable performance discrepancies. SciDQA represents a rigorously curated, naturally derived scientific QA dataset, designed to facilitate research on complex scientific text understanding.

1 Introduction

Question-answering (QA) datasets are valuable for evaluating the reading comprehension, reasoning, and document understanding capabilities of language models (Dua et al., 2019; Dasigi et al., 2021; Rogers et al., 2023). The scientific QA task involves reading a research paper and answering questions, drawing on the paper content and some background knowledge. This task mirrors how humans engage with academic literature (Lo et al., 2023; Palani et al., 2023).

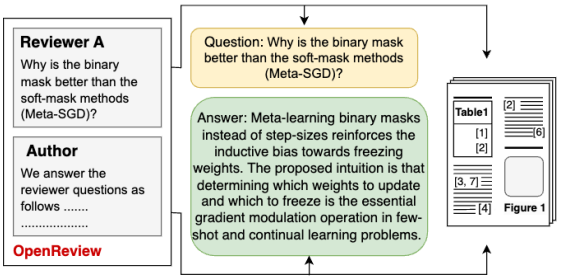


Figure 1: An instance in the SciDQA dataset. The question and answer corresponding to the paper are extracted from the reviewer-author discussion on OpenReview.

Scientific literature is inherently dense and typically requires a deep understanding and significant background knowledge to fully comprehend and engage with. To address this challenge, the NLP community has developed various datasets for question-answering (QA) from research papers to aid in development and evaluation of AI systems for comprehending the research papers. Methods range from manual question generation by domain experts (Möller et al., 2020; Dasigi et al., 2021; Lee et al., 2023) to automated extraction of questions using machine learning from selected texts (Saikh et al., 2022, 2020; Pappas et al., 2020; Jin et al., 2019; Pappas et al., 2018). However, many of these datasets focus on surface-level information and are often limited to questions that are written from titles and abstracts, which restricts the complexity and deeper engagement with the full papers.

We introduce SciDQA, a novel deep reading comprehension dataset for scientific papers. It is specifically tailored to the scientific articles in the machine learning (ML) domain and sourced from peer reviews on the OpenReview platform (OpenReview, 2023). Peer reviews frequently include questions or comments from reviewers who seek information or clarification on aspects they are confused about or do not fully understand. Answering many of such questions necessitate a deep and com-

Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 20908–20925
November 12-16, 2024.

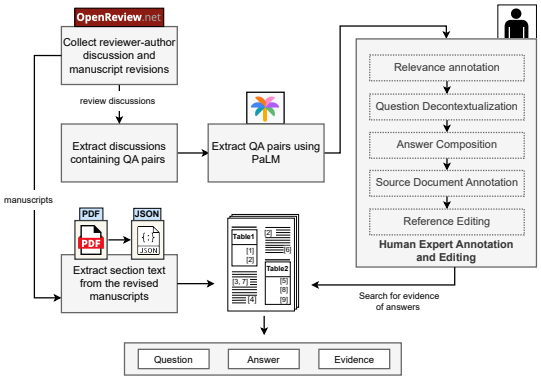


Figure 2: Dataset curation pipeline for SciDQA. LLM-based QA extraction from peer reviews is followed by a comprehensive human expert annotation and editing. As discussed, we only include evidence for a subset of the dataset due to high annotation cost.

prehensive understanding of the research and the background, such as a critical view of the approach and results, implications of the findings, and comparisons with previous works. Moreover, peer reviews are acompanied by responses from authors, who have carefully tried to address and clarify the reviewers questions. As both authors and reviewers are domain experts, responding to these inquiries necessitates a deep understanding of the paper and its broader research field. Consequently, we believe these questions are an excellent source for probing deep comprehension of research papers, contrasting with prior work that often targets shallow information-extraction or surface-level facts. However, not all such questions expressed in a review are useful. In addition they also need rewriting to stand alone as clear, self-contained queries suitable for a reading comprehension dataset. To ensure the quality and relevance of our dataset, we implement a human annotation process by domain experts, highlighted in Figure 2.

Our dataset features long-form questions and answer pairs, as shown in Table 1. It is diverse and also some questions require comprehension of figures, tables, equations, and references in addition to the paper text. Approximately 11% of the questions necessitate reasoning over at least one explicitly mentioned reference paper in addition to the candidate paper. We evaluate several open-source and proprietary LLMs under various configurations (including closed-book, retrieval-based setup and long-context reasoning) to benchmark their capabitlies on this task. Our findings suggest that our dataset presents a significant challenge, as several LLMs struggle to generate accurate factual answers across a variety of experimental setups. Our dataset, code, and model outputs to reproduce our results are available on the github repository. [1]

2 Building the SciDQA Dataset

We present the pipeline for the collection of the SciDQA dataset and the preprocessing, manual filtering and rewriting steps involved. A schematic denoting the pipeline is presented in Figure 2. We present the various stages of data curation next.

2.1 Curation from OpenReview

We selected top-tier ML and DL venues, designated as A* rankings by ICORE Portal (CORE), with publicly accessible reviewer-author discussions on OpenReview (Appendix A). We curate 11400 papers from ICLR (2018-2022) and NeurIPS (2021-2022), with a major focus on including newer papers to decrease the risk of contamination with LLM pretraining datasets.

2.2 Processing the reviews

PDF to Text Conversion OpenReview portal hosts multiple submitted PDF versions of a submitted manuscript which are curated. Nougat (Blecher et al., 2023), a visual transformer model designed for scientific OCR tasks (details in Appendix A), is used for PDF to text conversion.

Regex Filtering OpenReview has nested discussions, i.e. authors and reviewers reply to messages, creating a time-stamp chain of discussion. We extract 18,658 reviewer-author discussions for 11,400 papers that contain questions and answers, by regex pattern matching (details in Appendix A).

LLM-based QA Extraction Next, we extract explicit questions that reviewers asked the authors from the reviews. For QA extraction, we utilized the PaLM API (Google, 2023) to extract specific question-answer pairs within the reviewer-author discussions.[2] Initial attempts to extract questions and answers using non-LLM methods faced challenges, as authors and reviewers employ various patterns for posing questions and answers, making it difficult to comprehensively

[1] https://github.com/yale-nlp/SciDQA

[2] We chose to use PaLM because it consistently delivered high-quality extractions and offered an available API, capable of handling up to 60 requests per minute.

Dataset	Curation	Size	Source	Question length	Answer length	Multiple Docs	% Short Answers
QASA (2023)	Manual	1,554	Full-Text	15.86	44.95	×	1.61%
QASPER (2021)	Manual	5,089	Title/Abstract	9.33	18.19	×	39.94%
Covid-QA (2020)	Manual	2,019	Full-Text	10.61	15.79	×	32.64%
ScholarlyRead† (2020)	Synthetic	10,000	Abstract	NA	NA	×	NA
BioRead (2018)	Synthetic	16.4M	Full-Text	42.90	1.92	×	98.70%
BioMRC (2020)	Synthetic	700,000	Title/Abstract	16.01	1.73	×	99.38%
PubMedQA (2019)							
Annotated	Manual	1,000	Title/Abstract	14.42	43.23	×	0%*
Unlabeled	Synthetic	61,249	Title/Abstract	14.98	45.88	×	0%*
Artificial	Synthetic	211,269	Title/Abstract	16.35	40.97	×	0%*
SciDQA (Ours)	Hybrid	2,937	Full-Text	23.92	104.67	✓	1.74%

Table 1: Comparison of the related datasets. †ScholarlyRead dataset is unavailable publicly, hence we skip its statistics. *PubMedQA features two types of answers: a long answer, which is the last sentence of the abstract, and a short answer, which is yes/no. Here, we report statistics of long answers as all short answers are less than 5 words.

cover all instances. Through this approach, we extracted 26,085 question-answer pairs. Details of the prompts are in the Appendix A Figure 3.

2.3 Human Expert Annotation and Editing

In initial investigations, we found that many of the extracted questions are *not* useful and they would need additional revisions to be appropriate for a QA dataset. Therefore, to ensure the quality of the QA pairs in the SciDQA dataset, we employed an extensive manual annotation process by domain experts.[3] This included determining and keeping only the most relevant questions, rewriting both questions and answers, and editing references in the QA pairs. We briefly discuss annotation and editing stages.

Relevance Annotation This task selects information-seeking questions, whose answers are identifiable within the research paper text, from a set of synthetically generated QA pairs. Questions referencing figures, tables, equations, specific sections, or lines, and inquiries requiring data from multiple papers were categorized as relevant. Conversely, questions asking for edits, summaries, or subjective judgments about the paper's quality, or those based on the authors' personal experiences, were classified as irrelevant. To expedite the annotation process, we introduced an 'ambiguous' category for cases where the relevance of a question-answer pair was challenging to ascertain. Questions necessitating experimental validation for answers, and where it remained unclear whether the authors had conducted such experiments based on reviewer suggestion during reviewer-author discussion, were classified as ambiguous. We present a few samples for each category in Table 6 in Appendix A.

