DocHop-QA: Towards Multi-Hop Reasoning over Multimodal Document Collections

Anonymous ACL submission

Abstract

Despite rapid progress in large language models (LLMs), current QA benchmarks still overlook the core challenge of real-world scientific information seeking: synthesizing multimodal evidence scattered across multiple documents and structural formats. Existing QAs remain narrow in scope, relying on unimodal text and short-span reasoning that fail to capture the complexity of real information-seeking. We introduce **DocHop-QA**, a benchmark of 11,379 instances for evaluating multimodal, multi-document, multi-hop scientific QA. Built from publicly available PubMed articles, DocHop-QA incorporates textual passages, tables, and layout cues, enabling cross-document inference without explicit hyperlinks. To scale realistic QA construction, we develop an LLM-driven generation pipeline grounded in 11 scientific reasoning concepts, producing diverse and coherent question–answer pairs. To highlight the dataset's utility and versatility, we propose a task-driven evaluation framework spanning four settings, including structured index prediction, multimodal evidence integration, and generative answering. Experiments show that current models struggle with DocHop-QA's long-context, multi-evidence demands, establishing it as a rigorous testbed for advancing next-generation scientific QA systems.

1 Introduction

Recent advances in Large Language Models (LLMs) have substantially improved Question Answering (QA) performance, particularly in single-document and short-context settings. However, real-world information-seeking rarely conforms to such simplified conditions. In practice, answering complex questions often requires integrating fragmented evidence across multiple documents and heterogeneous formats, including narrative text, tables, and structured layouts (Yoon et al., 2022). This setting naturally gives rise to *multi-hop QA*, where models must retrieve and reason over multiple pieces of evidence rather than rely on localized textual cues. In scientific domains, for example, answering questions such as "What proteins are involved in pathway X, and what experiments validate their roles?" typically demands synthesizing various evidences across several publications.

Despite recent progress, deploying LLMs for multi-hop QA in realistic document-centric environments remains challenging. These challenges are not merely engineering limitations, but reflect fundamental NLP failure modes that become amplified in long-horizon reasoning. Multi-hop QA is highly sensitive to retrieval quality: omitting a single critical evidence node can break the reasoning chain, while retrieving many loosely related passages often introduces distraction and confusion (Zhu et al., 2025). In addition, scientific documents frequently encode key facts in semi-structured tables, where implicit headers, abbreviations, and layout-dependent semantics make reliable table grounding difficult. As context length increases and reasoning spans more hops, such errors accumulate, and maintaining coherent intermediate states across documents and modalities becomes increasingly fragile (Bai et al., 2025b). Together, retrieval failures, table grounding ambiguity, and long-horizon error accumulation form a recurring error taxonomy in realistic multi-hop QA, yet remain insufficiently captured by existing benchmarks.

Existing multi-hop QA datasets such as HotpotQA (Yang et al., 2018) and MuSiQue (Trivedi et al., 2022) have enabled important progress, but are ill-suited for studying these failure modes. They are predominantly sourced from Wikipedia, restrict answers to short text spans, and assume unimodal, text-only inputs, diverging from real-world documents that are multimodal and semi-structured, with layout cues essential for grounding and composition. Moreover, most datasets rely on manual annotation or remain confined to single-document

settings, limiting scalability and generality. While recent work such as LIQUID (Lee et al., 2023) explores automated QA generation, it remains restricted to text-only, single-document scenarios.

To fill this gap, we introduce DocHop-QA, a large-scale benchmark for complex, multi-hop QA over multimodal, multi-document corpora, constructed from scientific articles. Its generation methodology is domain-agnostic and scalable, making it applicable to a wide range of scientific domains. Constructed from PubMed articles, it includes unstructured text, structured tables, and layout features, without relying on predefined hyperlinks or annotated reasoning chains. To efficiently generate diverse and challenging QA pairs, we present a scalable LLM-driven pipeline based on 11 real-world scientific reasoning concepts. To evaluate the applicability and flexibility of DocHop-QA, we benchmark its performance across four representative QA tasks: (1) BBox Entity Index Extraction, (2) XML Entity Index Extraction, (3) Structured Generative Answering, and (4) Generative Text Extraction. These tasks span both discriminative and generative paradigms and encompass structured prediction, multimodal understanding, and free-form answer generation. Our key contributions are as follows: 1) We introduced **DocHop-QA**, a large-scale benchmark for multi-hop QA over multimodal, multi-document scientific corpora, incorporating text, tables, and layout features. 2) We developed a **scalable, LLM-driven generation pipeline** guided by 11 reasoning concepts inspired by real-world scientific inquiry, enabling the automated creation of diverse and challenging multi-hop QA instances without requiring gold chains or hyperlinks. 3) We **evaluated DocHop-QA across four core QA tasks**, ranging from structured index prediction to generative answering, demonstrating its broad applicability for assessing both discriminative and generative reasoning over textual, structural, and visual information.

2 Related Works

Multi-hop QA datasets Multi-hop QA tasks span multiple documents, pages, and modalities, reflecting a realistic information-seeking process. Early datasets such as QAngaroo (Welbl et al., 2018) and HotpotQA (Yang et al., 2018) introduced multi-hop reasoning by requiring models to combine facts from multiple passages, but remained limited to text-only and short-context. HybridQA (Chen et al., 2020) and WebQA (Chang et al., 2022) extended this space by incorporating semi-structured data such as tables alongside text. However, these datasets draw from Wikipedia or web-derived sources and lack structural complexity and document variety observed in real-world applications. More recent benchmarks, including FanOutQA (Zhu et al., 2024), MEQA (Li et al., 2024), LLeQA (Louis et al., 2024), Loong (Wang et al., 2024) and NOVELHOP (Gupta et al., 2025), further explore multi-hop reasoning under longer contexts and more controlled evaluation settings, for example, by varying hop depth or context length. While these efforts advance evaluation rigor, many multimodal QA benchmarks still rely on explicitly structured representations, such as relational DB(Zhu et al., 2025) and knowledge graphs(Kim et al., 2025) to support complex reasoning. In contrast, our work presents a domain-agnostic, multimodal, and multi-document QA benchmark that scales automatically using LLMs and supports rich, compositional reasoning across various document components and modalities.

Visually Rich Document QA Visually rich document (VRD) QA extends question answering beyond flat text to real-world documents with complex layouts, tables, and figures. Early datasets such as DocVQA (Mathew et al., 2021) and InfographicVQA (Mathew et al., 2022) focus on single page document images, often framed as span extraction or retrieval tasks. SlideVQA (Tanaka et al., 2023) and BLIVA (Hu et al., 2024) introduce multi-hop reasoning, but are primarily studied in single-document settings. More recent benchmarks such as ECG-QA(Oh et al., 2023), SPIQA (Pramanick et al., 2024) and MMVQA (Ding et al., 2024) incorporate scientific papers and richer modality mixes. However, they largely emphasize retrieval-oriented evaluation and offer limited support for multi-document or cross-page reasoning. Moreover, most VRD QA datasets are manually constructed and tailored to specific application settings or document types, which limits their scalability and flexibility. Table 1 compares prior datasets with ours.

Table-Oriented QA Several prior works explore table-centric QA, but they differ fundamentally from our setting. Pal et al. (2024) generate table QA for low-resource languages, yet their method relies on structured web pages and does not evaluate multi-hop or path-level reasoning. Likewise, FORTAP(Cheng et al., 2022) focuses on numerical reasoning over tabular data, but the framework lacks support for textual inference or multimodal evi-

Dataset	M Hop	VRD	#QA	Q Gen	Source	M Doc	Modality
HotpotQA	O	X	113K	H	Wiki	X	Tx
MEQA	O	X	2.2K	LLM + H	WikiEvents	X	Tx
NOVELHOPQA	O	X	3.8K	LLM	Gutenberg[2]	X	Tx
HybridQA	O	X	70K	H	Wiki	O	Tx+Tab
Loong	O	X	1.6K	LLM + H	Industry	O	Tx
FanOutQA	O	X	1K	H	Wiki	O	Tx
DocVQA	X	O	50K	H	Industry	X	Tx+Img
SlideVQA	X	O	14K	H	Slideshare	X	Tx+Img
ECG-QA	X	O	414K	H	PTB-XL	X	Tx+Img
SPIQA	X	O	270K	LLM + H	arXiv	X	Tx+Tab+Img
MMVQA	X	O	263K	LLM + H	PubMed	X	Tx+Tab+Img
Ours	O	O	11K	LLM + H	PubMed	O	Tx+Tab+Img

Table 1: **Comparison of QA dataset benchmarks.** M Hop: Multi Hop, VRD: Visually Rich Document, Q Gen.: Question Generation method (H: Human, LLM: Large Language Model), M Doc: Multiple Document, Modality (Tx: Text, Tab: Table, Img: Image). Further details about Multi-hop and reasoning are in Table 10

dence integration. More recently, Wu et al. (2025) propose MMQA, which evaluates multi-hop reasoning at the schema level over relational tables, rather than document-centric evidence integration. Our DocHop-QA targets document-centric multi-hop reasoning, where evidence spans heterogeneous elements such as paragraphs, tables, and visual layouts across multiple documents.

Our Contribution We introduce DocHop-QA, a new benchmark that enables end-to-end multi-hop reasoning over multimodal, visually rich documents spanning multiple pages and sources. Unlike prior datasets, DocHop-QA supports inference across unlinked documents using layout-aware features, semantic similarity, and compositional evidence[1] from text, tables, and structural formats. Constructed via an automated LLM-powered pipeline and grounded in 11 question concepts, our dataset provides a scalable and realistic foundation for evaluating complex reasoning in next-generation multimodal QA systems.

3 Dataset Construction

We constructed DocHop-QA through a pipeline with multiple stages, designed to generate high quality multi-hop question answer pairs grounded in real world documents. The overall process involves four key stages: (1) proposing diverse question concepts, (2) selecting document pairs suitable for multi-hop reasoning, (3) generating questions and (4) matching answers. As the foundation of our dataset, we leverage the PubMed Central (PMC) open access repository, which contains publicly available academic papers and provides structured XML and PDF versions of biomedical and life science literature. From this corpus, we collected about 2M full-text scientific articles and applied a combination of filtering heuristics and semantic similarity-based pairing strategies(Appendix A.1) to identify suitable document pairs for constructing multi-hop QA. After refinement, we retained around 20K document pairs, covering 9,250 unique documents, for question generation.

3.1 Question Concept Proposal

To enable systematic and scalable multi-hop question generation, we defined a set of structured question concepts grounded in the reasoning patterns frequently observed in PubMed articles. We first curated 90 cross-document seed QA pairs with assistance from NotebookLM and conducted a qualitative structural analysis. This process identified 15 core semantic keywords, such as `Problem`, `Solution`, and `Effect`, that capture fundamental components of scientific questions. Based on frequently co-occurring keyword pairs, we designed *scaffolds*: conceptual templates that represent common reasoning flows such as moving from a `problem` to a `Solution` and then to an `Effect` and serve as guiding structures for prompt formulation. Using these scaffolds, we generated an additional 1,000 multi-hop QA pairs with ChatGPT through prompts tailored to elicit multi-step reasoning. We then consolidated structurally overlapping or infrequent scaffolds and selected 11 representative question concepts, balancing empirical frequency, semantic distinctiveness, and scalability. These concepts function as practical abstractions that support controlled yet expressive large-scale QA generation. We empirically verify that these concepts induce diverse and meaningful reasoning behaviors. As shown in Appendix D.1, distance distributions reveal that DocHop-QA spans a broad range of reasoning difficulty, from local evidence aggregation to integration across distant document segments. While section-level hop distances remain relatively constrained, paragraph-level distances vary widely, including that reasoning depth primarily arises from fine-grained evidence aggregation rather than arbitrary context jumps. Moreover, fan-hop concepts encourage breadth-oriented aggregation across structural positions, whereas chain-hop concepts capture ordered, step-wise transitions be-

[1]For technical precision, we refer to the specific segments extracted from documents (incl. textual passages and tables) as "snippets," while using "evidence" in the broader conceptual sense of information supporting multi-hop reasoning.

[2]Project Gutenberg: public-domain for novels, https://www.gutenberg.org/

Type	Subtype	Keyword	Q Type	Question Concept	Question Example
Paragraph-Oriented (75.3%)	Non-Comparison	<**P**roblem><sep><**S**olution>	PS(1215)	What are the main challenges in [ ], and how can [ ] address these issues?	What are the main challenges in understanding the dual role of the transcription factor Sp3 in tumor progression, and how can the forkhead transcription factor Foxi1 address these issues by regulating the expression of key subunits of the vacuolar H+ ATPase?
		<**P**roblem><sep><**S**olution><sep><**M**echanism>	PSM (720)	What are the challenges in [ ], what solutions exist to address these challenges, and how do these solutions help resolve the issues?	What are the challenges in the relationship between leisure physical activity and the risk of osteoporotic fractures in men, what solutions exist to address these challenges, and how do these solutions help resolve the issues?
		<**P**roblem><sep><**S**olution><sep><**R**esult/effect>	PSR (977)	What are the main challenges in [ ], what strategies can be implemented to address them, and what are the observed effects of these strategies?	What are the main challenges in accurately diagnosing corticobasal syndrome, what strategies can be implemented to address them, and what are the observed effects of these strategies?
		<**P**roblem><sep><**S**olution><sep><**L**imitation>	PSL (1,269)	What are the main challenges in [ ], and how can [ ] address these issues? Additionally, what are the limitations of this approach?	What are the main challenges in developing consumer health vocabulary (CHV) due to the heterogeneity and ambiguity of consumer expressions, and what is the solution to this challenge? Additionally, what are the limitations of this approach?
		<**M**echanism><sep><**E**ffect>	ME (1,155)	How does [ ] contribute to [ ], and what is its impact on [ ]?	How do soy isoflavones contribute to estrogen-like effects, and what is their impact on male breast cancer incidence?
		<**S**tudytopic><sep><**A**dvantage><sep><**L**imitation>	SAL (850)	What are the [ ], what advantages does it offer, and what limitations exist?	What are the Gene Ontology term predictions, what advantages does it offer, and what limitations exist?
	Comparison	<**F**eature><sep><**F**eature>	FF (1,229)	How does [ ] compare to [ ], and what are the key similarities and differences?	How do the effects of inorganic arsenic exposure on children's intellectual function compare to those of fluoride exposure, and what are the key similarities and differences?
		<**S**olution><sep><**S**olution>	SS (1,152)	How do [ ] and [ ] differ in addressing [ ], and which approach is more effective?	How do traditional laboratory and clinical research methods and non-invasive abdominal recordings differ in addressing the prediction of preterm labor, and which approach is more effective?
Table-Oriented (24.7%)	Non-Refer Table	<**P**roblem><sep><**S**olution><sep><**T**able>	PST (874)	What are the main challenges in [ ], what strategies can be implemented to address them? Is there any table additionally describing it? Which table is it and what is the main point of the table?	What are the main challenges in regularized gene selection in cancer microarray meta-analysis, what strategies can be implemented to address them? Is there any table additionally describing it? Which table is it and what is the main point of the table?
	Refer Table	<**T**able><sep><**S**tudytopic><sep><**L**imitation>	TSL (573)	What is the main point of Table [ ] in document [ ], and what is the main topic of the referenced paper? Also, what limitations remain in the referenced paper?	What is the main point of table Quantitative synthesised results in given document, and what is the main topic of the referenced paper? Also, what limitations remain in the referenced paper?
		<**T**able><sep><**S**tudytopic>	TS (1,365)	What is the main point of Table [ ] in document [ ], and what is the main topic of the referenced paper?	What is the main point of table Summary of included study characteristics in given document, and what is the main topic of the referenced paper?

