A Unified Framework For LIm-Powered Knowledge Management Systems Using Rag And Knowledge Graphs

[1]Dr N. Jayashri, [2] Dr.Janani Selvam , [3]Dr.Divya Midhun Chakkaravarthy,

[1]PDF Scholar, Post Doctoral Fellowship (PDF) programme in Computer *Science Engineering, Lincoln University College, pdf.jayashri@lincoln.edu.my*
[2](corresponding author), Faculty of Engineering, Lincoln university college, janani@lincoln.edu.my
[3]Faculty of Engineering
Lincoln university college, divya@lincoln.edu.my

Abstract
The sheer proliferation of the unstructured organizational data, such as emails, documents, meeting notes, and chat logs, has put the usefulness of the traditional Knowledge Management Systems (KMS), which are mostly structured and inactive, into question. Recent developments in Large Language Models (LLMs) provide considerable possibilities in processing and extracting insights of such data, but their application in business contexts does not have a standardized framework that ensures accuracy of retrieval, contextual relevance, governance, and explainability. The present paper suggests a holistic design of an LLM-based Knowledge Management System that combines Retrieval-Augmented Generation (RAG) with Knowledge Graphs (KG) to facilitate hybrid knowledge retrieval and structured reasoning. The presented system is executed with the help of a cloud architecture written in Python which includes vector databases and graph databases to facilitate scalable and efficient access to knowledge. The framework leads to accuracy of answers, reduction in hallucination, and increases in the interpretability through integration of semantic retrieval and relational representation of knowledge. Accuracy, precision, latency, and explainability-based evaluation proves the usefulness of the proposed approach compared to traditional and standalone systems based on the LLM. This piece of work adds to the scalable and enterprise-ready next-generation knowledge management solution.

Keywords: Knowledge Management Systems, Large Language Models, Retrieval-Augmented Generation, Knowledge Graphs, Enterprise AI, Information Retrieval.

1. Introduction

The high rate of organizational process digitization has caused unstructured data such as emails, reports, meeting transcripts, and chat logs to increase at an unprecedented scale. The Early Knowledge Management Systems (KMS) were built to handle structured and semi-structured data and it heavily relied on the predefined schema and keyword-based data retrieval system. Consequently, such systems find it hard to capture, organize and retrieve valuable insights on unstructured sources of knowledge, resulting in inefficiencies in decision-making and knowledge reuse [1], [2].

Large Language Models (LLMs) are a groundbreaking technology in recent years, with the ability to process, generate, and reason natural language data. These models have shown outstanding performance in most tasks such as question answering, summarization and information extraction. Nonetheless, even with their capabilities, LLMs have inherent limitations due to their dependence on pre-trained knowledge, which can be outdated or incomplete, and the fact that they will produce hallucinated or unprovable answers [3], [4].

To overcome these shortcomings, the hybrid method of combining LLMs with external knowledge retrieval systems has been proposed as Retrieval-Augmented Generation (RAG). RAG makes the LLM outputs more factual and contextually relevant by basing responses off of the retrieved knowledge

sources or documents. This method has demonstrated high potential in the case of enterprise knowledge management, where it is important to have access to the latest and domain specific information [2], [5].

Along with these trends, Knowledge Graphs (KGs) have been on the spotlight as a format of structured knowledge representation in terms of entities and their relationship. KGs facilitate semantic reasoning, exploration of relationships, and enhanced interpretability of data. Knowledge Graphs can bring structured context when combined with LLMs to boost reasoning capabilities and explainable AI systems (especially in complex organizational settings) [1], [6].

More recent studies have been done on integrating LLMs, RAG and Knowledge Graphs into unified constructions, also known as GraphRAG architectures. These systems are based on semantic similarity search and graph reasoning in order to enhance information retrieval and generation of answers. Research shows that these hybrid systems are more accurate, have greater explainability, and understand context more well than standalone LLMs and classical retrieval systems [7], [8].