Two annotators, also the authors of this paper, annotated the dataset, starting with a common subset of 200 instances and achieving an 85% agreement rate. The disagreements were discussed and resolved, and the rest of the questions were annotated by a single annotator. In total, the annotators reviewed 7,000 instances, identifying 2,937 QA pairs as relevant, equivalent to a relevancy rate of approximately 41%. Additional details about the annotations are in Appendix A.1.

Decontextualizing Questions and Answers Originally, questions were directed towards the authors of the paper and authors provided answers from their perspective. We rewrote these QA pairs in the third-person point of view to make them universally applicable and to avoid biasing language models to generate answers in the first person when trained on SciDQA. This is also necessary for the models to understand that the question does not ask for their personal opinion, but is a factual question seeking information about the author's reasoning in the paper. We also add contextual information to the questions where the question is incomplete or incomprehensible without contextual information present in the review text. We present an example in Figure 4 showcasing scenarios where decontextualization and editing the narrative is necessary to comprehend the question. The perplexity of questions before and after rewriting, when evaluated with the GPT-2 model, exhibits a difference

[3]Students with extensive experience in NLP and ML.

of 16.3 points, suggesting that decontextualization contributes to an enhancement in dataset quality.

Annotating the Source Document Certain conferences like NeurIPS and ICLR allow authors to submit revised manuscripts during the author-reviewer discussion period. For simplicity, we focus only on the initial submitted copy and the final camera-ready manuscript. For rejected papers, the last submitted manuscript is considered the final version, which may sometimes be identical to the initial submission. Establishing the source document between the initial and final manuscripts presents challenges, as author-reviewer discussions often result in added details like tables, figures, and text, making the camera-ready version a suitable source document. However, reviewers' questions may prompt authors to rewrite paper text to explicitly mention the answer, simplifying the dataset if the final version is used. We depict two such scenarios in Figure 6. To manage these variations, each question-answer pair is annotated with the version of the document used as the source, typically the initial or final version. If author responses indicate additions in a revision, the final version is marked as the source document. If no specific information is given, the initial version is defaulted as the source. This approach addresses potential ambiguities arising from updates in table, figure, and section numbers in the revised final manuscript.

Reference Editing Finally, to prevent language models from taking shortcuts by extracting answers based on reference text markers within the papers, we edited the references in the QA pairs, as shown in Figure 5. This process involved replacing specific reference markers with placeholders and providing a list of necessary references at the end of the question and the answer.

3 Dataset Details and Analysis

The SciDQA dataset comprises 2,937 question-answer pairs. We present the statistics of SciDQA in comparison to other related existing QA datasets in Table 1. Next, we discuss the diversity of answer sources, and fuzzy searching for answers, and the statistics of changes in initial and revised manuscripts.

Diversity of Answer Sources Our dataset features questions necessitating reasoning across multiple modalities beyond mere text, including figures, tables, equations, and both appendix and supplementary materials.[4] This design ensures that comprehensive reasoning over the full-text of the paper is essential for answering the questions accurately. The statistics are presented in Table 2.

Information Source	% in Dataset
Tables	14.03%
Multiple documents	10.9%
Appendix and Supplementary	10.01%
Equations and Symbols	10.32%
Figures	6.98%

Table 2: Distribution of various modalities (text, figures, tables, equations, appendix, and supplementary) which are required to answer the questions in the dataset.

Fuzzy Search for Answers We search for answers in the research paper texts and find sections with at least 80% unigram overlap between answers and paragraphs. Such a high degree of overlap suggests that the text from the research papers is directly utilized as answers to questions, simplifying the question-answering process to the identification of pertinent paragraphs. This implies a reduced necessity for reasoning or inferential thinking compared to scenarios where answers must be derived from an analysis of the text. Our findings reveal that only 25% of the answers in our dataset can be identified with an overlap exceeding 80%. By contrast, the QASA dataset (Lee et al., 2023), features 52% of answers that demonstrate more than 80% unigram overlap with the paper text, indicating a higher reliance on direct text retrieval for answering questions.

Edits in Initial and Revised Manuscripts We conducted an analysis of differences between PDF versions for each QA pair.[5] Our dataset of 576 unique papers shows that 66.3% vary in figure mentions, and 54.9% vary in table counts between initial and final manuscripts, highlighting the need to maintain separate versions.

4 Experimental Setup

We design four task configurations to evaluate the capabilities of LLMs in answering the questions in SciDQA. We use two separate setups, closed-

[4]For experiments, we use table and figure captions and do not use multi-modal models for direct processing of figures. We'll leave that as a future direction.

[5]This is because authors often update their manuscripts in response to comments and questions by reviewers.

book (Roberts et al., 2020), and open-book. We experiment with a wide-range of open-source LLMs (Falcon (Almazrouei et al., 2023), Galactica (Taylor et al., 2022), Gemma (Team et al., 2024), Llama 2 (Touvron et al., 2023), Llama 3.1 (Dubey et al., 2024) Mistral (Jiang et al., 2023), Phi-2 (Javaheripi et al., 2023), Qwen v2.5 (Team, 2024), Vicuna (Zheng et al., 2024), and Zephyr (Tunstall et al., 2023)) and two frontier closed models Gemini Pro (Google et al., 2023) and GPT-4 (Achiam et al., 2023) models. For open-source models, we experiment with various model sizes from ∼2B to ∼70B parameters.

Priming with the Question Only (`closed-book`) Can LLMs answer the questions in a closed-book setting (Roberts et al., 2020) when primed only with the question text and without explicitly providing the paper? LLMs have the ability to retain knowledge and in this sense, it's conceivable that LLMs might be able to generate answers directly based solely on the question text, without any context from the associated research papers.[6] Further, LLMs might already have internalized the knowledge related to papers to be able to answer some specific questions without explicitly providing the context. To investigate this possibility, in the `closed-book` configuration, LLMs are presented with only the questions and instructions, without any information about the relevant paper.

Priming with Question, and Paper's Title and Abstract (`title-abs`) In this setting, we provide the LLM with the question text, along with the Title/Abstract of the paper. This mimicks a "partially" `closed-book` setting. The objective is to ascertain whether the inclusion of limited additional information, such as the paper's Title and Abstract, enhances the LLM's ability to accurately retrieve and recall the knowledge to correctly answer the question. Unlike the fully `closed-book` setting, it is not entirely infeasible to answer some questions with the information provided in the abstract. However, given that our dataset comprises questions that require complex reasoning, the answers to the majority of questions will not be found in the abstract alone.

[6]Comphrehensive evaluation of this setting is challenging, as it is difficult to disentangle potential effect of contamination, from knowledge retrained by LLMs, especially in models where source of training data isn't disclosed. While we source our questions from peer reviews, our questions and answers are significantly revised and re-written, so exact-match contamination is less likely.

Retrieval-Augmented Generation with LLMs (RAG) We follow a retrieval-augmented generation setup for this configuration. Research paper texts exceed the typical model context length with exception of few long-context models (which we will discuss in the next experimental setup). To accomodate processing such documents we employ a RAG setup, where we first divide the document into smaller and slightly overlapping chunks, retrieve the most relevant chunks to the question using a BM25 ranker,[7] and subsequently input the top ranked chunks to the LLM, tasked with generating the response. The operational flow of this pipeline is depicted in Appendix Figure 9 and the chunking algorithm is presented in Appendix Algorithm 1.

Comprehending the Full-text using LLMs (`full-text`) In this experimental setup, LLMs are provided with the full-text of scientific papers and are tasked with answering a specific question. The length of scientific texts could exceed the context length limit of many LLMs. In such cases, we divide the full-text into segments. Each segment, along with the question and instructions, is then presented to an LLM (referred to as base-LLM), which generates answers for each segment.

This setup produces multiple answer candidates for a single question, contingent on the number of passes required to present all chunks to the LLM. To distill these into a singular, optimal response, we introduce an answer selection phase. During this phase, the Llama 3.1 70B model is prompted with the question and all answers generated by the base-LLM, with instructions to identify the most comprehensive response from the provided options. Details of this prompt are included in the Appendix B in Figure 11. We only segment paper's full-text when it exceeds the model's context length (pipeline presented in Appendix Figure 10).

For models with context length limit greater than the full-text (Gemini, GPT-4o, and GPT-4o-mini), the base-LLM directly generates the answer from the full text, and the answer-selection phase is not required. For Qwen v2.5 (1.5B and 7B) and Llama 3.1 (8B and 70B) models, the context length is 128k, however the prompt with the entire paper text does not fit into the cache, so we chunk the text and generate multiple answer candidates similar to other LLMs, however, the answer selection

[7]More advanced retrieval settings using dense retrievers or rerankers can be also employed to improve the performance of models in this setting. Our goal is mainly to provide a baseline setup for each of the experimental settings.