Table 2: **Illustrative examples of the proposed question concepts**, categorized by (1) **Type**, denoting the primary answer modality; (2) **Subtype**, indicating the reasoning structure; (3) **Keyword**, representing the core reasoning elements and semantic placeholders used in question generation; (4) **Q Type**, a shorthand for the placeholder composition; (5) **Question Concept**, the abstract question structure using placeholders; and (6) **Question Example**, a concrete instantiation of each question concept. This design enables controlled, diverse, and semantically grounded multi-hop QA generation across heterogeneous document modalities.[3]

tween snippets. Together, these results demonstrate that the proposed concepts systematically cover both easier and harder multi-hop regimes, enabling controlled and structurally diverse QA generation. The question concepts and examples are summarized in Table 2, with the full generation pipeline described in Section 3.3.

3.2 Document Selection

We selected document pairs that can support meaningful multi-hop QA. To ensure relevance and reasoning depth, we focused on pairs that exhibit either semantic similarity or structural connections, such as overlapping keywords, thematically related abstracts, or citation-based relationships.

Paragraph-Oriented Type For paragraph-oriented question types, document similarity was computed using TF-IDF representations of titles, abstracts, and YAKE (Campos et al., 2020) extracted keywords. We first filtered out pairs without any shared keywords, and then ranked the remaining candidates by the weighted relevance score combining keyword overlap and TF-IDF similarity. This lexical and semantic measure was used to identify document pairs likely to support cross-document reasoning without relying on computationally expensive dense embeddings. BERT-based similarity was reserved for later stages, such as snippet-level answer matching, where deeper semantic comparison is needed (Nuri and Senyurek, 2025). The detailed process is provided in Appendix A.1.

Table-Oriented Type For Table-Oriented types, we introduced additional filtering steps. For the PST type, we retained only document pairs in which at least one table appears in a results section, as these tables typically summarize intervention outcomes. For TS and TSL type, which model reasoning over tables citing external studies, we targeted systematic review articles containing citation-linked tables, using these explicit references as structural evidence of cross-document relevance.

3.3 Question Generation

3.3.1 Paragraph-Oriented Type

Our question generation pipeline has two phases: 1) fine-tune an instruction following model and 2) generate questions via prompt injection. The goal is to

[3] See Fig. 10 for concept-wise distributions of section and paragraph level answer snippet positions.

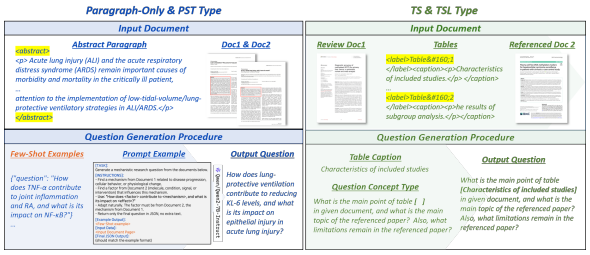


Figure 1: **Question Generation Procedure**. The process begins with **(1) Input documents** proceed via **(2) Question generation procedures** by type: Paragraph-Oriented types and PST type use LLMs; TS/TSL types follow a deterministic caption-insertion routine.

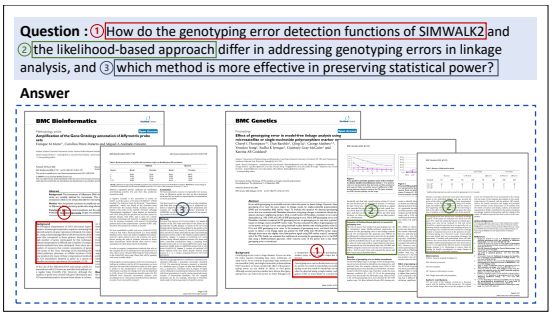


Figure 2: **Paragraph-Oriented QA with fan-hop reasoning.** Answer snippets are independently retrieved across documents and aggregated without explicit link.

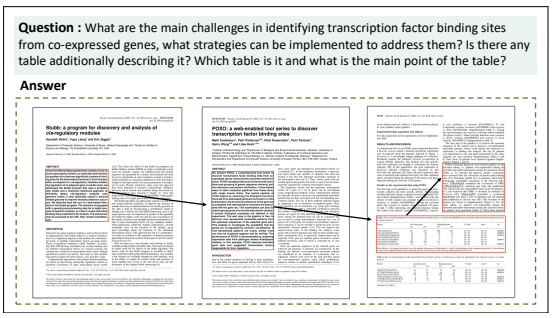


Figure 3: **Table-Oriented QA with chain-hop reasoning.** Answer snippets are retrieved step-by-step, linking descriptions with table evidence across documents.

create logically consistent multi-hop questions that weave document-specific information into question concepts. Training instances are JSON objects with instruction-based question concepts, paired abstract contexts, and gold answer instances. The instruction directs the model to replace the placeholders with information from the supplied contexts (Examples in Appendix A.2).

We fine-tuned the Qwen2-Instruct[4] model, chosen for its strong contextual reasoning, modest resource footprint, and permissive license, using Unsloth's 4-bit LoRA framework(Hu et al., 2021). To preserve controllability and semantic consistency, we trained separate models for each question category (comparison, non-comparison, and PST). Joint training proved suboptimal, frequently omitting contrastive cues in comparison questions and producing fragmented reasoning in non-comparison and PST cases. During inference, we supplied a few-shot prompt that includes the concept, 2 new abstracts, and 2 in-context example questions using a step-by-step reasoning skeleton shown in Figure 1.

Table-Oriented Type For the PST type, questions were generated using the same model-based approach as the Paragraph-Oriented type, but differed in the composition of the input to better reflect the table content. For the TS and TSL types, we did not use an LLM for question generation. Instead, we designed a deterministic procedure that constructs questions by directly inserting table captions into predefined placeholders. As shown in Figure 1, when a table was identified in the source document, its caption replaced the corresponding placeholder in the question concept. This design was chosen because the reasoning hops for these types are structured and explicitly defined. As a result, it allows greater controllability in question generation. We apply post processing filters to remove invalid generations for both Paragraph-Oriented and Table-Oriented types (Appendix A.3).

3.4 Answer Matching

In DocHop-QA, answer retrieval is guided by two distinct reasoning patterns[5]: *fan-hop* for Paragraph-Oriented types, where semantically related content is aggregated from documents (Zhu et al., 2024), and *chain-hop* for Table-Oriented types, which require explicit links between structured and unstructured content (Xu et al., 2021; Chen et al., 2019). Following the reasoning patterns illustrated in Figures 2 and 3, our matching pipeline adopts unified retrieval for Paragraph-Oriented types and tailored strategies for Table-Oriented ones.

Paragraph-Oriented Type For Paragraph-Oriented types, we extracted the top 5 semantically relevant snippets from documents using a weighted combination of TF-IDF and BERTScore. Then, we applied a similarity filter to remove low-quality snippets and retained only those QA pairs whose source content was sufficiently coherent. This

[4] https://huggingface.co/unsloth/Qwen2-7B-Instruct-bnb-4bit

[5] Further details are provided in Appendix A.4

improves evidence fidelity by aligning question intent with contextual meaning, without explicit document linking.

Table-Oriented Type For Table-Oriented types, we constructed up to four snippets per instance, connecting problem and solution paragraphs, tables, and referenced content across documents. To enhance retrieval precision, we combined semantic similarity with keyword filtering and applied logical chaining when anchor references were present. This structure reflects real-world scientific reasoning, and instances with low similarity were filtered to preserve dataset integrity. Detailed retrieval logic is provided in Appendix A.4.

4 Dataset Quality Assurance

To evaluate the quality of the DocHop-QA dataset, we define three criteria: ***Reality/Fluency*** ensures that questions are natural, plausible, and meaningful within the context. ***Accuracy*** assesses whether each snippet meaningfully addresses the question and contributes to forming a complete answer. ***Completeness*** evaluates whether the retrieved snippets collectively form a sufficient answer.

Human-LLM Quality assurance To ensure that dataset quality is not an artifact of automated self-evaluation, we place human judgment at the center of our quality assurance process. We first conduct a human evaluation on 50 carefully selected QA instances, covering all 11 question concepts, both paragraph- and table-oriented types, and diverse hop patterns. The sample intentionally includes both straightforward and challenging cases, and is designed to validate evaluation trends rather than to estimate absolute dataset quality. Human annotations are then compared against evaluations from three instruction-tuned LLM judges: Qwen-7B-Chat, GPT-4o, and Gemini 2.5 Flash, using the same 4-point rubric across *Reality/Fluency*, *Accuracy*, and *Completeness*. As shown in Figure 4, human judgments exhibit strong agreement with all three LLMs, with cosine similarities of 0.987 (GPT-4o) and 0.969 (Gemini 2.5 Flash), indicating that the observed quality patterns are not driven by model-specific bias. Based on this human-grounded validation, we extend the same evaluation protocol using Qwen-7B-Chat to the full dataset for scalability. The full-dataset results (Figure 5) show consistently high scores across all criteria, suggesting that the quality trends confirmed by human evaluation generalize to the entire DocHop-QA corpus. Detailed analyses are provided in Appendix C.

Figure 4: **Evaluation comparison on 50 QAs. (Human : pink, Qwen : blue, Gemini : green, GPT : yellow)** Cosine similarity between human and LLM evaluations (Gemini-2.5 Flash: 0.969; GPT-4o: 0.987).

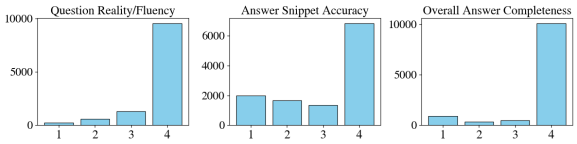
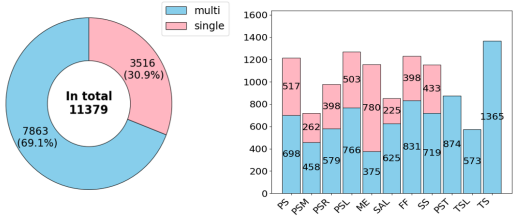


Figure 5: **Qwen evaluation across the full dataset**

5 Dataset Analysis

DocHop-QA contains evenly distributed 11 question concepts, each representing approximately 10% of dataset shown in Table 2 and Figure 29.



(a) Used Document Type (b) QA Distribution

Figure 6: **Overview of document usage across question types.** Multi-document reasoning dominates overall.

Used Document Analysis Our dataset comprises multi-page scientific documents with numerous tables, figures, and complex structures with multiple sections and subsections. As shown in Figure 7, most documents span 10 pages, contain 20 subsections and 60 paragraphs totaling 1,500 to 2,500 tokens closely reflecting the scale and complexity of real-world scientific literature. Figure 6 shows that the dataset includes both single- and multi-document instances across QA concept types.

Reasoning Hop Analysis We analyze answer snippet dispersion and reasoning depth to verify that our question concepts induce diverse, non-position-biased reasoning behaviors. Fan-hop questions exhibit broadly distributed evidence across documents, encouraging global aggregation, while chain-hop questions mainly involve local, stepwise transitions with occasional long-range dependencies. A structural analysis of chain-hop transitions further shows highly diverse hop patterns

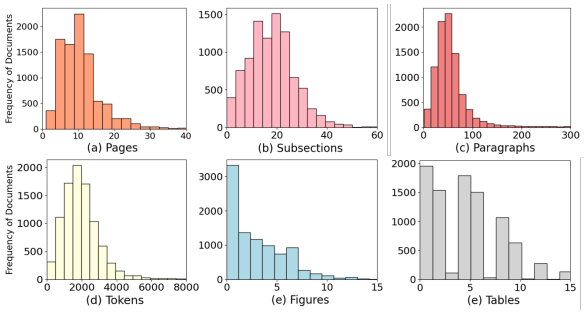


Figure 7: **Distribution of document component types.** DocHop-QA includes multi-page, multi-section VRDs, supporting complex multi-hop and multimodal QA.

across document sections and documents, where tables frequently function not only as terminal evidence but also as intermediate reasoning bridges. Compared to existing multi-hop QA benchmarks, DocHop-QA shows comparable hop statistics but requires richer semantic integration, as answers often span multiple paragraphs and tables and short localized focus as well. The further details are described in Appendix D.

Additional Dataset Analysis We further analyze our dataset from multiple perspectives to better characterize its linguistic, structural, and semantic properties, with detailed results provided in Appendix E. Specifically, we conduct a **Keyword Analysis** (Appendix E.1) by extracting salient lexical patterns from question texts to examine topical diversity and language usage. We also perform an **Answer snippet Distribution Analysis** (Appendix E.2) to investigate how answer snippets are distributed across document sections, revealing systematic alignment between question types and scientific discourse structure. Finally, we carry out a **Similarity Analysis** (Appendix E.3) to verify semantic coherence across documents and modalities, confirming that the constructed QA instances are grounded in meaningful cross-document and cross-modal relationships.

