GraphRAG-Bench: Challenging Domain-Specific Reasoning for Evaluating Graph Retrieval-Augmented Generation

Anonymous Author(s)
Affiliation
Address
email

Abstract

Graph Retrieval Augmented Generation (GraphRAG) has garnered increasing recognition for its potential to enhance large language models (LLMs) by structurally organizing domain-specific corpora and facilitating complex reasoning. However, current evaluations of GraphRAG models predominantly rely on traditional question-answering datasets. Their limited scope in questions and evaluation metrics fails to comprehensively assess the reasoning capacity improvements enabled by GraphRAG models. To address this gap, we introduce GraphRAG-Bench, a large-scale, domain-specific benchmark designed to rigorously evaluate GraphRAG models. Our benchmark offers three key contributions: (i) Challenging question design. Featuring college-level, domain-specific questions that demand multi-hop reasoning, the benchmark ensures that simple content retrieval is insufficient for problem-solving. For example, some questions require mathematical reasoning or programming. (ii) Diverse task coverage. The dataset includes a broad spectrum of reasoning tasks, multiple-choice, true/false, multi-select, open-ended, and fill-in-the-blank. It spans 16 disciplines in twenty core textbooks. (iii) Holistic evaluation framework. GraphRAG-Bench provides comprehensive assessment across the entire GraphRAG pipeline, including graph construction, knowledge retrieval, and answer generation. Beyond final-answer correctness, it evaluates the logical coherence of the reasoning process. By applying nine contemporary GraphRAG methods to GraphRAG-Bench, we demonstrate its utility in quantifying how graph-based structuring improves model reasoning capabilities. Our analysis reveals critical insights about graph architectures, retrieval efficacy, and reasoning capabilities, offering actionable guidance for the research community.

1 Introduction

Retrieval-Augmented Generation (RAG) [1; 2] has emerged as a key solution to ground large language models (LLMs) in external knowledge to mitigate both the hallucination problem and the lack of domain knowledge. By retrieving relevant text passages from corpora, RAG injects factual knowledge for a more reliable generation from LLMs. However, conventional RAG systems remain unsatisfactory when dealing with complex reasoning scenarios. The flat retrieval in RAG directly returns fragmentized chunks based on similarity matching, which limits their ability to model complex relationships between concepts to answer the questions requiring multi-hop reasoning [3; 4], i.e., 'What was the impact of [event] the 2008 Lehman Brothers bankruptcy on [person] Elon Musk's Tesla?' or global comprehension, i.e., 'What is the main idea of the [event] Trade Policy Change?'.

To address these limitations, Graph Retrieval-Augmented Generation (GraphRAG) has been extensively studied to capture the structured knowledge among concepts in the form of graphs [5; 6; 7],

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

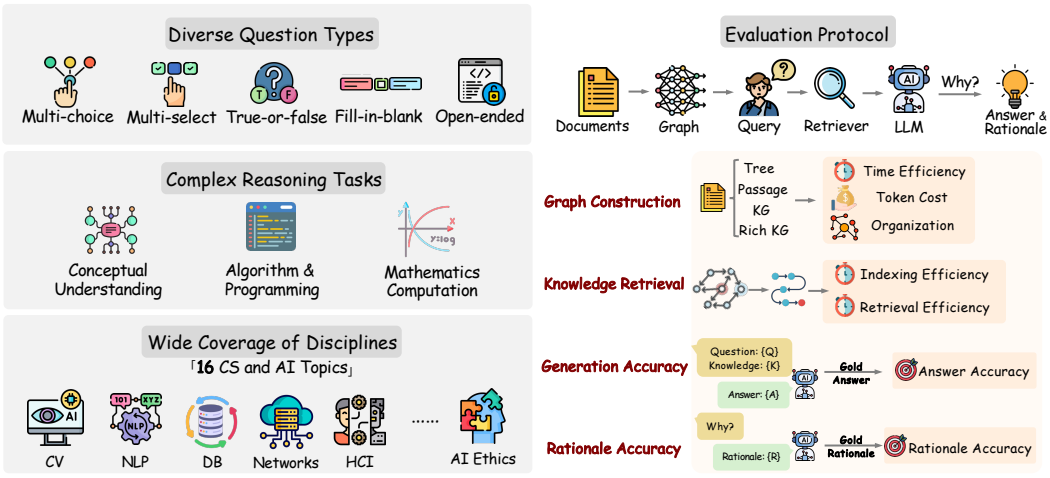

Figure 1: A sketched overview of our benchmark `GraphRAG-Bench`, illustrating the contributions.

where nodes represent concepts and edges are for the relations among them. Recent advances in GraphRAG can be categorized into three main directions. First, hierarchical graph construction methods like RAPTOR [8] and Microsoft's GraphRAG [5] organize knowledge through tree structures and community detection. Second, neural graph retrieval approaches, including GFM-RAG [9] and G-Retriever [10] employ graph neural encoders with specialized objectives for multi-hop reasoning. Third, dynamic knowledge integration systems such as DALK [11] and ToG [12] develop adaptive graph construction and traversal mechanisms that are tightly coupled with LLMs. By structuring knowledge as graphs, GraphRAG enables LLMs to both traverse and reason over explicit relational paths, but also supports deeper reasoning by inferring implicit relations based on the graph structure.