Model	CB	T/Abs	RAG	FT
2-3 B				
Gemma IT (2024)	40.75	31.50	39.47	30.33
Phi2 (2023)	45.24	43.16	42.20	40.95
Qwen 2.5 IT (2024)	34.86	37.81	33.01	35.70
6-7 B				
Falcon IT (2023)	28.49	19.70	44.28	42.25
Galactica (2022)	14.49	41.76	34.27	43.07
Llama 2 Chat (2023)	26.09	36.20	46.95	45.99
Llama 3.1 IT (2024)	20.46	42.56	38.59	45.73
Longchat 32k (2023)	25.77	22.59	44.98	40.58
Mistral IT (2023)	29.71	47.64	47.67	42.29
Qwen 2.5 IT (2024)	44.72	47.10	45.13	41.41
Vicuna (2024)	21.32	18.22	42.01	46.46
Zephyr β (2023)	29.20	41.66	**48.74**	42.13
13 B				
Llama 2 Chat (2023)	28.06	37.35	47.53	45.88
Vicuna (2024)	27.69	30.11	45.41	46.77
70 B				
Llama 2 Chat (2023)	43.33	39.69	40.71	30.14
Llama 3.1 IT (2024)	46.20	48.46	47.60	47.78
Proprietary LLMs				
Gemini Pro (2023)	28.31	32.01	38.03	37.59
GPT-4o	**48.48**	**50.61**	46.63	**54.03**
GPT-4o-mini	47.50	**50.32**	**48.90**	**54.02**
GPT-4o (2023)	-	-	-	49.3

Table 3: Average scores for all configurations. CB refers to `closed-book` and T/Abs referes to `title-abs`. Cells in blue indicate RAG or `full-text` (FT) settings where performance improves over both `closed-book` settings by atleast two points. For seven models, both RAG and `full-text` lead to better scores, while for four models (including GPT-4o and GPT-4o-mini) only one of the RAG/`full-text` settings performs significantly better than Closed settings.

phase uses the same base-LLM (Qwen and Llama versions respectively) instead of Llama 3.1 70B.

4.1 Evaluation

Surface-level Metrics: We first use surface-level metrics for evaluating the LLM generated answers, which compare the similarity of the generated long-form answer with the gold standard through textual overlaps. These include ROUGE score (Lin, 2004) (we compute ROUGE-1 (R-1), ROUGE-2 (R-2), and ROUGE-L (R-L); and report the average as R_{μ}), BLEURT-20 (Pu et al., 2021) (abbreviated as `BL`), and BERTScore (Zhang* et al., 2020) (BERTScore F1 score as `BS`).

LLM Judge: In addition to traditional surface-level metrics, we also use LLM-as-a-judge to evaluate the quality of the generated text, given potential unreliability issues of surface-level metrics (Liu et al., 2023). In particular, we employ Llama 3.1 70B (Dubey et al., 2024), GPT-4o, and GPT-4o-mini[8] to evaluate LLM-generated answers on four aspects, namely relevance, accuracy, completeness, and conciseness on a scale of 1-10. The LLM is also asked to report the overall quality scores by averaging the individual scores for each aspect. All LLM judge models (Llama 3.1 70B, GPT-4o, and GPT-4o-mini) are prompted to generate the explanations and the scores for each aspect, and subsequently, Llama 3.1 8B model is used to extract the overall quality score from the generated explanations (see appendix B.3 for the exact prompts used). The average scores, normalized to range 1-100, from Llama 3.1 70B, GPT-4o, and GPT-4o-mini judges are represented with `L70`, `4o`, and `4oM` respectively. The average of LLM-as-a-judge scores is presented as `ALS` in the tables. An average score of all traditional and LLM judge scores is presented in column `Avg`.

5 Results and Discussion

In this section we dicsuss the main results. We provide the main results as averages of all metrics for all the four experimental settings in Table 3. Then we present the detailed results of each setting in Tables 4 and 5.

Comparison among `closed-book`, `title-abs`, RAG, and `full-text` settings: We compare the average scores of all metrics for the models in four settings in Table 3, out of which two settings (`closed-book` and `title-abs`) are open-domain question answering settings, and the other two settings provide the LLMs with paper context either in relevant chunk format (RAG) or full-text of the paper (`full-text`). Seven models (Falcon Instruct 7B, Llama 2 Chat 7B and 13B, Longchat 7B, Vicuna 7B and 13B, and Gemini) perform better when provided with paper context (both RAG and `full-text` setup) over the other two open-domain question answering settings (Priming with Questions (`closed-book`) and Priming with Question and Title/Abstract (`title-abs`)). Other four

[8]Llama 3.1 allows full reproducible results, and is a highly capable open-source model in evaluating instruction-following (Liu et al., 2024), GPT-4o is a frontier LLM at time of writing, and GPT-4o-mini balances the cost and quality of evaluation.

Size	Model	closed-book (Prompt <INS, Q>)								title-abs (Prompt <INS, Q, TABS>)							
		R_μ	BL	BS	L70	4o	4oM	ALS	Avg	R_μ	BL	BS	L70	4o	4oM	ALS	Avg
2-3 B	Gemma IT (2024)	12.3	41.1	50.0	46.9	43.5	50.6	47.0	40.8	15.7	29.5	47.2	33.6	30.8	32.3	32.2	31.5
	Phi-2 (2023)	16.4	40.8	54.1	53.4	49.3	57.4	53.4	45.2	12.3	42.8	50.6	49.1	48.8	55.3	51.1	43.2
	Qwen 2.5 IT (2024)	5.9	53.1	42.5	31.7	32.3	43.7	35.9	34.9	6.3	54.1	42.4	37.8	37.5	48.8	41.4	37.8
6-7 B	Falcon IT (2023)	14.7	0.4	0.5	53.2	46.4	55.7	51.8	28.5	4.9	9.5	33.9	30.8	18.4	20.7	23.3	19.7
	Galactica (2022)	4.1	0.5	0.4	23.2	26.6	32.2	27.3	14.5	20.1	40.3	49.9	45.5	44.5	50.3	46.8	41.8
	Llama 2 Chat (2023)	7.5	0.5	0.5	46.8	46.2	55.1	49.4	26.1	10.6	37.1	48.9	40.2	37.5	42.9	40.2	36.2
	Llama 3.1 IT (2024)	1.7	11.0	32.8	33.4	21.1	22.8	25.8	20.5	6.1	46.4	45.8	48.3	51.7	57.1	52.4	42.6
	Longchat 32k (2023)	9.4	0.4	0.5	51.4	41.8	51.1	48.1	25.8	1.3	9.3	34.1	59.4	12.7	18.8	30.3	22.6
	Mistral IT (2023)	13.7	0.4	0.5	55.4	50.2	58.0	54.5	29.7	16.3	40.3	53.5	59.3	56.0	60.4	58.6	47.6
	Qwen 2.5 IT (2024)	6.5	46.2	46.3	54.2	54.1	61.0	56.4	44.7	7.5	45.1	47.4	60.7	59.0	62.9	60.9	47.1
	Vicuna (2024)	7.0	0.3	0.4	53.1	29.9	37.2	40.1	21.3	2.8	5.9	31.3	29.5	23.4	16.5	23.1	18.2
	Zephyr β (2023)	9.5	0.4	0.5	54.6	51.0	59.1	54.9	29.2	13.0	38.4	51.2	49.2	46.5	51.7	49.1	41.7
13 B	Llama 2 Chat (2023)	7.7	0.5	0.5	51.5	50.1	58.1	53.2	28.1	10.7	39.1	49.6	41.3	39.8	43.7	41.6	37.4
	Vicuna (2024)	9.3	0.4	0.5	54.2	47.5	54.2	52.0	27.7	7.9	22.5	41.1	40.7	32.6	35.8	36.4	30.1
70 B	Llama 2 Chat (2023)	8.2	44.7	49.0	50.2	49.8	58.0	52.7	43.3	6.2	49.5	43.7	40.2	46.7	51.8	46.2	39.7
	Llama 3.1 IT (2024)	10.5	44.2	50.0	57.0	55.4	60.1	57.5	46.2	12.7	42.7	52.1	61.3	60.0	62.0	61.1	48.5
UNK	Gemini Pro (2023)	5.3	21.1	39.1	38.9	31.7	33.8	34.8	28.3	12.0	24.6	44.2	40.0	36.6	34.7	37.1	32.0
	GPT-4o	9.0	42.6	49.1	64.3	60.8	65.1	**63.4**	**48.5**	11.7	42.3	51.5	66.7	64.9	66.6	66.1	**50.6**
	GPT-4o-mini-2	7.7	42.9	48.5	62.4	59.3	64.3	62.0	47.5	9.1	43.2	50.1	67.1	64.9	67.5	**66.5**	50.3

Table 4: closed-book evaluates LLMs in a closed book setting without the paper context. title-abs configuration evaluates if additional context (title and abstract) helps in answering the questions. The metrics are R_μ (average of rouge-1, rouge-2, and rouge-l), BLEURT-20 (BL), BERTScore F1 (BS), and LLM judges Llama 3.1 70B (L70), GPT-4o (4o), and GPT-4o-mini-2 (4oM). ALS is the average over LLM judges, and Avg is the average over all metrics.

models (Llama 3.1 8B, Zephyr 7B, GPT-4o, and GPT-4o-mini) perform better only in one of the RAG/full-text settings in comparison to open-domain question answering settings. The rest seven models show a degradation or similar scores when provided with paper context, likely indicating that either there could be contamination affecting the results or the models are able to generate shallow answers without reasoning about the question.