6 Experiments and Results

To evaluate the utility of DocHop-QA, we conduct experiments across four QA tasks covering different reasoning paradigms and input modalities, including discriminative and generative settings. These tasks assess models' ability to reason over complex, document-rich inputs involving structured index prediction, free-form generation, and multimodal integration. Experimental details, metrics, and ablations are provided in Appendix F.

Model	Sample			Accuracy		
	F1	Re	Pr	Be	Co	Ov
LayoutLMv3	**12.35**	34.97	8.47	0.00	0.04	95.52
LayoutXLM	7.13	4.49	24.06	15.64	0.00	33.96

Table 3: **Results on BBox Entity Index Extraction** reporting sample-level F1, Recall (Re), and Precision (Pr), along with spatial accuracy: Belong (Be), Contain (Co), and Overlap (Ov), for LayoutLMv3 and LayoutXLM.

Task 1: BBox Entity Index Extraction. Formulated as a multi-label classification task, this evaluates how well models can identify the relevant answer-contained bounding boxes (e.g., paragraphs, figures, tables) in OCR document layouts. We used two layout-aware vision-language models, LayoutLMv3 (Huang et al., 2022) and LayoutXLM (Xu et al., 2020), with classifier heads to predict bounding box indices. Evaluation includes spatial accuracy (belong/overlap/contain)[6] and sample-level precision, recall, and F1.

As shown in Table 3, LayoutLMv3 outperformed LayoutXLM, achieving the higher sample-level F1 score of 12.35, demonstrating its better capability in grounding questions to spatially localized bounding box entities. LayoutLMv3 achieved near-perfect overlap accuracy (95.52%), indicating that the model can still retrieve spatially proximate regions relevant to the answer, even if full answer containment is not achieved (0.04% in the "Contain" metric). These findings show DocHop-QA is a valuable benchmark for pushing the limits of document layout understanding in QA settings. It reveals critical challenges in handling long, multi-document inputs, while offering fine-grained bounding box supervision to support model development. Users can use DocHop-QA to design new training strategies (e.g., multi-image stitching, hierarchical encoding) or pretraining tasks that improve entity-level reasoning under spatial and modality constraints.

Task 2: XML Entity Index Extraction. This task predicts relevant XML entity indices directly, assessing how well models can reason over text-structured representations of documents. Evaluation metrics mirror Task 1.

We evaluated multi-label XML entity prediction using BigBird (Zaheer et al., 2020) and Longformer (Beltagy et al., 2020) across concatenated and per-entity setups, with and without CLIP-derived visual features (Radford et al., 2021). The best F1 score (23.24) was achieved using BigBird with both

[6] https://meta-pytorch.org/torcheval/stable/generated/torcheval.metrics.MultilabelAccuracy.html

Setup	Text	Img	Sample F1	Sample Re	Sample Pr	Accuracy Be	Accuracy Co	Accuracy Ov
Concat	BB	–	23.18	18.81	30.45	4.22	0.09	62.21
Concat	LF	–	21.83	20.09	24.12	0.22	0.00	66.17
PerEnt	BB	–	21.25	17.22	27.94	2.07	0.09	58.96
PerEnt	LF	–	22.99	18.71	30.08	3.25	0.00	63.80
PerEnt	BB	CL	**23.24**	21.29	25.77	0.48	0.09	64.98
PerEnt	LF	CL	23.18	18.81	30.45	4.22	0.09	62.21

Table 4: **Results for XML Entity Index Extraction.** We compared concatenated (Concat) vs. per-entity (PerEnt) setups across text-only and text+image CLIP (CL) inputs with BigBird (BB) and Longformer (LF). Metrics follow Table 3.

Model	S	Img	Sample F1	Sample Re	Sample Pr	Accuracy Be	Accuracy Co	Accuracy Ov
InternVL	Z	–	7.03	9.77	7.39	0.88	0.88	28.30
InternVL	Z	✓	4.24	6.72	4.03	0.35	0.75	17.44
InternVL	O	✓	4.34	5.77	4.68	0.44	0.62	17.31
Qwen2VL	Z	–	14.06	51.62	13.69	4.04	34.05	75.48
Qwen2VL	Z	✓	7.92	58.53	7.06	1.54	50.04	69.73
Qwen2VL	O	✓	8.12	18.55	7.41	0.04	9.89	41.08
GPT4o	Z	-	**16.21**	54.12	15.85	5.18	36.86	**79.38**
GPT4o	Z	✓	9.37	60.27	8.91	2.12	52.46	73.15
GPT4o	O	✓	9.15	21.43	8.75	0.97	12.21	45.62

Table 5: **Structured Generative Answering results.** S: Setup (Z: Zero-shot, O: One-shot), Img: Image input.

text and image inputs in the per-entity setup (Table 4), highlighting the value of layout-grounded visual features. Model comparisons reveal complementary strengths: BigBird excels in capturing long-range context in the concatenated setup, while Longformer performs better on segmented inputs. Visual inputs further boost per-entity performance, showing the benefit of aligning text with document layout. DocHop-QA exposes label bias and modality interaction challenges, offering a benchmark for developing models that integrate textual, visual, and positional cues in entity-level reasoning.

Task 3: Structured Generative Answering. For this task, models generate lists of answer entity indices in response to questions, using labeled XML content and optionally associated page images. We compared Qwen2.5-VL (Bai et al., 2025a), InternVL2 (Chen et al., 2024), and GPT-4o[7] in zero-shot and one-shot settings. Structured generation performance is measured by multi-label accuracy and sample-level F1. This task assesses whether generative models can produce structured answer indices from full-document inputs. GPT-4o achieves the highest F1 score (16.21) followed by Qwen2.5-VL (14.06) in a zero-shot, text-only setting (Table 5), showing strong performance without fine-tuning. InternVL2 performs poorly across sets and struggles with consistent output formatting, underscoring the need for task-specific pretraining. Adding document images generally hurts performance due to alignment challenges between text and visual layouts. The structured generative task highlights both the promise and limitations of using instruction-following LLMs for entity-index generation. DocHop-QA provides a challenging testbed for structured reasoning in generative settings, enabling deeper studies of prompt design, modality alignment, and decoding strategies.

[7] https://openai.com/index/hello-gpt-4o/

Task 4: Generative Text Extraction. This tests models' ability to generate natural-language answers given different combinations of text and image inputs. We assessed three prompting styles (question-only, zero-shot, and one-shot) and three input modalities (text-only, image-only, text+image), using models such as InternVL2, QWEN2.5, Gemini-2.5[8], and GPT-4o. Evaluation is based on BLEU (Papineni et al., 2002) and ROUGE (Lin, 2004) scores against gold reference answers. The generative text extraction task assesses the ability of large vision-language models to produce free-form answers from document-level context, without relying on predefined answer indices. GPT-4o demonstrates superior performance (BLEU: 28.92) with full-document ingestion, as opposed to input-processed architectures, where texts and images are incorporated into prompts, consuming valuable input context window. Detailed results are provided in Appendix F.

7 Conclusion

We presented DocHop-QA, a new benchmark designed to evaluate multi-hop reasoning over multimodal, multi-document scientific corpora. Unlike prior datasets, DocHop-QA captures realistic information seeking scenarios by integrating unstructured text, structured tables, and visual layout features, without relying on explicit links. Using an 11 concepts, we generated diverse QAs and evaluated performance on 4 representative tasks. Results highlight critical insights into model behavior, modality integration challenges, and prompt effects, underscoring the need for further research in multi-hop, multimodal reasoning. We hope DocHop-QA facilitates future research toward more robust QA systems for real world documents.

[8] https://cloud.google.com/vertex-ai/generative-ai/docs/models/gemini/2-5-flash

Limitations

DocHop-QA has a few limitations worth noting. First, parts of the dataset are generated using LLM-assisted pipelines, which may introduce stylistic regularities in question formulation despite concept-guided prompting and human validation. Second, we evaluate end-to-end reasoning over long, multi-document contexts without requiring access to explicit intermediate-reasoning annotations. Third, while table-oriented questions reflect common scientific reading behavior, they primarily emphasize high-level semantic interpretation rather than fine-grained numerical or cell-level operations. Finally, our evaluation targets general-purpose models in zero-shot or light prompting settings, leaving task-specific adaptation for future work.

Ethics Statement

DocHop-QA is constructed from publicly available scientific articles in the PubMed Central Open Access corpus. The dataset does not contain personal, sensitive, or private information, and all source documents are used in accordance with their original licenses. The benchmark is designed for research purposes to study multimodal, multi-document reasoning and is not intended for clinical decision-making or medical advice. While the dataset is automatically generated with assistance from large language models, we apply concept-guided generation, filtering, and human validation to mitigate hallucinations and low-quality instances.

References

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, and 8 others. 2025a. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Yushi Bai, Shangqing Tu, Jiajie Zhang, Hao Peng, Xiaozhi Wang, Xin Lv, Shulin Cao, Jiazheng Xu, Lei Hou, Yuxiao Dong, and 1 others. 2025b. Longbench v2: Towards deeper understanding and reasoning on realistic long-context multitasks. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3639–3664.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv:2004.05150*.

Ricardo Campos, Vítor Mangaravite, Arian Pasquali, Alípio Jorge, Célia Nunes, and Adam Jatowt. 2020. Yake! keyword extraction from single documents using multiple local features. *Information Sciences*, 509:257–289.

Yingshan Chang, Mridu Narang, Hisami Suzuki, Guihong Cao, Jianfeng Gao, and Yonatan Bisk. 2022. Webqa: Multihop and multimodal qa. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16495–16504.

Jifan Chen, Shih-ting Lin, and Greg Durrett. 2019. Multi-hop question answering via reasoning chains. *arXiv preprint arXiv:1910.02610*.

Wenhu Chen, Hanwen Zha, Zhiyu Chen, Wenhan Xiong, Hong Wang, and William Wang. 2020. Hybridqa: A dataset of multi-hop question answering over tabular and textual data. *arXiv preprint arXiv:2004.07347*.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, and 1 others. 2024. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24185–24198.

Zhoujun Cheng, Haoyu Dong, Ran Jia, Pengfei Wu, Shi Han, Fan Cheng, and Dongmei Zhang. 2022. Fortap: Using formulas for numerical-reasoning-aware table pretraining. *Preprint*, arXiv:2109.07323.

Yihao Ding, Kaixuan Ren, Jiabin Huang, Siwen Luo, and Soyeon Caren Han. 2024. Mvqa: A dataset for multimodal information retrieval in pdf-based visual question answering. *arXiv preprint arXiv:2404.12720*.

Abhay Gupta, Kevin Zhu, Vasu Sharma, Sean O'Brien, and Michael Lu. 2025. Novelhopqa: Diagnosing multi-hop reasoning failures in long narrative contexts. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 26145–26162.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *Preprint*, arXiv:2106.09685.

Wenbo Hu, Yifan Xu, Yi Li, Weiyue Li, Zeyuan Chen, and Zhuowen Tu. 2024. Bliva: A simple multimodal llm for better handling of text-rich visual questions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, page 2256–2264.

Yupan Huang, Tengchao Lv, Lei Cui, Yutong Lu, and Furu Wei. 2022. Layoutlmv3: Pre-training for document ai with unified text and image masking. In *Proceedings of the 30th ACM International Conference on Multimedia*, MM '22, page 4083–4091, New York, NY, USA. Association for Computing Machinery.

Yunsoo Kim, Yusuf Abdulle, and Honghan Wu. 2025. Biohopr: A benchmark for multi-hop, multi-answer reasoning in biomedical domain. *arXiv preprint arXiv:2505.22240*.

Seongyun Lee, Hyunjae Kim, and Jaewoo Kang. 2023. Liquid: A framework for list question answering dataset generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 13014–13024.

Ruosen Li, Zimu Wang, Son Tran, Lei Xia, and Xinya Du. 2024. Meqa: A benchmark for multi-hop event-centric question answering with explanations. *Advances in Neural Information Processing Systems*, 37:126835–126862.

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

Antoine Louis, Gijs van Dijck, and Gerasimos Spanakis. 2024. Interpretable long-form legal question answering with retrieval-augmented large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 22266–22275.

Minesh Mathew, Viraj Bagal, Rubèn Tito, Dimosthenis Karatzas, Ernest Valveny, and CV Jawahar. 2022. Infographicvqa. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1697–1706.

Minesh Mathew, Dimosthenis Karatzas, and CV Jawahar. 2021. Docvqa: A dataset for vqa on document images. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 2200–2209.

Yusra Nuri and Edip Senyurek. 2025. Research abstracts similarity implementation by using tf-idf algorithm. *IOSR Journal of Computer Engineering (IOSR-JCE)*, 27(1, Ser. 4):04–10.

Jungwoo Oh, Gyubok Lee, Seongsu Bae, Joon-myoung Kwon, and Edward Choi. 2023. Ecg-qa: A comprehensive question answering dataset combined with electrocardiogram. *Advances in Neural Information Processing Systems*, 36:66277–66288.

Vaishali Pal, Evangelos Kanoulas, Andrew Yates, and Maarten de Rijke. 2024. Table question answering for low-resourced Indic languages. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 75–92, Miami, Florida, USA. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, page 311–318, USA. Association for Computational Linguistics.

Shraman Pramanick, Rama Chellappa, and Subhashini Venugopalan. 2024. Spiqa: A dataset for multimodal question answering on scientific papers. *arXiv preprint arXiv:2407.09413*.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. *Preprint*, arXiv:2103.00020.

Ryota Tanaka, Kyosuke Nishida, Kosuke Nishida, Taku Hasegawa, Itsumi Saito, and Kuniko Saito. 2023. Slidevqa: A dataset for document visual question answering on multiple images. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 13636–13645.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2022. Musique: Multi-hop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554.

Minzheng Wang, Longze Chen, Cheng Fu, Shengyi Liao, Xinghua Zhang, Bingli Wu, Haiyang Yu, Nan Xu, Lei Zhang, Run Luo, and 1 others. 2024. Leave no document behind: Benchmarking long-context llms with extended multi-doc qa. *arXiv preprint arXiv:2406.17419*.

Johannes Welbl, Pontus Stenetorp, and Sebastian Riedel. 2018. Constructing datasets for multi-hop reading comprehension across documents. *Transactions of the Association for Computational Linguistics*, 6:287–302.

