

Article

Not peer-reviewed version

Hyper-RAG: Combating LLM Hallucinations using Hypergraph-Driven Retrieval-Augmented Generation

[Yifan Feng](#) , Hao Hu , Xingliang Hou , Shiquan Liu , [Shihui Ying](#) , Shaoyi Du , Han Hu , Yue Gao *

Posted Date: 9 April 2025

doi: 10.20944/preprints202504.0600.v2

Keywords: large language models; retrieval-augmented generation; hypergraph; hallucination mitigation



Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a Creative Commons CC BY 4.0 license, which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Article

Hyper-RAG: Combating LLM Hallucinations using Hypergraph-Driven Retrieval-Augmented Generation

Yifan Feng ¹, Hao Hu ², Xingliang Hou ³, Shiquan Liu ², Shihui Ying ⁴, Shaoyi Du ², Han Hu ⁵ and Yue Gao ^{1,*}

¹ School of Software, Tsinghua University, 100871, Beijing, China; e-mail@e-mail.com

² Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, 710049, Xi'an, China

³ School of Software Engineering, Xi'an Jiaotong University, 710049, Xi'an, China

⁴ Department of Mathematics, School of Science, Shanghai University, 200000, Shanghai, China

⁵ School of Information and Electronics, Beijing Institute of Technology, 100871, Beijing, China

* Correspondence: gaoyue@tsinghua.edu.cn

Abstract: Large language models (LLMs) have transformed various sectors, including education, finance, and medicine, by enhancing content generation and decision-making processes. However, their integration into the medical field is cautious due to hallucinations, instances where generated content deviates from factual accuracy, potentially leading to adverse outcomes. To address this, we introduce Hyper-RAG, a hypergraph-driven Retrieval-Augmented Generation method that comprehensively captures both pairwise and beyond-pairwise correlations in domain-specific knowledge, thereby mitigating hallucinations. Experiments on the NeurologyCrop dataset with six prominent LLMs demonstrated that Hyper-RAG improves accuracy by an average of 12.3% over direct LLM use and outperforms Graph RAG and Light RAG by 6.3% and 6.0%, respectively. Additionally, Hyper-RAG maintained stable performance with increasing query complexity, unlike existing methods which declined. Further validation across nine diverse datasets showed a 35.5% performance improvement over Light RAG using a selection-based assessment. The lightweight variant, Hyper-RAG-Lite, achieved twice the retrieval speed and a 3.3% performance boost compared with Light RAG. These results confirm Hyper-RAG's effectiveness in enhancing LLM reliability and reducing hallucinations, making it a robust solution for high-stakes applications like medical diagnostics. Our source code is available at <https://github.com/iMoonLab/Hyper-RAG>.

Keywords: large language models; retrieval-augmented generation; hypergraph; hallucination mitigation

1. Introduction

Large language models (LLMs) have revolutionized numerous sectors through their advanced content generation capabilities. In education, they enable personalized learning pathways[1,2]; in information retrieval, they enhance the precision and relevance of search results[3–5]; in finance, they improve predictive analytics and support strategic decision-making[6,7]; in medicine, they assist with preliminary diagnostics and patient management[8,9]; and in elder care, they facilitate cognitive engagement and support for daily living activities[10]. Despite these advancements, the integration of LLMs within the medical domain has been relatively cautious. This hesitancy primarily stems from concerns regarding the accuracy and reliability of the generated content, which can introduce uncertainty into clinical decision-making processes and potentially lead to adverse medical outcomes[11–13]. LLMs are adept at interpreting input data and generating responses based on their training data, often exhibiting high confidence in their outputs. However, this confidence does not inherently guarantee factual correctness, resulting in discrepancies commonly referred to as LLM hallucinations[14].

LLM hallucinations occur when the generated content diverges from established facts, colloquially termed as “bullshit.” For instance, in the diagnosis of neurological disorders, an LLM might incorrectly

attribute symptoms to an unrelated condition, potentially misleading healthcare professionals[11–13]. Extensive research has been conducted to elucidate the underlying causes of these hallucinations, with findings suggesting that they likely arise from the models' training methodology, characterized by "data compression[15]." The training process typically involves self-supervised tasks that compress and reconstruct vast datasets. While LLMs can accurately reconstruct approximately 98% of the training data, the remaining 2% may result in significantly inaccurate or misleading responses[16]. Enhancing the models' capabilities can mitigate the frequency of hallucinations; however, the persistent "last mile" challenge continues to impede their reliable application in contexts that demand stringent adherence to factual accuracy, such as in medical practice. However, strategies aimed at enhancing the capabilities of LLMs entail substantial costs, often necessitating significant computational resources to train new models from scratch. This resource-intensive process poses scalability challenges and limits the feasibility of frequent model updates. Moreover, these enhancement strategies do not fully mitigate the loss of knowledge induced by data compression during training. As a result, even with increased model capacity, certain informational gaps and inaccuracies persist, underscoring the need for alternative approaches to preserve and integrate comprehensive knowledge without incurring prohibitive costs[17].

To enhance LLMs' capacity to retain and comprehend critical knowledge, thereby mitigating hallucinations, retrieval-augmented generation (RAG)[4,18–21], strategies have garnered extensive scholarly attention. RAG operates by constructing domain-specific knowledge repositories and employing vector-based retrieval techniques to extract pertinent prior information related to a given query. By constraining the generative process with this external knowledge, RAG enables LLMs to produce more accurate and reliable content, particularly concerning sensitive data such as numerical values or product names[20]. For instance, in the medical domain, the application of RAG allows LLMs to precisely identify medication names, dosages, and administration schedules[19]. In scenarios where hallucinations might lead to erroneous key information, the model's output may appear coherent and logically sound, yet critical inaccuracies can result in severe repercussions, including medical errors[11–13]. Thus, RAG serves as a crucial mechanism to ensure the fidelity of LLM-generated information in high-stakes environments.

The efficacy of RAG hinges fundamentally on the representation of domain-specific knowledge, spawning a diverse array of methodologies. The most rudimentary form of RAG[20] involves partitioning the raw corpus into manageable chunks and employing keyword-based retrieval to identify segments pertinent to a given query. Advancements in this domain have led to graph-based organizational strategies, exemplified by seminal approaches such as GraphRAG[22] and LightRAG[23]. GraphRAG enhances retrieval precision by extracting comprehensive knowledge graphs from the corpus and establishing hierarchical correlations among entities through clustering techniques. In contrast, LightRAG introduces a dual-layered knowledge graph architecture, comprising both local and global structures, to effectively organize and index granular details alongside overarching concepts within the original knowledge base. The quintessential attribute of these classical RAG methodologies lies in their ability to structurally encode the knowledge embedded within raw textual data, facilitating rapid retrieval of relevant prior information in response to specific inquiries. By leveraging these meticulously curated knowledge points, RAG frameworks empower LLMs to anchor their generative outputs in verified data, thereby mitigating the incidence of hallucinations and enhancing the factual integrity of the responses, as opposed to relying solely on the inherently compressed knowledge acquired during model training.

Structuring raw corpus data can significantly enhance the efficiency of information retrieval; however, existing graph-based approaches to information architecture often result in substantial data loss[24,25]. Specifically, traditional graphs are constrained to representing pairwise correlations between entities, as illustrated in Figure 1. In medical contexts, for example, graphs can depict binary interactions between drugs but fail to capture the complex interactions involving multiple medications simultaneously[26–28]. Similarly, in narrative storytelling, while graphs can effectively model intricate

correlations between characters, they are inadequate for representing events that involve multiple characters interacting concurrently[29]. These beyond-pairwise correlations are typically lost during the construction of knowledge graphs, thereby depriving LLMs of comprehensive prior information. Consequently, developing more comprehensive methods for information representation is imperative to enable LLMs to access critical knowledge and effectively mitigate the occurrence of hallucinations.

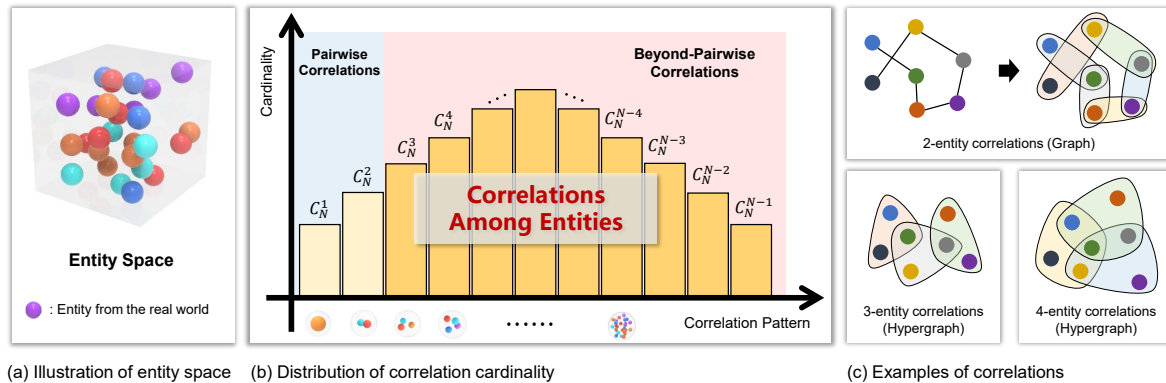


Figure 1. Illustration of Complex Correlation Modeling in Data. **a**, The real-world entity space, depicting the various entities present in the dataset. **b**, Potential complex correlations among these entities, including low-order correlations such as pairwise correlations or self-relations, and high-order correlations involving interactions among three or more entities. **c**, Visualization of entity correlations using circles to represent correlations between entities. The structure is modeled as a 2-uniform hypergraph, emphasizing pairwise connections. Another example illustrates correlations among three and four entities, with circles encompassing three and four entities, respectively.

To comprehensively capture both pairwise and multi-way correlations inherent in raw data, it is imperative to adopt a modeling approach that ensures complete coverage of these correlations, thereby providing LLMs with more robust and effective prior knowledge. Hypergraphs[30] emerge as a potent tool for modeling complex correlations due to their inherent flexibility. Unlike traditional graphs, where edges are limited to connecting two nodes and thus can only represent pairwise correlations, hypergraphs utilize hyperedges that can link any number of nodes, thereby facilitating the representation of multi-way correlations. As depicted in Figure 1, the correlations among points within the raw data space can be diverse, encompassing both pairwise and beyond-pairwise correlations. These varied connections collectively provide a comprehensive coverage of the possible interaction patterns within the data. Consequently, hypergraphs serve as an advanced framework for modeling inter-data correlations, enabling the complete and accurate representation of information contained within the data. This enhanced representation is crucial for empowering LLMs to access and utilize a more extensive and precise set of prior knowledge, thereby mitigating issues such as hallucinations and improving the reliability of generated outputs.

