Towards Building an RDF-based Deep Document Model and Retrieval Augmented Generation System for Enhanced Question Answering with Large Language Models

Runsong Jia[1], Bowen Zhang[2], Sergio J. Rodríguez-Méndez[1] and Pouya G. Omran[1]

[1]*Australian National University, Canberra ACT 2601, AU*

[2]*QDX Technologies Pte. Ltd., 33A Pagoda Street, Singapore 059192, SG*

Abstract

Knowledge Graphs (KGs) are crucial for Retrieval-Augmented Generation (RAG), but traditional methods have limitations in capturing details and querying academic KGs. The challenges lie in identifying the appropriate KG type for RAG, such as a Metadata KG, and optimizing the integration of Large Language Models (LLMs) with KGs to enhance retrieval and generation. This paper introduces a novel framework combining the Deep Document Model (DDM) concept and a KG-enhanced Query Processing (KGQP) mechanism. DDM provides a comprehensive, hierarchical representation of academic papers using advanced Natural Language Processing (NLP) techniques, while KGQP optimizes complex queries using the KG's structural information and semantic relationships. The framework also integrates KGs with state-of-the-art LLMs to improve knowledge utilization and downstream task performance. Evaluations show that the KG-based approach surpasses vector-based methods in relevance, accuracy, completeness, and readability. This research demonstrates the potential of combining KGs and LLMs for effective academic knowledge management and discovery. § *Submission type*: **Poster** §.

Keywords

Knowledge Graph, Deep Document Model, Large Language Model, Information Extraction, Knowledge Graph Construction

1. Introduction and Related Work

In the current era, where LLMs are extensively applied for complex question-answering tasks, these models serve as effective tools for understanding. However, relying solely on LLMs is not sufficient to meet the challenges, as they require appropriate methods to process and explore massive semi-structured data [1, 2]. KGs play a crucial role in this context, as they manage vast amounts of data and provide relevant contextual information to LLMs [3, 4]. Existing research has already utilized KGs to enhance the outputs of LLMs [5, 6, 7], highlighting the importance of integrating LLMs with KGs. Through KGs, relevant information can be effectively filtered and pinpointed, providing the precise context needed for LLMs to perform their tasks.

Posters, Demos, and Industry Tracks at ISWC 2024, November 13–15, 2024, Baltimore, USA

Runsong.Jia@anu.edu.au (R. Jia); Bowen.Zhang@qdx.co (B. Zhang); Sergio.RodriguezMendez@anu.edu.au (S. J. Rodríguez-Méndez); P.G.Omran@anu.edu.au (P. G. Omran)

0009-0003-3986-2679 (R. Jia); 0000-0001-6045-8599 (B. Zhang); 0000-0001-7203-8399 (S. J. Rodríguez-Méndez); 0000-0002-4473-3877 (P. G. Omran)

In this work, we propose utilizing the Metadata KG of a document set, such as ASKG [8], integrated with a RAG architecture and LLMs to provide relevant and diverse contexts from research papers for generating answers to complex queries. We utilize the following tools to construct the KG automatically. MEL (Metadata Extractor & Loader) and TNNT (The NLP-NER Toolkit) are powerful tools for extracting knowledge from unstructured sources[1]. MEL converts metadata and text into JSON objects [9], while TNNT enhances this data with Named Entity Recognition [10]. Additionally, the PARSE component [2] of the KGCP pipeline [3] employs web crawling and NLP models to enrich academic semantic knowledge bases in computer science [11]. The key contributions of this paper include: i. constructing KGs using DDM's fine-grained representation, ii. optimizing KG queries with KGQP for extensive knowledge graphs, and iii. enhancing query handling for complex scholarly domain inquiries.