However, despite the promise, existing benchmarks for GraphRAG methods fail to reflect the performance of reasoning on graphs. They predominantly leverage the traditional QA dataset, e.g., HotpotQA [13], 2WikiMultiHopQA [14] and MuSiQue [15], which only feature explicit factoid questions with limited complexity and short answers, e.g., 'Who is the grandchild of Dambar Shah?'. These datasets suffer from three critical limitations: (i) There are only commonsense questions that could be probably covered in the training corpus of LLMs. (ii) They typically require only single-hop or shallow multi-hop reasoning based on explicit connections, which inadequately probes the unique advantages of graph-structured knowledge. (iii) Narrow Answer Formats. Most answers are short (names, dates) or multiple-choice, which could hardly reflect the reasoning ability over graphs. To this end, we would like to ask a research question:

"Does graph augmentation truly enhance reasoning capabilities beyond simple retrieval?"

In this paper, we propose GraphRAG-Bench, the first challenging domain-specific benchmark particularly designed for GraphRAG. (i) Our dataset contains 1,018 college-level question spans 16 disciplines, e.g., computer vision, computer networks, human-computer interaction, AI ethics, etc, featuring the ability of conceptual understanding, e.g., "Given [theorem] A and B, prove [conclusion] C", complex algorithmic programming, e.g., coding with interlinked function calls) and mathematical computation, e.g., "Given [Input], [Conv1], [MaxPool], [FC], calculate the output volume dimensions." (ii) GraphRAG-Bench contains five types of diverse questions to thoroughly evaluate different aspects of reasoning, including multiple-choice (MC), multi-select (MS), true-or-false (TF), fill-in-blank (FB) and open-ended (OE). (iii) We offer a comprehensive multi-dimensional evaluation on each component of GraphRAG, including graph construction, knowledge retrieval, answer generation and rationale generation. We aim to provide unprecedented insights into how graph-structured knowledge enhances LLMs' reasoning capabilities compared to traditional RAG approaches.

Overall, we propose the first challenging domain-specific benchmark, particularly concentrating on GraphRAG. It contains 1,018 questions in 5 question types spanning 16 topics and a corpus of 7 million words from 20 computer science textbooks. A comprehensive evaluation protocol is designed to stress-test GraphRAG methods on graph construction, retrieval, and multi-hop answer generation and rationale generation. Extensive experiments have been conducted with nine state-of-the-art

GraphRAG models. We make insightful observations and provide the insights that: 1) GraphRAG substantially enhances the reasoning capabilities of LLMs, and - to the best of our knowledge - we are the first to quantify this improvement using concrete evaluation metrics. 2) GraphRAG's impact varies by question types: it yields significant gains on some types but offers limited benefit for others.

2 Related Work

GraphRAG. Recent work in GraphRAG has focused on integrating structured knowledge and advanced retrieval strategies to overcome the limitations of vanilla RAG in handling large, noisy corpora and complex reasoning. For example, RAPTOR [8] and Microsoft's GraphRAG [5] both employ hierarchical clustering, RAPTOR via recursive tree construction with multi-level summarization, and GraphRAG via community detection with LLM-generated synopses, to support coarse-to-fine retrieval and diverse, high-coverage responses. GFM-RAG [9], G-Retriever [10], and LightRAG [16] each combine graph neural encoders with specialized retrieval objectives, respectively a query dependent GNN trained in two stages for multi-hop generalizability, a Prize Collecting Steiner Tree formulation to reduce hallucination and improve scalability, and a dual level graph augmented index for efficient, incrementally updatable lookup, to enable accurate, scalable reasoning over document graphs. Inspired by hippocampal memory processes, HippoRAG [17] leverages Personalized PageRank to achieve single-step multi-hop retrieval, delivering state-of-the-art efficiency and performance on both path following and path finding QA tasks. DALK [11] and KGP [18] introduce dynamic KG construction and traversal agents, using LLMs to build domain specific graphs and self aware retrieval policies, to inject structural context while reducing noise. ToG [12] tightly couples LLMs with KGs via beam search exploration, enabling iterative graph reasoning and on the fly correction without additional training. Collectively, these methods exemplify the GraphRAG paradigm by uniting graph structures, generative language models, and novel retrieval formulations to enhance knowledge integration, scalability, and deep reasoning across diverse domains.

Prior benchmarks for GraphRAG. To date, no dataset has been specifically designed for GraphRAG tasks. Widely used datasets such as Quality [19], PopQA [20], and HotpotQA [13] are tailored for general question answering, where answers can often be directly extracted from corpora, failing to effectively measure the core capabilities of GraphRAG methods. Multi-hop QA datasets like MusiqueQA [15] and 2WikiMultiHopQA [14] contain questions artificially constructed via rules and logic, rather than natural queries from real-world scenarios. Additionally, their corpora are short and often derived from converting entities and descriptions of existing KGs, which deviates from practical application contexts. While DIGIMON [7] benchmarks some methods, it neither introduces new datasets nor evaluates the reasoning capabilities of GraphRAG. Critically, all aforementioned datasets neglect question type distinctions, focusing primarily on simple questions and thus unable to reflect GraphRAG's performance variations across different question categories. In summary, existing datasets lack long contexts and raw documents, mismatching real-world scenarios, and omit gold rationale, making it impossible to systematically evaluate GraphRAG's reasoning abilities.