To mitigate hallucinations in LLMs, we propose a Hypergraph-Driven Retrieval-Augmented Generation method (Hyper-RAG) by incorporating hypergraph modeling into the RAG framework. Unlike existing RAG[22,23] approaches that typically utilize traditional graph structures to represent pairwise correlations, our method leverages hypergraphs to capture the intricate and multifaceted correlations present in raw data. Specifically, low-order correlations are employed to delineate direct connections between entities, while high-order correlations and group correlations are utilized to characterize more complex interactions. The process begins with the extraction of entities from the raw dataset, which serve as nodes in the hypergraph. Subsequently, both low-order and high-order correlations between these entities are identified and integrated to construct a hypergraph-based knowledge repository. During the question-answering phase, key entities are first extracted from the input query, and relevant prior corpus information is retrieved from the knowledge base using the hypergraph structure. The inclusion of high-order correlations ensures a more comprehensive retrieval of pertinent information, thereby providing the LLM with a richer set of prior knowledge.

This approach effectively compensates for the information loss resulting from the compression inherent in model training, thereby enhancing the accuracy and reliability of the generated responses.

The core of Hyper-RAG lies in utilizing hypergraphs to achieve a comprehensive and structured representation of knowledge from raw data, thereby minimizing information loss. Figure 2 provides an example illustrating how entities, low-order correlations, and high-order correlations are extracted from the raw corpus. For instance, consider the following excerpt from the corpus: “Neurologic lesions that cause hyperventilation are diverse and widely located throughout the brain, not just in the brainstem. In clinical practice, episodes of hyperventilation are most often seen in anxiety and panic states. The traditional view of ‘central neurogenic hyperventilation’ as a manifestation of a pontine lesion has been brought into question by the observation that it may occur as a sign of primary cerebral lymphoma, in which postmortem examination has failed to show involvement of the brainstem regions controlling respiration.” From this passage, entities such as brain, neurologic lesions, anxiety states, and hyperventilation are identified. Low-order correlations, for example, the correlation between neurologic lesions and hyperventilation, are extracted as “Neurologic lesions can lead to episodes of hyperventilation by impacting brain regions that control breathing.” Furthermore, high-order correlations involving multiple entities, such as brainstem, primary cerebral lymphoma, neurologic lesions, and postmortem examination, are also identified. These high-order correlations encompass significant entities that illustrate the connections between brain regions, cancer pathology, and research methods involved in assessing neurogenic responses like hyperventilation. This comprehensive correlation modeling facilitates a more complete knowledge structure. In contrast, if only pairwise correlations are extracted using traditional graphs, the intricate correlations among multiple entities cannot be adequately represented, leading to potential information loss. Such omissions may result in incomplete prior knowledge being available to LLMs, thereby undermining the effectiveness of RAG in mitigating hallucinations.

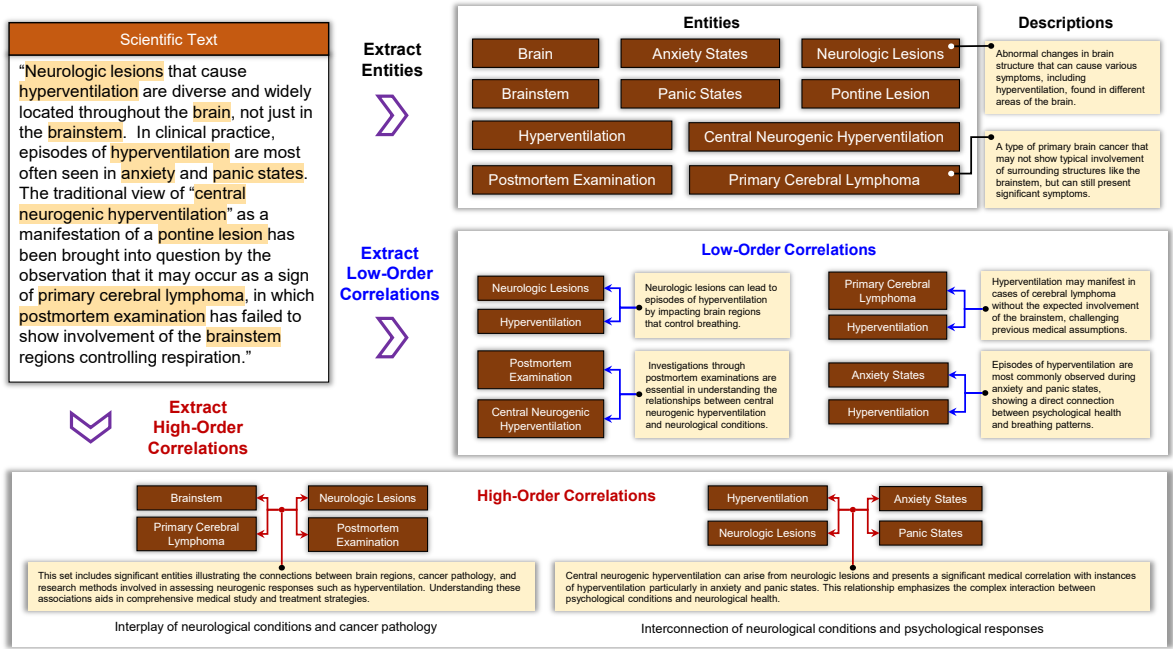


Figure 2. Illustration of Entity and Correlation Extraction from Raw Corpus: Dark brown boxes represent entities, blue arrows denote low-order correlations between entities, and red arrows indicate high-order correlations. Yellow boxes contain the original descriptions of the respective entities or their correlations.

Hallucinations in LLMs pose significant challenges, subtly undermining the logical coherence and expressive clarity of their outputs, and overtly distorting critical nouns and factual data. These phenomena are notoriously difficult to quantify due to their diverse manifestations and the complexity of natural language understanding. Existing research and benchmark evaluations have primarily

focused on assessing LLMs by posing questions with known, definitive answers to determine whether key terms and data points are accurately retrieved. While this approach provides valuable insights, it remains inherently limited and somewhat biased, as it predominantly addresses scenarios with closed-ended questions. In real-world applications, the vast majority of queries are inherently open-ended, lacking predetermined answers and often requiring nuanced, context-dependent responses of varying lengths. This discrepancy highlights a critical gap in current evaluation methodologies, which fail to capture the full spectrum of hallucination behaviors exhibited by LLMs in more complex, unconstrained environments. To bridge this gap and achieve a more comprehensive assessment of hallucinations in LLMs, we propose two novel evaluation strategies: Scoring-Based Assessment (Section 5.1) and Selection-Based Assessment (Section 5.2). The first strategy, Scoring-Based Assessment, employs five distinct metrics to evaluate model outputs across multiple dimensions, assigning scores ranging from 0 to 100. This method provides a multifaceted evaluation framework, allowing for horizontal comparisons across various enhancement strategies and effectively quantifying the extent of hallucinations present in different models. The second strategy, Selection-Based Assessment, introduces eight metrics designed to facilitate a voting mechanism between the responses of two different models. While this approach is constrained to scenarios involving the comparison of two specific models, it enables a more granular evaluation across multiple performance aspects, offering deeper insights into the relative strengths and weaknesses of each model. By implementing these two evaluation methodologies, we aim to quantitatively measure the effectiveness of various enhancement techniques in mitigating hallucinations across different LLMs. This comprehensive assessment framework not only addresses the limitations of existing evaluation methods but also provides a robust foundation for developing strategies that enhance the reliability of LLM outputs in diverse, real-world contexts.

Intuitively, our Hyper-RAG framework achieves comprehensive coverage of prior corpus knowledge by constructing a hypergraph-driven knowledge base. This comprehensive coverage effectively guides LLMs in addressing domain-specific questions, thereby enhancing the accuracy and reliability of their responses. We conduct experiments on the NeurologyCrop dataset to evaluate the augmentation effects of Hyper-RAG on six prominent LLMs: GPT-4o Mini[31], Qwen-Plus[32], LLaMa-3.3-70B[33], DeepSeek-V3[34], and Doubao-1.5-Pro[35]. The experimental results reveal that Hyper-RAG outperforms the direct application of LLMs by an average improvement of 12.3%. Furthermore, when compared to Graph RAG and Light RAG, Hyper-RAG demonstrated additional performance gains of 6.3% and 6.0%, respectively. A particularly intriguing finding emerged when we manipulated the difficulty of the questions by introducing nesting—where one question is followed by another to increase complexity. As question difficulty escalated, the performance of existing LLMs and RAG-based methods exhibited significant declines. In contrast, Hyper-RAG maintain stable performance levels. Specifically, as the difficulty increased, Hyper-RAG's improvement over direct LLM usage grow from 12.7% to 15%. This highlights Hyper-RAG's robustness in handling more complex queries.

To further validate our approach, we extend our experiments to nine diverse corpus datasets spanning multiple domains. Across these datasets, Hyper-RAG consistently outperform the conventional graph-based method, Light RAG, achieving an average performance improvement of 35.5% when evaluated using an alternative selection-based assessment method. Ablation studies are also conducted to assess the impact of different knowledge representations, original prior corpus, high-order correlations, and low-order correlations, on the capabilities of LLMs. The results indicated that the combined representation of high-order and low-order correlations effectively supplements information, thereby enhancing the performance of LLMs. Finally, in our performance analysis, Hyper-RAG demonstrate a balanced trade-off between speed and performance compared to graph-based methods. Notably, the lightweight variant, Hyper-RAG-Lite, which retains only the essential entity retrieval enhancements, achieved a twofold increase in retrieval speed and a 3.3% performance improvement over Light RAG. These findings collectively substantiate the effectiveness of our Hyper-RAG method in augmenting the capabilities of LLMs and mitigating the occurrence of hallucinations.

2. Results

To validate the effectiveness of the proposed Hyper-RAG method, we conduct experimental evaluations on nine corpus datasets across eight domains[19], with statistical details summarized in Table 1. While existing LLMs demonstrate strong performance on tasks with standardized answers, their performance on open-ended responses remains modest. Therefore, in this study, we employ domain-specific open-domain question-answering (QA) tasks to assess the Hyper-RAG strategy. We design two evaluation approaches for open-ended assessments: The first involve directly scoring each model’s output across five dimensions for comparative analysis, and the second entails conducting pairwise competitions where a large language model evaluates responses from two different models based on eight metrics and casts votes accordingly. Detailed procedures can be found in Section 5. We select prominent LLMs, including GPT-4o Mini, Qwen-Plus, LLaMa-3.3-70B, DeepSeek-V3, and Doubao-1.5-Pro, as baselines and applied various augmentation strategies, namely, RAG, GraphRAG, LightRAG, and our proposed Hyper-RAG—to evaluate their impact on model outputs. More information on those augmentation strategies is described in Section 4. Subsequently, we performed four sets of experiments to comprehensively assess our method.

Table 1. Statistical Information of the Corpus Dataset.

Dataset	Domain	#Token	#Chunk	#Ques
NeurologyCorp	Medicine	1,968,716	1,790	2,173
PathologyCorp	Medicine	905,760	824	2,530
MathCorp	Mathematics	3,863,538	3,513	3,976
AgricCorp	Agriculture	1,993,515	1,813	2,472
FinCorp	Finance	3,825,459	3,478	2,698
PhysiCorp	Physics	2,179,328	1,982	2,673
LegalCorp	Law	4,956,748	4,507	2,787
ArtCorp	Art	3,692,286	3,357	2,993
MixCorp	Mix	615,355	560	2,797

2.1. Performance of Integrating with Diversity LLMs

We first conduct experiments to evaluate the performance of the proposed Hyper-RAG when collaborating with various LLMs in order to verify whether it can effectively enhance the accuracy of LLM responses while mitigating hallucinations. Given that medical data is replete with specialized knowledge—and that even the slightest deviation in terminology can precipitate severe consequences such as misdiagnosis—we performed a comprehensive comparative study using the NeurologyCorp dataset. This dataset comprises extensive records of neuroscience knowledge and clinical practices, making it an ideal benchmark for assessing precision in a high-stakes domain.

