RDMSFR-RAG: Reasoning RAG based on Relation Disambiguation and Multi-Scale Fusion for complex multi-hop QA

1st Haihui Ren
College of Intelligence Science and Technology, National University of Defense Technology
Changsha, China
renhaihui23@nudt.edu.cn

2nd Sihang Zhou
College of Intelligence Science and Technology, National University of Defense Technology
National Key Laboratory of Equipment State Sensing and Smart Support.National University of Defense Technology
Changsha, China
shzhou@nudt.edu.cn

3rd Dong Wang
College of Intelligence Science and Technology, National University of Defense Technology
Changsha, China
hntd_wang@163.com

4th Duanyang Yuan
College of Intelligence Science and Technology, National University of Defense Technology
Changsha, China
ydy_n1@nudt.edu.cn

5th Zexin Li
College of Intelligence Science and Technology, National University of Defense Technology
Changsha, China
lizexin@nudt.edu.cn

6th Jianxing Gong*
College of Intelligence Science and Technology, National University of Defense Technology
National Key Laboratory of Equipment State Sensing and Smart Support.National University of Defense Technology
Changsha, China
fj_gjx@nudt.edu.cn

***Abstract*—Recent advancements in knowledge graph enhanced RAG methods have shown promise in delivering efficient and accurate responses to complex multi-hop QA tasks by leveraging both graph-based and text-based knowledge. However, existing methods suffer from the problem of insufficient extraction, utilization, and fusion of graph information. To address these limitations, this paper presents RDMSFR-RAG, a novel methodology that enhances RAG's performance in multi-hop QA task through three key improvements in indexing, retrieval, and generation in RAG pipeline. Specifically, the triplet extraction method based on relation disambiguation accurately captures entity-relation triplets, yielding precise and concise graph data; The multi-scale score fusion approach effectively combines retrieval results from graph and text databases; The CoT reasoning based on classification method utilizes the idea chain of thought to guide targeted reasoning in LLMs for different multi-hop question types. Extensive experiments demonstrate that RDMSFR-RAG significantly outperforms the HippoRAG baseline, achieving 6-9 percentage point improvements in both retrieval and QA accuracy. Comprehensive experiments and ablation studies demonstrate that each of the three proposed enhancements significantly contributes to the improvement of retrieval and generation within the graph-based RAG method for complex multi-hop QA tasks.**

Keywords—Multi-hop QA, Relation Disambiguation, Multi-scale Fusion, RAG, CoT

I. INTRODUCTION

In order to better assist users in obtaining valuable information from massive amounts of data, multi-source information can be rapidly collected and presented via question answering (QA) systems. Traditional QA models often struggle with complex questions in practical application scenarios. Therefore, the multi-hop question answering task is proposed, which puts higher demands on the multi-hop inference and information integration capabilities of the question answering system. Conventional multi-hop QA models exhibit limited generalization, performing well only on specific tasks. The advent of large language models (LLMs) has significantly improved QA performance through their strong generative and generalization capabilities. However, due to the limitations of their pre-training, LLMs lack domain specific expertise, and retraining is computationally expensive. To assist LLMs in QA tasks, existing research has introduced retrieval-augmented generation (RAG). RAG retrieves knowledge needed for QA and addresses the hallucination problem caused by LLMs' lack of relevant information [1]. The RAG method firstly obtaining accurate and complete knowledge fragments related to the question through retrieval, and then integrating knowledge with LLMs to generate answers. This approach enhances the accuracy and reliability of QA while preserving the strong generative capabilities of LLMs. This retrieval-generation

paradigm effectively addresses the performance limitations of traditional QA methods in handling complex QA tasks.

Traditional RAG method exhibits limited performance in handling global query tasks and summarizing questions by sequentially partitioning and vectorizing text. Those approaches are inadequate for effectively retrieving relevant information from multiple documents. The RAG method based on knowledge graph (KG) constructs high quality entity-relation knowledge graphs for question answering , thereby enhancing the comprehensiveness and logicality of responses. However, the accuracy of question answering relies heavily on the quality of the graph, and the construction of high quality KG comes with high costs. Moreover, an exclusive reliance on knowledge graphs may result in the loss of semantic information contained within the text. Existing research has sought to integrate both approaches, utilizing the semantic information from text to enhance knowledge graph-based RAG methods [2-5]. The hybrid RAG method, which amalgamates text and graph databases, shows advantages in multi-hop QA task. The extensive corpus information provided by a text database can supplement the search content, while the high-quality entities and relations offered by knowledge graphs facilitate large language models (LLMs) in capturing interactions across documents and establishing efficient inference paths. Consequently, these two databases significantly enhance the accuracy and depth of information retrieval in multi-hop QA tasks. Therefore, this RAG method effectively provides both structured and unstructured information, leading to improved performance in complex question answering scenarios. Nevertheless, there remains substantial potential for further enhancing the high-quality construction of knowledge graphs through LLMs, the integration of textual information with knowledge graphs, and the development of chain of thought for multi-hop question answering. This article introduces an enhanced methodology for automating knowledge graph construction, a refined strategy for seamlessly integrating textual information with knowledge graphs, and a more rational approach to constructing reasoning chains for multi-hop question answering. These advancements, developed within a RAG framework that combines text and graph databases, effectively boost the accuracy and comprehensiveness of answers in multi-hop QA scenarios.

Specifically, to effectively extract graph data information and integrate it with text information, thereby enhancing retrieval accuracy and question answering accuracy, this paper proposes an enhanced RAG pipeline incorporating three novel methods: a relation disambiguation-based triplet extraction method in the indexing module, a multi-scale score fusion method in the retrieval module, and a classification-based complex multi-hop reasoning method in the generation module. The main contributions of this article are as follows:

- We propose a high-quality triplet extraction method that employs relation disambiguation to significantly enhance the accuracy and consistency of graph database construction.
- We devise an efficient approach for integrating textual information with knowledge graphs, which effectively improves the accuracy of retrieval through a carefully designed multi-scale retrieval score fusion mechanism.
- We establish an effective multi-hop question answering chain of thought (CoT), leveraging the reasoning capabilities of LLMs to decompose complex problems across various scales, effectively enhancing the model's answer generation ability and accuracy.