We compute the maximum improvement in scores when provided with paper context (full-text setting), by computing the difference of best scores in closed-book and RAG/full-text settings. Vicuna 7B model shows the highest improvement in average scores (25 points) from closed-book to full-text setting, indicating it is able to effectively use the papers full-text to reason about the question.

Overall high score may not correlate with reasoning from the context. GPT-4o and GPT-4o-mini models perform the best among all evaluated models, and achieve the highest average score in the full-text setting. However, the GPT models also perform best in both the closed-book settings, which indicates that the model is also able to reason about the questions without providing the context. Priming with the paper title and abstract (title-abs) leads to 2-3 points improvement over the full closed-book setting for both models. In comparison to title-abs, the average score improves by four points in full-text setting. This might indicate the model is able to retrieve relevant knowledge to the question from its parameters without explicitly being provided with the paper, in which case the model had been trained on the source papers.[9]

Among the open-source LLMs, the best scores are achieved by Llama 3.1 70B Instruct model, however, its average score for full-text and RAG setup is within one point difference of other open-domain question answering setups, which means the model is not using the provided context from the paper for reasoning about the questions. Other models with performances similar to Llama 3.1 70B, and also showcasing significant improvements over the corresponding closed-book settings are Vicuna 7B, 13B and Llama 2 13B Chat.

[9]This also might suggest potential for contamination affecting the results. Although our question and answers are revised and rewritten, there's a chance that training on raw open-review data might help the models in this task.

Size	Model	RAG (Prompt <INS, Q, top-3 Chunks>)								full-text (Prompt <INS, Q, Full-text>)							
		R_μ	BL	BS	L70	4o	4oM	ALS	Avg	R_μ	BL	BS	L70	4o	4oM	ALS	Avg
2-3 B	Gemma IT (2024)	24.7	34.9	54.0	41.0	41.1	41.2	41.1	39.5	13.5	18.9	45.5	37.1	31.7	35.4	34.7	30.3
	Phi-2 (2023)	28.7	35.2	54.2	27.3	51.4	56.4	45.0	42.2	16.2	24.6	49.0	53.1	48.1	54.7	52.0	41.0
	Qwen 2.5 IT (2024)	7.1	43.1	44.4	32.5	30.1	40.8	34.5	33.0	17.5	24.7	46.8	45.7	37.7	41.9	41.8	35.7
6-7 B	Falcon IT (2023)	23.9	37.3	53.7	45.8	49.5	55.5	50.3	44.3	18.3	28.6	51.3	49.5	50.1	55.8	51.8	42.3
	Galactica (2022)	20.7	29.3	49.8	17.3	40.6	47.9	35.3	34.3	20.5	24.9	49.8	53.1	52.3	57.8	54.4	43.1
	Llama 2 Chat (2023)	24.1	35.6	53.8	56.8	53.8	57.6	56.1	47.0	15.1	30.2	52.0	59.4	56.7	62.5	59.5	46.0
	Llama 3.1 IT (2024)	5.9	51.6	42.7	40.7	39.6	51.1	43.8	38.6	16.6	30.5	51.3	57.8	57.4	60.8	58.7	45.7
	Longchat 32k (2023)	19.8	38.1	52.6	53.1	50.6	55.7	53.1	45.0	15.8	22.9	49.5	56.4	47.8	51.1	51.8	40.6
	Mistral IT (2023)	24.4	38.4	55.2	55.9	54.1	58.0	56.0	47.7	18.6	24.4	50.4	54.3	50.5	55.6	53.5	42.3
	Qwen 2.5 IT (2024)	8.9	48.4	45.7	54.3	53.5	60.0	55.9	45.1	17.0	29.3	48.0	53.8	49.5	50.9	51.4	41.4
	Vicuna (2024)	23.8	33.3	53.2	50.2	42.7	48.9	47.3	42.0	15.8	29.0	52.3	61.2	58.1	62.3	60.5	46.5
	Zephyr β (2023)	18.8	39.4	54.5	59.1	58.9	61.8	**59.9**	48.7	16.5	24.5	50.5	56.7	49.7	54.9	53.8	42.1
13 B	Llama 2 Chat (2023)	22.0	39.0	55.3	58.5	53.1	57.3	56.3	47.5	15.4	30.1	52.0	58.2	57.4	62.2	59.3	45.9
	Vicuna (2024)	25.6	35.4	54.6	53.7	49.5	53.6	52.3	45.4	16.0	28.2	51.9	62.1	59.5	62.9	61.5	46.8
70 B	Llama 2 Chat (2023)	16.0	26.4	43.1	56.3	52.4	50.1	52.9	40.7	11.7	18.7	45.3	35.7	33.6	35.9	35.1	30.1
	Llama 3.1 IT (2024)	23.0	37.9	54.0	57.0	55.5	58.2	56.9	47.6	18.2	28.9	52.7	61.4	61.6	63.9	62.3	47.8
UNK	Gemini Pro (2023)	24.2	30.2	49.9	42.0	41.5	40.4	41.3	38.0	6.9	27.5	42.9	51.3	48.5	48.4	49.4	37.6
	GPT-4o	28.5	36.5	55.6	51.9	53.4	53.9	53.1	46.6	26.2	40.3	57.4	66.2	66.6	67.5	66.8	**54.0**
	GPT-4o-mini-2	25.4	38.0	56.0	57.2	57.9	58.9	58.0	**48.9**	22.4	40.8	56.9	68.0	67.2	68.8	68.0	**54.0**
	GPT-4	-	-	-	-	-	-	-	-	10.0	33.1	48.4	68.2	67.6	68.5	**68.1**	49.3

Table 5: RAG setup prompts the LLM with top-3 chunks extracted from the paper. `full-text` evaluation - LLMs are provided with the paper's full-text. If the full-text exceeds LLM's context length, the base-LLM reasons over paper chunks and generates answer candidates, followed by Llama 3.1 70B for answer selection.

Instruction-tuned models perform better than their counterparts generally. The instruction-tuned counterparts of Gemma, Falcon, Llama 2, and Mistral perform better at retrieving the answers.

Overall, GPT-4o and GPT-4o-mini perform best among all evaluated models in all task settings. For `closed-book` and `title-abs` task settings, we report the scores with surface-level metrics, LLM judge scores and overall average scores in Table 4. GPT-4o and GPT-4o-mini perform best among evaluated models in both settings. Among open-source models, Llama 3.1 70B Instruct and Qwen2.5 7B Instruct models perform the best in both `closed-book` and `title-abs` settings. In RAG setting, Zephyr β 7B and GPT-4o-mini perform the best, followed by similar performances from GPT-4o, Llama 3.1 8B Instruct, Llama 2 Chat 13B, Llama 2 Chat 7B, Mistral 7B Instruct, and Qwen2.5 7B Instruct. However, the best overall score in RAG setting (Table 5) is similar to `closed-book` and `title-abs` settings, indicating that providing the context chunks does not lead to significantly higher scores.

Best performance is observed in `full-text` setting, with significant differences among scores of proprietary and open models. For the full-text setting (Table 5), a significant score difference is observed among proprietary models (GPT-4o, GPT-4o-mini) and best-performing open-source models (Llama 3.1 70B Instruct and Vicuna 13B).