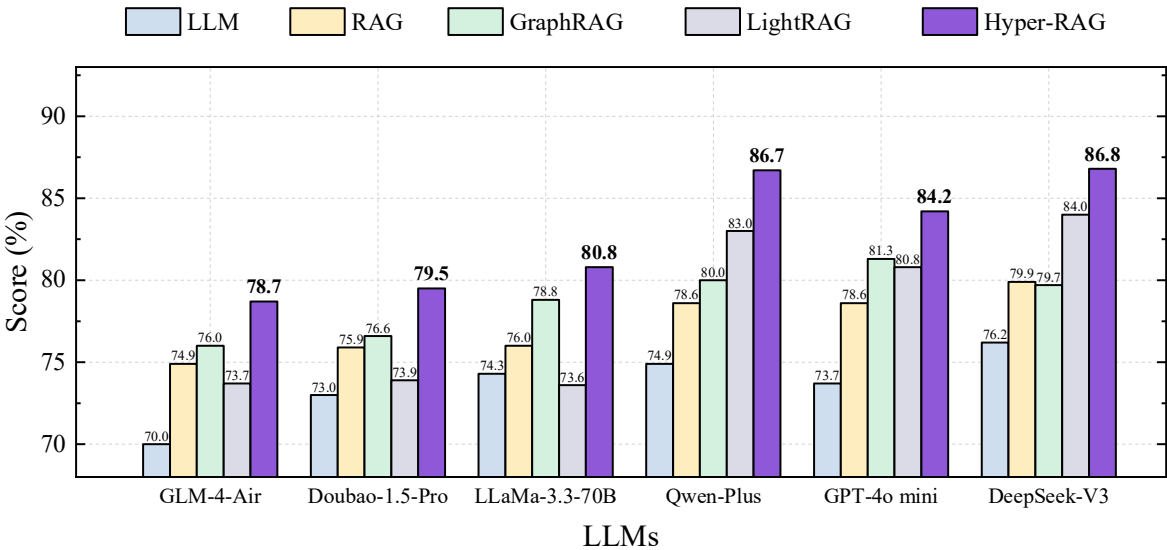


Figure 3. Results of Integrating Hyper-RAG with Different Large Language Models. Each LLM displayed on the x-axis represents the respective base model as indicated by its label. The other RAG methods shown are enhancements built upon these base models. The evaluation scores are calculated as the average of five scoring-based assessment metrics. The results demonstrate that Hyper-RAG consistently improves performance by an average of 12.3% across six LLMs, highlighting its effectiveness in enhancing model capabilities through integration with LLMs.

To set up the experiment, the corpus are segmented into 1,968,716 chunks, and distinct prior knowledge bases are constructed for each method. For the standard Retrieval-Augmented Generation (RAG) approach, embeddings are directly extracted from each chunk and stored in a vector database to facilitate knowledge retrieval. In contrast, both Graph RAG and Light RAG establish a graph-based knowledge base: from each chunk, entity information and paired correlations are extracted and stored in a graph database, with each entity and correlation accompanied by a brief textual description. Notably, when identical vertices or paired correlations emerge across multiple chunks, their descriptive texts are merged using a large model to ensure consistency. For Hyper-RAG, we built a hypergraph knowledge base from the original neuroscience corpus. From each chunk, we extracted not only entity information and paired correlations, but also higher-order correlations that transcend pairwise relationships, with every entity and correlation supplemented by a textual description. The primary distinction between Hyper-RAG and the conventional Graph RAG lies in its inclusion of non-paired, higher-order correlations, which results in a more comprehensive and structured representation of the source data. Detailed implementation specifics are provided in Section 4.

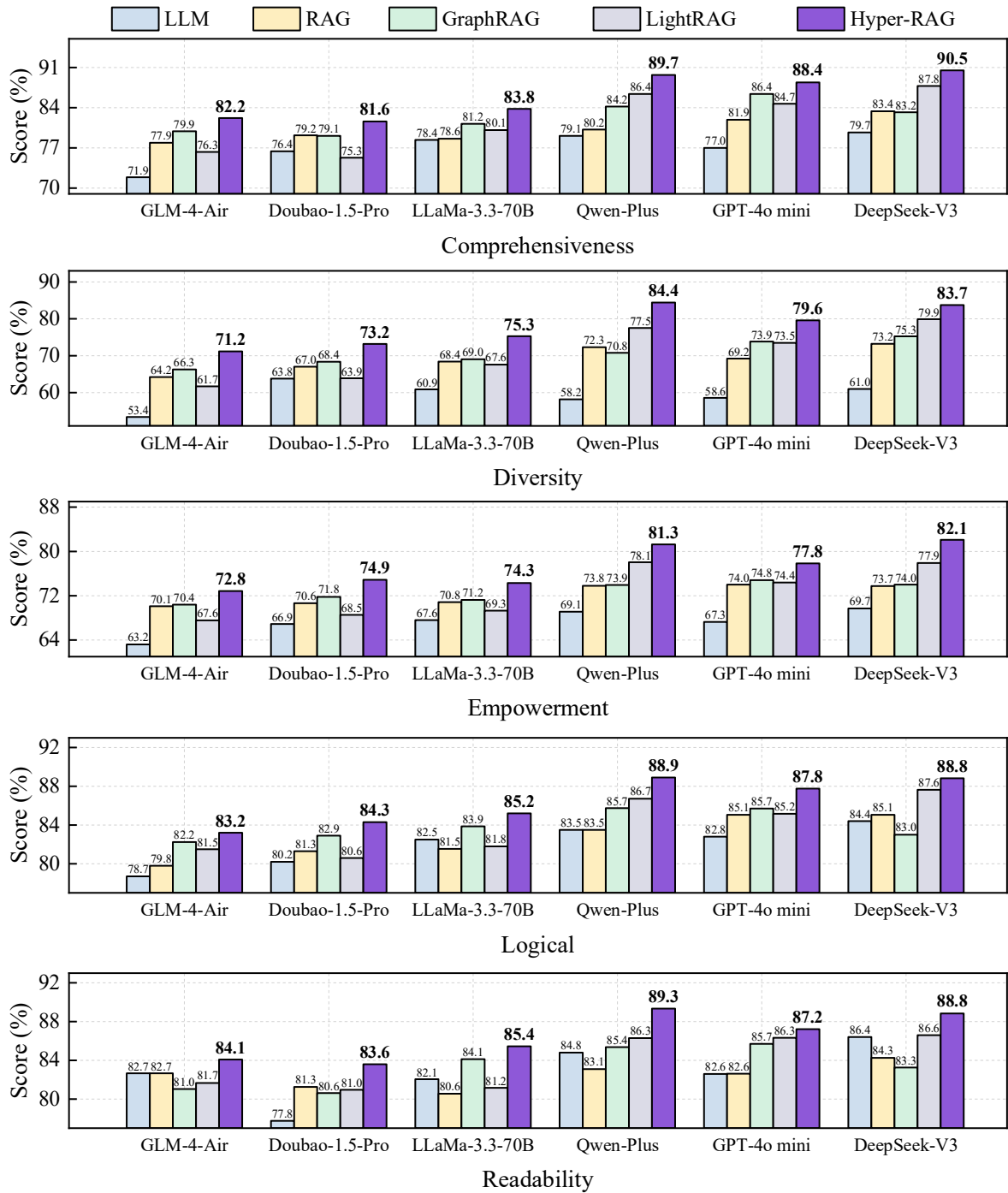


Figure 4. Detailed results of integrating Hyper-RAG with various LLMs.

To facilitate a robust horizontal comparison across different methods and LLMs, we adopt a scoring-based assessment that quantifies response quality using five distinct metrics (see Section 5.1 for additional details). Prior to the experiments, 50 unique questions are randomly sampled from different chunks using the large models, ensuring that each LLM and augmentation strategy is evaluated on an identical set of queries. The experimental results are shown in Figures 3 and 4, where the baseline LLMs are presented alongside the performance improvements achieved via the different augmentation approaches.

From Figures 3 and 4, we have six key observations. First, compared to direct LLM responses, our proposed Hyper-RAG method yields an average improvement of 12.3%, as shown in Figure 3. Notably, Hyper-RAG enhances performance by 15.8% relative to Qwen-Plus and by 14.3% in comparison to GPT-4o mini, underscoring the value of constructing a hypergraph-based prior knowledge repository

for elevating the quality of LLM outputs. Second, our findings corroborate that integrating a domain-specific prior knowledge base using the RAG strategy significantly boosts response quality. Specifically, a naive RAG approach improves baseline responses by 4.9% on average, while Graph RAG and Light RAG achieve enhancements of 6.3% and 6.0%, respectively. In contrast, Hyper-RAG delivers a 12.3% enhancement over the baseline, highlighting the critical role of domain knowledge in reinforcing LLM capabilities. Third, organizing the underlying corpus with structured associative frameworks markedly bolsters RAG performance. The introduction of relational structures yields a 7.0% improvement over unstructured methods. This increase likely stems from the fact that a well-structured representation assists in more efficient retrieval of pertinent information and promotes the diffusion of contextual cues along the relational network, thereby fostering a broader, more innovative spectrum of responses.

Fourth, both the baseline LLM and the RAG-augmented approaches demonstrate high scores in logical coherence and readability. This result reflects the extensive pre-training on large-scale corpora, which endows these models with inherent abilities to produce logically sound and accessible text regardless of the query context. Fifth, LLMs tend to score lower on metrics of comprehensiveness, diversity, and empowerment. These lower scores likely reflect intrinsic challenges in capturing nuanced domain-specific details and expressive capability. Encouragingly, the incorporation of prior corpus information via the RAG strategy results in an average improvement of 9.4% for these metrics, thereby partially offsetting these limitations. Sixth, baseline LLMs generally exhibit modest diversity scores—typically around 60, with model such as GLM, scoring as low as 53. In contrast, implementing Graph RAG elevates diversification by 19.3%, and our Hyper-RAG method further boosts the diversity score by 31.6%. This substantial gain can be attributed to the integration of additional correlation information, which more effectively steers responses towards greater divergence. Moreover, the comprehensive coverage of both lower-order and higher-order correlations cultivates a richer prior knowledge base, thereby driving significant improvements across all evaluation metrics.

2.2. Performance of Different Questioning Strategies

Given that our Hyper-RAG method provides a more comprehensive coverage of the knowledge embedded in the raw data, we further evaluated its capabilities by varying the difficulty of the questions. In our framework, questions are nested and asked progressively; the deeper the nesting, the greater the complexity of the task. This design is premised on the fact that a series of interdependent queries will magnify the impact of any inaccuracies in earlier responses, thereby serving as a stringent test of the LLM’s grasp of the domain knowledge. Table 2 illustrates examples of these progressive questions, where each subsequent inquiry is formulated based on the preceding one and connected by transitional terms (indicated in bold font). We categorized the questions into three tiers according to their escalation in difficulty: single-stage, two-stage, and three-stage questions. For this experiment, GPT-4o mini is employed as the baseline LLM, and we compare several enhancement strategies, including RAG, Graph RAG, Light RAG, and our Hyper-RAG. The experimental results are presented in Figure 5.