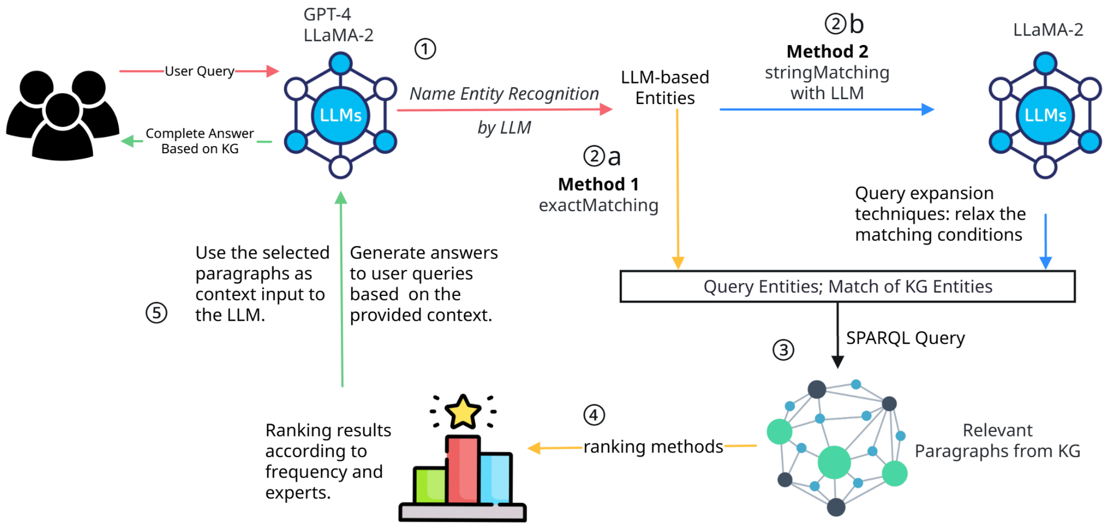
2. Methodology

Central to our approach is the DDM concept[4], which captures the logical hierarchical structure of documents from section, paragraph to sentence level. By employing NLP techniques, DDM conducts in-depth analysis of textual elements, identifies their hierarchical relationships, and creates a model that reflects the document's logical flow. Our process integrates PARSE and DDM methods to construct a KG from scientific papers (ASKG). This approach not only transcends the traditional role of knowledge graphs as mere fact providers but also offers comprehensive metadata and context for collections of research papers. DDM enhances the depth and richness of knowledge representation through its alignment with the Document Object Model Ontology (DOMO). In this way, the DDM's structure is materialized in an ontology-based metadata KG.

To address the challenge of "AI hallucination" faced by LLMs when dealing with complex questions, particularly those involving intricate fact verification, we have designed and implemented the KGQP (KG-enhanced Query Processing) workflow[5]. This process harnesses the potential advantages of LLMs and incorporates an academic KG (ASKG) aligned with a generic model (DOMO). As shown in Figure 1, this pipeline leverages KG-based context in LLM question-answering, utilizing structured information from ontology-based KGs like ASKG to provide accurate context to prompt engineering tasks with LLMs and reduce hallucinations. The workflow's main steps as shown in Figure 1 involves:

1. **Entity Identification:** We utilize GPT-4 for identifying entities in user queries. GPT-4 demonstrates exceptional performance in entity recognition and matching [12], accurately identifying entities and their corresponding quantities within user queries.
2. **Retrieval of Relevant Paragraphs:** We employ SPARQL queries with exact and string matching techniques:
    - **Exact Matching:** We construct SPARQL queries to retrieve paragraphs directly related to the identified entities in the query.

[1] https://w3id.org/kgcp/MEL-TNNT
[2] https://w3id.org/kgcp/PARSE
[3] https://w3id.org/kgcp/
[4] https://w3id.org/kgcp/DDM
[5] https://w3id.org/kgcp/KGQP



GPT-4
LLaMA-2
User Query
Complete Answer Based on KG
LLMs
①
Name Entity Recognition by LLM
LLM-based Entities
②b
Method 2
stringMatching with LLM
LLaMA-2
LLMs
②a
Method 1
exactMatching
Query expansion techniques: relax the matching conditions
Query Entities; Match of KG Entities
Use the selected paragraphs as context input to the LLM.
⑤
Generate answers to user queries based on the provided context.
SPARQL Query
③
④
ranking methods
Ranking results according to frequency and experts.
Relevant Paragraphs from KG

Figure 1: Flow chart of the KGQP mechanism: KG-LLM interaction process

- **String Matching:** When exact matching fails to retrieve sufficient relevant entities, we employ more flexible search techniques. These include case-insensitive searches and searches for any occurrences of the specified substring within the entity label, allowing for partial matches and retrieving more potentially relevant entities.

3. **Entity Matching:** We link academic entities with relevant paragraphs using GIST-Embedding. The semantic similarity between each entity and paragraph pair is calculated using cosine similarity, with a threshold set at 70%. For each entity, we identify the paragraph with the highest similarity score. Subsequently, we match the entities identified in the user's query with the corresponding entities in our KG.
4. **Selection of Most Relevant Paragraphs:** We employ Keyword Frequency Matching to select the most relevant paragraphs.
5. **Answer Generation:** We utilize LLaMA2 [13] to generate answers based solely on the selected paragraphs.

This workflow leverages the structured information in our KG to provide more accurate and relevant context for the LLM, thereby improving answer quality and reducing hallucinations.

3. Evaluation and Conclusions

We compare our KGQP method against a vector-based baseline. In the baseline experiment, a simple chunking approach was employed to preprocess the text data. The dataset used in this experiment consists of 10 scientific papers in the field of computer science. These papers were all published by the same two authors, who are also the experts involved in the subsequent human evaluation. The dataset was divided into chunks with a maximum of 100 tokens per chunk and a 5% overlap ratio between adjacent chunks. This ensures that the content length of

Table 1
Comparison of KG-based and Vector-based Systems on five questions

Evaluator	System	Relevance	Accuracy	Completeness	Readability
Evaluator 1	KG-based	3.4	**3.8**	**3.4**	4.0
	Vector-based	3.4	3.4	2.8	4.0
Evaluator 2	KG-based	3.6	**3.6**	**3.6**	4.2
	Vector-based	3.6	3.4	3.2	4.2
Evaluator 3	KG-based	**4.6**	**4.0**	**4.0**	**4.6**
	Vector-based	4.2	3.8	3.2	4.0
Average	KG-based	**3.9**	**3.8**	**3.7**	**4.3**
	Vector-based	3.7	3.5	3.1	4.1

each chunk obtained by "Simple Chunking" is comparable to the average length of Paragraphs[6] in KGQP process and provides richer contextual information. This chunking process yielded a total of 618 text chunks.