Human Performance Estimation: Evaluating human performance on the SciDQA dataset is challenging due to the complex and domain-specific nature of its questions. To assess human proficiency, the authors compared human-written responses from 28 annotated QA pairs against those generated by GPT-4.[10] An author performs the task by writing the answers to the given questions, by reading and examining the paper. The annotator found this task to be relatively challenging, particularly for papers outside their expertise. During evaluation, each instance included a question, a ground truth answer, an author-written answer after reading the paper, and a GPT-4 generated answer; with evaluations focusing on comprehensiveness, factuality, and clarity. Results showed that 32% of comparisons ended in a tie, indicating GPT-4's adequacy for simpler questions. Humans were preferred in

[10] GPT-4 shows simlilar performance to GPT-4o in our LLM judge metrics, and this experiment was done earlier than GPT-4o's release.

29% of cases, mainly due to factual inaccuracies in GPT-4 responses. GPT-4 outperformed humans in 21% of instances; these cases were mostly related to papers whose topics were outside the authors' expertise. However, 18% of both answers were rejected as unsatisfactory, particularly for complex questions. Detailed performance metrics are available in the Appendix Table 7.

6 Related Work

Manually Curated Scientific QA Datasets: The QASPER dataset (Dasigi et al., 2021) involves NLP practitioners creating questions from paper titles/abstracts, with answers derived from full-texts by separate annotators. The QASA dataset (Lee et al., 2023) is generated by AI/ML practitioners and paper authors who formulate surface, testing, and deep questions. In contrast, the COVID-QA dataset (Möller et al., 2020) is crafted by 15 biomedical experts, who develop questions and annotate corresponding text as answers, focusing on COVID-19 research. QASPER has 40% questions answered in less than five words, while 30% of QASA QA pairs are sourced from only the introductions and abstracts, with 52% of answers showing high unigram overlap with the text, indicating easier retrieval. The ExpertQA dataset (Malaviya et al., 2024) features 2,177 questions across 32 fields, created by 524 experts to simulate complex, web-based information-seeking scenarios. BioASQ-QA dataset (Krithara et al., 2023; Tsatsaronis et al., 2015) involves expert-curated questions ranging from yes/no, factoid, list, and summary types, growing from 310 to 4,721 instances over ten years. Since 2016, BioASQ-QA has focused solely on titles and abstracts, reflecting the high effort in manual curation.

Synthetically Generated Scientific QA Datasets: BioRead (Pappas et al., 2018) and BioMRC (Pappas et al., 2020) are cloze-style biomedical MRC datasets that utilize text entities as answers, masking these entities in texts (passages in BioRead, abstracts in BioMRC) and forming questions from the last passage line or title. ScholarlyRead (Saikh et al., 2020) generates QA pairs by extracting noun phrases from abstracts and using a question-generation model. As shown in Table 1, these synthetically generated QA datasets generally feature shorter answers than ours. PubMedQA (Jin et al., 2019) starts with a labeled dataset where the title is a question and the last abstract line is the answer, creating 1000 instances with short answers (yes/no/maybe) annotated based on the abstract. Its synthetic counterpart uses syntax heuristics and modification rules to craft similar QA pairs.

Other datasets: The ARIES dataset (D'Arcy et al., 2023) compiles review comments and associated paper edits. Its synthetic subset uses a method similar to ours to extract comment-edit pairs based on textual overlap. Our dataset diverges by extracting questions from review comments using LLMs, not just from quoted responses but also from author rewrites. We employ human-expert annotation to refine questions and answers, avoiding reliance solely on textual overlap. This allows us to include high-quality queries involving tables, equations, and multi-paragraph reasoning. In a parallel direction, Kang et al. (2018) collect peer review datasets for paper acceptance prediction and score prediction for review aspects tasks.

SciDQA stands out among QA datasets as its questions are sourced directly from the peer review process, ensuring they are natural, evaluative, and of high quality due to the scientific discourse among domain experts. This sourcing ensures that the questions require a deep understanding of the content, emphasizing depth as well as quality.

7 Conclusion and Future Work

We curate SciDQA, dataset designed to challenge language models on the QA task aiming to evaluate their understanding of scientific papers. The dataset consists of 2937 QA pairs, and extracts QA asked by reviewers and answered by paper authors during reviewer-author discussion during manuscript review on OpenReview. Our multi-stage refinement pipeline annotates for quality, decontextualizes the QA pairs, edits references, and establishes the source document from different manuscript versions. Our dataset features questions necessitating reasoning across multiple modalities beyond mere text, including figures, tables, equations, appendix and supplementary materials. SciDQA also provides a testbed for evaluation of multi-document comprehension properties of LLMs. We evaluate the performance of several open-source and proprietary LLMs in generating the answer to questions after comprehending the research paper. We posit that SciDQA will serve as a useful resource to benchmark the performance of LLMs in scientific text comprehension.

8 Limitations

Multiple questions in our dataset necessitate comprehension and reasoning over multiple documents. The questions in the dataset often mention the reference text for previous works that need to be referred to for answering the question. However, in our experiments we do not search and include those documents for answer generation. Additionally, 7% questions are dependent on figures, but the Nougat parser does not extract images and only extracts the figure captions. We do not evaluate any visual or multimodal LLM. We extract figures for the specific figure-related questions using PDFFigures (Clark and Divvala, 2016), summarize it using Llama 3.2 and make that available. Large-scale evaluation of free-form generation is still challenging. We provided both surface-level and LLM-as-a-judge metrics to show the full picture of the performance, however, extensive meta-evaluation studies might be needed to carefully understand the limitations of such metrics in our setting. Additionally, the dataset could be used to generate difficult questions from a manuscript. Our dataset does not have any judgment statements about paper acceptance/rejection. However, the questions dataset could still be used for training a question generator, and complex questions could be misused by reviewers as grounds for rejection. Another limitation is disentangling the effect of potential contamination from performance of various evaluated models, which is difficult to do for models that don't discuss their training data (which includes majority of both closed and open weight models). Finally, similar to other existing datasets, our dataset focuses on curating QA pairs from a specific domain (machine learning), rather than all scientific fields of study.

Acknowledgments

We extend our gratitude to Mike D'Arcy for providing feedback and valuable discussion on the paper. We also thank the anonymous reviewers and area chairs for their feedback. We are grateful for the compute support provided by the Microsoft Research's Accelerate Foundation Models Research (AFMR) program and Google's TRC program. Shruti is grateful for the support received through the Fulbright-Nehru Doctoral Research Fellowship to visit Yale University. Shruti is also supported by the Prime Minister's Research Fellowship (PMRF-1701251) awarded by the Government of India.

References

2008–2024. Grobid. https://github.com/kermitt2/grobid.

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*.

Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Mérouane Debbah, Étienne Goffinet, Daniel Hesslow, Julien Launay, Quentin Malartic, et al. 2023. The falcon series of open language models. *arXiv preprint arXiv:2311.16867*.

Lukas Blecher, Guillem Cucurull, Thomas Scialom, and Robert Stojnic. 2023. Nougat: Neural optical understanding for academic documents. *Preprint*, arXiv:2308.13418.

Christopher Clark and Santosh Divvala. 2016. Pdffigures 2.0: Mining figures from research papers. In *2016 IEEE/ACM Joint Conference on Digital Libraries (JCDL)*, pages 143–152.

CORE. Icore conference portal. https://portal.core.edu.au/conf-ranks/. Accessed: 2024-04-15.

Mike D'Arcy, Alexis Ross, Erin Bransom, Bailey Kuehl, Jonathan Bragg, Tom Hope, and Doug Downey. 2023. Aries: A corpus of scientific paper edits made in response to peer reviews. *arXiv preprint arXiv:2306.12587*.

Pradeep Dasigi, Kyle Lo, Iz Beltagy, Arman Cohan, Noah A Smith, and Matt Gardner. 2021. A dataset of information-seeking questions and answers anchored in research papers. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4599–4610.

Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. 2019. DROP: A reading comprehension benchmark requiring discrete reasoning over paragraphs. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2368–2378, Minneapolis, Minnesota. Association for Computational Linguistics.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Google. 2023. Palm text-bison api. https://cloud.google.com/vertex-ai/generative-ai/docs/model-reference/text.

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

Mojan Javaheripi, Sébastien Bubeck, Marah Abdin, Jyoti Aneja, Sebastien Bubeck, Caio César Teodoro Mendes, Weizhu Chen, Allie Del Giorno, Ronen Eldan, Sivakanth Gopi, et al. 2023. Phi-2: The surprising power of small language models. *Microsoft Research Blog*.

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*.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. 2019. PubMedQA: A dataset for biomedical research question answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2567–2577, Hong Kong, China. Association for Computational Linguistics.

Dongyeop Kang, Waleed Ammar, Bhavana Dalvi, Madeleine van Zuylen, Sebastian Kohlmeier, Eduard Hovy, and Roy Schwartz. 2018. A dataset of peer reviews (PeerRead): Collection, insights and NLP applications. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1647–1661, New Orleans, Louisiana. Association for Computational Linguistics.