Jian Wu, Linyi Yang, Dongyuan Li, Yuliang Ji, Manabu Okumura, and Yue Zhang. 2025. Mmqa: Evaluating llms with multi-table multi-hop complex questions. In *The Thirteenth International Conference on Learning Representations*, page 1.

Weiwen Xu, Yang Deng, Huihui Zhang, Deng Cai, and Wai Lam. 2021. Exploiting reasoning chains for multi-hop science question answering. *arXiv preprint arXiv:2109.02905*.

Yiheng Xu, Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, and Ming Zhou. 2020. Layoutlm: Pre-training of text and layout for document image understanding. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '20, page 1192–1200. ACM.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. *arXiv preprint arXiv:1809.09600*.

Wonjin Yoon, Richard Jackson, Aron Lagerberg, and Jaewoo Kang. 2022. Sequence tagging for biomedical extractive question answering. *Bioinformatics*, 38(15):3794–3801.

Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and 1 others. 2020. Big bird: Transformers for longer sequences. *Advances in Neural Information Processing Systems*, 33.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with BERT. *CoRR*, abs/1904.09675.

Andrew Zhu, Alyssa Hwang, Liam Dugan, and Chris Callison-Burch. 2024. Fanoutqa: A multi-hop, multi-document question answering benchmark for large language models. *arXiv preprint arXiv:2402.14116*.

Rongzhi Zhu, Xiangyu Liu, Zequn Sun, Yiwei Wang, and Wei Hu. 2025. Mitigating lost-in-retrieval problems in retrieval augmented multi-hop question answering. *arXiv preprint arXiv:2502.14245*.

A Dataset and Code Implementation Details

Code & Data We released the full implementation and reproduction scripts for Dataset Construction (Section 3), Dataset Quality Assurance (Section 4), and Experiments and Results (Section 6) at the following link.
Access it here: https://shorturl.at/rqfcM

A.1 Document Selection Implementation Details

As mentioned in Section 3.2, we formalized the semantic similarity-based document pairing strategy into a concrete algorithmic procedure, as illustrated in **Algorithm 1**. Candidate pairs are first identified using the following two filters: (1) at least one overlapping keyword, and (2) a TF-IDF cosine similarity score of at least 0.3[9]. After this first-stage filtering, we compute a total matching score for each remaining pair as a weighted combination of normalized keyword overlap and TF-IDF similarity. For each source document, we then kept only the top three scoring partners for downstream question generation, limiting repeated reuse of the same documents and promoting document diversity.

Algorithm 1: **Document Selection**

Input: Source corpus $\mathcal{C} = \{d_1, \ldots, d_N\}$ (PubMed XMLs)
Parameter: Keyword extractor (YAKE) Φ ; TF–IDF vectorizer ψ ; similarity threshold τ ; weight α ; top-L
Output: Paired set $\mathcal{P} \subseteq \{(i, j) \mid 1 \le i < j \le N\}$

$\mathcal{P} \leftarrow \varnothing$; initialize $S_{ij} \leftarrow -\infty$ for all $i < j$
for $i \leftarrow 1$ **to** N **do**
  extract *pmc id*$_i$, *title*$_i$, *abstract*$_i$ from d_i
  fulltext$_i \leftarrow$ *title*$_i \parallel$ *abstract*$_i$
  $\mathcal{K}_i \leftarrow \Phi($*fulltext*$_i)$; remove stopwords
$\mathbf{X} \leftarrow \psi($*fulltext*$_1, \ldots,$ *fulltext*$_N)$
$\mathbf{M} \leftarrow \text{cosine}(\mathbf{X}\mathbf{X}^\top)$
for $i \leftarrow 1$ **to** N **do**
  for $j \leftarrow i + 1$ **to** N **do**
    $C_{ij} \leftarrow \mathcal{K}_i \cap \mathcal{K}_j$
    if $|C_{ij}| > 0$ **and** $\mathbf{M}_{ij} \ge \tau$ **then**
      $s_{\text{kw}} \leftarrow \dfrac{\log(1 + |C_{ij}|)}{\log(1 + \max_k |\mathcal{K}_k|)}$
      $s_{\text{sim}} \leftarrow \mathbf{M}_{ij}$
      $S_{ij} \leftarrow \alpha\, s_{\text{kw}} + (1 - \alpha)\, s_{\text{sim}}$
  $\mathcal{N}_i \leftarrow$ Top-L indices j by S_{ij}
  for all $j \in \mathcal{N}_i$ **do**
    $\mathcal{P} \leftarrow \mathcal{P} \cup \{(i, j)\}$
return $\mathcal{P}$

[9] After testing multiple thresholds (0.1–0.9) and manually inspecting retrieved documents, 0.3 was selected for consistently yielding the most relevant results.

A.2 Question Generation Training Data

We presented training instances for three types of question generation: Non-Comparison, Comparison, and Non-Refer-Table (PST). Each instance follows a unified schema comprising a prompt, an instruction, two abstracts, and the resulting question. The prompt encapsulates the underlying question concept, the instruction clarifies how the concept should be instantiated, and Contexts 1 and 2 provide the input abstracts. Although the schema is identical across types, we train a distinct model for each category because our preliminary experiments revealed that type-specific models generate a higher proportion of valid questions than a single unified model. Representative examples of each question type are shown in Tables 6, 7, and 8.

Prompt
Instruction: How does [ ] compare to [ ], and what are the key similarities and differences?
Chain-of-Thought (internal use only): Generate a question by replacing only the placeholder [ ] in the question concept below using document-specific information derived from the provided contexts. You must use two documents.

Instruction
How does [ ] compare to [ ], and what are the key similarities and differences?

Input : {
Context 1
`Regionalization of pediatric emergency care in the US. The 2006 IOM Report highlighted uneven care. Pediatric emergency services are concentrated in a limited number of children's hospitals and trauma centers. EMS-based categorization models like EDAP exist, but family preference for nearby EDs limits effectiveness.`

Context 2
`Current state of pediatric emergency care in Korea. A 2010 survey showed widespread unpreparedness: many EDs lacked essential pediatric equipment, and consultations were often handled by inexperienced residents. Monitoring for sedated children was frequently unavailable.` }

Output
How do the characteristics of the healthcare systems in the United States compare to Korea, and what are the key similarities and differences?

Table 6: **An example of our dataset instance based on comparison-type multi-hop question generation.** Fields such as prompt, instruction, and context are shown in full.

Prompt
Instruction: What are the main challenges in [ ], and how can [ ] address these issues?
Generate a question by replacing only the placeholder [ ] in the question concept below using document-specific information derived from the provided contexts.
Chain-of-Thought (internal use only): Generate a question by replacing only the placeholder [ ] in the question concept below using document-specific information derived from the provided contexts. You must use two documents.

Instruction
What are the main challenges in [ ], and how can [ ] address these issues?

Input : {
Context 1
`Anaemia is a common haematological disorder in chronic disease, often complicating diagnosis and worsening quality of life and survival. This paper reviews its pathogenesis and potential therapeutic targets.`

Context 2
`Cloning the human EPO gene (1984) enabled ESA therapies for cancer-related anaemia. Trials show ESAs reduce transfusion needs, provide sustained correction, lower transfusion-related risks, and improve blood flow and quality of life.` }

Output
What are the main challenges in iron availability for red cell precursors in cancer-related anaemia, and how can erythropoiesis-stimulating agents address this issue?

Table 7: **An example of our dataset instance based on Non-comparison type multi-hop question generation.** Fields such as prompt, instruction, and context are shown in full.

Prompt
Instruction: What are the main challenges in [ ], what strategies can be implemented to address them? Is there any table additionally describe it? Which table is it and what is the main point of the table?
Chain-of-Thought (internal use only): Generate a question by replacing only the placeholder [ ] in the question concept below using document-specific information derived from the provided contexts. You must use two documents.

Instruction
What are the main challenges in [ ], what strategies can be implemented to address them? Is there any table additionally describe it? Which table is it and what is the main point of the table?

Input : {
Context 1
`Inflammatory diseases, including rheumatic disorders, are a major global health burden. While multiple treatments exist, long-term steroidal or non-steroidal therapy can cause serious adverse effects.`

Context 2
`A randomized, double-blind, placebo-controlled trial enrolled 80 UC patients in remission (Sep 2009–Oct 2010). Participants received SM (140 mg/day) or placebo alongside standard therapy for six months.` }

Output
What are the main challenges in traditional anti-inflammatory in treating inflammatory diseases, what strategies can be implemented to address them? Is there any table additionally describe it? Which table is it and what is the main point of the table?

Table 8: **An example of our dataset instance based on PST type multi-hop question generation.** Fields such as prompt, instruction, and context are shown in full.

A.3 Post-processing and Filtering

To ensure that generated questions strictly follow the intended question concepts and remain usable for downstream QA, we apply a lightweight post-processing step to filter invalid generations. We use the following criteria:

(1) Generation failures. We remove instances where the model produces empty outputs or outputs that do not conform to the expected format (e.g., malformed JSON fields or missing required components).

(2) Question–concept mismatch. We filter out questions that fail to instantiate the target concept, operationalized by missing the required concept keywords/placeholders specified in the corresponding prompt. This step prevents concept drift and enforces consistency between the intended reasoning structure and the generated question.

After applying these filters, we retain only question–context pairs that pass all checks, forming the final dataset used in our experiments.

A.4 Detailed Answer Matching

Reasoning Patterns In DocHop-QA, multi-hop reasoning follows two paradigms. Fan-hop aggregates semantically complementary information from multiple, independently retrieved documents. Each snippet contributes to the final answer without explicit cross-referencing. Chain-hop, on the other hand, performs step-by-step inference where information in one document directly leads to or supports content in another, often forming a logical link-such as when a table summarizes results from an external study.

Paragraph-Oriented Type To ground each QA instance with relevant context, we parsed the full-text files of both documents. The subsection-level content was extracted, and segments with fewer than two sentences were discarded to avoid sparse inputs. For each segment, we computed semantic similarity to the question using a hybrid scoring method: a weighted combination of TF-IDF (weight 0.2) and BERT-based(Zhang et al., 2019) sentence embeddings (weight 0.8). This configuration prioritized deeper semantic understanding while preserving term-level sensitivity. We aggregated similarity scores across both documents and retrieved the top-5 segments with the highest scores. To ensure quality, only answer instances with all retrieved snippets above a similarity score of 0.4 were retained. This threshold was empirically selected based on preliminary validation, which showed that higher similarity was strongly associated with answer plausibility. All filtered QA-context pairs were saved in structured JSON format(Appendix B).

Table-Oriented Type We employed two strategies depending on the snippet's role: some snippets were retrieved via semantic-similarity scoring augmented with keyword-based filtering, whereas others were extracted using predefined logic. To support the semantic-similarity route, we curated three keyword lists targeting problem-, solution-, and limitation-related paragraphs; these lists served as an initial paragraph filter in each context and were then combined with semantic-similarity scoring to select answer snippets. If no paragraphs are filtered we computed similarity scores between questions and all paragraphs.

1) Problem-related Keywords. burden, challenge, gap, unmet need, lack of, insufficient, controversy, uncertainty, poor outcomes, high prevalence, incidence, rising rates, increasing trend, public health concern, problem, barrier, limitation, disparity, inequity, inaccessibility, misdiagnosis, delay, complication, risk factor, vulnerability, deficiency, inconsistency, overuse, underuse, shortage, low adherence, noncompliance, failure, inadequacy, ineffectiveness, complexity, fragmentation, obstacle, hazard, disease burden.

2) Solution-related Keywords. approach, strategy, intervention, treatment, therapy, model, framework, tool, instrument, protocol, program, solution, innovation, novel, effective, efficacy, improvement, enhancement, implementation, guideline, practice, plan, policy, recommendation, management, design, structure, evaluation, algorithm, process, assessment, mechanism, procedure, initiative, rollout, deployment, trial, interdisciplinary, collaborative, multimodal, integrated, systematic, scalable, feasible, evidence-based, optimization.

3) Limitation-related Keywords limitation, limit, lack, weakness, shortcoming, bias.

Logical Answer Chain We designed logical answer chains tailored to each QA type (PST Type and TS/TSL Type), which are explained as follows. Green-colored snippets used the semantic similarity strategy, while others were retrieved via predefined logic. The detailed procedure can be found in Figure 8.

1) PST Type

- **Snippets 1 & 2**: Problem and Solution paragraphs were keyword filtered. The table is assumed to be in the solution document; the problem document provides motivation. We then selected the most relevant paragraph per document using 0.2 * TF-IDF + 0.8 * BERTScore.
- **Snippet 3**: The most relevant table content, selected based on similarity to Snippet 2.
- **Snippet 4**: Paragraph in the document referring the table.

2) TS & TSL Type

- **Snippet 1**: Table metadata from the source document.
- **Snippet 2**: Paragraph referring to the table.
- **Snippet 3**: Abstract of the referenced document, linked via citation markers in the table.
- **Snippet 4**: Paragraph describing the cited document's limitations. Snippet 4 was retrieved via a query-driven similarity search using the prompt "What is the limitation of this paper?" and subsequently keyword-filtered. If the similarity score for snippet 4 fell below 0.4, the snippet was discarded, and the QA instance was reclassified as TS type, containing only the first three snippets.

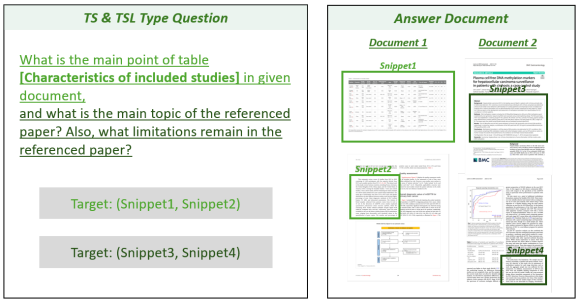


Figure 8: **Answer matching logic for TS & TSL types:** Snippets 1–3 follow a fixed sequence, while Snippet 4 is selected using the same method as in other Q types.

A.5 Computation Efficiency and Scaling Cost

We measured generation and matching costs to assess the computational scalability of our pipeline. As summarized in **Table 9**, the measured question/answer generation time and GPU usage indicate that once prompts are prepared, scaling up the QA set can be done efficiently with moderate resource requirements.This demonstrates that the proposed LLM-driven pipeline is computationally scalable for large-scale QA dataset construction.