II. Related work

A. Graph Retrieval-Augmented Generation

With the widespread application of LLMs in complex reasoning tasks, RAG has become a research hotspot due to its ability to integrate external knowledge. Traditional RAG alleviates the illusion problem of LLMs by retrieving and querying relevant document fragments as contextual inputs. However, it still faces limitations in multi - hop reasoning tasks, such as fragmented retrieval and insufficient contextual relevance [2,6].

To address these issues, researchers have proposed a graph-based RAG framework. By combining structured knowledge representation and graph traversal mechanisms, this framework significantly enhances the coherence and accuracy of multi-hop inference. Graph-based RAG combines text retrieval with graph structure inference. HippoRAG [4], inspired by hippocampal indexing theory, constructs an Open KG and uses the PPR algorithm for multi - hop inference in a single search, achieving efficiency 6-13 times higher than traditional methods. It integrates cross - document knowledge through entity path propagation probability, showing 20%-30% higher accuracy than traditional RAG on MuSiQue and 2WikiMultiHopQA datasets. LightRAG [3] introduces a graph-enhanced text indexing method with a two-layer retrieval paradigm, outperforming GraphRAG on large - scale datasets and reducing index maintenance costs with its incremental update algorithm. These studies indicate that graph structures effectively capture long - range inference paths and reduce irrelevant content generation by LLMs [7-8].

In multi-hop QA scenarios, knowledge - graph - based RAG methods show significant advantages. KET-RAG [9] balances cost and quality via a multi - granularity indexing framework. It designs a knowledge graph skeleton, constructs a lightweight KG, and uses a text keyword bipartite graph to simulate KG retrieval. Experiments show 82.6% coverage on MuSiQue and a 34.6% improvement in generation quality over Hybrid RAG. GEAR [10], introducing SyncGE and Gist Memory mechanisms, guides subgraph extension through LLM agents, achieving 10% higher R@15 on HotpotQA than HippoRAG and reducing computational costs. KAG [11], with a logic-form-guided hybrid solver, improves F1 score by 19.6% in professional QA but relies on expert rules for graph construction. HippoRAG 2.0 [12] enhances context awareness by integrating dense - sparse encoding but requires additional maintenance of reset probabilities for phrase and document nodes. Future research needs to balance efficiency and accuracy and explore universal solutions for dynamic graph updates.

Despite these advancements, existing graph-enhanced RAG methods still face specific limitations. Methods like HippoRAG and LightRAG rely heavily on pre-existing KGs or text-derived

indices but lack sophisticated mechanisms for robust, real-time triple extraction directly from unstructured text within the RAG pipeline. This restricts their ability to dynamically integrate novel information or handle evolving domains effectively. Also multimodal fusion mechanisms integrating heterogeneous data sources into a unified, traversable graph structure for RAG are lacking. Our proposed method RDMSFR-RAG addresses these limitations by introducing a refined relation disambiguation mechanism and an enhanced multimodal fusion framework.

B. *Triple extraction in RAG indexing*

The RAG framework's performance is highly dependent on the indexing quality of the retrieval module. LLMs have introduced a new paradigm for triplet extraction in RAG indexing through knowledge representation and generation [13]. Traditional triplet extraction methods, which rely on supervised learning and manual annotation, struggle with generalization in low-resource scenarios. Recent research has shifted toward leveraging LLMs' pre-training capabilities to extract zero/few-shot triplets via prompt engineering or fine-tuning, enabling the construction of dynamically updated knowledge graphs [14-16].

In knowledge representation, COMET [17] pioneered a generative KG construction approach by fine-tuning GPT to generate high-quality commonsense triplets, achieving 77.5% accuracy on the ATOMIC dataset, with 59.25% of triplets being novel. Similarly, EDC [18] proposed an Extract-Define-Normalize framework that dynamically extends ontologies through a schema retriever, improving the F1 score by 12.5% over the baseline on the Wiki NRE dataset. These studies demonstrate that LLMs can overcome traditional IE limitations and directly generate structured knowledge [19]. SF-GPT [20] further introduced an untrained triplet extraction method, achieving an F1 score of 85.5% through a self-fusing subgraph strategy. Its entity alias generation (EAG) module effectively addressed semantic ambiguity issues.

In RAG index optimization, LLMs enhance knowledge atomization and retrieval relevance. PIKE-RAG [3] designed a knowledge-aware task decomposition algorithm that dynamically breaks down complex queries into atomic problems, improving accuracy by 15% in legal precise clause referencing tasks. This framework achieves multi-granularity retrieval via multi-layer heterogeneous graph indexing, with its iterative retrieval-generation mechanism boosting the HotpotQA EM index by 7.3%. RAMIE [21] constructed a four-layer medical diagnostic KG and integrated EHR records dynamically through an active questioning mechanism, reducing the misdiagnosis rate by 22% on the DDXPlus dataset. These applications highlight LLMs' dual role in domain knowledge representation: acting as generators to output triplets directly or as inference engines to optimize retrieval paths [22].

However, LLMs face a trade-off between noise and computational cost in triplet extraction. MQUAKE [23] noted that while existing editing methods accurately recall single-hop facts, their accuracy drops sharply to 7% in multi-hop problems, necessitating recalibration via external memory. KET-RAG [9] employs a core text block selection strategy to reduce indexing costs by an order of magnitude, though retrieval quality depends on the KNN graph's initialization accuracy. PURE [15] proposed a voting strategy for LLM re-evaluation of uncertain samples, achieving a 93.7% F1 score in materials science literature. THINK-ON-GRAPH [24] introduced topic and relation pruning algorithms to reduce redundant calculations, increasing three-hop inference speed by 40%.