Anastasia Krithara, Anastasios Nentidis, Konstantinos Bougiatiotis, and Georgios Paliouras. 2023. Bioasq-qa: A manually curated corpus for biomedical question answering. *Scientific Data*, 10(1):170.

Yoonjoo Lee, Kyungjae Lee, Sunghyun Park, Dasol Hwang, Jaehyeon Kim, Hong-in Lee, and Moontae Lee. 2023. Qasa: advanced question answering on scientific articles. In *International Conference on Machine Learning*, pages 19036–19052. PMLR.

Dacheng Li, Rulin Shao, Anze Xie, Ying Sheng, Lianmin Zheng, Joseph Gonzalez, Ion Stoica, Xuezhe Ma, and Hao Zhang. 2023. How long can context length of open-source llms truly promise? In *NeurIPS 2023 Workshop on Instruction Tuning and Instruction Following*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Yixin Liu, Alex Fabbri, Pengfei Liu, Yilun Zhao, Linyong Nan, Ruilin Han, Simeng Han, Shafiq Joty, Chien-Sheng Wu, Caiming Xiong, and Dragomir Radev. 2023. Revisiting the gold standard: Grounding summarization evaluation with robust human evaluation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4140–4170, Toronto, Canada. Association for Computational Linguistics.

Yixin Liu, Kejian Shi, Alexander R. Fabbri, Yilun Zhao, Peifeng Wang, Chien-Sheng Wu, Shafiq Joty, and Arman Cohan. 2024. Reife: Re-evaluating instruction-following evaluation. *Preprint*, arXiv:2410.07069.

Kyle Lo, Joseph Chee Chang, Andrew Head, Jonathan Bragg, Amy X. Zhang, Cassidy Trier, Chloe Anastasiades, Tal August, Russell Authur, Danielle Bragg, Erin Bransom, Isabel Cachola, Stefan Candra, Yoganand Chandrasekhar, Yen-Sung Chen, Evie (Yu-Yen) Cheng, Yvonne Chou, Doug Downey, Rob Evans, Raymond Fok, F.Q. Hu, Regan Huff, Dongyeop Kang, Tae Soo Kim, Rodney Michael Kinney, Aniket Kittur, Hyeonsu B Kang, Egor Klevak, Bailey Kuehl, Michael Langan, Matt Latzke, Jaron Lochner, Kelsey MacMillan, Eric Stuart Marsh, Tyler C. Murray, Aakanksha Naik, Ngoc-Uyen Nguyen, Srishti Palani, Soya Park, Caroline Paulic, Napol Rachatasumrit, Smita R Rao, Paul Sayre, Zejiang Shen, Pao Siangliulue, Luca Soldaini, Huy Tran, Madeleine van Zuylen, Lucy Lu Wang, Christopher Wilhelm, Caroline M Wu, Jiangjiang Yang, Angele Zamarron, Marti A. Hearst, and Daniel S. Weld. 2023. The semantic reader project: Augmenting scholarly documents through ai-powered interactive reading interfaces. *ArXiv*, abs/2303.14334.

Chaitanya Malaviya, Subin Lee, Sihao Chen, Elizabeth Sieber, Mark Yatskar, and Dan Roth. 2024. ExpertQA: Expert-curated questions and attributed answers. In *2024 Annual Conference of the North American Chapter of the Association for Computational Linguistics*.

Timo Möller, Anthony Reina, Raghavan Jayakumar, and Malte Pietsch. 2020. COVID-QA: A question answering dataset for COVID-19. In *Proceedings of the 1st Workshop on NLP for COVID-19 at ACL 2020*, Online. Association for Computational Linguistics.

OpenReview. 2023. Openreview. `https://openreview.net/about`. Accessed: 2024-04-15.

Srishti Palani, Aakanksha Naik, Doug Downey, Amy X. Zhang, Jonathan Bragg, and Joseph Chee Chang. 2023. Relatedly: Scaffolding literature reviews with existing related work sections. *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*.

Dimitris Pappas, Ion Androutsopoulos, and Haris Papageorgiou. 2018. BioRead: A new dataset for biomedical reading comprehension. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Dimitris Pappas, Petros Stavropoulos, Ion Androutsopoulos, and Ryan McDonald. 2020. BioMRC: A dataset for biomedical machine reading comprehension. In *Proceedings of the 19th SIGBioMed Workshop on Biomedical Language Processing*, pages 140–149, Online. Association for Computational Linguistics.

Amy Pu, Hyung Won Chung, Ankur Parikh, Sebastian Gehrmann, and Thibault Sellam. 2021. Learning compact metrics for MT. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 751–762, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Adam Roberts, Colin Raffel, and Noam Shazeer. 2020. How much knowledge can you pack into the parameters of a language model? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5418–5426, Online. Association for Computational Linguistics.

Anna Rogers, Matt Gardner, and Isabelle Augenstein. 2023. Qa dataset explosion: A taxonomy of nlp resources for question answering and reading comprehension. *ACM Computing Surveys*, 55(10):1–45.

Tanik Saikh, Asif Ekbal, and Pushpak Bhattacharyya. 2020. ScholarlyRead: A new dataset for scientific article reading comprehension. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 5498–5504, Marseille, France. European Language Resources Association.

Tanik Saikh, Tirthankar Ghosal, Amish Mittal, Asif Ekbal, and Pushpak Bhattacharyya. 2022. Scienceqa: A novel resource for question answering on scholarly articles. *International Journal on Digital Libraries*, 23(3):289–301.

Ross Taylor, Marcin Kardas, Guillem Cucurull, Thomas Scialom, Anthony Hartshorn, Elvis Saravia, Andrew Poulton, Viktor Kerkez, and Robert Stojnic. 2022. Galactica: A large language model for science. *arXiv preprint arXiv:2211.09085*.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, et al. 2024. Gemma: Open models based on gemini research and technology. *arXiv preprint arXiv:2403.08295*.

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

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

George Tsatsaronis, Georgios Balikas, Prodromos Malakasiotis, Ioannis Partalas, Matthias Zschunke, Michael R Alvers, Dirk Weissenborn, Anastasia Krithara, Sergios Petridis, Dimitris Polychronopoulos, et al. 2015. An overview of the bioasq large-scale biomedical semantic indexing and question answering competition. *BMC bioinformatics*, 16:1–28.

Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, et al. 2023. Zephyr: Direct distillation of lm alignment. *arXiv preprint arXiv:2310.16944*.

Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2024. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36.

A Curation of SciDQA- Data Pre-processing, Annotation and Editing

Curation from OpenReview We selected top-tier machine learning and deep learning venues, designated as A* rankings by ICORE Portal (CORE), with publicly accessible reviewer-author discussions on OpenReview. During the dataset compilation phase, NeurIPS, ICLR, ICML, and TMLR were the A* venues with available discussions. However, only discussions from ICML workshop papers and accepted papers from TMLR were accessible, with rejected papers from TMLR not being included. To ensure high quality, we excluded ICML workshop papers. Further, TMLR was also excluded to maintain diversity and avoid a narrow focus on only accepted papers. We curate 11400 papers from ICLR (2018-2022) and NeurIPS (2021-2022), with major focus to include newer papers to decrease the risk of contamination with LLM pretraining datasets.

PDF to Text Conversion OpenReview portal hosts the multiple versions of PDF files for papers submitted to ICLR and NeurIPS, which also includes the versions uploaded during the discussion phase. We downloaded the last manuscript submitted prior to the conference deadline, and refer to it as the initial version, as well as the final manuscript, known as the camera-ready version. In case of rejected manuscripts, the camera-ready version is not uploaded, and hence, we either take the latest version submitted during discussion with reviewers,

or take the initially submitted manuscript. For converting PDFs to text, we employed Nougat (Blecher et al., 2023), a visual transformer model designed for the optical character recognition (OCR) task. Nougat parses research paper PDFs into markdown format and has been trained on a dataset comprising papers from arXiv, Pubmed Central, and the Industry Document Library. We opted for Nougat as it is the current state-of-the-art, showcasing superior performance in extracting tables, mathematical text (equations), and general text compared to GROBID (GRO, 2008–2024), another widely used OCR tool.

Regex Filtering OpenReview has nested discussions, i.e. authors and reviewers reply to corresponding messages, creating a time-stamp chain of discussion. Reviewers post the initial review message, generally consisting of paper summary, strengths and weakness, questions to authors, and a recommendation score. Segments of reviewer messages may be quoted in markdown or paraphrased by the authors in their replies to address specific content. Subsequently, reviewers may ask additional clarifying questions based on the authors' responses, or express satisfaction or dissatisfaction. There are instances where, despite the reviewers' questions, the authors do not provide responses. To extract nested discussions containing at least one question and answer, we employed regex pattern matching, searching for cues such as 'Question:', 'Q', etc. Using this method, we extracted 18,658 reviewer-author discussions for 11,400 papers that contained questions and answers. We use the following regex pattern to identify discussions that contain some questions:

Regex for Extraction

```
"que[ 0-9]*?[:-] .*[^\n]"
"Q[ 0-9]*?[:-] .*[^\n]"
"question[ 0-9]*?[:-] .*[^\n]"
"^> .*[^\n]"
```

LLM-based QA Extraction The prompt provided to PaLM text-bison-001 model to extract QA pairs is presented in Figure 3.