Table 2. Examples of questions with different difficulty.

Type	Examples
One-Stage Question	What is the role of the ventrolateral preoptic nucleus in the flip-flop mechanism described for transitions between sleep and wakefulness?
Two-Stage Question	Identify the anatomical origin of the corticospinal and corticobulbar tracts, and explain how the identified structures contribute to the control of voluntary movement.
Three-Stage Question	How does the corticospinal system function in terms of movement control, and specifically , what are the roles and interconnections of the basal ganglia and the thalamus in modulating these movements, including the effects of lesions in these areas on movement disorders?

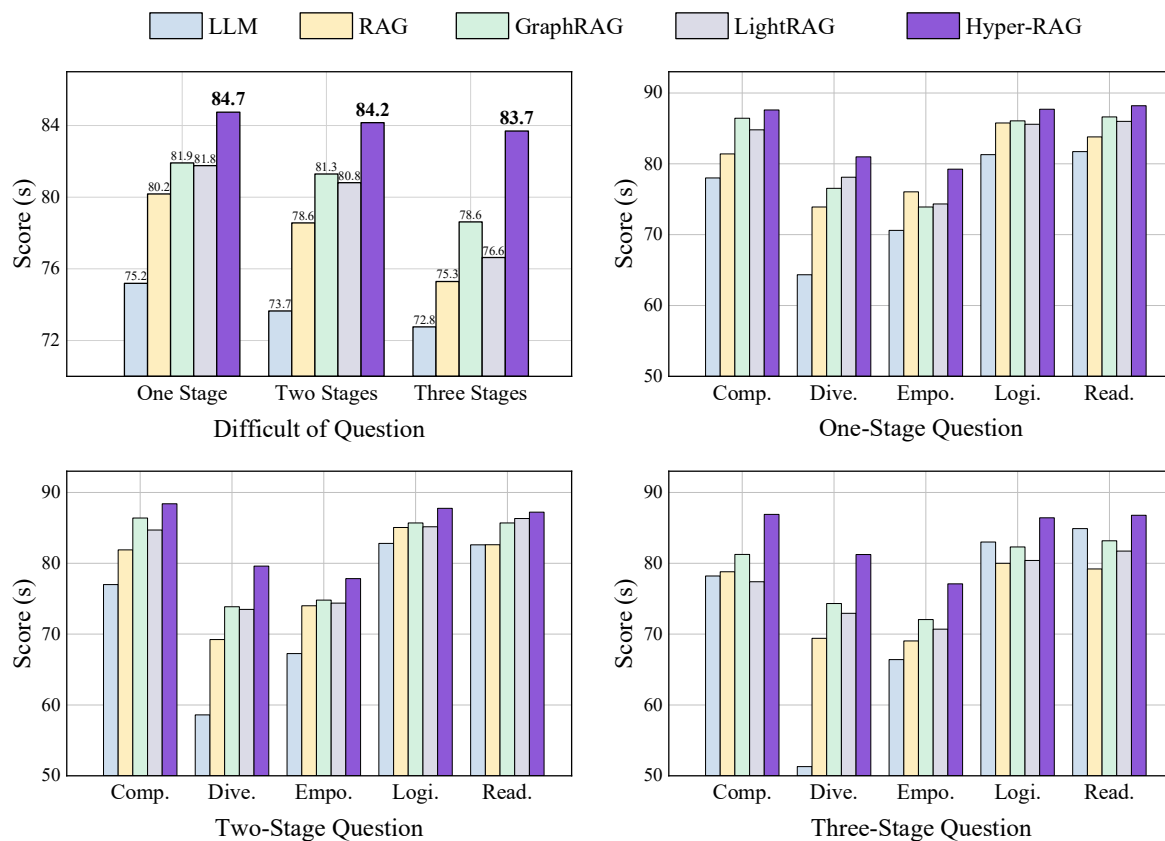


Figure 5. Experimental results of questions with different difficulty. The first subplot summarizes the experimental results across three different difficulties, with each score representing the average of five dimension-based assessments. The subsequent three subplots display the response quality scores across five dimensions for different methods, each targeting a specific difficulty. The x-axis displays six evaluation metrics: Comp. (Comprehensiveness), Dive. (Diversity), Empo. (Empowerment), Logi. (Logical), Read. (Readability), and Overall (the average of these five metrics).

Figure 5 yields three key insights from our experiments. First, as evident from the first panel, Hyper-RAG consistently demonstrates stable performance improvement across various levels of question difficulty. This indicates that employing hypergraphs to represent the full spectrum of prior corpus knowledge effectively captures domain information and guides the LLM toward more accurate responses. Second, we observe that as the complexity of the questions increases, the performance of the baseline LLM gradually declines, from 75.2 to 73.7 and then to 72.8. A similar trend is observed in other RAG methods, such as Graph RAG, which reinforces the notion that more heavily nested queries place a higher demand on prior knowledge. Although all methods exhibit some degradation in performance with increasing difficulty, RAG-based approaches still manage to enhance performance relative to the original LLM. Notably, Light RAG, which omits clustering steps, loses a portion of this vital information, and its performance deteriorates more significantly as the questions become more complex.

Finally, our Hyper-RAG shows a more pronounced improvement as the difficulty increases. Specifically, relative to the baseline LLM, our method achieves incremental gains of 12.7%, 14.3%, and 15.0% as the question complexity escalates, while, when contrasted with Light RAG, the improvements are 8.7%, 9.7%, and 5.3%, respectively. These results substantiate the superiority of a hypergraph-based, comprehensive information representation. For straightforward queries, direct responses from an LLM or simple pairwise (i.e., low-order) correlations may suffice. However, as queries become more intricate, the availability of complex higher-order correlations becomes essential to constrain and enrich the model's outputs. **This experimental trend underscores the importance of developing**

hypergraph-based structural representations and retrieval methods to meet the challenges posed by increasingly complex questions.

2.3. Performance in Diversity Domains

To validate the adaptability of Hyper-RAG across various data domains, we further evaluate its effectiveness using nine corpora spanning eight different fields (for statistical details, see Table 1). In this experiment, we select GPT-4o mini as the baseline architecture and compared our method against Light RAG, the latest graph-based RAG approach. Notably, we adopt a Selection-Based Assessment to comprehensively compare the performance of the graph-based and hypergraph-based RAG methods across these diverse domains. This assessment involved voting across eight distinct evaluation metrics that collectively capture the strengths and weaknesses of both approaches. For further details on the evaluation criteria, please refer to Section 5.2. The experimental results are presented in Figure 6.



Figure 6. Experimental results across diverse domain datasets. This figure presents the experimental outcomes of two methods evaluated on datasets from various domains. We utilize a Selection-Based Assessment approach, employing eight distinct indicators to measure and compare the performance of the methods. The Overall score is calculated as the average of these eight evaluation metrics, providing a comprehensive assessment of each method's effectiveness. The results illustrate how the two methods perform across different domain-specific challenges, highlighting their relative strengths and areas for improvement based on the aggregated evaluation criteria.

Based on experimental results shown in Figure 6, our Hyper-RAG method has demonstrated impressive improvements across nine datasets, with an average performance increase of 35.5%. Specifically, the method delivers a 55.3% improvement on LegalCrop, 41.3% on AgricCrop, and 37.5% on FinCrop, underscoring its effectiveness and adaptability across diverse domains. Although the enhancements in Accuracy and Relevance are relatively modest—averaging 29.8% and 32.0% respectively, indicating that existing low-order correlations provide a substantial baseline, Hyper-RAG exhibits notably stronger gains in Comprehensiveness and Coherence, with average improvements of 35.1% and 39.6%. These results can be attributed to Hyper-RAG’s unique capability to leverage both low- and high-order correlations, offering a more complete representation of the underlying data and enhancing response consistency through embedding-based retrieval from a vector database. Overall, while graph-based RAG methods may suffice for tasks primarily focused on accuracy and relevance, our Hyper-RAG method shows significant promise for more complex tasks that demand a broader and deeper domain understanding, paving the way for extensive future applications.

2.4. Experiments of Different Knowledge Representations

In this paper, we claim that using hypergraphs to structurally extract information from raw corpus data can more completely represent the inherent structure of the data. In this subsection, we perform an ablation study on our data organization methods based on three types of prior knowledge representations: \mathcal{D} , \mathcal{E}_{low} , and $\mathcal{E}_{\text{high}}$. Here, \mathcal{D} refers to directly splitting the raw corpus into chunks and using the embedding representation of each chunk for data retrieval; \mathcal{E}_{low} captures pairwise knowledge correlations to construct a knowledge graph; and $\mathcal{E}_{\text{high}}$ extracts non-pairwise higher-order correlations to build a knowledge hypergraph. Based on these three fundamental representations, we can construct eight different types of knowledge representations, as shown in Table 3. Evidently, directly using the LLM involves no additional data organization; using only \mathcal{D} corresponds to a simple RAG; combining \mathcal{D} with \mathcal{E}_{low} creates a variant similar to Graph RAG (with clustering removed from the original Graph RAG for a fair comparison); and employing all three yields our proposed Hyper-RAG. We employ GPT-4o mini as the foundational LLM and evaluated the various prior knowledge representation strategies using a scoring-based assessment (Section 5.2). The experimental results are presented in Table 3.

Table 3. Results of different knowledge representation strategies.

Method	\mathcal{D}	\mathcal{E}_{low}	$\mathcal{E}_{\text{high}}$	Comp.	Dive.	Empo.	Logi.	Read.	Overall	Rank
LLM	✗	✗	✗	77.00	58.60	67.26	82.80	82.60	73.65	8
-	✗	✓	✗	83.40	74.96	74.72	86.06	86.24	81.08	6
-	✗	✗	✓	84.40	75.00	75.68	86.46	86.22	81.55	5
-	✗	✓	✓	85.90	78.34	77.14	87.02	86.70	83.02	3
RAG	✓	✗	✗	81.90	69.24	74.00	85.06	82.62	78.56	7
-	✓	✓	✗	85.80	77.20	76.56	86.58	86.84	82.60	4
-	✓	✗	✓	88.26	78.80	77.52	87.34	87.04	83.79	2
Hyper-RAG	✓	✓	✓	88.40	79.60	77.84	87.76	87.22	84.16	1

From the experimental results presented in Table 3, we draw the following four key observations. First, employing the comprehensive data representation strategy, referred to as Hyper-RAG, yields the highest performance with a score of 84.16. This superiority is attributed to the holistic organization of data, which effectively imparts prior knowledge to the large language model. Second, we observe that augmenting the model with knowledge representations significantly enhances performance compared to the baseline without such augmentation, as illustrated in the first row. Specifically, the use of any single knowledge representation method results in an improvement of at least 4.9%. This underscores the efficacy of knowledge augmentation strategies in enhancing the model’s ability to respond accurately within domain-specific contexts. Third, our findings indicate that utilizing the original corpus as supplementary information leads to better performance than relying solely on descriptions

of entities and their correlations. The latter approach may introduce errors or hallucinations due to summarization by the large model, thereby negatively impacting the augmentation effectiveness. Lastly, when enhancing the model with a single type of correlation, high-order correlations outperform low-order ones. High-order correlations encompass more extensive information and cover a broader spectrum of knowledge within the correlation representation space, as depicted in Figure 1. In our current experiments, approximately 4,000 high-order and 13,000 low-order correlations were extracted from the prior corpus. Remarkably, the use of only high-order correlations resulted in superior performance, demonstrating a more effective enhancement of the large model. This indicates that a relatively smaller set of high-order correlations can encapsulate more substantial knowledge, thereby offering a promising new direction for the future development of RAG techniques.