B Dataset Description

DocHop-QA is a multi-hop QA dataset built from PubMed articles. Each record contains a single QA pair, along with supporting paragraphs drawn from one or more scientific papers. In total, the dataset comprises 11,359 instances : Paragraph-Oriented (8,547), Table-Oriented (2,812)

Data Structure Each instance is represented as a JSON object with the following fields:

- id: Unique identifier for each QA instance
- task_type: Type
- hop_type: Hop type (“fan" or “chain")
- question_concept_type: 11 Question concept types
- question: The natural language multi-hop question
- used_doc: Indicates if snippets are from a single or multiple documents (“single" or “multi")
- context_list: List of answer snippets

Context List Structure Each element in context_list is a paragraph or table object with the following attributes:

- pmc_id: PubMed Central article ID
- section: Top-level section
- subsection: Subsection title
- type: Type of content (“text" or “table")
- Raw content: Original XML tagged paragraph content
- content: Cleaned plain text content

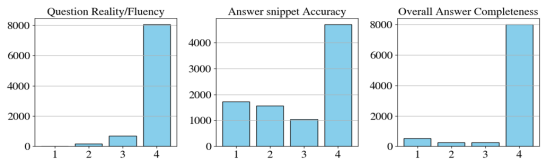
Question Type	Time per Q gen	GPU peak usage (Q gen)	Time per A matching	GPU peak usage (A matching)
Comparison	6.84s	11.82GB	5.34s	0.99GB
Non-comparison	2.22s	12.10GB	—	—
PST	12.56s	14.89GB	0.05s	15.38GB
TSL	24.03s	—	4.45s	—

Table 9: **Computation efficiency of QA generation and matching.** Time and GPU usage measured per question (Q gen) and answer matching (A matching) step for each question type. The results show that once prompts are prepared, QA scaling can be efficiently achieved with moderate GPU memory usage.

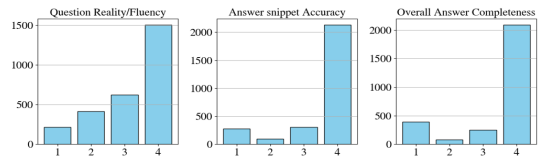
C Dataset Quality Assurance

C.1 LLM-based Dataset Quality Assurance

As shown in **Figure 9**, Paragraph-Oriented QA performed well, with most scores rated 3 or 4. Some instances had lower accuracy, likely due to snippets covering only parts of the question, reducing alignment with its overall intent. Table-Oriented QA showed slightly lower Reality/Fluency due to its format, but still maintained high overall quality. These results confirm that DocHop-QA exhibits robust quality and is suitable for complex QA tasks.



(a) Paragraph-Oriented



(b) Table-Oriented

Figure 9: **LLM based Quality Assurance result**, (1) Question Reality/Fluency (2) Answer Snippet Accuracy (3) Overall Answer Completeness. Scores are based on a 4-point scale (1:poor, 4: excellent).

C.2 Human-based Dataset Quality Assurance

Annotator Demographics Fifteen annotators participated in the human evaluation.The sample was gender-balanced (7F/8M) and included 6 undergraduates, 6 master's, and 3 doctoral students. The participants had academic backgrounds in business (n=3), industrial and management engineering (n=4), computer engineering unrelated to AI (n=2), and AI-related computer science (n=6). The age distribution was as follows: 20 years (n = 8), 30 years (n = 5) and older than 50 years (n = 2).

Institutional Review Board Statement The study was approved by the Institutional Review Board (Exemption Number: PIRB-2025-E020, Date: June 9, 2025).

Evaluation Interface To ensure consistency and transparency in the human evaluation process, annotators were asked to rate each question-answer pair using a structured Google Form interface. Figure 32 shows a screenshot of the actual form used in the study, which included Likert-scale items for three evaluation criteria and mandatory comment fields for low ratings. This setup helped guide annotators through a standardized evaluation flow and facilitated the collection of both quantitative and qualitative feedback.

D Hopping Analysis of Dataset

D.1 Answer Snippets Dispersion and Reasoning Depth

In this section, we quantify how answer snippets are distributed across the document structure to verify that our 11 question concepts yield diverse, non-position-biased snippet layouts (Fig. 11). While fan-hop and chain-hop characterize two complementary reasoning patterns based on answer snippet aggregation and dependency structure, the underlying 11 question concept types span these patterns and induce distinct structural retrieval and traversal behaviors. We report fan-hop and chain-hop results separately using four subplots: *Fan-hop (Absolute)*, *Fan-hop (Ratio)*, *Chain-hop (Element)*, and *Chain-hop (Average)*. For **fan-hop**, we plot *Section/Paragraph Absolute Positional Distance* and *Normalized Section/Paragraph Distance Ratio* over answer snippets (element-wise). For **chain-hop**, we concatenate the two documents into a single sequence to account for cross-document distances, and compute the snippet-to-snippet *distance* between consecutive steps based on snippet's position. We report distances at the element-wise and as an instance-wise average. All box plots retain only values within the interquartile range (IQR; 25th–75th percentile).

Fan-hop Results For fan-hop QA, answer snippets are broadly distributed across the document without strong positional bias, encouraging breadth-oriented aggregation over the document rather than relying on a narrow region. In Fig. 11, *Fan-hop (Absolute)* exhibits broad *Section/Paragraph Positional Distance* distributions: while the median tends to lie in earlier regions, the whiskers extend across multiple sections/paragraphs even after IQR filtering, showing that a non-trivial portion of instances involve snippets from later parts of the document. Consistently, *Fan-hop (Ratio)* shows widely spread *Normalized Section/Paragraph Distance Ratio*, indicating that snippets can appear across a wide range of relative positions within a Super-Section.

Chain-hop Results In contrast, chain-hop QA exhibits a heavy-tailed distribution of snippet-to-snippet *distance*, where most instances involve short-range dependencies but a non-negligible fraction requires long-range, dependency-sensitive reasoning. As shown in Fig. 11, *Chain-hop (Element)* displays a broader distance distribution than *Chain-hop (Average)*, implying that many hops are local while some hops constitute substantially larger jumps within an instance.
Overall, chain-hop mainly consists of local transitions while still including longer-range connections, and this mixed pattern is most clearly reflected in the element-wise statistics.

Chain-Hop Structure Analysis To analyze the structural diversity of chain-hop reasoning in DocHop-QA, we visualize ground-truth hop transitions across documents and document sections using a Sankey-style representation shown in Figure 12.
This analysis reveals that chain-hop questions exhibit diverse hop patterns, spanning document sections such as abstracts, background, methods, results, and conclusions, often across different documents. Notably, table entities appear not only as terminal answer evidence but also as intermediate nodes within reasoning chains, indicating that tables frequently serve as both sources and bridges of information.

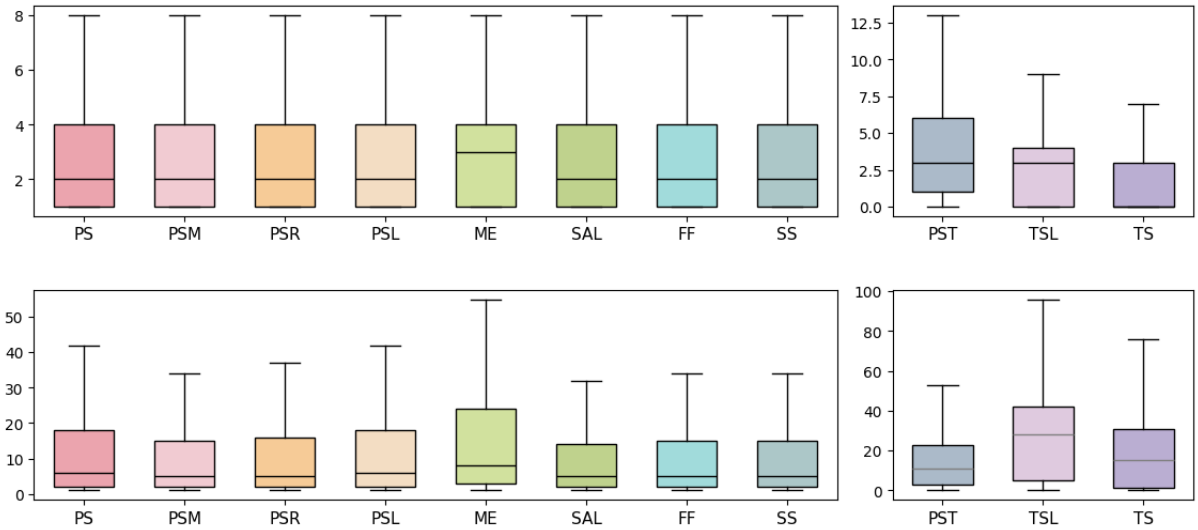
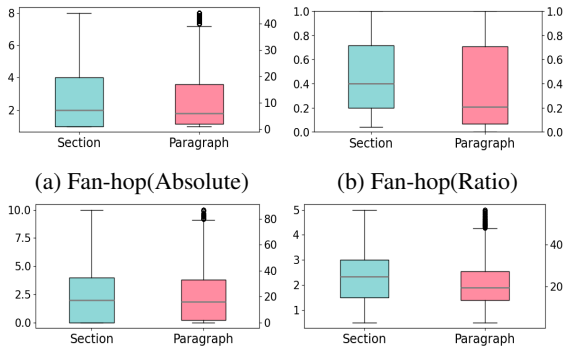


Figure 10: **Structural evidence distribution across question concept types.** We show section-level (top) and paragraph-level (bottom) positions of answer snippets for each concept. Fan-hop questions exhibit broadly distributed positions within documents, indicating global aggregation, whereas chain-hop questions show progressive structural transitions between snippets, reflecting ordered multi-step reason-chaining processes. Notably, PSM, ME tend to retrieve evidence from relatively localized yet deeper structural regions, while PST, TSL, TS span wider structural ranges, often bridging descriptive and analytical sections. Overall, the diverse structural ranges covered across question concept types highlight the heterogeneity of reasoning difficulty and retrieval behaviors encoded in our taxonomy.



(a) Fan-hop(Absolute) (b) Fan-hop(Ratio)

(c) Chain-hop(Element) (d) Chain-hop(Average)

Figure 11: **Structural distance analysis of fan-hop and chain-hop questions at the section and paragraph levels.** For **fan-hop** questions, we visualize the distribution of answer-relevant snippet distances using both absolute positional distance and normalized distance ratios within documents, capturing how broadly evidence is distributed across the document. For **chain-hop** questions, distances are measured as inter-snippet distances between consecutive reasoning steps, reported both element-wise and as their average distance. Overall, these distances characterize local and long-range reasoning patterns in DocHop-QA.

These observations highlight that DocHop-QA chain-hop questions require models to perform structured, dependency-sensitive reasoning across heterogeneous document components, rather than relying on linear or section-local inference. The diversity of hop transitions underscores the need for multi-document understanding that integrates both textual and tabular evidence in a coherent reasoning process.

Reasoning Depth Analysis. The dispersion patterns above reflect the *spatial* complexity of our QA instances. To complement this structural view, we additionally compare hop statistics with benchmarks (Table 10) as a proxy for reasoning depth. We conduct a quantitative comparison of hop statistics with multi-hop QA benchmarks (Table 10). While DocHop-QA shows comparable hop statistics to HotpotQA (Yang et al., 2018), HybridQA (Chen et al., 2020), FanOutQA (Zhu et al., 2024), MEQA (Li et al., 2024) and NOVELHOPQA (Gupta et al., 2025), the QA difficulty is higher due to the complexity of evidence and answers.

Compared to Loong (Wang et al., 2024), which primarily emphasizes large scale document traversal via citation links, DocHop-QA incorporates semantic similarity when aligning answer snippet across hops. Unlike many Wikipedia-based datasets where answers are often short spans or lexical units, DocHop-QA frequently requires synthesizing information across paragraphs and tables,

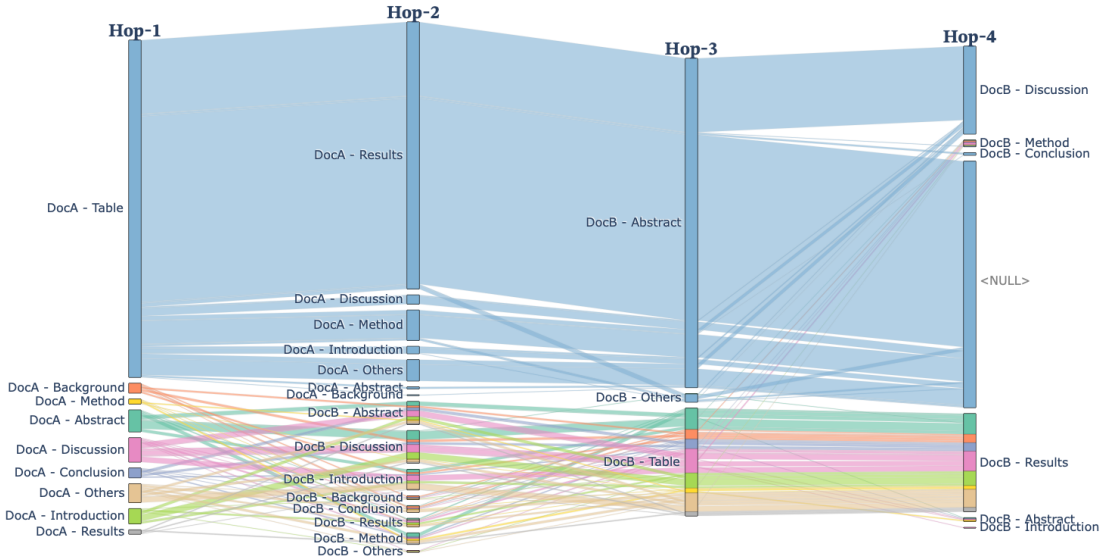


Figure 12: **Ground-truth chain-hop transitions across documents and sections (Sankey Diagram).** Each node denotes the document-section location of an answer snippet at each hop, and links represent hop-to-hop transitions in the ground-truth chain. Tables appear not only as endpoints but also as intermediate bridges. This diversity highlights the structural complexity of multi-document reasoning required by DocHop-QA

yielding richer and challenging answer content.