Prompt for QA Extraction using PaLM

You are a helpful assistant. Read the following paragraph and find all question-answer pairs in it.

Author Response to Reviewer

Add 'Q:' before each question and 'A:' before answers. The question-answer pairs are:

Figure 3: Prompt for PaLM model to extract question-answer pairs from Reviewer-Author discussions.

A.1 Annotation details

The annotators achieved an 85% agreement rate in filtering the type of questions as relevant, irrelevant or ambiguous. Half of the disagreements pertained to the ambiguous category, with discrepancies arising from one annotator marking instances as 'ambiguous' to speed up annotation versus another favoring detailed assessment. In such cases, the annotation disagreement does not imply disagreement regarding the inclusion of the instance in the dataset. The annotators resolved the remaining disagreements through discussion and refined the annotation guidelines to eliminate ambiguities before proceeding with the rest of the dataset.

The annotation process was facilitated by providing details such as the paper title, submission venue, area chair recommendations, and the extracted questions with their corresponding answers. To minimize the workload, questions from the same paper but different reviewers were assigned to the same annotator. Annotators were encouraged to consult the original review texts for additional context, enhancing the accuracy of their annotations. Some instances of QA pairs that are marked as relevant, irrelevant, or ambiguous are presented in Table 6.

We present scenarios depicting the requirement of editing QA pairs, and the references text to improve dataset quality in Figure 4 and Figure 5.

Source Document Annotation Scenarios depicting cases where initial vs revised manuscripts are appropriate for answering the reviewer questions are presented in Figure 6.

Evidence Extraction We extract evidential paragraphs, figures, tables, and lines in paper text from the author responses. We also extract evidences for a smaller subset of the dataset automatically where there is a high overlap between a paper section and the answer.

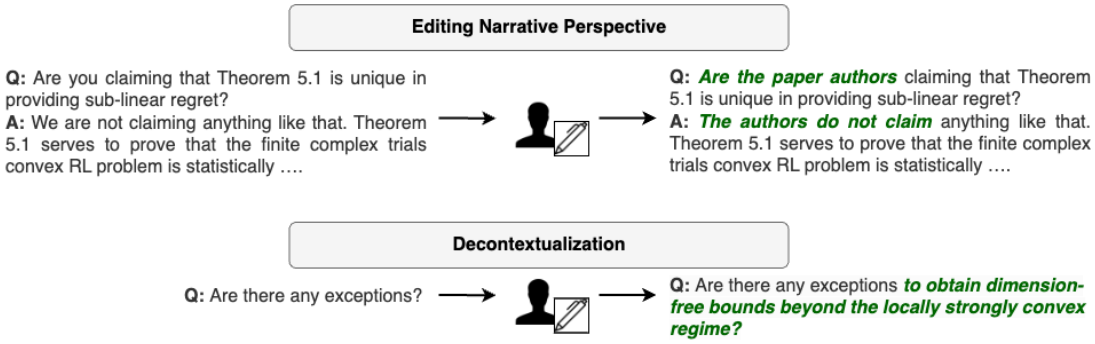


Figure 4: Rewriting QA pairs in a third-person narrative is crucial for models to recognize that questions seek factual answers based on the author's reasoning in the paper, rather than personal opinions. Furthermore, incorporating contextual information enhances the comprehension of questions that necessitate prior contextual knowledge for accurate interpretation.

Relevant for SciDQA
Q: How is the expectation of TCE algorithm computed in Equation 18? A: The expectation is calculated with respect to the ... Q: In section 3.4.1, is it possible to apply ReMERT to non-episodic or continuing task? A: ReMERT might not provide proper weights to ....
Irrelevant for SciDQA
Q: Can the inversion method by Chen et al. 2022 be used to improve the latency? A: We believe that this may be possible, however it will require further analysis. Q: Can you correct the typos in Section 3.4? A: Yes, we will correct them in the revised version.
Ambiguous
Q: Can this inversion method be used in tandem with online filtering/smoothing (e.g. 4DVar, EnKF)? A: We believe that this may be possible, potentially leveraging ideas from Chen et al. [2021].
Q: Why don't the authors compare to PINNs? A: PINNs are typically employed to retrieve individual solutions, not learn distributions over data sets. When using them to solve the individual problems, inference is much slower since the network needs to be trained for each inferred solution. Iterative solvers seem like a better alternative in our setting.

Table 6: Categorization of questions for inclusion in the SciDQA dataset. Information-seeking questions, whose answers are ascertainable within the research paper text, from a collection of synthetically extracted question-answer pairs using PaLM text-bison-001 model are categorized as relevant, and added to the SciDQA dataset.

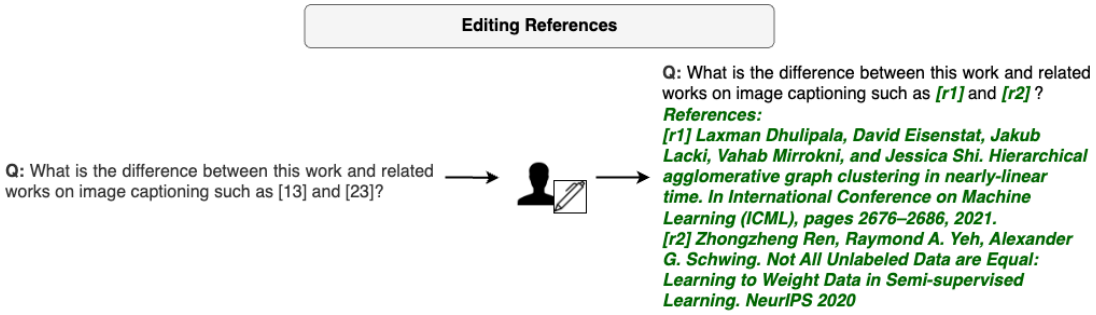
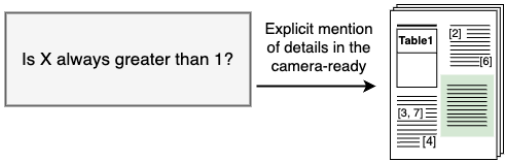
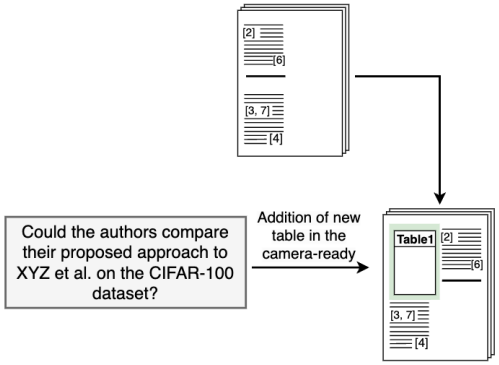


Figure 5: References in question and answer texts are uniformly renumbered (e.g., r1, r2, or 1, 2, or A, B) to preclude the LM from leveraging specific reference markers as shortcuts for answer retrieval. To facilitate accurate answer formulation by the LM, textual information pertaining to paper references is incorporated into questions, deterring reliance on mere reference numbers. Similarly, references in answers are renumbered and supplemented with the relevant reference text as necessary.



(a) Initial version is preferred as the revised copy explicitly mentions the answer.



(b) Revised version is preferred as the answer is originally absent.

Figure 6: We present scenarios where the initial and the revised manuscript versions are most appropriate for answering the reviewer's question. For each question in the dataset, we annotate the preferred manuscript version.

Figure 7: Priming LLMs with Questions (`closed-book`). This task evaluates the ability of LLM to recall the answer without any relevant context.

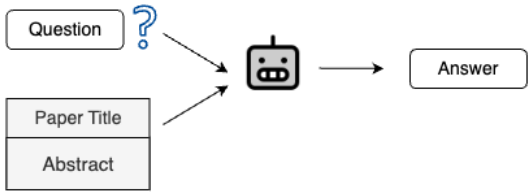


Figure 8: Open-Domain Question Answering - Priming with Question and Title/Abstract (`title-abs`). This task evaluates the impact of additional context on LLM ability to recall the answer without reasoning about the question.