3 GraphRAG-Bench: Challenging Reasoning Benchmark for GraphRAG

3.1 Question design

To evaluate the GraphRAG framework on college-level reasoning, we first assembled an authoritative textbook corpus. Beginning with over 100 publications spanning 16 distinct subfields in computer science, we systematically identified the most representative 20 textbooks. We defined five types of questions, each targeting a different aspect of GraphRAG's reasoning capabilities, which are detailed in Tab. 1. After rigorous screening and refinement by several domain experts, we selected 1,018 high-quality challenging questions, covering a broad spectrum of topics.

By design, each question type is explicitly mapped to the core competencies of GraphRAG, with individual questions meticulously crafted for application in college-level instructional or assessment contexts. Should GraphRAG demonstrate improved performance on these tasks, it would establish itself as a highly effective tool in education, significantly enhancing teaching and learning efficiency.

Question Type	Description
Fill-in-blank (FB)	Requires completing context-dependent statements with semantically precise terms. These assess the model's ability to generate contextually coherent content by leveraging local semantic dependencies and entity grounding within graph-structured knowledge.
Multi-choice (MC)	Presents a question with 4 options, including linguistically plausible distractors. These assess the model's capacity to discern correct answers through discriminative reasoning, integrating entity information and edge relationships to reject semantically similar but factually incorrect options.
Multi-select (MS)	Demands selecting 2–4 correct answers from 4 options, often requiring reasoning over interconnected concepts. The inclusion of overlapping distractors tests the model's ability to handle complex query semantics, aggregating evidence from multi-hop graph paths and resolving conflicts between related but non-essential attributes.
True-or-false (TF)	Involves verifying the correctness of statements. These measure the model's factual accuracy assessment, requiring logical inference over knowledge.
Open-ended (OE)	OE questions allow for a wide range of responses, requiring methods to formulate detailed and comprehensive answers. These evaluate the model's holistic knowledge synthesis, demanding the integration of multi-subfield knowledge to generate structured, logically coherent long-form responses.

Table 1: The description of different question types.

3.2 Corpus collection and processing

Extracting accurate content from the 20 PDF-format core textbooks presents significant challenges. We implement a multi-stage pipeline comprising preprocessing, content parsing, post-processing, and hierarchy construction.

Textbook Preprocessing. 1) PDF Classification: To distinguish text-based pages from scanned (image-based) pages, we analyze each page's text density and image area proportion. Text-based pages are processed by extracting text directly using PyMuPDF, while scanned pages require optical character recognition (OCR) to extract their textual content. 2) Metadata Extraction: We extract metadata for each textbook, including its outline, total page count, and the page ranges for each chapter or section. This metadata supports the later construction of the document's logical structure.

Content Parsing. After preprocessing, we analyze each page's layout to extract textual and non-textual elements. 1) Layout Analysis: We apply LayoutLMv3 [21] for multimodal document layout analysis. LayoutLMv3 is pre-trained with masked language modeling, masked image modeling, and cross-modal alignment, enabling it to learn rich representations of document pages. The model classifies page regions into semantic categories such as titles, paragraphs, figures, tables, or decorative/irrelevant elements. This segmentation yields coherent content blocks on each page. 2) Formula Recognition: Mathematical formulas embedded in text are often misrecognized by OCR. To prevent this, we first detect inline formulas using a pre-trained YOLO-based model [22] from PDF-Extract-Kit. This model identifies the bounding boxes of formula regions so that formula images can be extracted separately, ensuring that OCR does not garble the formula content. 3) OCR: In scanned PDFs, OCR is applied to recognize text regions. We use PaddleOCR to transcribe text from the regions labeled as titles and body paragraphs via layout analysis. This step produces the page's textual content in the correct reading order, while preserving non-text elements as separate objects.

Post-Processing. After parsing, the extracted elements (text blocks, formula, figures, tables, etc.) may be disordered due to overlapping bounding boxes or fragmented text lines. We resolve these issues by reordering and merging page regions according to human reading order. Concretely, we use MinerU [23] for post-processing, which partitions each page into logical reading regions and sequences them so that the final text flow matches the natural reading sequence.

Hierarchy Construction. Finally, we organize the extracted content into a hierarchical textbook-tree structure. We map the textbook metadata (e.g., chapter titles, section divisions, and page ranges) to a

four-level hierarchy: Book Title → Chapter → Section (Subchapter) → Knowledge Content Unit. Each node in this hierarchy is annotated with its contextual metadata and its structural role. This textbook-tree provides an intuitive, pedagogical navigation framework aligned with the textbook's organization. The resulting corpus – with its accurate content extraction, structural annotation, and hierarchical organization – forms a robust basis for evaluating GraphRAG's ability to leverage organized textbook knowledge for context-rich reasoning and retrieval-augmented generation.

3.3 Expert-crafted rationale

Existing benchmarks typically supply only final answers or explicit graph paths; by contrast, our dataset supplies expert-crafted rationales that articulate the complete logical progression necessary to solve each problem. These rationales go beyond mere corpus aggregation; they are structured narratives that (i) isolate prerequisite concepts, (ii) describe the relationships among these concepts, and (iii) specify the inferential operations applied during problem solving. By tracing each step of logical inference and knowledge interaction, we can assess whether GraphRAG models truly generate contextually grounded explanations or simply exploit surface-level patterns.

To enable fine-grained, topic-specific evaluation, each question in our dataset carries two hierarchical labels: a broad subfield (Level 1, e.g., "Machine Learning") and a more granular concept (Level 2, e.g., "Unsupervised Learning"). These annotations structure our post-hoc analyses. For each topic, we measure not only the accuracy of the model's answer but also the degree to which its generated rationale aligns with the gold one. In this way, we convert evaluation into a multidimensional process, requiring models to produce both correct solutions and faithful reasoning patterns.