Dataset	Context	Answer	Num Hops	Reasoning Depth
HotpotQA	Wiki	short span and supporting facts	2.0	1.42
HybridQA	Wiki	short span	3.03	2.05
Loong	Industry	short spans or NL sentence	10.18	3.25
FanOutQA	Wiki	collections of short span	6.40	2.96
MEQA	Wiki	short span	2.53	2.53
NovelHopQA	Gutenberg	1 -2 sentences	2.50	2.50
Ours	Pubmed	collection of paragraphs	4.63	2.46

Table 10: **Comparison of QA benchmark dataset** described in Table 1 in terms of context granularity, answer form, and reasoning complexity

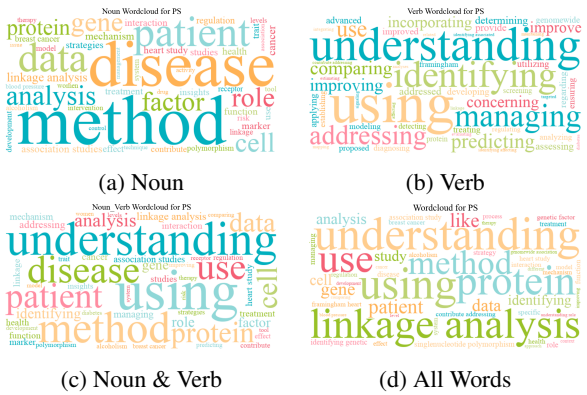
E Additional Dataset Analysis

This section presents additional analyses to support the robustness of our dataset, including keyword frequency patterns, answer snippet distributions, document, table similarity and dataset distribution. These analyses collectively examine linguistic diversity, structural evidence placemet, providing complementary validation of the dataset construction and its intended reasoning characteristics.

E.1 Common Keywords Analysis

We conducted word cloud analysis to examine the patterns of the question texts. After removing stopwords and question concept-specific terms, we extracted nouns, verbs, noun-verb combinations, and all remaining words from each question. Word clouds were generated for each question concept (See Figure 13 to 23). As expected from PubMed-based data, medical terms such as *patient*, *gene* and *cell* appeared, confirming biomedical relevance. Each question type exhibited distinct keyword distributions, reflecting diverse reasoning styles.

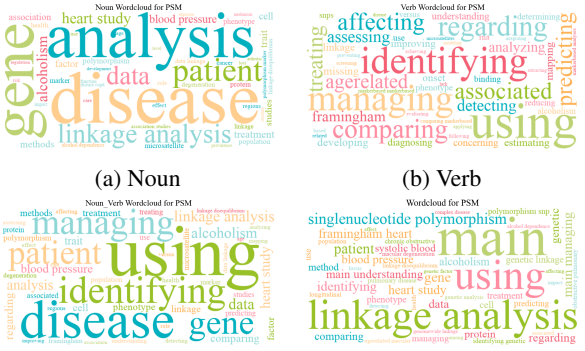
For instance, non-comparison questions focused on specific mechanisms, while comparison questions highlighted contrasting entities or outcomes. Frequent terms like *compare* and *associated* further illustrates the comparative emphasis. These findings suggest that our dataset captures both conceptual diversity and domain specificity, providing a foundation for training QA models capable of handling a wide range of biomedical reasoning tasks.



(a) Noun (b) Verb

(c) Noun & Verb (d) All Words

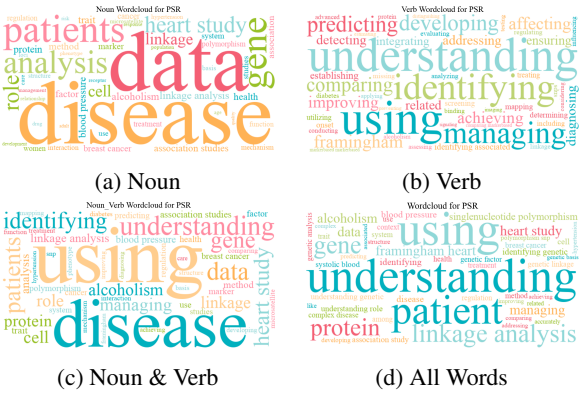
Figure 13: **Frequent words in PS-type questions:** Showing terms like *"method"* and *"disease,"* indicating conceptual understanding in biomedical topics.



(a) Noun (b) Verb

(c) Noun & Verb (d) All Words

Figure 14: **Frequent words in PSM-type questions:** Showing terms like *analysis*, *affecting*, and *identifying*, highlighting analytical reasoning about genetic mechanisms and disease traits.



(a) Noun (b) Verb

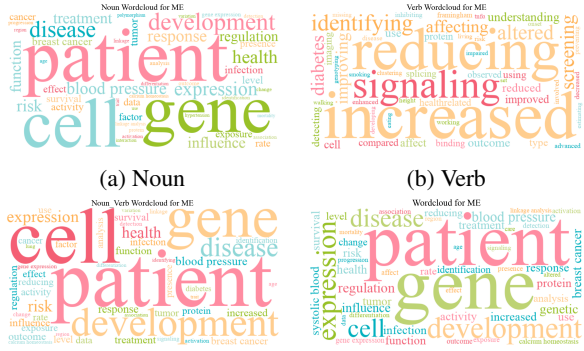
(c) Noun & Verb (d) All Words

Figure 15: **Frequent words in PSR-type questions:** Suggesting a focus on result-oriented reasoning about biomedical solutions or interventions.

(a) Noun (b) Verb

(c) Noun & Verb (d) All Words

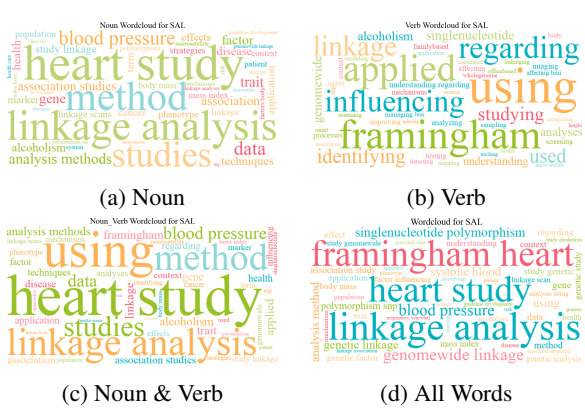
Figure 16: **Frequent words in PSL-type questions:** Showing terms like *conducting*, *addressing*, and *diagnosing*, highlighting evaluation limitations and follow up aspects of solutions.



(a) Noun (b) Verb

(c) Noun & Verb (d) All Words

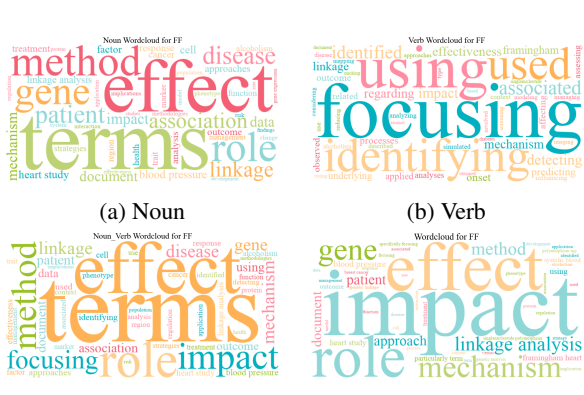
Figure 17: **Frequent words in ME-type questions:** Showing terms like *increased*, *reducing*, and *signaling*, indicating a focus on causal mechanisms and biological effects.



(a) Noun (b) Verb

(c) Noun & Verb (d) All Words

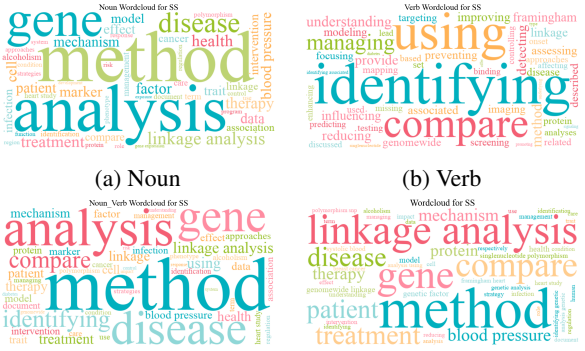
Figure 18: **Frequent words in SAL-type questions:** Showing terms like *heart* and *blood*, indicating strengths and limitations of biological studies.



(a) Noun (b) Verb

(c) Noun & Verb (d) All Words

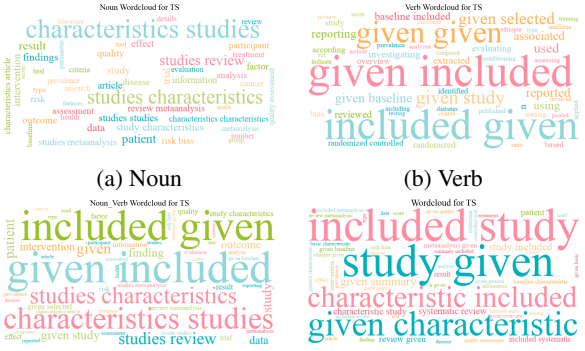
Figure 19: **Frequent words in FF-type questions:** Showing terms like *associated*, *effect*, and *mechanism*, indicating functional comparisons between two entities.



(a) Noun (b) Verb

(c) Noun & Verb (d) All Words

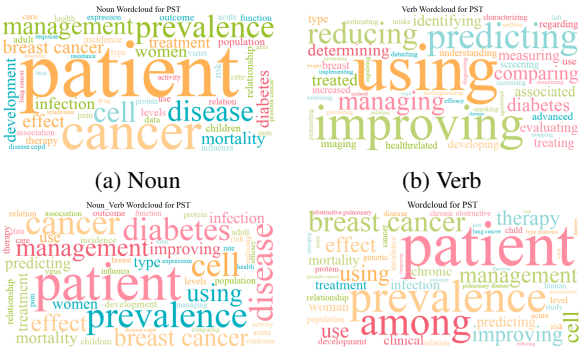
Figure 20: **Frequent words in SS-type questions:** Showing terms like *compare*, *method*, and *analysis*, reflecting contrast of scientific approaches to biomedical problems.



(a) Noun (b) Verb

(c) Noun & Verb (d) All Words

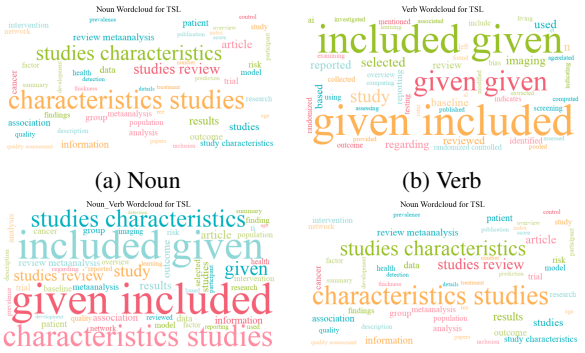
Figure 23: **Frequent words in TS-type questions:** Showing terms like *given*, *included*, and *study*, reflecting their deterministic, table-driven design with limited lexical diversity.



(a) Noun (b) Verb

(c) Noun & Verb (d) All Words

Figure 21: **Frequent words in PST-type questions:** Showing terms like *predicting*, *patient*, and *infection*, highlighting challenges and strategies based on tabular data.



(a) Noun (b) Verb

(c) Noun & Verb (d) All Words

Figure 22: **Frequent words in TSL-type questions:** Showing terms like *given*, *included*, and *characteristics*, reflecting their deterministic, caption-based generation with limited lexical variety.

E.2 Detailed Answer Snippet Distribution

Super-Section Mapping Analysis To facilitate section-level analysis, we standardized a diverse set of original section titles into 8 unified Super-Section tags. This mapping aimed to reduce variation caused by inconsistent section titles (e.g., “Materials and Methods", “Subjects and Methods", “Methodology") and to improve interpretability of section-based statistics. The results of standardization are described in Table 24 and Figure 25.

Answer Snippet Distribution Analysis Figure 27 presents the distribution of answer snippet across QA types. Paragraph-Oriented types predominantly retrieved answers from the Abstract, aligning with their fan-hop retrieval strategy. In contrast, Table-Oriented types mostly extracted snippets from the Results and Discussion sections, reflecting the presence of structured, interpretive content found in them. These patterns highlight that each Q type draws from distinct sections of the document, underscoring the need for models to understand section-level semantics when reasoning over scientific papers.

Figure 26 illustrate the distribution of answer snippets for each type(Paragraph-Oriented and Table-Oriented), with Figure 27 presenting the results separated by each Q type. The Paragraph-Oriented type, which adopts fan-hop based answer retrieval strategy, most frequently extracts from the “Abstract" section. In contrast, the Table-Oriented type most frequently extracts from the “Results" section, due to its document filtering strategy.

Table Oriented Section Analysis We analyzed the origin of tables (Table 11) and found that most

Super-Section	Detail Section Title
Abstract	**abstract**, etc.
Introduction	**introduction**, etc.
Background	**background**, **background and recent developments**, background and overview, etc.
Method	**method(s)**, material(s) and method(s), method and implementation, etc.
Results	**result(s)**, **results and discussion**, implementation and results, result analysis, etc.
Discussion	**discussion**(s), **discussion and conclusions**, limitations, summary and discussion, etc
Conclusion	**conclusion(s)**, conclusions and future directions, conclusion and perspectives, etc.
Others	case report, **review**, dataset and evaluation strategy, **abbreviations**, etc.

Figure 24: **Detailed results of Super-Section Mapping**, with frequent section titles bolded.

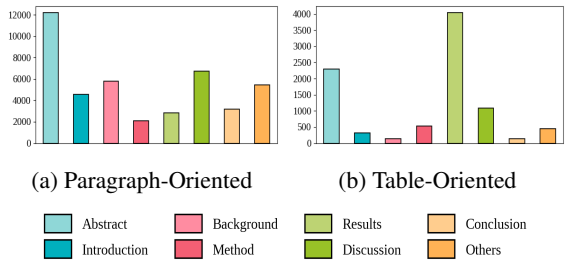


Figure 26: **Barplot of answer snippet distribution for each type of question**; **Abstract** the most for Paragraph-Oriented & **Result** the most for Table-Oriented).