Our evaluation employed both human assessment and quantitative metrics. Three evaluators, including two human experts and Claude [14], scored answers from KG-based and vector-based (Simple Chunking) methods across four dimensions: Relevance, Accuracy, Completeness, and Readability, using a 5-point scale.

We also conducted entity extraction, calculating overlap and Jaccard distance between the two methods. Semantic distance was measured using GIST-Embedding [15]. Additionally, we analyzed context diversity by recording the number of article sources per answer and calculated similarity between answers and user query embeddings.

As shown in Table 1, the KG-based method consistently outperformed the vector-based approach across all evaluated dimensions. The most significant improvements were observed in Completeness and Accuracy, demonstrating the KG-based method's effectiveness in providing more comprehensive and precise answers. These results suggest that our proposed approach effectively enhances question-answering performance in academic contexts.

Our approach shows promise despite challenges posed by varied document structures. Future work will focus on better integrating DDM with the PARSE Pipeline for full automation and extending our framework to support Multimodal Knowledge Graphs and Question Answering for figures and tables.

In conclusion, our KG-based method demonstrates consistent superiority in producing relevant, accurate, and comprehensive answers while maintaining high readability. The observed improvements across all metrics underscore the potential of our approach in enhancing question-answering performance within academic contexts. As we address current limitations and expand the system's capabilities, we anticipate further advancements in academic knowledge management and discovery.

[6] These paragraph entities were extracted through the DDM pipeline.

References

[1] J. Wu, Q. Gao, Z. Liu, W. Wei, A Survey on Embedding Techniques for Scholarly Knowledge Graphs, in: International Conference on Web Information Systems Engineering, Springer, Singapore, 2020, pp. 150–160.

[2] X. Wang, C. Zhang, X. Li, M. Zhang, S. Ma, Review on Knowledge Graph Techniques for Retrieving Scientific Publications, Frontiers of Computer Science 14 (2020) 143301.

[3] B. Abu-Salih, Domain-specific knowledge graphs: A survey, Journal of Network and Computer Applications 185 (2021) 103076.

[4] A. B. Cano, C. Gómez-Rodríguez, A. O. Tijani, Combining Text Embeddings and Knowledge Graphs for Scholarly Document Classification, in: European Conference on Information Retrieval, Springer, Cham, 2021, pp. 249–263.

[5] D. Sanmartin, KG-RAG: Bridging the Gap Between Knowledge and Creativity, arXiv preprint arXiv:2405.12035 (2024). URL: https://doi.org/10.48550/arXiv.2405.12035.

[6] G. Agrawal, T. Kumarage, Z. Alghamdi, H. Liu, Can Knowledge Graphs Reduce Hallucinations in LLMs? : A Survey, NAACL (2024). URL: https://doi.org/10.48550/arXiv.2311.07914.

[7] S. Pan, L. Luo, Y. Wang, C. Chen, J. Wang, X. Wu, Unifying Large Language Models and Knowledge Graphs: A Roadmap (2023).

[8] R. Jia, B. Zhang, S. J. Rodríguez Méndez, P. G. Omran, Leveraging Large Language Models for Semantic Query Processing in a Scholarly Knowledge Graph, arXiv preprint arXiv:2405.15374 (2024). URL: https://doi.org/10.48550/arXiv.2405.15374.

[9] S. J. Rodríguez Méndez, P. G. Omran, A. Haller, K. Taylor, MEL: Metadata Extractor & Loader, in: ISWC (Posters/Demos/Industry), 2021.

[10] S. Seneviratne, S. J. Rodríguez Méndez, X. Zhang, P. G. Omran, K. Taylor, A. Haller, TNNT: The Named Entity Recognition Toolkit, in: Proceedings of the 11th on Knowledge Capture Conference, 2021, pp. 249–252.

[11] B. Zhang, S. J. Rodríguez Méndez, P. G. Omran, ASKG: An Approach to Enrich Scholarly Knowledge Graphs through Paper Decomposition with Deep Learning, in: ISWC 2023 Posters and Demos: 22nd International Semantic Web Conference, Athens, Greece, 2023.

[12] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, et al., GPT-4 technical report, arXiv preprint arXiv:2303.08774 (2023).

[13] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, G. Lample, LLaMA: Open and Efficient Foundation Language Models, 2023. arXiv:2302.13971.

[14] Anthropic, Claude: Introducing Claude, https://www.anthropic.com/index/introducing-claude, 2023.

[15] A. V. Solatorio, GISTEmbed: Guided In-sample Selection of Training Negatives for Text Embedding Fine-tuning, arXiv preprint arXiv:2402.16829 (2024). URL: https://arxiv.org/abs/2402.16829. arXiv:2402.16829.