III. METHOD

A. *Overall framework*

This article adopts a two-stage retrieval method similar to the HippoRAG method [4], which indexes the database in the offline stage and performs retrieval and generation in the online stage. By using this two-stage method to process the database, the efficiency of retrieval and question answering can be effectively improved. The overall framework of this article is shown in Fig. 1.

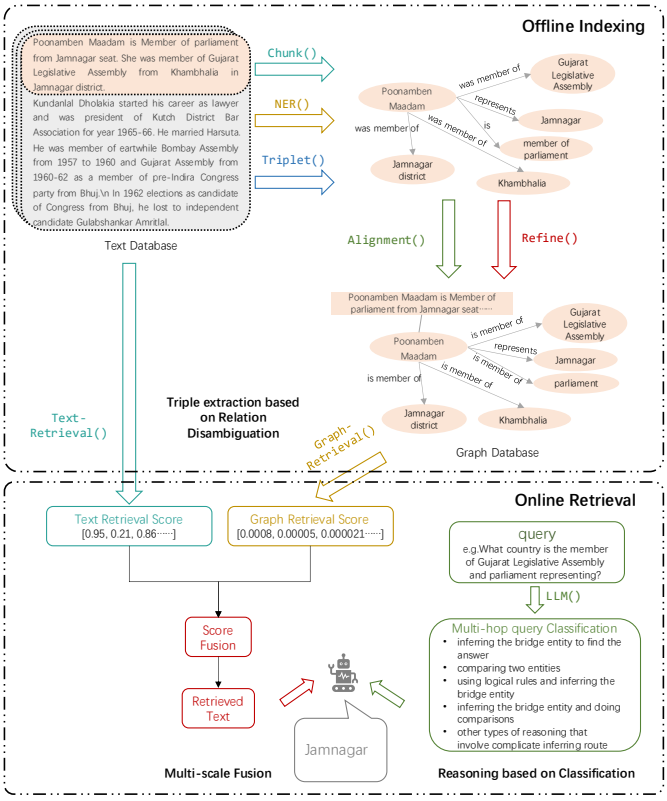

Fig. 1. RDMSFR-RAG framework

The offline indexing stage is shown in the upper part of Fig. 1, our method effectively achieves database simplification, removes redundant relations, and improves the efficiency of retrieval and question answering by designing a triplet extraction method based on relation disambiguation. Specifically, we propose a triplet extraction method based on relation disambiguation to construct a graph database. Based on the relation characteristics of the dataset itself, targeted disambiguation operations are performed on the relations, effectively reducing the number of relations and the sparsity of the graph database. By chunking the data in the text database, extracting entities, and extracting triples, preliminary triple extraction results are obtained. Subsequently, through alignment and refinement of the extracted results, the final triple results are derived.

In online retrieval stage, this article divides the online stage into a retrieval stage and a generation stage as shown in Fig. 1. In the retrieval stage, two databases are used for retrieval simultaneously. To better integrate the retrieval results from the two databases, this article proposes an effective score fusion method. This method comprehensively considers article rankings and retrieval scores from both the graph and text databases, based on their characteristics. In the generation stage, this article classified existing multi-hop questions and utilized the reasoning ability of LLMs' chain of thought, combined with the retrieved knowledge, to answer different characteristics of multi-hop question. Detailed elaboration of the three research points is as follows.

B. Triplet extraction based on relation disambiguation

During the indexing phase, this article adopts an offline approach to construct the index using a large amount of text data as input. To extract effective structured information and avoid the decrease in retrieval speed and accuracy caused by excessive redundant information, this article designs a triplet extraction method based on relation disambiguation on the basis of OpenIE. The implementation process is shown in Table Ⅰ.

TABLE I. TABLE TYPE STYLES TRIPLET EXTRACTION BASED ON RELATION DISAMBIGUATION ALGORITHM

Algorithm 1:Triplet extraction based on relation disambiguation

Input: All Text Data *Passages*

Output: Triple result *Triplet*

```
Chunk the article into sections to obtain small paragraphs containing multiple statements;
Paragraphs = chunk(Passages)
Definition_list = []
Triplet = []
for paragraph in Paragraphs:
  NER = LLM(paragraph, Prompt_ner)
  sparse_Triplets = LLM(paragraph, NER, Prompt_triplet)
  for sparse_Triplet in sparse_Triplets:
    relation = sparse_Triplet[1]
    Definition = LLM(relation, Prompt_defination)
    Scores=sim_cos(embedding(Definition), embedding(Definition_list))
    Obtain the top four scores and their corresponding relations;
    Top_scores = sorted(Scores)[:4]
    Top_indices = sorted(range(len(Scores)), key=lambda i: Scores[i])[:4]
    Top_definitions = [Definition_list[i] for i in Top_indices]
    Final_relation = LLM(Top_scores, Top_definitions, Prompt_relation)
    if Final_relation==None:
      Definition_list.append(Definition)
    else:
      relation = Final_relation
    Triplet.append(sparse_Triplet)
Return Triplet
```

Specifically, first of all, a large number of input text data contained in different articles are chunked. Here, we employ the LangChain framework to perform chunking based on semantic similarity; Then, OpenIE is utilized to extract entities from each text segment. The extracted entities and the text segments are then re-inputted into a large language model, where OpenIE is invoked once more to obtain preliminary triple extraction results. The entities in the triplet are used as nodes and the relations are used as edges to form the knowledge graph; Next, leveraging the semantic comprehension capabilities of the large language model, the relations within the triples are defined to generate relation-concept pairs in the form of "Relation: Definition." The definition expression is standardized through the specification of output formats and exemplification within the prompt. The similarity between different definitions is calculated, and the similarity scores are used to determine whether the relations extracted from different textual expressions share identical meanings. In order to calculate the similarity score and combine semantic information, we use the pre-training embedding model e5-mistral-7b-instruct to embed the relation-concept pairs and calculate the similarity score. After obtaining the similarity scores, we select the top four related relations with the highest scores as options A-D. Additionally, we include option E: None of the above. These options are then input into the large language model for the final determination of whether to replace the relation. For relation with the same meaning, disambiguation is performed and the relation will be replaced, else the relation and its concept will be added to a list with relation-concept pairs. By pre-constructing a graph database offline and performing disambiguation on the database, the size of the dataset can be effectively reduced, redundant information can be eliminated, and the efficiency and accuracy of retrieval and question-answering can be further enhanced.