4 Experiments

We conduct experiments on each submodule following GraphRAG's pipeline, which includes the **graph construction** (or similar specialized structures), **knowledge retrieval**, and **generation**. Additionally, since our dataset contains a gold rationale for each query, we require the GraphRAG method to generate **rationales** during the generation phase to evaluate its reasoning capabilities.

Metrics. We provide a succinct introduction to the core ideas of each metric; the full evaluation protocol and details can be found in the Appendix.

- **Graph construction.** We evaluate graph construction across three aspects: 1) Efficiency: the time required to build a complete graph. 2) Cost: the number of tokens consumed during graph construction. 3) Organization: the proportion of non-isolated nodes within the constructed graph.
- **Knowledge retrieval.** We evaluate retrieval from two dimensions: 1) indexing time, defined as the duration required to construct the vector database for retrieval; 2) average retrieval time, representing the mean time consumed for knowledge retrieval per query. Additionally, we summarize the retrieval operators employed by each method to assess the complexity of their retrieval mechanisms.
- **Generation.** We argue that the existing exact match metric is inappropriate, as correct answering does not necessitate word-by-word correspondence. Therefore, this paper introduces a new metric, Accuracy, defined as follows: 1) For OE and FB questions, both the generated output and groundtruth are fed into an LLM via our designed prompt, which assigns a score based on semantic alignment and correctness. 2) For MC and TF, 1 point for the correct answer, 0 points for otherwise. 3) For MS, 1 point for a fully correct answer; 0.5 points for a subset; 0 points for incorrect answers.
- **Rationale.** We designed a prompt to feed both the rationale generated by the GraphRAG method and the gold rationale into a LLM, which assigns a reasoning score R to evaluate their semantic correspondence and reasoning consistency. Simultaneously, we developed an additional assessment metric, namely the AR metric, to determine whether the model is able to provide correct reasoning when it answers the question accurately. This metric serves to distinguish whether the model has merely guessed the correct answer or has actually engaged in proper logical reasoning to reach the correct answer, thereby offering a more comprehensive understanding of the model's performance.

Experiment setups. In our experiments, we evaluated the performance of nine state-of-the-art GraphRAG methods, including: 1) RAPTOR [8]; 2) LightRAG [16]; 3) GraphRAG [5]; 4) G-Retriever [10]; 5) HippoRAG [17]; 6) GFM-RAG [9]; 7) DALK [11]; 8) KGP [18]; 9) ToG [12]. To ensure a fair comparison across all methods, we adopted the same GPT-4o-mini as the default large

language model. We imposed no max token length to limit the performance of individual methods. For methods requiring top-k selection, we uniformly set k=5. Regarding text chunking, the chunk size was consistently set to 1200 tokens. Except for the parameters standardized for fair comparison, all other hyperparameters were configured to the optimal values reported in the original papers.

4.1 Evaluation of graph construction

Method	Token cost of graph construction	Time cost of graph construction	Organization
RAPTOR (2024)	10,142,221	20396.49s	-
KGP (2024)	15,271,633	17318.07s	46.03%
LightRAG (2024)	83,909,073	12976.22s	69.71%
GraphRAG (2025)	79,929,698	11181.24s	72.51%
G-Retriever (2024)	32,948,161	5315.27s	89.95%
HippoRAG (2024)	33,006,198	5051.41s	89.58%
DALK (2024)	33,007,324	**4674.30s**	89.49%
ToG (2024)	33,008,230	5235.30s	89.95%
GFM-RAG (2025)	32,766,094	5631.10s	**89.97%**

Table 2: Comparison of graph construction process.

Graph construction aims to transform corpus into structured, storable objects, serving as the foundational step in GraphRAG. Current mainstream graph construction methods can be categorized into four classes: 1) Tree: RAPTOR leverages this structure, where each leaf node represents a chunk. By generating summaries via LLMs and applying clustering methods, parent nodes are iteratively created to form a hierarchical tree structure. 2) Passage Graph: Adopted by KGP, this structure represents each chunk as a node, with edges established through entity linking tools. 3) Knowledge Graph: Used in G-Retriever, HippoRAG, GFM-RAG, and DALK, this structure extracts entities and relationships from chunks using open information extraction (OpenIE) tools to construct knowledge graphs. 4) Rich Knowledge Graph: Employed by GraphRAG and LightRAG, this structure enriches standard knowledge graphs with additional information (e.g., summarizing descriptions for nodes or edges).

Experimental results in Tab. 2 show that the tree structure incurs the lowest token count, as it only invokes LLMs for summary generation, but requires the longest time due to iterative clustering. The passage graph has suboptimal token cost, invoking LLMs only for summarizing entities or relationships, with the second-longest time consumption attributed to the time-intensive entity linking process. The knowledge graph has moderate token usage, requiring LLMs for both entity extraction from corpora and triple generation from entities, yet achieves the shortest time consumption due to rapid knowledge graph construction after triple acquisition. The rich knowledge graph consumes the most tokens, as it generates additional descriptions for entities and relationships via LLMs on top of standard knowledge graphs, leading to increased time costs. For evaluating graph construction quality, we use the non-isolated nodes ratio as the metric. Since the Tree structure contains no isolated nodes, this metric is inapplicable to it. Experimental results show that the Knowledge Graph achieves the best performance, with its non-isolated nodes ratio maintained at approximately 90%. The Rich Knowledge Graph performs suboptimally; while it incorporates additional information, it inevitably introduces more noise. The Passage Graph exhibits the lowest non-isolated nodes ratio, indicating that entity linking tools fail to effectively establish edges between most entity pairs.