2.5. Efficiency Analysis

We further conduct an efficiency analysis of the proposed method. Utilizing GPT-4o mini as the base model, we perform efficiency experiments on the NeurologyCrop dataset, comparing our Hyper-RAG approach with the fundamental RAG, Graph RAG, and Light RAG methods. To ensure a fair comparison unaffected by network latency, we exclusively evaluate the time required for local retrieval from the database to acquire relevant knowledge and the construction of the prior knowledge prompt. For the standard RAG, this primarily involves the direct retrieval time of chunk embeddings. In contrast, Graph RAG, Light RAG, and our Hyper-RAG method encompass both the retrieval time from node and correlation vector databases and the time required for a single layer of graph or hypergraph information diffusion. Since the retrieval time is influenced by both the specific questions and the methods employed, we calculate the average response time for each method by posing 50 questions from the dataset. Consistent with our selection metric outlined in Section 5, we employ a scoring-based assessment as the evaluation criterion. Additionally, to accommodate practical applications, we develop a lightweight variant of Hyper-RAG, termed Hyper-RAG-Lite, which preserves the essential enhancements for entity retrieval. The experimental results, presented in Figure 7, demonstrate the comparative efficiency of each method.

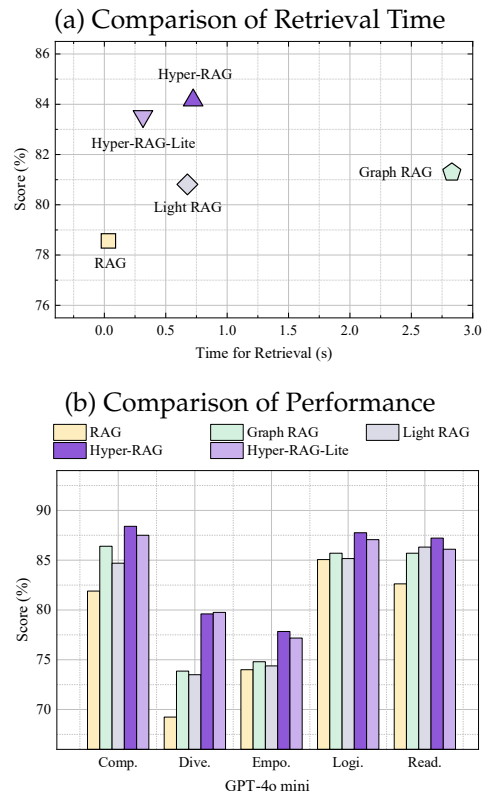


Figure 7. Efficiency Comparison of Different Augmentation Methods. **a**, Comparison of performance and time. The performance is obtained by scoring-based assessment on the NeurologyCrop dataset, where each method’s performance is the average of five indicators. The average retrieval time for RAG, Graph RAG, Light RAG, Hyper-RAG, and Hyper-RAG-Lite are 0.033s, 2.83s, 0.676s, 0.723s, and 0.315s respectively. **b**, Specific scores of the five indicators on the NeurologyCrop dataset.

Figure 7 presents a comparative analysis of performance versus time, where points closer to the top-left corner indicate faster speeds and superior performance. From the figure, we derive the following four key observations: Firstly, we observe that both the proposed Hyper-RAG and Hyper-RAG-Lite are positioned near the top-left corner of the plot, indicating that these methods excel in both speed and performance. This demonstrates the efficacy of our approach in maintaining high efficiency without compromising answer quality. Secondly, we note that RAG is situated at the far left of the plot. This positioning is attributed to its sole reliance on document corpus retrieval without incorporating structural diffusion, which confers an efficiency advantage. However, this method’s performance significantly lags behind other structure-based enhanced methods, highlighting a trade-off between speed and accuracy. Thirdly, Graph RAG achieves a performance level that is second only to our Hyper-RAG method, yet it incurs considerable time delays. The primary reason for this sluggishness is the necessity of retrieving community information in addition to node information retrieval and diffusion. Community information is derived through hierarchical clustering of nodes and lacks indexing via vector databases, necessitating layer-by-layer matching and retrieval, thereby slowing down the process. Nevertheless, the inclusion of community information, which embodies high-order correlations, effectively complements pairwise graph correlations, thereby enhancing performance. Additionally, we observe that Light RAG omits the retrieval of community information, resulting in a reduction in performance. However, this omission leads to a substantial increase in processing speed, as the computational overhead associated with managing high-order correlations is eliminated. Lastly, our Hyper-RAG method exhibits performance comparable to Light RAG while maintaining superior speed. Both methods essentially rely on prompt-based correlation extraction and indexing through vector databases and graph/hypergraph databases. However, Hyper-RAG differentiates itself by extracting both low-order and high-order correlations via prompts and utilizing a hypergraph database for indexing, thereby achieving similar efficiency levels. Crucially, Hyper-RAG compensates for the

information loss inherent in Light RAG by incorporating additional high-order correlations, resulting in enhanced performance. It is noteworthy that the Hyper-RAG-Lite variant, although retaining only entity information retrieval, still implements diffusion through high-order correlations. This ensures that Hyper-RAG-Lite introduces additional high-order information, thereby achieving performance improvements over both Light RAG and Graph RAG.

3. Discussion

In our study, we integrate Hyper-RAG with six widely used LLMs, demonstrating a significant enhancement in performance. On average, Hyper-RAG improves the models' accuracy by 12.3% compared to their direct application without retrieval augmentation. When juxtaposed with the conventional Graph RAG approach, our method yielded an additional 5.3% improvement. This superior performance can be attributed to Hyper-RAG's comprehensive coverage of domain-specific knowledge. By modeling both low-order (pairwise) and high-order (beyond-pairwise) correlations within the data, Hyper-RAG facilitates a more complete and structured representation of domain knowledge, thereby reducing information loss and enhancing the quality of the generated responses.

We also evaluate the robustness of Hyper-RAG by increasing the complexity of the questions through added nesting. The experimental results reveal that existing methods, including RAG and Graph-RAG, experienced a noticeable decline in performance under these more challenging conditions. In stark contrast, Hyper-RAG maintains its performance levels, underscoring the pivotal role of high-order correlations in enabling LLMs to handle complex queries effectively. This finding suggests that current LLMs possess untapped potential for improvement in complex question-answering scenarios and that the incorporation of high-order relational modeling can significantly bolster their ability to provide accurate and reliable responses.

Our investigations into knowledge representation highlight that the impact of prior knowledge on model performance varies across different scenarios. In simpler contexts, leveraging low-order correlations alongside the original prior corpus suffices to cover the necessary information. However, in more intricate scenarios, the inclusion of high-order correlations becomes imperative to enhance the accuracy of the model's responses. This adaptability in knowledge representation allows for the selection of appropriate prior knowledge based on the complexity of the task at hand, thereby optimizing the model's performance across diverse application domains.

Despite its advantages, Hyper-RAG presents certain limitations. The construction of the knowledge base necessitates the extraction of high-order correlations, which introduces additional steps into the knowledge base development process. Nonetheless, the number of high-order correlations is considerably smaller compared to low-order ones, mitigating the overall impact. Moreover, these extraction processes can be performed offline, thereby not impeding the real-time application of the models. In comparison, the classic Graph RAG approach relies on clustering to represent group correlations within the data, a process that is both time-consuming and resource-intensive. Light RAG, while alleviating the computational burden by omitting clustering, consequently loses high-order relational information, leading to diminished performance.

In the knowledge retrieval phase, Hyper-RAG offers distinct advantages over Graph-RAG by eliminating the need for redundant local and global retrieval processes. Instead, it allows for direct retrieval of nodes or relational structures. The retrieval of relational structures is optional; incorporating it can enhance performance but at the cost of additional computational resources. Alternatively, when only node information is retrieved, the system operates in a mode we designate as Hyper-RAG-Lite. In this mode, the integration of both low-order and high-order correlations enables the diffusion of information, thereby utilizing high-order knowledge embedded within the data. Consequently, Hyper-RAG-Lite not only accelerates the retrieval process but also improves the quality of responses generated by the LLMs, presenting a promising avenue for future research.

However, our current approach to knowledge construction faces challenges in directly extracting correlations across different data chunks, necessitating post-processing steps to merge these relations.

A significant portion of the relevant information spans multiple chunks, making it inadequate to capture through a single chunk alone. Future research should focus on developing methods for the fusion and extraction of cross-chunk correlations. Additionally, modeling relationships between different documents could further enhance the dimensionality and scalability of the knowledge base. This study has demonstrated that improved representation and organization of domain knowledge can significantly enhance the capabilities of large language models. Future work may explore automating data organization and knowledge representation techniques, fostering a deeper integration with large language models to further mitigate issues such as hallucinations.

4. Methods

In this section, we provide a comprehensive overview of the proposed Hypergraph-Driven Retrieval-Augmented Generation (Hyper-RAG) framework, encompassing the processes of knowledge extraction, indexing, and retrieval. Subsequently, we present the architecture of the open-domain question answering tasks utilized to evaluate Hyper-RAG, alongside two distinct evaluation metrics that assess both the accuracy and comprehensiveness of the generated responses. These metrics are chosen to rigorously measure the reduction in hallucinations and the enhancement in answer reliability provided by our approach. Finally, we describe the data collection and construction procedures, detailing the sources and criteria for dataset assembly. Additionally, we illustrate the various prompt templates employed at different stages of the Hyper-RAG pipeline, demonstrating how they are tailored to optimize knowledge retrieval and generation processes.

Table 4. The comparison of different LLMs augmentation strategies.

Method	Formulation	Reference
LLM	$\text{response} = \text{LLM}(\mathcal{P}_q(q))$	[31–36]
RAG	$\text{response} = \text{LLM}(\mathcal{P}_q(q, \mathcal{D}))$	[4,18–21,37–39]
Graph-RAG	$\text{response} = \text{LLM}(\mathcal{P}_q(q, \text{RAG}(q, \mathcal{D}, \mathcal{V}, \mathcal{E}_{\text{low}})))$	[22,23,40–42]
Hyper-RAG	$\text{response} = \text{LLM}(\mathcal{P}_q(q, \text{RAG}(q, \mathcal{D}, \mathcal{V}, \mathcal{E}_{\text{low}}, \mathcal{E}_{\text{high}})))$	This work

Table 4 presents a mathematical comparison of various LLMs enhancement strategies. Here, \mathcal{P}_q denotes the function that transforms the input into a prompt, where q represents the input query and \mathcal{D} is the prior corpus. The symbols \mathcal{V} , \mathcal{E}_{low} , and $\mathcal{E}_{\text{high}}$ correspond to the vertices, low-order correlation knowledge, and high-order correlation knowledge of the knowledge base constructed from the corpus, respectively. It is evident that the original LLM generates responses directly based on the input question q . In contrast, RAG retrieves relevant data from the prior corpus \mathcal{D} to assist the LLMs in providing answers. Graph-RAG further extracts low-order structural information from the prior knowledge. Our proposed Hyper-RAG simultaneously constructs both low-order and high-order correlation information from the prior corpus, enabling a more comprehensive representation of knowledge. This enhanced knowledge representation effectively reduces information loss, thereby mitigating the occurrence of hallucinations in LLMs.