C. Multi-scale Score Fusion

In the retrieval stage, this article retrieves content from both the text database and the graph database, obtaining two types of retrieval results and their corresponding retrieval scores. The existing score fusion methods, such as the reciprocal ranking fusion method [25], only consider the ranking information of the retrieval results and ignore the retrieval scores. In order to simultaneously utilize the semantic information contained in the text database, the relational information contained in the graph database, and the retrieval ranking information, this paper designs a multi-scale score fusion method that comprehensively considers the ranking information and retrieval scores. Specifically, the search result for the graph database is $Scores_{Graph} = [s_{g1}, s_{g2}, s_{g3}, \dots, s_{gm}]$, and the search result for the text database is $Scores_{Text} = [s_{t1}, s_{t2}, s_{t3}, \dots, s_{tm}]$. The sorting results are calculated using the reciprocal sorting fusion method, and the specific calculation formula is:

$$RRF(d \in D) = \frac{1}{k + Rank(d)} \tag{1}$$

Here, following the commonly used settings for RRF, we set the hyperparameter k to 60. Using the above formula, we obtain the rank of text database $RRF_{Text} = [r_{t1}, r_{t2}, r_{t3}, \dots, r_{tm}]$, and the rank of graph database $RRF_{Graph} = [r_{g1}, r_{g2}, r_{g3}, \dots, r_{gm}]$, The multi-scale fusion method proposed in this paper effectively fuses the above retrieval scores, and its specific calculation formula is:

$$Scores = norm\left(\sum_{i=1}^{n}\sum_{j=1}^{m} w_i \left(Scores_{ij} + \alpha RRF_{ij}\right)\right) \tag{2}$$

n represents the number of retrieval scores, m represents the number of all retrieved articles; $Scores_{ij}$ and RRF_{ij} represent retrieval scores and ranking scores, respectively; w_i indicating the weight hyperparameters of different databases, and it can be

adjusted according to the importance of different databases; α represents the weight hyperparameters of retrieval scores and ranking scores, which can be adjusted according to different task requirements. At the end of score fusion, Min-Max normalization is added to ensure that the range of the final retrieval score is between [0,1], making the final score smooth and continuous, and meeting the subsequent calculation requirements. This article combines the search results and ranking scores of multiple databases, effectively integrating the search results of multiple scales and databases, fully utilizing score information and ranking information, and effectively improving the accuracy of retrieval. In addition, for different databases with different ranges of search scores, adjusting the hyperparameter w_i can better balance the impact of the search scores.

D. CoT Reasoning based on classification

Due to the characteristics of dependency reasoning inherent in complex multi-hop problems, such as the need to consult multiple reference documents, the requirement for lengthy inference chains, and the involvement of intricate logical relations, this paper advances a classification - based reasoning approach tailored for complex multi-hop problems. The primary objective of this method is to precisely address questions that are contingent upon diverse logical relations. To elaborate, this study meticulously investigates the construction methodologies and classification schemes of existing multi-hop datasets [26 - 28]. By analyzing the reasoning patterns of multi-hop questions, the existing problems are systematically categorized into five distinct types: (1) inferring answers via intermediate entities, (2) inferring answers by comparing two entities, (3) inferring answers as intermediate entities through logical rules, (4) inferring answers by combining intermediate entity inference and comparison, and (5) other more complex reasoning paths. Table Ⅱ provides examples for each of these five categories.

To address the different multi-hop problems mentioned above, this article designs a classification prompt. Based on the structural and semantic characteristics of various multi-hop problems, a large language model is employed to classify the issues into the five categories outlined above. This classification approach allows for a thorough understanding and utilization of the semantic information within the text. By preprocessing complex multi-hop problems with this method, it becomes possible to adopt different answer generation strategies tailored to the distinct logical reasoning methods during the reasoning process.

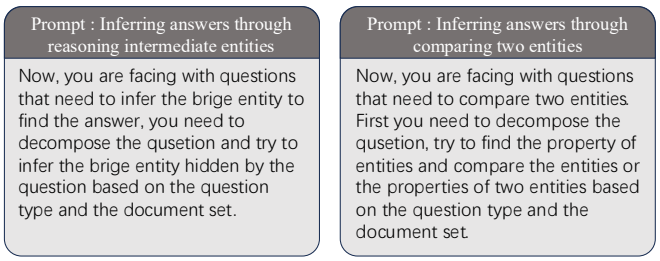

Fig. 2. Comparison of prompt words of different categories

Therefore, this paper proposes five distinct answer generation prompts, each corresponding to one of the five categories of questions. These prompts facilitate the LLM in employing a chain-of-thought approach to reason and answer complex multi-hop questions, thereby improving the accuracy of the answers. Examples of the prompts for "inferring answers through intermediate entities" and "inferring answers through comparing two entities" are illustrated in Fig. 2.

TABLE II. EXPLANATION OF MULTI-HOP QUSETIONS

Question type	Question example
Inferring answers through reasoning intermediate entities	Why did the composer of song Waspman die? When was the last time the team Matthew Webb was a member of beat the winner of the 1894-95 FA Cup? Who was honored with the award Dhondo Keshav Karve received prior to becoming president of India?
Inferring answers through comparing two entities	Who lived longer, Theodore Mcmillian or Hillel Slovak? Are Ural Federal University and California State Polytechnic University, Pomona both public universities? Did LostAlone and Guster have the same number of members?
Inferring answers as intermediate entities through logical rules	Who is Margaret Of Baux's father-in-law? What team is the highest goal scorer in the EPL a member of? The Unwinding author volunteered for which organisation? What relationship does Fred Gehrke have to the 23rd overall pick in the 2010 Major League Baseball Draft?
Inferring answers through inferring intermediate entities and comparing them	Are both directors of films Target Zero and Midnight Court (Film) from the same country? Which film has the director born later, Arr\u00eate Ton Cin\u00e9ma or Agni (2004 Film)? Which film has the director who died later, Love, Honor And Oh-Baby! or I Cover The Underworld?
Other more complex reasoning paths	Who was the football manager that played in the Football League Cup in 1985 and managed to lead the Birmingham City Football Club's 103rd season to finish in the 18th position? How were the same people who the Somali Muslim Ajuran Empire declared independence from expelled from the natural boundary between Thailand and the country Setthathirath is a citizen of? When did the explorer reach the city where the headquarters of the only group larger than Vilaiyaadu Mankatha's record label is located?