4.2 Evaluation of knowledge retrieval

As shown in Tab. 3. GFM-RAG incurs the shortest indexing time; it does not construct a traditional vector database to store entities but instead stores question-corresponding entities exclusively during graph construction. Among methods using vector databases, KGP, RAPTOR, and DALK exhibit lower costs due to minimal stored information; ToG, G-Retriever, and LightRAG have moderate costs, as relationship storage is inherently time-consuming; GraphRAG further increases indexing time by additionally storing community reports. HippoRAG demands the longest indexing time, attributed to its extra construction of entity<->relationship and relationship<->chunk mappings. Regarding average retrieval time, RAPTOR achieves the fastest speed, as its tree structure enables rapid information localization. GFM-RAG and HippoRAG follow, leveraging GNNs and PageRank algorithms for retrieval, respectively. G-retriever employs a prize-collecting Steiner forest algorithm, while LightRAG relies on relationship-based retrieval, both introducing additional latency. GraphRAG

needs to utilize community information for retrieval, which leads to its time-consuming. KGP, ToG, and DALK incur substantial time costs due to their dependence on LLM invocations during retrieval.

Method	Retrieval operators	Indexing time	Average retrieval time
KGP	Node	204.10s	89.38s
ToG	Node+Relationship	1080.43s	70.53s
GraphRAG	Node+Relationship+Chunk+Community	1796.65s	44.87s
DALK	Node+Subgraph	407.10s	26.80s
G-Retriever	Node+Relationship+Subgraph	920.39s	23.77s
LightRAG	Node+Relationship+Chunk	1430.32s	13.95s
HippoRAG	Node+Relationship+Chunk	4695.29s	2.44s
GFM-RAG	Node	**93.55s**	1.96s
RAPTOR	Node	451.03s	**0.02s**

Table 3: Comparison of knowledge retrieval process.

4.3 Evaluation of generation accuray

Method	Accuracy					
	Fill-in-blank	Multi-choice	Multi-select	True-or-false	Open-ended	Average
GPT-4o-mini	74.29	81.11	76.68	75.95	52.23	70.68
TF-IDF	75.71	77.88	72.52	84.17	50.18	71.71↑
BM-25	74.28	78.80	71.17	**84.49**	50.00	71.66↑
DALK	70.00	78.34	71.62	77.22	51.49	69.30↓
G-Retriever	70.95	77.42	71.62	78.80	52.04	69.84↓
LightRAG	65.24	78.80	73.42	82.59	53.16	71.22↑
ToG	70.48	78.80	**78.38**	79.75	54.28	71.71↑
KGP	74.29	79.26	74.77	82.28	51.49	71.86↑
GFM-RAG	72.38	80.65	72.07	82.59	52.79	72.10↑
GraphRAG	75.24	**81.57**	77.48	80.70	52.42	72.50↑
HippoRAG	70.48	80.18	74.32	81.65	**56.13**	72.64↑
RAPTOR	**76.67**	80.65	77.48	82.28	54.83	**73.58**↑

Table 4: Comparison of generation process.

As shown in Tab.4. Given that GPT-4o-mini already exhibits strong question-answering capabilities, not all GraphRAG methods effectively enhance its performance. Notably, DALK and G-Retriever degrade LLM performance; their over-reliance on structural information at the expense of semantic content introduces excessive noise during generation, impairing LLM judgment accuracy. LightRAG, ToG, and KGP achieve slight performance improvements, indicating their retrieved content provides marginal assistance for generation tasks. In contrast, GFM-RAG, GraphRAG, and HippoRAG significantly boost LLM performance by effectively integrating graph structural information with chunk-level semantics: GFM-RAG leverages large-scale pretraining to obtain a robust foundation model, GraphRAG optimizes retrieval using community-based information, and HippoRAG enhances retrieval efficiency via PageRank algorithm. The top-performing method in experiments is RAPTOR, which constructs a tree structure through iterative clustering, a design that aligns with the natural hierarchical organization of textbook data, enabling efficient retrieval of relevant information. Additionally, most GraphRAG methods outperform traditional RAG baselines such as BM-25 and TF-IDF, highlighting the utility of graph-based architectures in improving generation accuray.

4.4 Evaluation of reasoning capabilities

As shown in Tab.5. In contrast to the high accuracy in generation tasks, GPT-4o-mini exhibits a notable decline in reasoning performance. The decrease in R score indicates that LLMs often fail to perform correct reasoning, instead selecting answers through conjecture or pattern matching in many cases. The drop in AR score suggests that even when LLMs provide correct answers, their reasoning processes may be flawed; alternatively, they might generate correct reasoning but choose incorrect answers. Importantly, all GraphRAG methods significantly enhance the reasoning capabilities of LLMs: through distinct algorithmic designs, these methods retrieve not only semantically relevant corpus for questions but also identify multi-hop dependent corpus in the knowledge base, providing evidential support for LLM reasoning. This enables LLMs to reason based on external information rather than relying solely on internal knowledge for conjecture. In terms of algorithm performance, the

distribution aligns with that of generation tasks: HippoRAG and RAPTOR remain the top performers, which is intuitive, since retrieving useful information is inherently correlated with enabling correct reasoning. Additionally, most GraphRAG methods still outperform traditional RAG baselines.