4.1. Framework Schema

Figure 8 illustrates the proposed Hyper-RAG framework, encompassing both the offline construction of the knowledge database and the online retrieval-augmented response generation processes. Initially, we collect domain-specific corpora, which may include manuals, books, reports, and other relevant documents. These raw corpora are then processed using LLMs to segment the text and extract entities and their relationships. In the Hyper-RAG framework, relationships are categorized into pairwise low-order correlations and beyond-pairwise high-order correlations that represent correlations among groups of entities. The extracted knowledge is subsequently stored in a database to facilitate rapid retrieval during the query phase.

During the question-answering process, consider an example where a user with neurological disorders poses a question to the LLMs, as shown in Figure 8. When using a naive LLM, the model responds directly to the patient's query without additional context. In contrast, the Hyper-RAG strategy involves a two-step approach: first, we extract keywords from the user's question; second, we retrieve knowledge related to these similar keywords from the knowledge database. The retrieved relevant knowledge is then provided as supporting information alongside the original question to the LLMs, resulting in more accurate and reliable responses. The subsequent sections will provide a detailed description of each component and process within the Hyper-RAG.

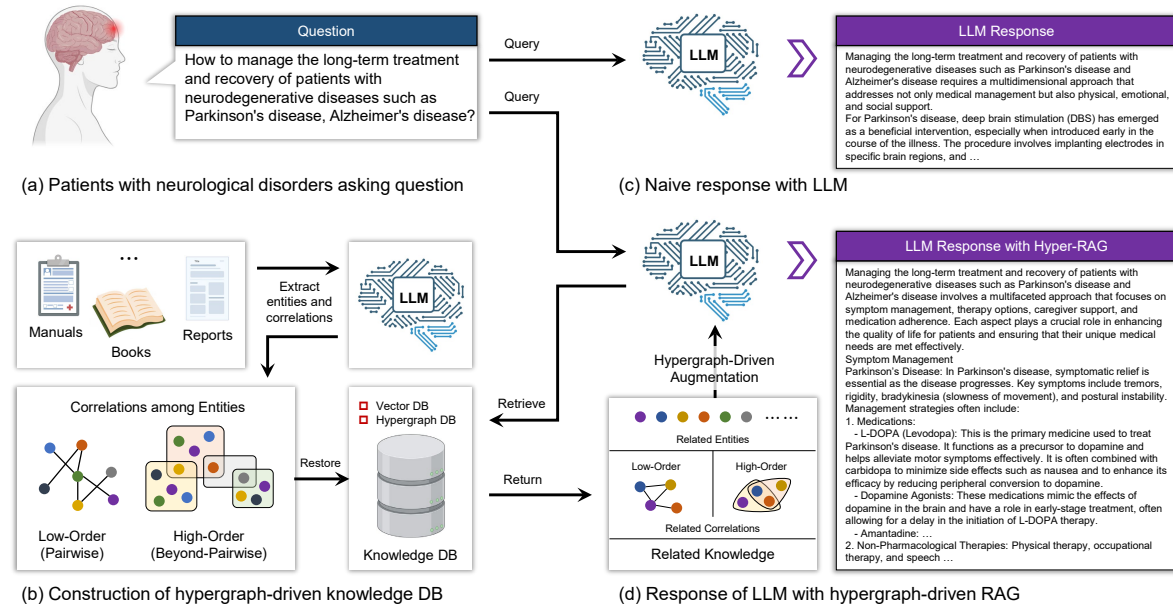


Figure 8. Schematic diagram of the proposed Hyper-RAG architecture. **a**, The patient poses a question. **b**, A knowledge base is constructed from relevant domain-specific corpora. **c**, Responses are generated directly using LLMs. **d**, Hyper-RAG generates responses by first retrieving relevant prior knowledge from the knowledge base and then inputting this knowledge, along with the patient's question, into the LLMs to formulate the reply.

4.2. Knowledge Extraction

The objective of knowledge extraction is to systematically organize raw corpora into a structured format, thereby enabling more efficient retrieval of prior information. In our approach, the corpus data can comprise various types of documents, including books, manuals, reports, and other relevant texts. We begin by preprocessing the original documents and partitioning them into uniformly sized chunks, denoted as D_i , thereby forming the corpus collection:

$$\mathcal{D} = \{D_1, D_2, \dots, D_N\}. \quad (1)$$

Subsequently, a document structuring function, ϕ , is employed to extract structural information from the corpus, resulting in a hypergraph \mathcal{G} :

$$\mathcal{G} = \phi(\mathcal{D}) \quad \text{and} \quad \mathcal{G} = \{\mathcal{V}, \mathcal{E}_{\text{low}}, \mathcal{E}_{\text{high}}\}, \quad (2)$$

where \mathcal{G} represents the hypergraph structure extracted from the documents, comprising a set of vertices \mathcal{V} , low-order correlations \mathcal{E}_{low} , and high-order correlations $\mathcal{E}_{\text{high}}$. The elements within the vertex set \mathcal{V} can be of various types, such as names of entities, task titles, or skills. The relation sets \mathcal{E}_{low} and $\mathcal{E}_{\text{high}}$ describe the connections between entities, where \mathcal{E}_{low} captures pairwise relationships and $\mathcal{E}_{\text{high}}$ encapsulates correlations involving multiple entities. For each chunk $D_i \in \mathcal{D}$, we extract entities and their descriptions using LLMs as follows:

$$\mathcal{K}_v = \text{LLM}(\mathcal{P}_{\text{ext_entity}}(D_i)) \quad \text{for} \quad D_i \in \mathcal{D}, \quad (3)$$

where $\mathcal{K}_v = \{v_1, v_2, \dots\}$ denotes the set of entities, each accompanied by a generated description, as illustrated in Figure 2. It is important to note that if multiple chunks contain the same entity, their descriptions are merged using the LLMs to ensure consistency and completeness. The function $\mathcal{P}_{\text{ext_entity}}$ serves as the prompt filler that converts the input into an appropriate prompt for entity extraction, which is detailed in Section 6.

Following entity extraction, we proceed to identify the corresponding low-order and high-order correlations within each chunk based on the extracted entities:

$$\begin{cases} \mathcal{K}_e^{\text{low}} = \text{LLM}(\mathcal{P}_{\text{ext_low}}(D_i, \mathcal{K}_v)) \\ \mathcal{K}_e^{\text{high}} = \text{LLM}(\mathcal{P}_{\text{ext_high}}(D_i, \mathcal{K}_v)) \end{cases} \quad \text{for } D_i \in \mathcal{D}, \quad (4)$$

where $\mathcal{K}_e^{\text{low}} = \{(u, v), \dots\}$ represents the set of low-order correlations between pairs of entities, while $\mathcal{K}_e^{\text{high}} = \{(u, v, \dots), \dots\}$ denotes the high-order correlations involving multiple entities. Each relation within the knowledge base is also accompanied by a descriptive narrative, as depicted in Figure 2. When identical correlations are extracted from different chunks, their descriptions are amalgamated to maintain a unified representation. This comprehensive extraction of both low-order and high-order relational information from the corpus ensures a robust and detailed knowledge base, which is critical for minimizing information loss and enhancing the retrieval process in the Hyper-RAG framework.

4.3. Knowledge Indexing

Hyper-RAG utilizes two distinct types of databases to effectively organize and manage the extracted knowledge: a vector database for storing the embedding representations of vertices and a hypergraph database for maintaining both high-order and low-order relational structures.

Vector Database

The vector database stores fixed-dimensional vector representations derived from the descriptions of each entity. These embeddings are organized into a matrix M , where each row corresponds to the vector representation of an entity or a hyperedge. During retrieval, a query vector q is compared against the vectors in the database using distance metrics such as cosine similarity or Euclidean distance. The system then retrieves the top- k nearest vectors, which may correspond to relevant entities. This approach ensures that the most semantically similar entries are efficiently identified and utilized to enhance the generation process.

Hypergraph Database

The hypergraph database stores the structural information extracted from the raw corpus, encompassing both low-order and high-order correlations. This database comprises two primary components: vertex adjacency lists and relation adjacency lists. A hypergraph extends a traditional graph by allowing hyperedges to connect more than two vertices, thereby uniformly storing both low-order (pairwise) and high-order (beyond-pairwise) correlations within this structure. Each hyperedge is represented as a tuple of vertex names, facilitating the representation of complex relationships among multiple entities. During retrieval, given a specific vertex name, the hypergraph database can swiftly access the vertex's descriptive information as well as its connected relational structures. Additionally, the database supports relational queries where, upon inputting a particular relation structure, it returns the corresponding descriptive information and the neighboring vertices associated with that relation. This dual capability allows Hyper-RAG to efficiently navigate and utilize intricate relational data, thereby enhancing the accuracy and contextual relevance of responses generated by the LLMs.

Integration of Vector and Hypergraph Databases

The integration of the vector database and the hypergraph database within the Hyper-RAG framework provides a comprehensive mechanism for knowledge storage and retrieval. While the vector database excels in capturing and retrieving semantically similar entities and correlations through

embedding spaces, the hypergraph database ensures that the structural and relational integrity of the knowledge base is maintained and easily accessible. Together, these databases enable Hyper-RAG to leverage both the semantic richness and the structural complexity of the underlying knowledge, thereby effectively mitigating hallucinations in large language models by providing accurate and contextually relevant information.