IV. EXPERIMEMTS

A. Experimental Settings

Datasets. This article conducted experiments on three widely used multi hop question answering datasets, HotpotQA [26], 2WikiMultiHopQA [27], and MuSiQue [28]. In order to improve validation efficiency, we extracted 1000 questions from each validation set and used paragraphs related to the selected questions as the retrieval corpus.

Metrics. This article uses recall @ 2, recall @ 5, and recall @ 10 as evaluation metrics for retrieval performance, representing the proportion of correctly retrieved results to all results (R @ 2, R @ 5, and R @ 10 will be used later in the article, respectively). Exact Match (EM) and F1 score are used as evaluation metrics for question answering.

Benchmark. This article compares widely used and effective single retrieval methods, including BM25 [29],

Contractor [30], GTR [31], and ColBERTv2 [32]; Two benchmark models, Propositioner [33] and RAPTOR [34], based on large language model enhancement, NativeRAG, which directly uses the Contractor [30] retriever to retrieve documents, and HippoRAG [4], which performs single retrieval based on graphs and texts, serve as benchmark models.

Implementation details. The primary LLM utilized in this study's experiments is Llama3-8B. During the retrieval phase, the Competitor model serves as the main retrieval method. In the QA stage, we select the top 10 retrieved documents to use as reference materials for answering questions. The experiments are conducted on a device equipped with an Intel (R) Core (TM) i9-14900K processor, 125GB of RAM, and two NVIDIA RTX 4090 GPUs (24GB each). Hyperparameter w_{Graph} and w_{Text} are all set at 0.5 because we think the text database and graph database equally important. And we set α=1 for our QA task.

B. Overall Results

The main experimental search results of this article are presented in Table Ⅲ. For the HotpotQA dataset, the RDMSFR-RAG model achieves the best performance across three metrics: R@2, R@5, and R@10. Compared to HippoRAG, it improves by 7.10 on R@2, 9.70 on R@5, and 9.84 on R@10. This further demonstrates that RDMSFR-RAG has significant advantages in retrieval accuracy and recall ability in this dataset. However, the traditional BM25 method underperforms, especially under stricter retrieval conditions. On the 2WikiMultiHopQA dataset, RDMSFR-RAG is comparable to HippoRAG in R@5 and R@10, outperforming other models like BM25 and Competitor. For the MuSiQue dataset, despite the decreased retrieval performance across all models due to the dataset's complexity, RDMSFR-RAG still yields the highest retrieval results, indicating its superior ability to capture semantic information and locate relevant knowledge.

The main experimental QA results are displayed in Table Ⅳ. Our method attains the best results across all datasets, showing its effectiveness in transforming retrieved knowledge into high-quality answers. On the HotpotQA dataset, RDMSFR-RAG outperforms HippoRAG by 6.10 in EM and 8.23 in F1. For the 2WikiMultiHopQA dataset, it ranks first in both EM and F1 metrics. Although its retrieval results are not optimal on this dataset, it still achieves the best QA outcomes. On the MuSiQue dataset, despite the overall low EM and F1 scores due to the dataset's complexity, RDMSFR-RAG leads with an EM of 17.40 and an F1 of 26.12, indicating its better problem - understanding and answer - generation capabilities compared to other models.

Based on the above analysis, our proposed RDMSFR-RAG method consistently delivers strong retrieval and QA performance across different datasets, with an index improvement of around 10, effectively demonstrating its effectiveness and robustness.

TABLE III. RETRIEVAL RESULTS OF MAIN EXPERIMENT

	HotpotQA			2WikiMultiHopQA			MuSiQue		
	R@2	R@5	R@10	R@2	R@5	R@10	R@2	R@5	R@10
BM25	55.47	72.26	80.41	51.89	61.91	68.38	32.33	41.26	51.27
Contriever	57.10	75.45	83.60	46.57	57.53	63.48	34.82	46.62	55.12
GTR	59.35	73.30	80.10	60.22	67.93	71.10	37.36	49.08	57.02
ColBERTv2	64.70	79.30	85.68	59.23	68.29	70.36	37.91	49.28	57.64
RAPTOR	58.17	71.25	79.03	46.36	53.87	63.24	35.74	45.39	54.39
Proposition	58.70	71.05	78.25	56.40	63.10	65.18	37.63	49.25	56.92
NativeRAG	60.30	77.50	84.50	69.30	83.60	87.20	40.39	52.04	59.75
HippoRAG	57.60	74.30	81.35	**71.25**	86.45	**90.25**	39.61	50.33	57.11
RDMSFR-RAG	**64.70**	**84.03**	**91.19**	68.35	**87.55**	90.15	**42.97**	**58.03**	**67.42**

TABLE IV. QA RESULTS OF MAIN EXPERIMENT

	HotpotQA		2WikiMultiHopQA		MuSiQue	
	EM	F1	EM	F1	EM	F1
BM25	28.60	37.14	27.50	32.87	8.70	13.72
Contriever	30.80	41.63	25.90	31.50	11.90	19.40
GTR	26.00	35.59	29.40	34.96	13.00	20.46
ColBERTv2	33.50	45.69	30.70	35.17	12.60	18.75
RAPTOR	25.30	33.21	25.80	31.44	10.80	15.53
Proposition	25.70	35.05	30.20	35.88	10.60	17.18
NativeRAG	29.90	40.11	37.50	44.36	13.70	21.41
HippoRAG	29.30	39.32	34.50	41.66	10.30	18.12
RDMSFR-RAG	**35.40**	**47.55**	**42.90**	**50.64**	**17.40**	**26.12**