Method	Reasoning											
	FB		MC		MS		TF		OE		Average	
	R	AR	R	AR	R	AR	R	AR	R	AR	R	AR
GPT-4o-mini	64.76	53.33	55.07	50.92	54.50	39.19	58.23	53.40	49.26	9.76	55.45	39.78
TF-IDF	68.09	52.61	52.76	49.19	56.30	43.02	64.08	61.23	50.37	10.50	57.61	42.38
BM-25	69.04	56.42	57.14	53.11	57.20	42.79	65.18	62.18	50.74	11.52	59.18	44.15
DALK	70.95	55.24	54.15	50.35	59.01	46.40	62.18	58.23	54.09	9.67	58.89	42.12
KGP	64.29	49.29	56.45	52.07	58.11	44.37	64.08	60.68	52.42	8.92	58.74	42.22
GraphRAG	**71.43**	55.24	56.22	52.42	57.66	45.72	63.61	60.13	53.16	10.50	59.43	43.30
G-Retriever	70.00	55.00	**57.60**	**53.46**	60.81	48.20	64.24	60.21	53.35	10.04	60.17	43.66
LightRAG	66.19	47.86	57.14	52.30	**61.71**	**49.10**	66.61	63.45	53.16	10.13	60.46	43.81
ToG	70.00	53.10	56.00	51.73	57.21	45.72	65.66	62.26	54.46	12.08	60.17	44.01
GFM-RAG	70.00	54.76	56.22	52.07	58.11	45.50	66.46	**63.69**	53.72	10.69	60.36	44.30
HippoRAG	66.67	50.48	56.68	52.30	59.91	47.52	**67.25**	63.61	**55.02**	12.36	**60.90**	44.55
RAPTOR	**71.43**	**57.86**	56.45	52.07	60.36	**49.10**	66.30	62.90	53.90	**13.57**	60.81	**45.53**

Table 5: Comparison of reasoning ability.

4.5 Topic-specific generation accuracy analysis

Given our dataset spans 16 distinct topical domains, we conducted a fine-grained analysis of GraphRAG's impact on LLM generation accuracy. Overall, GraphRAG yields consistent improvements in most areas; However, several intriguing findings emerge: **1) Mathematics Domain.** All GraphRAG methods degrade the LLM's generation accuracy in mathematics. This is attributed to the critical reliance of mathematical problems on rigorous symbolic manipulation and precise reasoning chains; models must internally "compute" each deductive step rather than relying on keyword matching from external texts. Most documents retrieved through GraphRAG are explanatory or conceptual, with symbolic notation, formula layouts, and contextual structures often misaligned with the problem requirements, leading to ambiguities or loss of key steps during the extraction and transformation of information. **2) Ethics Domain.** Both GraphRAG and the LLM itself exhibit mediocre performance in ethics. We posit that ethical problems fundamentally involve subjective value judgments, whose meanings depend on dynamic contexts of moral trade-offs and social norms. The symbolic representations captured by LLMs through statistical learning struggle to accurately model ambiguous ethical constructs, introducing intrinsic limitations in reasoning. **3) Robustness.** Excellent GraphRAG approaches such as RAPTOR substantially enhance LLM generation accuracy across most topics, demonstrating robust performance that validates their cross-domain effectiveness.

Figure 2: Comparison of Generation Accuracy by Topic.

4.6 Observation

'Can GraphRAG improve performance across all question types?'

Accuracy drop of MC questions. LLMs have internalized vast amounts of knowledge through extensive training on large corpora, enabling them to often correctly select answers in multiple-choice tasks. However, GraphRAG's retrieval-based augmentation may introduce redundant or loosely related information that does not precisely match the question context. Such retrieval noise can interfere with the model's decision-making ability, ultimately reducing its accuracy on MC questions.

Improvement in TF questions. TF questions require binary judgments about factual or logical statements. LLMs may contain blind spots or incomplete knowledge for certain facts, leading

to incorrect answers. By retrieving relevant factual evidence, GraphRAG helps the model verify statements before answering. These supplementals improve the model's accuracy on TF questions.

Improvement in OE questions. Open-ended questions allow for expansive, detailed responses, which can be challenging for LLMs that rely solely on their internal knowledge. GraphRAG mitigates this challenge by providing additional context and facts from external corpora. The retrieved information enriches the model's responses, improves subject-matter detail and expressiveness, and reduces instances of hallucination by grounding answers in explicit evidence.

Different effects in FB & MS questions. Fill-in-blank questions demand precise contextual understanding to correctly predict missing words. GraphRAG's retrieved corpora often fail to match exact contexts, introducing noise that degrades the model's performance on FB questions. Multi-select questions require choosing multiple correct answers from a set and involve reasoning over complex combinations of options; if GraphRAG's retrieval omits relevant answer options or includes irrelevant details, it can confuse the model. As a result, these question types place high demands on retrieval precision; GraphRAG may have limited benefit unless its retrieval is highly accurate.

'Can GraphRAG effectively enhance LLMs' reasoning ability?'