Section	PST	TS	TSL
Abstract	5	3	3
Introduction	1	16	5
Background	1	2	0
Method	33	132	47
Results	**830**	**1057**	**478**
Discussion	1	41	4
Conclusion	0	2	1
Others	3	112	35

Table 11: **Distribution of Table Origins Across Table-Oriented types** (PST, TS, TSL) with Section Mapping

were extracted from the *Results* section. This supports our decision to include only documents with at least one table in result section for PST. TSL, TS tables also mainly came from Results, reflecting their common role in summarizing outcomes across studies. These patterns show that the dataset aligns with typical scientific reporting, where key findings are often presented in results tables.

Figure 25: **Section Mapping Results in Illustrative Chart.**

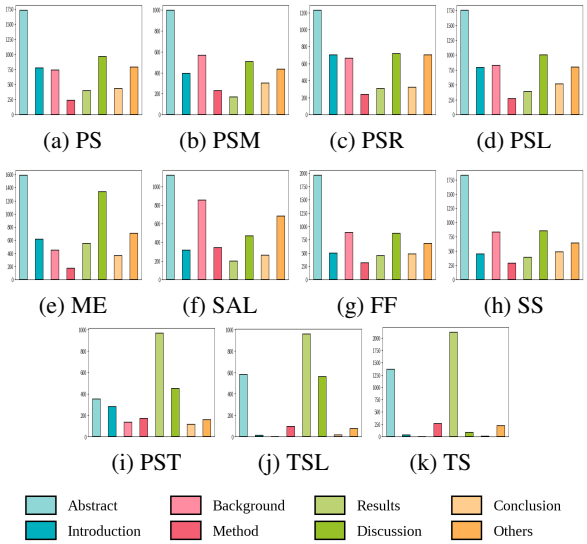
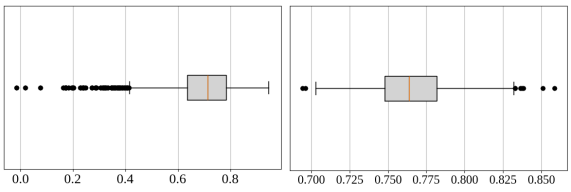


Figure 27: **Section-level distribution of answer spans across the 11 Q types.**

E.3 Detailed Similarity Analysis

TS, TSL: Document Similarity Analysis To assess the semantic similarity between document pairs in the TSL, TS type, we computed the BERT-based similarity shown in Figure 28a we used the title and abstract of both the main and referenced documents. Unlike TF-IDF, we adopted BERT similarity to capture deeper conceptual relationships, since references are typically cited based on semantic rather than lexical overlap. The resulting mean similarity score of 0.69 confirms that the paired documents in the TSL, TS type are highly related,



(a) Document Similarity (b) Solution-Table Similarity

Figure 28: **Similarity analysis**, **(a)** Boxplot of title similarity for TS & TSL type's document pairs **(b)** Boxplot of weighted similarity score (TF-IDF: 0.2, BERTScore : 0.8) between snippet 2 and snippet 3's table caption.

consistent with the pairing strategy used for other QA types. Also, this semantic alignment ensures that multi-hop questions in our dataset reflect realistic cross-document reasoning patterns, enhancing both coherence and training utility.

PST: solution-table Similarity Analysis We analyzed the semantic similarity between the solution snippet and the corresponding table caption in PST type questions shown in Figure 28b. We computed a weighted average of TF-IDF and BERTScore and the results showed a high average similarity of 0.76, indicating strong semantic alignment between the solution content and its associated table. This suggests that the dataset effectively links textual and table snippet, reinforcing its suitability for tasks requiring cross-modal understanding.

E.4 Question Concept Distribution

DocHop-QA contains a total of 11,379 samples, consisting of Paragraph-Oriented types (75.3%) and Table-Oriented types (24.7%). The 11 question concepts are evenly distributed, each representing approximately 10% of the dataset.

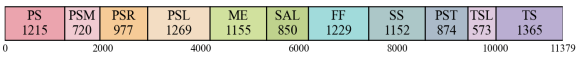


Figure 29: **Q Types distribution (Mean: 1034, Std: 241).**

F Additional Experiments and Results

General Preprocessing All documents are converted from PDF to XML and image formats. OCR and table detection are applied to extract layout-aware content. We use a 70/10/20 train/dev/test split, ensuring that each document appears in only one split to prevent information leakage.

At the document level, we first extracted structured entities, including titles, abstracts, paragraphs, figures, and tables, from the XML version. For visual grounding, we align these XML entities with their corresponding positions in the PDF using the PaddleX OCR toolkit, and save each page of the PDF as an image. This enables us to generate both text-based and image-based inputs across tasks.

At the sample level, we combined all entities from the involved context documents into a unified list, assigning global indices. For instance, if Document 1 has 10 entities and Document 2 has 32, entities are indexed sequentially from [Entity 0] (e.g., <doc1_title>) to [Entity 41] (e.g., <doc2_paragraph20>). Answer labels are represented as index lists pointing to this combined sequence. In tasks involving image-based models, such as LayoutLMv3, all pages from the context documents are merged into a single image and resized to comply with the model's input constraints.

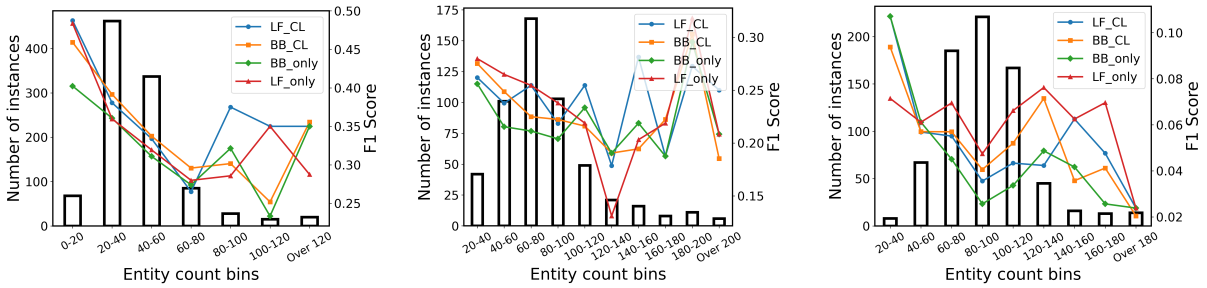
For fine-tuning, all classification-based models are trained with binary cross-entropy loss with the AdamW optimizer for up to 50 epochs. Early stopping is applied with a patience of either 5 or 10 epochs, based on validation performance measured by sample-level recall. Training is conducted locally using a single NVIDIA A100 GPU, while Gemini-based inference tasks are executed via API.

F.1 Additional Task Description

Task 1: BBox Entity Index Extraction We map each XML entity to one or more OCR-detected bounding boxes. These boxes are compiled into a single 1000×1000 image per document set. LayoutLMv3 and LayoutXLM receive bounding box coordinates, associated OCR text, and document images to predict relevant indices. Entities that cannot be matched are marked with index 0. Evaluation includes multi-label accuracy and sample-level F1 based on spatial containment.

Task 2: XML Entity Index Extraction We evaluate two setups: 1) Concatenated input: the question and all XML entities are fed as one sequence into BigBird or Longformer. 2) Per-entity input: Separately encoded each entity with pooled text (BigBird/Longformer) and optional CLIP image embeddings. Predictions are made via a classifier over these embeddings. Metrics are identical to Task 1.

Task 3: Structured Generative Answering This task uses generative models (Qwen2.5-VL-7B-Instruct and InternVL2-4B) to produce lists of entity indices. Inputs include labeled XML entities and document images. Prompts follow zero-shot or



(a) Paragraph & Single (b) Paragraph & Multi (c) Table & Multi

Figure 30: **Task 2: Entity Count vs. F1.** Mean F1 per entity-count bin (lines) with bin sizes (bars). The top row reports results over all instances, while the bottom row excludes zero-F1 cases. Performance generally degrades as entity counts increase, with a more pronounced effect for paragraph-oriented settings.

one-shot formats. To avoid memory issues, we cap the number of entities and pages. Evaluation uses the same index-based metrics as in classification tasks.

Task 4: Generative Text Extraction We test generative LLMs under three prompting strategies (question-only, zero-shot, one-shot) and three input types (text-only, image-only, text+image). Separate runs using Gemini-2.5-Flash and GPT-4o handle XML and PDF ingestion. Generated answers are evaluated with BLEU and ROUGE (1, 2, L, Lsum) against reference answers extracted during dataset construction.

F.2 Additional Result Analysis

Results of Task1: BBox Entity Index Extraction We treat this task as multi-label classification over OCR-extracted bounding boxes. Input includes the resized page image (1000×1000), bounding box coordinates, and OCR text. LayoutLMv3 and LayoutXLM are evaluated using spatial metrics(Belong, Contain, and Overlap) and sample-level F1, precision, and recall. LayoutLMv3 significantly outperforms LayoutXLM in overlap-based accuracy (95.52%) and F1 (12.35), showing its robustness in spatial grounding. However, full containment accuracy is near zero, suggesting difficulties in aligning complex layouts.

Results of Task2: XML Entity Index Extraction This task also uses multi-label classification but predicts indices directly from XML-defined entities. We examine: 1) Concatenated input setup (question + all XML entities) and 2) Per-entity setup, where text and CLIP-based image features are pooled for each entity. Models include BigBird and Longformer. Best performance (F1: 23.24) is from BigBird (per-entity) with CLIP embeddings, indicating that visual context improves model understanding. Model architecture choice matters: BigBird shows advantages in long-range context, while Longformer performs better on per-entity coherence. The task reveals dataset bias towards frequently occurring entity positions.

Impact of Context Entity Cardinality on Model Performance To investigate how contextual complexity affects model performance, we analyze Task 2 performance as a function of the number of context entities involved in each question instance described in Figure 30. Across both text-only and text-plus-image settings, we observe a general degradation in F1 score as the number of entities increases, particularly for paragraph-oriented instances. For single-document questions, performance declines noticeably as entity counts approach 80, while for multi-document questions, a similar trend appears around 100 entities. This degradation likely reflects increased noise from repetitive or overly detailed contextual information, such as experimental settings or hyperparameters that are irrelevant to the question. Interestingly, a modest performance increase is observed beyond these thresholds, which we attribute to the small number of instances in higher entity-count bins. For table-oriented, multi-document instances, overall scores remain lower than paragraph-oriented counterparts. This gap underscores the difficulty of contextualizing tabular information within large entity sets and suggests that improved mechanisms for entity selection and table-aware reasoning are crucial for addressing complex DocHop-QA instances.

Results of Task 3: Structured Generative Answering We prompt generative vision-language models (InternVL2, Qwen2.5-VL, GPT-4o) to out-

put structured entity indices. Inputs include labeled entity texts and optionally, document page images. Prompts follow zero-shot or one-shot formats. GPT-4o in the zero-shot, text-only configuration yields the highest F1 (16.21). Visual input degrades performance due to layout misalignment. One-shot prompts reduce input space, harming performance on long documents. This task highlights the potential of LLMs for structured reasoning, but also the limits of current multimodal alignment.

Results of Taks 4: Generative Text Extraction As shown in Table 12, the best performance is from GPT-4o in a one-shot setting using whole PDF file inputs, which achieves a BLEU score of 28.92, and strong performance across ROUGE 1, 2, L, and LSum (45.79, 23.81, 31.26, 34.12). This suggests that modern vision-language models can effectively extract answer passages grounded in complex document images, even without textual context.

Comparing input-processed architectures, Qwen2.5 consistently outperforms InternVL2 in setups that utilize image input. This advantage likely stems from Qwen's greater input capacity, which allows it to process up to 100 document pages, while InternVL2 is limited to 10 with the simplest configuration. As a result, Qwen2.5 is better suited for large-scale document understanding, particularly when full visual context is required. However, InternVL2 demonstrates better performance when using text-only inputs, achieving a BLEU of 12.65 and ROUGE-2 of 11.11, which is slightly surpassing Qwen's text-only setup (BLEU: 9.66, ROUGE-2: 10.93). This implies that InternVL2's language modeling capabilities are competitive when visual grounding is not required, and reaffirms the importance of architecture-modality alignment.

Interestingly, Qwen2.5 performs best when using image input alone, and worst when using only text. This indicates that Qwen is likely optimized for visual reasoning and may struggle to integrate text-only signals in the absence of layout structure. However, adding both image and text did not significantly boost performance, suggesting that naive concatenation of modalities may introduce redundancy or noise, and that more sophisticated cross-modal fusion strategies are needed for optimal performance on DocHop-QA. Prompt design plays a critical role in generative QA performance. For Qwen2.5, zero-shot prompting consistently outperforms one-shot setups, likely due to the substantial length of context documents. Including a one-shot example in the prompt consumes valuable input space, which forces truncation of relevant information, especially under token limits. This finding highlights an important trade-off in prompting long-context models: example-rich prompts can backfire in high-volume settings like full-document QA.

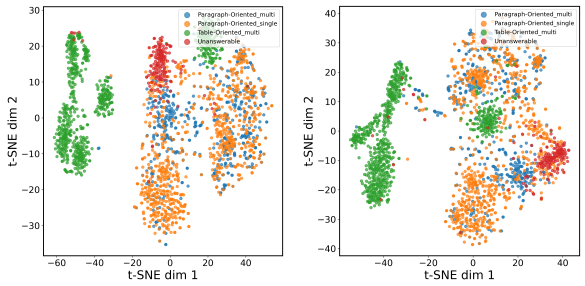
We also evaluated Gemini 2.5 across XML and PDF inputs and GPT-4o with PDF inputs. Unlike Qwen and InternVL2, these models were accessed via API and thus were not bound by local memory constraints, allowing full documents without severe truncation. As a result, one-shot prompting performed best across most configurations, with top BLEU score from GPT-4o of 28.92 using PDF input. GPT-4o outperforms Gemini 2.5 significantly, demonstrating it's superior ability in document understanding and multimodal QA-based text extraction. These results illustrate the potential of cloud-based LLMs for processing full-document QA tasks without aggressive input filtering, further expanding the applicability of DocHop-QA in real-world settings.

In summary, this task underscores how DocHop-QA enables robust benchmarking of free-form QA grounded in multimodal and multi-page documents. Users can evaluate generative models under various prompting and input modality combinations, uncovering limitations in memory, alignment, and format-following behavior. The dataset also reveals that vision-language models optimized for document images (e.g., Qwen2.5) are currently better suited for zero-shot generative answering than generic LLMs trained on short passages or unimodal data.