TABLE V. RETRIEVAL RESULTS OF DIFFERENT MODELS

	HotpotQA			2WikiMultiHopQA			MuSiQue		
	R@2	R@5	R@10	R@2	R@5	R@10	R@2	R@5	R@10
Llama3-8b	64.70	84.03	91.19	68.35	87.55	90.15	42.97	58.03	67.42
Deepseek-r1-8B	68.05	88.85	94.00	70.45	84.62	87.75	38.52	48.70	54.67
Qwen2.5-7B	69.50	88.50	93.50	69.77	85.38	88.68	45.17	60.72	70.80
Qwen2.5-14B	69.70	90.35	95.00	71.13	86.48	89.58	46.67	64.63	74.75

TABLE VI. QA RESULTS OF DIFFERENT MODELS

	HotpotQA		2WikiMultiHopQA		MuSiQue	
	EM	F1	EM	F1	EM	F1
Llama3-8B	35.40	47.55	42.90	50.64	17.40	26.12
Deepseek-r1-8B	9.30	14.52	14.30	26.53	5.30	10.08
Qwen2.5-7B	18.20	30.00	25.00	29.54	8.10	17.13
Qwen2.5-14B	39.30	51.23	46.20	55.49	20.10	28.96

In order to evaluate the robustness of our method on different models, this paper compared Deepseek-r1-8B, Qwen2.5-7B, Qwen2.5-14B , and the Llama3-8B model used in the baseline experiments The retrieval and QA results are presented in Tables Ⅴ and Ⅵ. With the expansion of the model scale (from Llama3-8B to Qwen2.5-14B), there is a clear upward trend in the performance metrics for both retrieval and QA tasks.

The Qwen2.5-14B model achieves the best performance on the HotpotQA, 2WikiMultiHopQA, and MuSiQue datasets, demonstrating its superiority in handling complex knowledge-intensive tasks. In contrast, Deepseek-r1-8B performs relatively poorly across all datasets, while Qwen2.5-7B shows intermediate performance between Llama3-8B and Qwen2.5-14B. This indicates that increasing model size generally enhances performance on knowledge-intensive tasks.

C. Ablation Studies

Ablation experiments were conducted on the three proposed methods: tuple extraction method based on relation disambiguation (T), multi-scale score fusion method (R), and complex multi hop problem inference method based on classification (P), and RDMSFR w/o X represents the experimental results without using the X method. The search results and QA results of the experiment are shown in Tables Ⅶ and Ⅷ. On three multi hop question answering datasets, HotpotQA, 2WikiMultiHopQA, and MuSiQue, the complete RDMSFR-RAG method integrating all three methods achieved the best performance in retrieval and question answering tasks, fully verifying the effectiveness of the method combination.

When any two methods are removed (such as RDMSFR w/o R/P), the system performance shows a significant decrease: on the R@5 index of HotpotQA, the complete model (83.25%) improves by 3.8 percentage points compared to RDMSFR w/o R/P (79.45%); The EM score of the complete model has increased by 4.2%. In addition, it is worth noting that when T and R modules are removed at the same time (RDMSFR w/o T/R), MuSiQue's R@10 index fluctuates abnormally (62.63% vs. 67.93% of the complete model), which indicates that the triple extraction and score fusion methods have strong coupling in complex reasoning scenarios, and removing any module alone will destroy the integrity of semantic association modeling.

In addition, between the 2WikiMultiHopQA and MuSiQue datasets with the largest difference in data distribution, the relative standard deviation of the R@10 index of the complete model is only 1.28% (89.55% vs. 67.93%), significantly lower than the benchmark model (standard deviation 3.75%). This indicates that the method proposed in this article can effectively alleviate the negative transfer problem caused by differences in knowledge base structures. By dynamically adjusting the triplet granularity, correlation weights, and prompt word generation strategy, cross domain stability is achieved, further revealing the effective generalization of the method proposed in this article.

TABLE VII. RETRIEVAL RESULTS OF ABLATION STUDIES

	HotpotQA			2WikiMultiHopQA			MuSiQue		
	R@2	R@5	R@10	R@2	R@5	R@10	R@2	R@5	R@10
RDMSFR w/o T/R/P	57.60	74.30	81.35	71.25	86.45	**90.25**	39.61	50.33	57.11
RDMSFR w/o R/P	59.50	77.65	85.60	69.93	86.10	89.10	39.22	50.79	57.40
RDMSFR w/o T/P	59.90	78.40	85.60	**71.43**	86.15	89.32	41.40	53.96	61.28
RDMSFR w/o T/R	57.50	75.90	81.30	68.13	85.92	90.08	40.49	53.28	62.63
RDMSFR w/o T	63.15	83.25	88.80	68.70	86.42	89.55	41.67	57.78	67.93
RDMSFR w/o R	60.75	79.45	87.35	68.50	87.52	**90.83**	42.25	55.29	63.44

	HotpotQA			2WikiMultiHopQA			MuSiQue		
	R@2	R@5	R@10	R@2	R@5	R@10	R@2	R@5	R@10
RDMSFR w/o P	61.10	79.85	88.20	70.20	86.10	89.10	40.82	54.14	61.51
RDMSFR-RAG	**64.70**	**84.03**	**91.19**	68.35	**87.55**	**90.15**	**42.97**	**58.03**	**67.42**