Experiments demonstrate that GraphRAG effectively enhances the reasoning capabilities of LLMs across diverse question types, increasing the probability of generating correct rationales alongside answers. This is attributed to their efficient retrieval mechanisms, which not only identify highly relevant corpora for questions but also provide robust evidential support for LLM reasoning processes. In particular, existing benchmarks lack systematic evaluation of GraphRAG's reasoning capabilities, an aspect of critical importance in real-world applications. For example, in the college-level educational context targeted in this document, users seeking professional knowledge expect not only correct answers, but also explicit rationales to facilitate understanding and knowledge acquisition. Similarly, in medical scenarios, patients require clear rationales for medication along with treatment recommendations to ensure transparency in decision-making. Thus, an effective GraphRAG approach should aim not only for high accuracy in answer generation but also for strong reasoning and explainability.

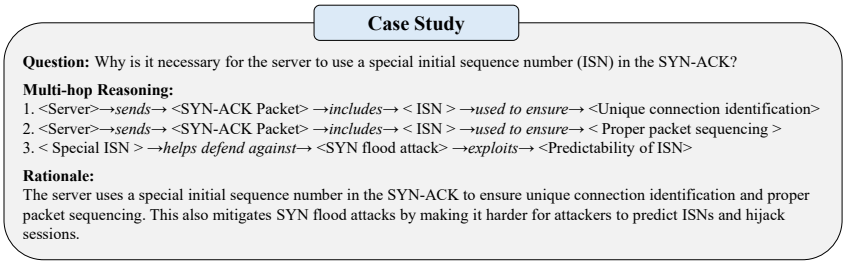

Figure 3: A case study in the topic of computer networks.

4.7 Case Study

As illustrated in Fig 3, we present a case study highlighting specific challenges within our dataset. Our questions span 16 core topics in undergraduate computer science; here, we focus on a sample from the Computer Networks section. This example demonstrates that (i) the questions demand specialized, college-level knowledge, and (ii) the correct answer cannot be retrieved through simple lookup. Instead, solving the problem requires synthesizing multiple reasoning steps to construct a coherent rationale before generating the final answer.

5 Conclusion

In this paper, we present GraphRAG-Bench, the first domain-specific benchmark designed for GraphRAG, comprising a 16-discipline dataset that challenges methods with multi-hop reasoning, complex algorithmic/programming tasks, mathematical computing, and varied question types. Our comprehensive, multi-dimensional evaluation, spanning graph construction, knowledge retrieval, generation and reasoning, quantifies the enhancement of LLM reasoning when augmented with structured knowledge. Extensive experiments on nine state-of-the-art GraphRAG methods reveal the significant role of graph integration in improving reasoning and generation performance.

References

[1] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, S. Riedel, and D. Kiela, "Retrieval-augmented generation for knowledge-intensive nlp tasks," in *Advances in Neural Information Processing Systems* (H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, eds.), vol. 33, pp. 9459–9474, Curran Associates, Inc., 2020.

[2] Y. Gao, Y. Xiong, X. Gao, K. Jia, J. Pan, Y. Bi, Y. Dai, J. Sun, M. Wang, and H. Wang, "Retrieval-augmented generation for large language models: A survey," 2024.

[3] Q. Zhang, S. Chen, Y. Bei, Z. Yuan, H. Zhou, Z. Hong, J. Dong, H. Chen, Y. Chang, and X. Huang, "A survey of graph retrieval-augmented generation for customized large language models," 2025.

[4] J. Dong, Q. Zhang, X. Huang, K. Duan, Q. Tan, and Z. Jiang, "Hierarchy-aware multi-hop question answering over knowledge graphs," in *WWW*, pp. 2519–2527, 2023.

[5] D. Edge, H. Trinh, N. Cheng, J. Bradley, A. Chao, A. Mody, S. Truitt, D. Metropolitansky, R. O. Ness, and J. Larson, "From local to global: A graph rag approach to query-focused summarization," 2025.

[6] B. Peng, Y. Zhu, Y. Liu, X. Bo, H. Shi, C. Hong, Y. Zhang, and S. Tang, "Graph retrieval-augmented generation: A survey," 2024.

[7] Y. Zhou, Y. Su, Y. Sun, S. Wang, T. Wang, R. He, Y. Zhang, S. Liang, X. Liu, Y. Ma, and Y. Fang, "In-depth analysis of graph-based rag in a unified framework," 2025.

[8] P. Sarthi, S. Abdullah, A. Tuli, S. Khanna, A. Goldie, and C. D. Manning, "RAPTOR: Recursive abstractive processing for tree-organized retrieval," in *The Twelfth International Conference on Learning Representations*, 2024.

[9] L. Luo, Z. Zhao, G. Haffari, D. Phung, C. Gong, and S. Pan, "Gfm-rag: Graph foundation model for retrieval augmented generation," 2025.

[10] X. He, Y. Tian, Y. Sun, N. V. Chawla, T. Laurent, Y. LeCun, X. Bresson, and B. Hooi, "G-retriever: Retrieval-augmented generation for textual graph understanding and question answering," in *Advances in Neural Information Processing Systems* (A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, eds.), vol. 37, pp. 132876–132907, Curran Associates, Inc., 2024.

[11] D. Li, S. Yang, Z. Tan, J. Y. Baik, S. Yun, J. Lee, A. Chacko, B. Hou, D. Duong-Tran, Y. Ding, H. Liu, L. Shen, and T. Chen, "DALK: Dynamic co-augmentation of LLMs and KG to answer Alzheimer's disease questions with scientific literature," in *Findings of the Association for Computational Linguistics: EMNLP 2024* (Y. Al-Onaizan, M. Bansal, and Y.-N. Chen, eds.), (Miami, Florida, USA), pp. 2187–2205, Association for Computational Linguistics, Nov. 2024.