4.4. Knowledge Retrieval and LLMs Augmentation

After constructing the hypergraph knowledge base offline, we detail the methodology for augmenting the LLMs' response capabilities using knowledge databases. Given a user query q , we first extract two distinct sets of keywords: the entity keyword set \mathcal{X}_{ent} (fundamental components) and the correlation keyword set \mathcal{X}_{cor} (complex interdependencies), as follows:

$$\mathcal{X}_{\text{ent}}, \mathcal{X}_{\text{cor}} = \text{LLM}(\mathcal{P}_{\text{ext_key}}(q)) \quad \text{and} \quad \mathcal{X}_* = \{x_1, x_2, \dots\}, \quad (5)$$

where $\mathcal{P}_{\text{ext_key}}$ is the prompt used to extract keywords from the input question, as detailed in Section 6. The entity keyword set comprises specific detailed nouns, such as personal names, locations, and other discrete identifiers. In contrast, the correlation keyword set encompasses more sophisticated descriptions, typically involving interactions between two or more entities. These correlations often capture narratives, systems, responses, reactions, and various forms of interactions that emerge from the relationships among entities. By distinguishing between these two types of keyword sets, our approach effectively models both the fundamental components and the complex interdependencies within the data. This comprehensive representation enhances the retrieval-augmented generation process, enabling the LLMs to leverage a richer and more nuanced foundation of prior knowledge. Subsequently, based on these two categories of extracted keywords, we retrieve relevant information from the hypergraph database. It is important to note that entity keyword retrieval targets vertices, while correlation keyword retrieval targets hyperedges. This distinction arises because entity keywords predominantly describe individual entities, making vertices the appropriate retrieval objects. In contrast, correlation keywords describe abstract information that typically involves relationships among multiple entities, thereby necessitating hyperedges as retrieval targets. For entity information retrieval, we employ the following formulation:

$$\begin{cases} \mathcal{V}_{\text{rel}} = \{\psi_{\text{ret}}(x_i, \mathcal{V}) | x_i \in \mathcal{X}_{\text{ent}}\} & // \text{Entity information} \\ \mathcal{E}_{\text{more}} = \{e | v \in e \text{ and } v \in \mathcal{V}_{\text{rel}}\} & // \text{Extended information via diffusion} \end{cases}, \quad (6)$$

where ψ_{ret} denotes the vector-based retrieval function, which retrieves vertices similar to the input x_i from the vertex vector database. Subsequently, $\mathcal{E}_{\text{more}}$ is used to diffuse through the associated structural relationships, thereby obtaining hyperedges connected to these vertices as supplementary information.

Similarly, for correlation information retrieval, we use the following formulation:

$$\begin{cases} \mathcal{E}_{\text{rel}} = \{\psi_{\text{ret}}(x_i, \mathcal{E}) | x_i \in \mathcal{X}_{\text{cor}}\} & // \text{Correlation information} \\ \mathcal{V}_{\text{more}} = \{v | v \in e \text{ and } e \in \mathcal{E}_{\text{rel}}\} & // \text{Extended information via diffusion} \end{cases}, \quad (7)$$

where $\psi_{\text{ret}}(x_i, \mathcal{E})$ retrieves hyperedges related to the correlation keywords from the hyperedge vector database. Through one-step diffusion, the vertices associated with these hyperedges are acquired as supplementary information. Due to the constraints on the LLM's input context length, we aggregate and rank the retrieved entity and correlation information, selecting the most relevant data based on the maximum permissible context length to serve as prior knowledge for augmenting the LLM. In practical applications, we incorporate the textual content from the original chunks associated with the relevant vertices and hyperedges as prior information. This approach is employed to mitigate the

potential hallucinations that may arise from descriptions synthesized by the LLM, thereby ensuring the reliability of the supplementary knowledge.

4.5. Dataset

To assess the efficacy of the proposed Hyper-RAG framework, we curate an extensive collection of corpora encompassing both domain-specific and mixed-domain datasets. Recognizing that domain-specific data is more susceptible to hallucinations, owing to the heightened demands for lexical precision in specialized fields, we selected nine corpora across eight distinct domains: medicine, mathematics, agriculture, finance, physics, law, and art, the comprehensive statistics of which are detailed in Table 1. Additionally, to evaluate the model's performance in managing general knowledge across diverse areas, we construct a mixed-domain dataset. These corpora, referenced in [19,43], are primarily extracted from books, reports, academic papers, narratives, and encyclopedias, with an average token count of 2,733,191 per dataset. Each raw corpus underwent preprocessing to eliminate special symbols and non-textual elements, retaining solely the textual information. Subsequently, the sanitized corpora were partitioned into fixed-size chunks of 1200 tokens, with an overlapping segment of 100 tokens between consecutive chunks to ensure contextual coherence. For performance evaluation, a LLMs (GPT-4o mini) is employed to generate 50 questions per dataset. The question generation leverage the $\mathcal{P}_{\text{ext}_q}(q)$ prompt tailored to each chunk, facilitating automatic question formulation by the LLM. Furthermore, the origin chunk for each question are recorded to enable Scoring-Based Assessment, wherein the corresponding source chunk served as the reference answer for evaluating the responses.

5. Evaluation Criteria

The evaluation of LLMs has predominantly been conducted using benchmarks with predefined answers, such as SQuAD, GPT-3 Benchmarks, and others. These benchmarks typically involve generating concise responses that align with standard answers. However, in real-world applications, obtaining standard answers is often impractical, especially for open-ended questions where responses can vary widely. In such scenarios, the absence of supervisory information makes LLMs more prone to hallucinations, leading to the confusion of critical entities like names, dates, and locations. This issue is particularly detrimental in sensitive domains such as medicine, where inaccuracies can result in significant consequences. To effectively evaluate LLMs in open-ended contexts, we introduce two assessment strategies: Scoring-Based Assessment and Selection-Based Assessment.

5.1. Scoring-Based Assessment

Scoring-Based Assessment is designed to facilitate the comparative evaluation of multiple model outputs by quantifying their performance across various dimensions. This approach allows for a nuanced assessment of model capabilities by providing scores on several key metrics. However, a notable limitation is its reliance on reference answers. In our preprocessing steps, we leverage the source chunks from which each question is derived as reference answers. Using these references, we construct a scoring prompt, denoted as $\mathcal{P}_{\text{eval_score}}$, which directs the LLM to evaluate open-ended responses based on five dimensions:

1. **Comprehensiveness (0-100):** Assesses whether the response sufficiently addresses all relevant aspects of the question without omitting critical information.
2. **Diversity (0-100):** Evaluates the richness of the content, including additional related knowledge beyond the direct answer.
3. **Empowerment (0-100):** Measures the credibility of the response, ensuring it is free from hallucinations and instills confidence in the reader regarding its accuracy.
4. **Logical (0-100):** Determines the coherence and clarity of the response, ensuring that the arguments are logically structured and well-articulated.

5. **Readability (0-100):** Examines the organization and formatting of the response, ensuring it is easy to read and understand.

Each evaluation dimension is scored on a scale from 0 to 100, with higher scores indicating better performance. Recognizing the challenges associated with assigning broad numerical scores directly, we implemented a hierarchical scoring system by dividing each dimension into five distinct levels. Each level corresponds to specific criteria that provide clear and consistent guidelines for scoring. To illustrate, we present the classification for the **Comprehensiveness** dimension:

1. **Level 1 | 0-20:** The answer is extremely one-sided, leaving out key parts or important aspects of the question.
2. **Level 2 | 20-40:** The answer has some content but misses many important aspects and is not comprehensive enough.
3. **Level 3 | 40-60:** The answer is more comprehensive, covering the main aspects of the question, but there are still some omissions.
4. **Level 4 | 60-80:** The answer is comprehensive, covering most aspects of the question with few omissions.
5. **Level 5 | 80-100:** The answer is extremely comprehensive, covering all aspects of the question with no omissions, enabling the reader to gain a complete understanding.

While the detailed five-level classification is exemplified for the **Comprehensiveness** dimension, similar hierarchical structures have been established for the other evaluation metrics (**Diversity**, **Empowerment**, **Logical**, and **Readability**) to ensure uniformity and precision in the scoring process. Finally, an overall performance score is calculated as the average of the individual dimension scores. A higher overall performance score indicates greater accuracy in expression and a lower probability of hallucinations.

5.2. Selection-Based Assessment

Selection-Based Assessment is tailored for scenarios where preliminary candidate models are available, enabling a comparative evaluation through a binary choice mechanism. This method does not require reference answers, making it suitable for diverse and open-ended questions. However, its limitation lies in its comparative nature, as it only allows for the evaluation of two models at a time.

In this strategy, the outputs from two methods, denoted as A_{out} and B_{out} , are simultaneously presented to the LLM, denoted as $\mathcal{P}_{\text{eval_select}}$. The model is then instructed to select the better response based on eight evaluation criteria:

1. **Comprehensiveness:** How much detail does the answer provide to cover all aspects and details of the question?
2. **Empowerment:** How well does the answer help the reader understand and make informed judgments about the topic?
3. **Accuracy:** How well does the answer align with factual truth and avoid hallucination based on the retrieved context?
4. **Relevance:** How precisely does the answer address the core aspects of the question without including unnecessary information?
5. **Coherence:** How well does the system integrate and synthesize information from multiple sources into a logically flowing response?
6. **Clarity:** How well does the system provide complete information while avoiding unnecessary verbosity and redundancy?
7. **Logical:** How well does the system maintain consistent logical arguments without contradicting itself across the response?
8. **Flexibility:** How well does the system handle various question formats, tones, and levels of complexity?

For each of these eight criteria, the LLM selects the superior response between A_{out} and B_{out} . The cumulative votes across all criteria determine the overall score for each model. This voting mechanism ensures a balanced evaluation based on multiple facets of response quality, thereby providing a robust assessment of the models' relative performance without the need for predefined reference answers.

6. Prompts

6.1. Extracting Entities, Correlations and Keywords

Extracting Entities

Formulation: $\mathcal{P}_{\text{ext_entity}}(D_i)$
 D_i denotes the text chunk.

Prompt: *Identify all entities. For each identified entity, extract the following information:*

- *entity_name*: Name of the entity, use same language as input text. If English, capitalized the name.
- *entity_type*: One of the following types: [entity_types]
- *entity_description*: Comprehensive description of the entity’s attributes and activities.
- *additional_properties*: Other attributes possibly associated with the entity, like time, space, emotion, motivation, etc.

Extracting Low-Order Correlations

Formulation: $\mathcal{P}_{\text{ext_low}}(D_i, \mathcal{K}_v)$
 D_i denotes the text chunk, \mathcal{K}_v denotes the extracted entities.

Prompt: *From the entities identified in $\{\mathcal{K}_v\}$, identify all pairs of (source_entity, target_entity) that are *clearly related* to each other.*

For each pair of related entities, extract the following information:

- *entities_pair*: The name of source entity and target entity, as identified in $\{\mathcal{K}_v\}$.
- *low_order_relationship_description*: Explanation as to why you think the source entity and the target entity are related to each other.
- *low_order_relationship_keywords*: Keywords that summarize the overarching nature of the relationship, focusing on concepts or themes rather than specific details.
- *low_order_relationship_strength*: A numerical score indicating the strength of the relationship between the entities.)

Extracting High-Order Correlations

Formulation: $\mathcal{P}_{\text{ext_high}}(D_i, \mathcal{K}_v)$
 D_i denotes the text chunk, \mathcal{K}_v denotes the extracted entities.

Prompt: *Extract high-level keywords that summarize the main idea, major concept, or themes of the important passage.*
(Note: The content of high-level keywords should capture the overarching ideas present in the document, avoiding vague or empty terms).

For the entities identified in \mathcal{K}_v , based on the entity pair relationships and the high-level keywords, find connections or commonalities among multiple entities and construct high-order associated entity set as much as possible.

(Note: Avoid forcibly merging everything into a single association. If high-level keywords are not strongly associated, construct separate association).

Extract the following information from all related entities, entity pairs, and high-level keywords:

- *entities_set:* The collection of names for elements in high-order associated entity set, as identified in \mathcal{K}_v .
- *high_order_relationship_description:* Use the relationships among the entities in the set to create a detailed, smooth, and comprehensive description that covers all entities in the set, without leaving out any relevant information.
- *high_order_relationship_generalization:* Summarize the content of the entity set as concisely as possible.
- *high_order_relationship_keywords:* Keywords that summarize the overarching nature of the high-order association, focusing on concepts or themes rather than specific details.
- *high_order_relationship_strength:* A numerical score indicating the strength of the association among the entities in the set.