TABLE VIII. QA RESULTS OF ABLATION STUDIES

	HotpotQA		2WikiMultiHopQA		MuSiQue	
	EM	F1	EM	F1	EM	F1
RDMSFR w/o T/R/P	29.30	39.32	34.50	41.66	10.30	18.12
RDMSFR w/o R/P	30.10	40.98	37.00	45.63	13.90	20.85
RDMSFR w/o T/P	30.50	41.95	37.50	44.78	15.40	23.69
RDMSFR w/o T/R	29.30	40.43	35.40	41.67	14.40	22.12
RDMSFR w/o T	34.80	46.05	39.70	46.32	16.50	24.46
RDMSFR w/o R	36.70	46.85	37.50	45.59	15.60	23.63
RDMSFR w/o P	29.70	40.76	37.60	46.39	14.50	24.05
RDMSFR-RAG	**35.40**	**47.55**	**42.90**	**50.64**	**17.40**	**26.12**

This article also conducts a horizontal comparison to explore the effectiveness of the three methods:

The absence of the **T module** significantly impacts long - range reasoning. In the MuSiQue dataset (average question length of 8.2 words), the F1 score of RDMSFR w/o T/P decreases by 6.65% (from 67.93% to 61.28%) compared to the complete model. This highlights the crucial role of the relation disambiguation-based triplet extraction method in modeling multi-entity and multi-relation scenarios.

The **R module** primarily optimizes the credibility evaluation of the evidence chain. When the R module is removed (RDMSFR w/o R/P), the R@5 index of 2WikiMultiHopQA decreases by 1.35% (from 87.52% to 86.17%). This indicates that dynamic fractional fusion effectively suppresses noise fragment interference.

The **P module** significantly enhances knowledge adaptability. In the HotpotQA dataset, removing the P module (RDMSFR w/o T/R) causes a 5.8% drop in EM values due to the model's failure to correctly understand the problem. This further demonstrates that our method effectively enhances the model's ability to understand problems and utilize retrieval results by classifying and decomposing answers for different question types.

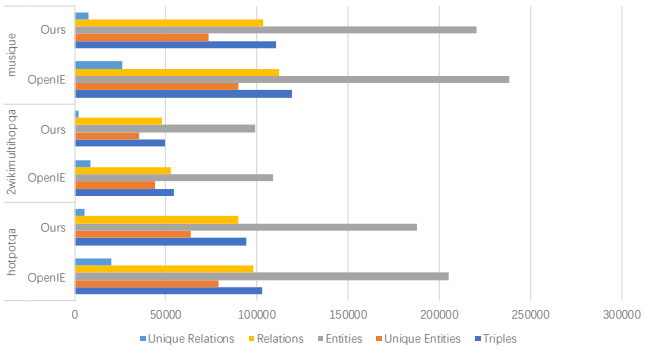

Fig. 3. Number of entities and relations extracted by our relation disambiguation method and OpenIE

To verify the effectiveness of the relation disambiguation based triplet extraction method, this paper conducted statistical and comparative analysis on the proposed method and OpenIE method on three different datasets, including the sorting of non-duplicated relations, the number of relations, the number of entities, the number of non-duplicated entities, and the number of triples extracted. The results are shown in Fig. 3. When the number of relations, entities, non-repeating entities, and triples are similar, the number of non-repeating relations has significantly decreased. This enhances graph consistency and node connectivity, improving the effectiveness of structured information and boosting retrieval and QA accuracy.

V. CONCLUSION

In order to further improve the accuracy of RAG in generating multi-hop QA results, this paper proposes the RDMSFR-RAG method, which is a reasoning RAG method based on relation disambiguation, multi-scale fusion and CoT for complex multi hop question answering. The proposed approach encompasses three key aspects: indexing, retrieval, and generation. Firstly, a relation disambiguation-based triplet extraction method is put forward to refine the extraction of knowledge triplets. Secondly, a multi-scale score fusion method is devised to integrate retrieval results from multiple sources effectively. Thirdly, a classification-based reasoning method for complex multi-hop questions is developed to improve the logical inference process. The experimental results far exceed the baseline method HippoRAG, across five evaluation metrics in both retrieval and generation stage. In addition, this paper conducted a LLM comparison experiments and ablation experiments to further demonstrate the effectiveness of the three improvement points proposed.

However, the study has limitations. Firstly, the relation-disambiguation-based triplet extraction method, while improving triplet enhancement accuracy, increases time consumption and demands more experimental resources. Secondly, although the method performs well on the complex MusiQue dataset, retrieval and QA results remain suboptimal, indicating room for further improvement.

REFERENCES

[1] Y. Gao, Y. Xiong, X. Gao, K. Jia, J. Pan, Y. Bi, et al, "Retrieval-augmented generation for large language models: A survey," arXiv preprint arXiv:2312.10997, 2, 1, 2023.

[2] J. Wang, J. Fu, R. Wang, L. Song, and J. Bian, "PIKE-RAG: sPecIalized KnowledgE and Rationale Augmented Generation," arXiv preprint arXiv:2501.11551, 2025.

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

[4] B. Jimenez Gutierrez, Y. Shu, Y. Gu, M. Yasunaga, and Y. Su, "HippoRAG: Neurobiologically Inspired Long-Term Memory for Large Language Models," Advances in Neural Information Processing Systems, 37, 59532-59569, 2024.

[5] D. Edge, H. Trinh, N. Cheng, J. Bradley, A. Chao, A. Mody, et al, "From local to global: A graph rag approach to query-focused summarization," arXiv preprint arXiv:2404.16130, 2024.

[6] J. Kim, and M. Min, "From rag to qa-rag: Integrating generative ai for pharmaceutical regulatory compliance process," arXiv preprint arXiv:2402.01717, 2024.

[7] X. Zhao, S. Liu, S. Y. Yang, and C. Miao, "MedRAG: Enhancing Retrieval-augmented Generation with Knowledge Graph-Elicited Reasoning for Healthcare Copilot," In Proceedings of the ACM on Web Conference 2025, pp. 4442-4457, April 2025.

[8] L. Luo, Y. F. Li, G. Haffari, and S. Pan, "Reasoning on graphs: Faithful and interpretable large language model reasoning," arXiv preprint arXiv:2310.01061, 2023.