B Experiments

B.1 Experimental Setup

We present figures for the experimental setup of the following:

1. Open-Domain Question Answering - Priming with Questions (`closed-book`) in Figure 7.
2. Open-Domain Question Answering - Priming with Question and Title/Abstract (`title-abs`) in Figure 8.
3. Retrieval Augmented Generation (RAG) in Figure 9.
4. Comprehending the full-text (`full-text`) in Figure 10. The figure presents the scenario where the full-text cannot fit into the models' context length.

We experimented with the parameters (temperature=0.1, 0.9, top_p=0.1, 0.5, 0.9) on a smaller subset of 20 QA pairs, and selected temperature=0.1 and top_p=0.9 after manually inspecting LLM answers. We carried out three runs initially, but upon observing no significant difference in performance, we reported the final numbers in the paper using a single run.

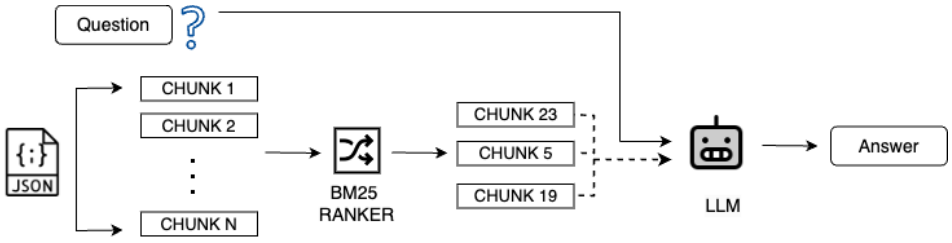


Figure 9: RAG setup ranks paper subsections based on their relevance to the question, and top-3 subsections are provided to the base-LLM, which generates the answer.

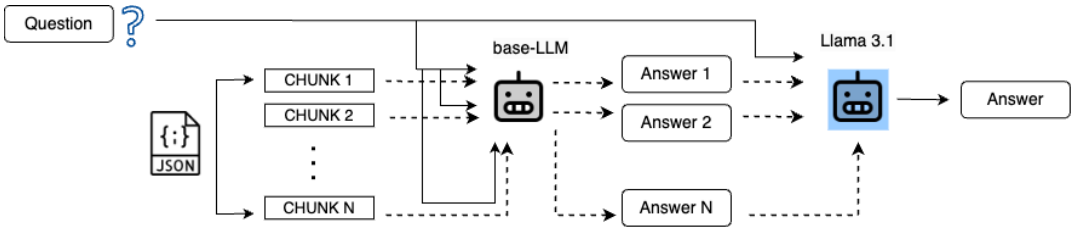


Figure 10: Comprehending the full-text (`full-text`) of the research paper by passing model context-length segments to the base-LLM and generating answers from each segment. Llama 3.1 70B selects the best answer among generated candidate answers. For models with 128k context length, such as Gemini, GPT-4o, and GPT-4o-mini, the entire text is provided to the base-LLM in a single chunk and answer selection phase is not required.

B.1.1 Chunk Creation Algorithm

Chunking for RAG Setup RAG setup ranks top-k chunks from the full-text which are then provided to the LLM to generate the answer. The chunking strategy is presented in Algorithm 1, and ensures that the individual chunks fit into the model context length. It also ensures that the collective top-k chunk lengths also fit the model context length, and sentences from different paper sections or paragraphs are not collated together in a single chunk. We found this setting to work better than naive chunking and truncating by paragraphs or sections.

Chunking for FT Setup In the `full-text` setting, the chunk length is determined by the LLM's context length. If the model context length is N, we reserved 500 tokens for the instruction and the question, and utilized the rest N - 500 tokens for context. The chunking strategy is presented in Algorithm 2.

For Gemini, GPT-4o, GPT-4o-mini, and GPT-4; chunking is not required and the answer selection phase is not included in final answer generation. For models with 128k context-length, Qwen v2.5 models (1.5B and 7B) and Llama 3.1 70B models (8B and 70B), the prompt with entire full-text does not fit on our GPUs, so we create chunks as it is done with other smaller context-length LLMs. However, for `full-text` generations with Qwen v2.5 and Llama 3.1 models, the base-LLM is used for final answer selection. With other base-LLMs, all generated candidate answers concatenated together exceed the context-length of base-LLM so Llama 3.1 70B is used for final answer selection.

Algorithm 1 Chunk Creation Algorithm for RAG

Input: Full-text document
Output: List of chunks
Split the full-text into paragraphs (demarcated by \n).
for each paragraph P **do**
    Split P into individual sentences (using the NLTK library).
    Initialize an empty list $chunks$
    for every 10 consecutive sentences in P **do**
        Join the sentences to build a chunk.
        Add the chunk to $chunks$
        Slide the window by nine sentences (i.e., keep a single overlapping sentence between consecutive chunks).
    end for
end for

B.2 Answer Selection Prompt for Llama 3.1 70B

The prompt provided to Llama 3.1 70B model to generate a single answer during the answer selection phase in `full-text` setup is presented in Figure 11.

Algorithm 2 Chunk Creation Algorithm for `full-text`

Split the full-text into paragraphs (demarcated by \n).
for each paragraph **do**
  if the length of paragraph is less than $N - 500$ **then**
    The entire paragraph is treated as a chunk
  else
    Split the paragraph into a list of sentences, say $S = [s_1, s_2, \ldots, s_n]$
    Initialize an empty $chunks\_list = [\,]$
    Initialize an empty string chunk $c = ""$
    for sentence s in S **do**
      if $token\_count(c) + token\_count(s) < N - 500$ **then**
        Add sentence s to the chunk c
        continue
      else
        Add chunk c to the $chunks\_list$
        Reinitialize the empty chunk c
        Add sentence s to the chunk c
        continue
      end if
    end for
  end if
end for

Answer Selection Prompt - Llama 3.1

You are provided with a question and some potential answers about a research paper submitted to a top-tier computer science conference in the domain of ML and DL. Your task is to select the best answer from the provided answer options, which comprehensively answers the question. Do not include any additional text other than the answer and select only one answer from the provided options.

Figure 11: Prompt provided to Llama 3.1 70B to select one candidate answer among multiple candidates generated from multiple chunks.

B.3 LLM Judge Prompts

The prompt provided to models to generate an evaluation of relevance, accuracy, completeness, and conciseness aspects is presented in Figure 12. The prompt provided to the Llama 3.1 8B model to extract the overall score from the evaluation statement is presented in Figure 13.

Preferred Answer → / Score ↓	Tie	Human	GPT-4	None
GPT-4	32.5	30.4	37.0	34.6
Human	34.8	34.4	38.5	34.0

Table 7: Average scores of Human and GPT-4 generated answers on a subset of SciDQA dataset across instance categories. The average score (R-1, R-2, R-L, BL, BS) of human and GPT-4 generated answers are grouped by preference.

B.4 Comparison with human-written answers

Table 7 demonstrates the comparison of human-written answers with GPT-4. We present the average scores of surface-level metrics for GPT-4 answers and human-written answers, which are further grouped by categories that indicate which answer was preferred.

Prompt for LLM Judge

You are an expert evaluator tasked with assessing the quality of a model-generated answer compared to a gold standard correct answer in a long-form question-answering context. Your goal is to provide a quantified evaluation across multiple dimensions. Please follow these steps:
Carefully read the original question, the model-generated answer, and the gold correct answer.
Evaluate the model-generated answer on the following dimensions, providing a score from 1-10 for each (where 1 is poor and 10 is excellent): a) Relevance (1-10): How well does the answer address the specific question asked? b) Accuracy (1-10): To what extent is the information provided correct and aligned with the gold answer? c) Completeness (1-10): How thoroughly does the answer cover all aspects of the question compared to the gold answer? d) Conciseness (1-10): Does the answer provide information efficiently without unnecessary details?
Calculate an overall quality score by taking the average of the five dimension scores. In your answer for each dimension, provide a justification why not a higher score and why not a lower score.
Structure your response as follows:
Evaluation:
1. Relevance: [Score] - [Explanation]
2. Accuracy: [Score] - [Explanation]
3. Completeness: [Score] - [Explanation]
4. Conciseness: [Score] - [Explanation]

Overall Quality Score: [Average of the four above scores]

Figure 12: Prompt provided to Llama 3.1 70B model during answer selection phase in `full-text`.

Prompt for Extraction of Scores from LLM Evaluation Statements

You are provided with an evaluation of an answer in the following format:

Evaluation:
1. Relevance: [Score] - [Explanation]
2. Accuracy: [Score] - [Explanation]
3. Completeness: [Score] - [Explanation]
4. Conciseness: [Score] - [Explanation]
Overall Quality Score: [Average of the four above scores].

Carefully read the evaluation provided next, and extract the final overall quality score from the discussion. Do not include any explanation, you should only provide the final numeric score for overall quality from the evaluation statement.

Figure 13: Prompt to extract the final overall quality score from the evaluation statements generated by LLM Judges.