Embedding Analysis for Question Types We analyze the last-layer hidden states of the Qwen model using t-SNE in Figure 31 to understand how LLMs and LVLMs internally differentiate question types and context complexities. Clear separation emerges between table-oriented and paragraph-oriented questions for both text-only and image-only inputs, suggesting that the model can distinguish question types at a coarse level, likely driven by surface-level cues in question formulation rather than explicit tabular reasoning. This indicates an awareness of question format and required evidence type, but does not imply effective reasoning over tabular content, which is consistent with the substantially lower performance observed on table-oriented tasks. Additionally, we identify a distinct cluster corresponding to unanswerable questions,

Model	Setup	Text	Image	Document	BLEU	Rouge1	Rouge2	RougeL	RougeLSum
InternVL	Question	-	✓	-	6.22	30.02	7.24	15.95	19.04
InternVL	Zero-shot	-	✓	-	5.54	28.67	7.03	14.62	19.46
InternVL	One-shot	-	✓	-	6.39	28.74	6.99	16.12	18.50
InternVL	Question	✓	-	-	11.20	32.69	10.02	18.49	20.32
InternVL	Zero-shot	✓	-	-	12.65	33.36	11.11	18.93	21.40
InternVL	One-shot	✓	-	-	14.53	32.11	11.99	19.56	20.91
InternVL	Question	✓	✓	-	6.91	30.09	8.05	15.85	19.79
InternVL	Zero-shot	✓	✓	-	6.91	30.06	8.48	15.55	20.60
InternVL	One-shot	✓	✓	-	7.48	31.23	8.55	16.46	20.26
QWEN	Question	-	✓	-	15.06	36.64	13.43	20.79	23.89
QWEN	Zero-shot	-	✓	-	21.56	38.45	16.92	23.90	25.84
QWEN	One-shot	-	✓	-	12.40	34.48	11.49	19.50	22.53
QWEN	Question	✓	-	-	7.77	28.62	9.33	15.71	20.44
QWEN	Zero-shot	✓	-	-	9.66	30.81	10.93	17.28	21.96
QWEN	One-shot	✓	-	-	7.32	27.66	8.30	15.45	19.44
QWEN	Question	✓	✓	-	9.80	31.89	10.70	17.82	22.12
QWEN	Zero-shot	✓	✓	-	12.78	33.69	12.86	19.80	23.66
QWEN	One-shot	✓	✓	-	9.45	31.35	9.96	17.94	21.40
Gemini	Question	-	-	PDF	7.57	28.76	10.71	15.41	20.92
Gemini	Zero-shot	-	-	PDF	9.03	29.62	12.08	16.20	21.88
Gemini	One-shot	-	-	PDF	11.96	25.47	10.96	15.33	18.68
Gemini	Question	-	-	XML+PDF	7.38	28.85	10.55	15.15	20.46
Gemini	Zero-shot	-	-	XML+PDF	8.89	30.70	12.24	17.07	22.63
Gemini	One-shot	-	-	XML+PDF	11.71	24.57	12.61	16.30	18.95
GPT4o	Question	-	-	PDF	24.80	41.25	19.46	26.85	29.92
GPT4o	Zero-shot	-	-	PDF	26.45	43.16	21.05	28.63	31.55
GPT4o	One-shot	-	-	PDF	**28.92**	**45.79**	**23.81**	**31.26**	**34.12**

Table 12: **Comprehensive results for generative text extraction task** using InternVL2, QWEN 2.5, Gemini 2.5, and GPT-4o across different prompt setups and input configuration. We measure performance through BLEU and ROUGE metrics.



(a) Image-only input (b) Text-only (XML) input

Figure 31: **Task 4: t-SNE of Qwen last layer hidden representations** Visualization under image-only and text-only (XML) inputs, analyzing how LLMs and VLMs differentiate question types and contextual complexity. Table and paragraph-oriented instances form separable clusters in both settings, and unanswerable cases show a distinct grouping, suggesting modality-consistent internal structuring of question types.

demonstrating that recent LLMs can partially detect answerability without fine-tuning. However, this separation is less pronounced for paragraph-oriented and fan-hop questions, where partial answers may be available but full reasoning chains cannot be completed. This observation reflects the inherent difficulty of DocHop-QA unanswerable cases and highlights the tendency of dataset-generation LLMs to introduce hallucinated intermediate hops. Overall, the similarity of clustering patterns across text-only and image-only inputs suggests consistent internal representations across modalities, while reinforcing the need for more robust reasoning and answerability-aware modeling.

F.3 Hyperparameters

We provide the best hyperparameter setups for each model in Tables 13 to 16. For fine-tuned models in Task 1 and Task 2, we explore learning rate values of 2e-01, 2e-02, 2e-03, 2e-04, and 2e-05, early stop patience of 10, 5, or 0 (no early stop), and warm up of 10 or 0 (no warm up). For learning rates, 2e-02 generally produces a better F1 score, however, predicted indices tend to skew to the more frequent entity indices, raising recall but sacrificing precision. Using 2e-05 tend to produce more varied predictions with better precision but recall suffers greatly. For the generative models, we limited the number of input entities and page images. Being a smaller model, InternVL2 only ran suc-

cessfully with a maximum of 25 entities and 10 pages. QWEN2.5 could handle up to 200 entities and 100 page images but, interestingly, performed best mostly with only 10 pages.

Model	Learning Rate	Batch Size	Max Len	Early Stop	Warm Up
LayoutLMv3	2e-02	32	512	10	10
LayoutXLM	2e-05	64	512	10	10

Table 13: **BBox Entity Index Extraction best model hyperparameters.**

Setup	Text Emb	Img Emb	Max Len	Learning Rate	Early Stop	Batch Size	Warm Up
Concat	BB	–	2048	2e-02	5	16	10
Concat	LF	–	2048	2e-02	5	8	10
PerEnt	BB	–	4096	2e-02	0	64	0
PerEnt	LF	–	4096	2e-02	0	64	0
PerEnt	BB	CLIP	4096	2e-02	5	64	10
PerEnt	LF	CLIP	4096	2e-02	5	64	10

Table 14: **XML Entity Index Extraction best model hyperparameters.** Concat: Concatenated, PerEnt: Per Entity, BB: BigBird, LF: Longformer

Model	Setup	Text	Img	Max Entities	Max Pages	Max Tokens
InternVL	Zero-shot	✓	–	25	–	1024
InternVL	Zero-shot	✓	✓	25	5	1024
InternVL	One-shot	✓	–	25	–	1024
InternVL	One-shot	✓	✓	25	5	1024
Qwen	Zero-shot	✓	–	100	–	1024
Qwen	Zero-shot	✓	✓	50	10	1024
Qwen	One-shot	✓	–	50	–	1024
Qwen	One-shot	✓	✓	50	10	1024

Table 15: **Structured generative answering best model hyperparameters.**

F.4 Experiment Prompts

Table 17 summarizes the prompt structures used for generative models. The question-only prompt includes the context and question, the zero-shot prompt inserts an instruction between them. The one-shot prompt adds an example to the zero-shot structure. For structured generative answering, the instructions are changed to *"Extract the entity IDs of the paragraphs, figures, and/or tables that answer the question:"*

Model	Setup	Text	Img	Max Entities	Max Pages	Max Tokens
InternVL	Question	–	✓	–	10	1024
InternVL	Zero-shot	–	✓	–	10	1024
InternVL	One-shot	–	✓	–	5	1024
InternVL	Question	✓	–	25	–	1024
InternVL	Zero-shot	✓	–	25	–	1024
InternVL	One-shot	✓	–	25	–	1024
InternVL	Question	✓	✓	25	5	1024
InternVL	Zero-shot	✓	✓	25	5	1024
InternVL	One-shot	✓	✓	25	5	1024
Qwen	Question	–	✓	–	10	1024
Qwen	Zero-shot	–	✓	–	10	1024
Qwen	One-shot	–	✓	–	10	1024
Qwen	Question	✓	–	200	–	1024
Qwen	Zero-shot	✓	–	200	–	1024
Qwen	One-shot	✓	–	50	–	1024
Qwen	Question	✓	✓	100	10	1024
Qwen	Zero-shot	✓	✓	100	10	1024
Qwen	One-shot	✓	✓	50	10	1024

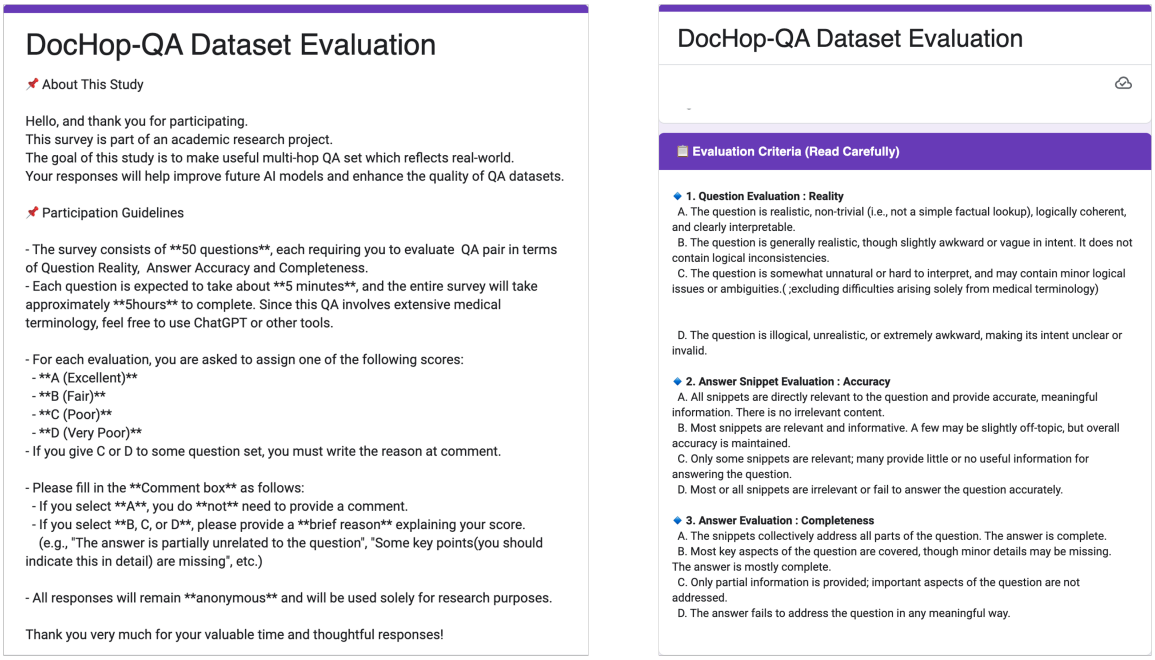
Table 16: **Generative text extraction best model hyperparameters.**

Segment	Prompt

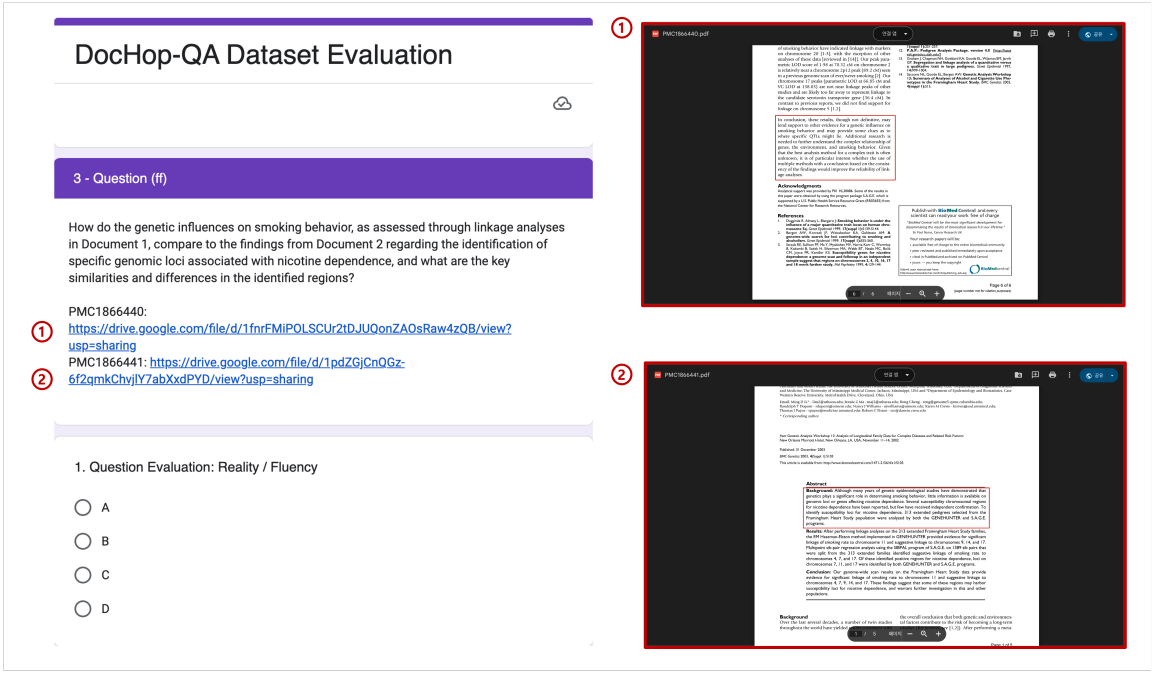
Example	The following `<input_desc>` from one or more documents that may be used to answer a question by extracting paragraphs, figures, and/or tables. For example:
	`[example_context_pages]`
	`[example_context_entities]`
	Question: `<example_question>`
	Answers: `<example_answer>`
	Consider the following to answer the question at the end. Extract the paragraphs, figures, and/or tables that answer the question.

Context Input	`[instance_context_pages]` `[instance_context_entities]`
Instruction	The following `<input_desc>` from one or more documents. Extract the paragraphs, figures, and/or tables that answer the question:
Question	`<instance_question>`
`<input_desc>`	{"images are pages", "texts are xml", "xml files", "pdf files"}

Table 17: **General prompt structure for generative tasks.**



(a) Overview page providing the study purpose and outlining the evaluation criteria.



(b) Example QA instance with hyperlinks to PDF documents displaying the answer snippet with a bounding box.

Figure 32: **Screenshots of the structured Google Form interface used for human evaluation.**