[9] Y. Huang, S. Zhang, and X. Xiao, "KET-RAG: A Cost-Efficient Multi-Granular Indexing Framework for Graph-RAG," arXiv preprint arXiv:2502.09304, 2025.

[10] Z. Shen, C. Diao, P. Vougiouklis, P. Merita, S. Piramanayagam, D. Graux, and et al, "GeAR: Graph-enhanced Agent for Retrieval-augmented Generation," arXiv preprint arXiv:2412.18431, 2024.

[11] L. Liang, M. Sun, Z. Gui, Z. Zhu, Z. Jiang, L. Zhong, and et al, "Kag: Boosting llms in professional domains via knowledge augmented generation," arXiv preprint arXiv:2409.13731, 2024.

[12] B. J. Gutiérrez, Y. Shu, W. Qi, S. Zhou, and Y. Su, "From rag to memory: Non-parametric continual learning for large language models," arXiv preprint arXiv:2502.14802, 2025.

[13] X. Wu, and K. Tsioutsiouliklis, "Thinking with Knowledge Graphs: Enhancing LLM Reasoning Through Structured Data," arXiv preprint arXiv:2412.10654, 2024.

[14] Z. Tan, X. Zhao, W. Wang, and W. Xiao, "Jointly extracting multiple triplets with multilayer translation constraints," In Proceedings of the AAAI conference on artificial intelligence, Vol. 33, No. 01, pp. 7080-7087, July 2019.

[15] Y. Liu, Y. Ma, Y. Zhang, R. Yu, Z. Zhang, Y. Meng, and Z. Zhou, "Interactive optimization of relation extraction via knowledge graph representation learning," Journal of Visualization, 27(2), 197-213, 2024.

[16] Y. Yang, S. Chen, Y. Zhu, X. Liu, W. Ma, and L. Feng, "Intelligent extraction of reservoir dispatching information integrating large language model and structured prompts," Scientific Reports, 14(1), 14140, 2024.

[17] A. Bosselut, R H. ashkin, M. Sap, C. Malaviya, A. Celikyilmaz, and Y. Choi, "COMET: Commonsense transformers for automatic knowledge graph construction," arXiv preprint arXiv:1906.05317, 2019.

[18] B. Zhang, and H. Soh, "Extract, define, canonicalize: An llm-based framework for knowledge graph construction," arXiv preprint arXiv:2404.03868, 2024.

[19] T. Nayak, N. Majumder, P. Goyal, and S. Poria, "Deep neural approaches to relation triplets extraction: a comprehensive survey," Cognitive computation, 13(5), 1215-1232, 2021.

[20] L. Sun, P. Zhang, F. Gao, Y. An, Z. Li, and Y. Zhao, "SF-GPT: A training-free method to enhance capabilities for knowledge graph construction in LLMs," Neurocomputing, 613, 128726, 2025.

[21] Z. Zhan, S. Zhou, M. Li, and R. Zhang, "RAMIE: retrieval-augmented multi-task information extraction with large language models on dietary supplements," Journal of the American Medical Informatics Association, ocaf002, 2025.

[22] Q. Geng, J. You, H. Guo, X. Huang, J. Tao, and J. Yi, "Document-Level Iterative Entity and Relation Extraction for Materials Scientific Literature," In International Conference on Intelligent Computing (pp. 499-510). Singapore: Springer Nature Singapore, August 2024.

[23] Z. Zhong, Z. Wu, C. D. Manning, C. Potts, and D. Chen, "Mquake: Assessing knowledge editing in language models via multi-hop questions," arXiv preprint arXiv:2305.14795, 2023.

[24] S. Ma, C. Xu, X. Jiang, M. Li, H. Qu, and J. Guo, "Think-on-graph 2.0: Deep and interpretable large language model reasoning with knowledge graph-guided retrieval" arXiv e-prints, arXiv-2407, 2024.

[25] G. V. Cormack, C. L. Clarke, and S. Buettcher, "Reciprocal rank fusion outperforms condorcet and individual rank learning methods," In Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval, pp. 758-759, July 2009.

[26] Z. Yang, P. Qi, S. Zhang, Y. Bengio, W. W. Cohen, R. Salakhutdinov, and C. D. Manning, "HotpotQA: A dataset for diverse, explainable multi-hop question answering," arXiv preprint arXiv:1809.09600, 2018.

[27] X. Ho, A. K. D. Nguyen, S. Sugawara, and A. Aizawa, "Constructing a multi-hop qa dataset for comprehensive evaluation of reasoning steps," arXiv preprint arXiv:2011.01060, 2020.

[28] H. Trivedi, N. Balasubramanian, T. Khot, and A. Sabharwal, "♫ MuSiQue: Multihop Questions via Single-hop Question Composition," Transactions of the Association for Computational Linguistics, 10, 539-554, 2022.

[29] V. Karpukhin, B. Oguz, S. Min, P. S. Lewis, L. Wu, S. Edunov, and W. T. Yih, "Dense Passage Retrieval for Open-Domain Question Answering," In EMNLP (1), pp. 6769-6781, November 2020.

[30] G. Izacard, M. Caron, L. Hosseini, S. R iedel, P. Bojanowski, A. Joulin, and E. Grave, "Unsupervised dense information retrieval with contrastive learning," arXiv preprint arXiv:2112.09118, 2021.

[31] J. Ni, C. Qu, J. Lu, Z. Dai, G. H. Ábrego, J. Ma, and Y. Yang, "Large dual encoders are generalizable retrievers," arXiv preprint arXiv:2112.07899, 2021.

[32] K. Santhanam, O. Khattab, J. Saad-Falcon, C. Potts, and M. Zaharia, "Colbertv2: Effective and efficient retrieval via lightweight late interaction," arXiv preprint arXiv:2112.01488, 2021.

[33] T. Chen, H. Wang, S. Chen, W. Yu, K. Ma, X. Zhao, and D. Yu, "Dense x retrieval: What retrieval granularity should we use?," In Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pp. 15159-15177, November 2024.

[34] 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, May 2024.