[12] J. Sun, C. Xu, L. Tang, S. Wang, C. Lin, Y. Gong, L. Ni, H.-Y. Shum, and J. Guo, "Think-on-graph: Deep and responsible reasoning of large language model on knowledge graph," in *The Twelfth International Conference on Learning Representations*, 2024.

[13] Z. Yang, P. Qi, S. Zhang, Y. Bengio, W. Cohen, R. Salakhutdinov, and C. D. Manning, "HotpotQA: A dataset for diverse, explainable multi-hop question answering," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing* (E. Riloff, D. Chiang, J. Hockenmaier, and J. Tsujii, eds.), (Brussels, Belgium), pp. 2369–2380, Association for Computational Linguistics, Oct.-Nov. 2018.

[14] X. Ho, A.-K. Duong Nguyen, S. Sugawara, and A. Aizawa, "Constructing a multi-hop QA dataset for comprehensive evaluation of reasoning steps," in *Proceedings of the 28th International Conference on Computational Linguistics*, (Barcelona, Spain (Online)), pp. 6609–6625, International Committee on Computational Linguistics, Dec. 2020.

[15] H. Trivedi, N. Balasubramanian, T. Khot, and A. Sabharwal, " MuSiQue: Multihop questions via single-hop question composition," *Transactions of the Association for Computational Linguistics*, vol. 10, pp. 539–554, 2022.

[16] Z. Guo, L. Xia, Y. Yu, T. Ao, and C. Huang, "Lightrag: Simple and fast retrieval-augmented generation," 2024.

[17] B. J. Gutiérrez, Y. Shu, Y. Gu, M. Yasunaga, and Y. Su, "Hipporag: Neurobiologically inspired long-term memory for large language models," in *Advances in Neural Information Processing Systems* (A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, eds.), vol. 37, pp. 59532–59569, Curran Associates, Inc., 2024.

[18] Y. Wang, N. Lipka, R. A. Rossi, A. Siu, R. Zhang, and T. Derr, "Knowledge graph prompting for multi-document question answering," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, pp. 19206–19214, 2024.

[19] R. Y. Pang, A. Parrish, N. Joshi, N. Nangia, J. Phang, A. Chen, V. Padmakumar, J. Ma, J. Thompson, H. He, and S. Bowman, "QuALITY: Question answering with long input texts, yes!," in *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, (Seattle, United States), pp. 5336–5358, Association for Computational Linguistics, July 2022.

[20] A. Mallen, A. Asai, V. Zhong, R. Das, D. Khashabi, and H. Hajishirzi, "When not to trust language models: Investigating effectiveness of parametric and non-parametric memories," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (A. Rogers, J. Boyd-Graber, and N. Okazaki, eds.), (Toronto, Canada), pp. 9802–9822, Association for Computational Linguistics, July 2023.

[21] Y. Huang, T. Lv, L. Cui, Y. Lu, and F. Wei, "Layoutlmv3: Pre-training for document ai with unified text and image masking," in *Proceedings of the 30th ACM International Conference on Multimedia*, MM '22, (New York, NY, USA), p. 4083–4091, Association for Computing Machinery, 2022.

[22] A. Wang, H. Chen, L. Liu, K. Chen, Z. Lin, J. Han, and G. Ding, "Yolov10: Real-time end-to-end object detection," in *Advances in Neural Information Processing Systems* (A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, eds.), vol. 37, pp. 107984–108011, Curran Associates, Inc., 2024.

[23] B. Wang, C. Xu, X. Zhao, L. Ouyang, F. Wu, Z. Zhao, R. Xu, K. Liu, Y. Qu, F. Shang, B. Zhang, L. Wei, Z. Sui, W. Li, B. Shi, Y. Qiao, D. Lin, and C. He, "Mineru: An open-source solution for precise document content extraction," 2024.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The main claims in the abstract and introduction precisely outline the paper's core contributions and scope, aligning with the theoretical analysis and experimental results.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: The paper includes a dedicated "Limitations" section in appendix that candidly discusses the limitations of this work.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

Answer: [NA]

Justification: The paper does not include theorems and formulas. It is a benchmark and dataset paper.

Guidelines:

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

4. **Experimental result reproducibility**

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

Answer: [Yes]

Justification: The paper provides detailed descriptions of experimental protocols, hyperparameters and evaluation metrics, enabling reproduction of main results; Dataset is also submitted.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

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

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

   Answer: [Yes]

   Justification:The paper provides the dataset proposed.

   Guidelines:

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

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: The paper specifies all the details, such as hyperparameters and base LLM, etc.

   Guidelines:

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

7. **Experiment statistical significance**

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

   Answer: [NA]

   Justification: The experiments are not related to statistical significance.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: We provide it in the supplementary materials.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: The research conforms with the NeurIPS Code of Ethics in every respect.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [Yes]

    Justification: We have discussed in the paper that this research will have a great positive impact on the field of education.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

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

    Answer: [NA]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: Please refer to supplementary materials.

    Guidelines:

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

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

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

    Answer: [Yes]

    Justification: Please refer to supplementary materials.

    Guidelines:

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

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

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

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing or research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

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

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

    Answer: [NA]

    Justification: The paper does not involve research with human subjects or crowdsourcing.

    Guidelines:

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

16. **Declaration of LLM usage**

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

Answer: [NA]

Justification: LLMs is not applied to any original content about the manuscript.

Guidelines:

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