Extracting Keys from User Query

Formulation: $\mathcal{P}_{\text{ext_key}}(q)$
 q denotes user input.

Prompt: *You are a helpful assistant tasked with identifying both high-level and low-level keywords in the user's query.*
—Goal—

Given the query, list both high-level and low-level keywords. High-level keywords focus on overarching concepts or themes, while low-level keywords focus on specific entities, details, or concrete terms.

6.2. Evaluation

Evaluation of Scoring-Based Assessment

Formulation: $\mathcal{P}_{\text{eval_scoring}}(q, R, T_o)$
 q denotes user input, R denotes LLM response, T_o denotes the original text chunk that generated the question.

Prompt: *You are an expert tasked with evaluating answers to the questions by using the relevant documents based on five criteria: Comprehensiveness, Diversity, Empowerment, Logical, and Readability.*

—Goal—
You will evaluate the answers to the questions by using the relevant documents based on five criteria: Comprehensiveness, Diversity, Empowerment, Logical, and Readability.

-Comprehensiveness-
Measure whether the answer comprehensively covers all key aspects of the question and whether there are omissions.

Level	score range	description
Level 1	0-20	The answer is extremely one-sided, leaving out key parts or important aspects of the question.
Level 2	20-40	The answer has some content, but it misses many important aspects of the question and is not comprehensive enough.
Level 3	40-60	The answer is more comprehensive, covering the main aspects of the question, but there are still some omissions.
Level 4	60-80	The answer is comprehensive, covering most aspects of the question, with few omissions.
Level 5	80-100	The answer is extremely comprehensive, covering all aspects of the question with no omissions, enabling the reader to gain a complete understanding.

....
For each indicator, please give the problem a corresponding Level based on the description of the indicator, and then give a score according to the score range of the level.

Here are the question: q
Here are the relevant document: T_o
Here are the answer: R

Evaluate all the answers using the five criteria listed above, for each criterion, provide a summary description, give a Level based on the description of the indicator, and then give a score based on the score range of the level.

Evaluation of Selection-Based Assessment

Formulation: $\mathcal{P}_{\text{eval_scoring}}(q, R_a, R_b)$
 q denotes user input, R_a denotes the response from one LLMs, R_b denotes the response from another LLMs.

Prompt: *You will evaluate two answers to the same question based on eight criteria: Comprehensiveness, Empowerment, Accuracy, Relevance, Coherence, Clarity, Logical, and Flexibility.*

—Goal—
You will evaluate two answers to the same question by using the relevant documents based on eight criteria: Comprehensiveness, Empowerment, Accuracy, Relevance, Coherence, Clarity, Logical, and Flexibility.

-Comprehensiveness: *How much detail does the answer provide to cover all aspects and details of the question?*
-Empowerment: *How well does the answer help the reader understand and make informed judgments about the topic?*
...,
-Flexibility: *How well does the system handle various question formats, tones, and levels of complexity?*

For each criterion, choose the better answer (either Answer 1 or Answer 2) and explain why. Then, select an overall winner based on these ten categories.

*Here are the question: q
Here are the two answers:
Answer 1: R_a ;
Answer 2: R_b*

Evaluate both answers using the eight criteria listed above and provide detailed explanations for each criterion.

This section is not mandatory, but can be added to the manuscript if the discussion is unusually long or complex.

References

1. Razafinirina, M. A., Dimbisoa, W. G. & Mahatody, T. Pedagogical alignment of large language models (llm) for personalized learning: a survey, trends and challenges. *Journal of Intelligent Learning Systems and Applications* **16**, 448–480 (2024).
2. Naseer, F., Khan, M. N., Tahir, M., Addas, A. & Aejaz, S. H. Integrating deep learning techniques for personalized learning pathways in higher education. *Heliyon* **10** (2024).
3. Dagdelen, J. *et al.* Structured information extraction from scientific text with large language models. *Nature Communications* **15**, 1418 (2024).
4. Prince, M. H. *et al.* Opportunities for retrieval and tool augmented large language models in scientific facilities. *npj Computational Materials* **10**, 251 (2024).
5. Zhao, W. X., Liu, J., Ren, R. & Wen, J.-R. Dense text retrieval based on pretrained language models: A survey. *ACM Transactions on Information Systems* **42**, 1–60 (2024).
6. Cao, L. Ai in finance: challenges, techniques, and opportunities. *ACM Computing Surveys (CSUR)* **55**, 1–38 (2022).
7. Nahar, J., Hossain, M. S., Rahman, M. M. & Hossain, M. A. Advanced predictive analytics for comprehensive risk assessment in financial markets: Strategic applications and sector-wide implications. *Global Mainstream Journal of Business, Economics, Development & Project Management* **3**, 39–53 (2024).

8. Ullah, E., Parwani, A., Baig, M. M. & Singh, R. Challenges and barriers of using large language models (llm) such as chatgpt for diagnostic medicine with a focus on digital pathology—a recent scoping review. *Diagnostic pathology* **19**, 43 (2024).
9. Savage, T., Nayak, A., Gallo, R., Rangan, E. & Chen, J. H. Diagnostic reasoning prompts reveal the potential for large language model interpretability in medicine. *NPJ Digital Medicine* **7**, 20 (2024).
10. Lukkien, D. R. *et al.* Toward responsible artificial intelligence in long-term care: a scoping review on practical approaches. *The Gerontologist* **63**, 155–168 (2023).
11. Quinn, T. P., Senadeera, M., Jacobs, S., Coghlan, S. & Le, V. Trust and medical ai: the challenges we face and the expertise needed to overcome them. *Journal of the American Medical Informatics Association* **28**, 890–894 (2021).
12. Hussain, S. *et al.* Modern diagnostic imaging technique applications and risk factors in the medical field: a review. *BioMed research international* **2022**, 5164970 (2022).
13. Hager, P. *et al.* Evaluation and mitigation of the limitations of large language models in clinical decision-making. *Nature medicine* **30**, 2613–2622 (2024).
14. Huang, L. *et al.* A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems* **43**, 1–55 (2025).
15. Buciluă, C., Caruana, R. & Niculescu-Mizil, A. Editor, D. (ed.) *Model compression*. (ed.Editor, D.) *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, 535–541 (2006).
16. Jones, N. Ai hallucinations can't be stopped—but these techniques can limit their damage. *Nature* **637**, 778–780 (2025).
17. Ke, Z., Liu, B., Ma, N., Xu, H. & Shu, L. Achieving forgetting prevention and knowledge transfer in continual learning. *Advances in Neural Information Processing Systems* **34**, 22443–22456 (2021).
18. Es, S., James, J., Anke, L. E. & Schockaert, S. Editor, D. (ed.) *Ragas: Automated evaluation of retrieval augmented generation*. (ed.Editor, D.) *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, 150–158 (2024).
19. Xiong, G., Jin, Q., Lu, Z. & Zhang, A. Editor, D. (ed.) *Benchmarking retrieval-augmented generation for medicine*. (ed.Editor, D.) *Findings of the Association for Computational Linguistics ACL 2024*, 6233–6251 (2024).
20. Gao, Y. *et al.* Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997* **2** (2023).
21. Miao, J., Thongprayoon, C., Suppadungsook, S., Garcia Valencia, O. A. & Cheungpasitporn, W. Integrating retrieval-augmented generation with large language models in nephrology: advancing practical applications. *Medicina* **60**, 445 (2024).
22. Edge, D. *et al.* From local to global: A graph rag approach to query-focused summarization. *arXiv preprint arXiv:2404.16130* (2024).
23. Guo, Z., Xia, L., Yu, Y., Ao, T. & Huang, C. Lightrag: Simple and fast retrieval-augmented generation. *arXiv preprint arXiv:2410.05779* (2024).
24. Srinivasan, G. *et al.* Quantifying topological uncertainty in fractured systems using graph theory and machine learning. *Scientific reports* **8**, 11665 (2018).
25. Santos, A. *et al.* A knowledge graph to interpret clinical proteomics data. *Nature biotechnology* **40**, 692–702 (2022).
26. Pais, C. *et al.* Large language models for preventing medication direction errors in online pharmacies. *Nature medicine* **30**, 1574–1582 (2024).
27. Li, T. *et al.* Cancergpt for few shot drug pair synergy prediction using large pretrained language models. *NPJ Digital Medicine* **7**, 40 (2024).
28. Singhal, K. *et al.* Large language models encode clinical knowledge. *Nature* **620**, 172–180 (2023).
29. Labatut, V. & Bost, X. Extraction and analysis of fictional character networks: A survey. *ACM Computing Surveys (CSUR)* **52**, 1–40 (2019).
30. Gao, Y., Feng, Y., Ji, S. & Ji, R. HGNN⁺: General hypergraph neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **45**, 3181–3199 (2022).
31. Achiam, J. *et al.* Gpt-4 technical report. *arXiv preprint arXiv:2303.08774* (2023).
32. Bai, J. *et al.* Qwen technical report. *arXiv preprint arXiv:2309.16609* (2023).
33. Touvron, H. *et al.* Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971* (2023).
34. Liu, A. *et al.* DeepSeek-v3 technical report. *arXiv preprint arXiv:2412.19437* (2024).

35. Gong, L. *et al.* Seedream 2.0: A native chinese-english bilingual image generation foundation model. *arXiv preprint arXiv:2503.07703* (2025).
36. Du, Z. *et al.* Gln: General language model pretraining with autoregressive blank infilling. *arXiv preprint arXiv:2103.10360* (2021).
37. Yu, Y. *et al.* Rankrag: Unifying context ranking with retrieval-augmented generation in llms. *Advances in Neural Information Processing Systems* **37**, 121156–121184 (2024).
38. Yu, H. *et al.* Editor, D. (ed.) *Evaluation of retrieval-augmented generation: A survey.* (ed.Editor, D.) *CCF Conference on Big Data*, 102–120 (2024).
39. Wang, M. *et al.* Editor, D. (ed.) *Leave no document behind: Benchmarking long-context llms with extended multi-doc qa.* (ed.Editor, D.) *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 5627–5646 (2024).
40. Zhu, X., Guo, X., Cao, S., Li, S. & Gong, J. Editor, D. (ed.) *Structugraphrag: Structured document-informed knowledge graphs for retrieval-augmented generation.* (ed.Editor, D.) *Proceedings of the AAAI Symposium Series*, Vol. 4, 242–251 (2024).
41. Jiang, X. *et al.* Ragraph: A general retrieval-augmented graph learning framework. *Advances in Neural Information Processing Systems* **37**, 29948–29985 (2024).
42. He, X. *et al.* G-retriever: Retrieval-augmented generation for textual graph understanding and question answering. *Advances in Neural Information Processing Systems* **37**, 132876–132907 (2024).
43. Qian, H., Zhang, P., Liu, Z., Mao, K. & Dou, Z. Memorag: Moving towards next-gen rag via memory-inspired knowledge discovery. *arXiv preprint arXiv:2409.05591* (2024).

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.