Regardless of these developments, there are still major issues when it comes to implementing LLM-based Knowledge Management Systems in the real-world enterprise environment. These are data governance, privacy, scalability, system integration, and inability to have standardized evaluation metrics. Moreover, most of the solutions available have been unintegrated, addressing single elements like retrieval or graph reasoning, but not offering an overall implementation of the solution on an enterprise level [9], [10].

The lack of an overarching architectural design that incorporates data ingestion, knowledge representation, retrieval mechanisms, and LLM-based reasoning into a unified system is another important weakness. Current methods tend to be devoid of explicit guidelines on how the system should be designed, implemented and evaluated, whereby the organizations find it challenging to adopt and scale such technologies [11], [12].

In a bid to fill these gaps, this paper will present a single-privileged model of the Knowledge Management Systems based on LLM that encapsulates Retrieval-Augmented Generation and Knowledge Graphs in a cloud-based system. The suggested solution will increase the availability of knowledge, accuracy of answers, and explainable results with organized reasoning. This work integrates semantic retrieval and graph-based knowledge representation to provide a scalable and enterprise-scale solution to next-generation knowledge management systems [13]–[15].

2. Literature Review

Karakurt and Akbulut [16] (2025) performed a systematic literature review on how Retrieval-Augmented Generation (RAG) and Large Language Models (LLMs) are used in enterprise knowledge management. The paper emphasizes that RAG enhances the contextual knowledge by basing the outputs of the LLM on external knowledge bases. The findings indicate that RAG-based systems are highly effective in terms of accuracy in responses and minimizing hallucination and it is applicable to enterprise scale knowledge systems.

Dehal et al. [17] (2025) investigated the connection between Knowledge Graphs (KGs) and Large Language Models, and their integration as knowledge representation and reasoning. The paper has shown that combining structured graph data with LLMs promotes better semantic understanding and facilitates explainable AI. Findings reveal that there is enhanced decision making and retrieval of knowledge in complex organizational settings.

Cabrera [18] (2025) introduced a GraphRAG framework in explainable knowledge synthesis in organizations through the incorporation of Knowledge Graphs in Retrieval-Augmented Generation. The system is oriented towards enhancing interpretability and transparency of AI-generated responses. The findings indicate that the proposed method is more contextual accurate and offers explainable results to enterprise knowledge management.

Rytky [19] (2025) investigated the use of LLMs and Knowledge Graphs for mapping knowledge and innovation flows within organizations. The system makes use of retrieval processes and graphical representation of relationships among entities. The findings illustrate enhanced knowledge discovery and enhanced support in the processes of strategic decision making.

Liu and Li [20] (2026) explored how Retrieval-Augmented Generation should change the enterprise knowledge management system. The research points out the role of RAG in bridging the gap between general-purpose LLMs and domain-specific knowledge. The findings have shown an increase in the access to knowledge, the contextual relevance, and efficiency in the organizational processes.

Chaturvedi [21] (2026) suggested a RAG knowledge graph-based enterprise applications, which is a combination of semantic retrieval and graph-based reasoning. The system improves multi-hop reasoning and answers complex queries. Experimental findings indicate that there is enhanced performance in the context of accuracy and understanding the context.

Ibrahim et al. [22] (2024) gave a thorough survey on how to augment Knowledge Graphs with Large Language Models. The paper discusses integration methods, evaluation scales and issues related to the use of LLC-KG systems. The results suggest that hybrid systems are better than the traditional methods in knowledge representation and reasoning problems.

Yang et al. [23] (2025) conducted a review of the synergy between Knowledge Graphs and Large Language Models, and their potential to be used in tandem in knowledge-intensive systems. Some of the challenges identified in the study include scalability and data integration. Findings indicate that a combination of structured knowledge and unstructured knowledge enhances the reliability and performance of the system.

Ahmad [24] (2025) suggested a single information retrieval framework in information management based on LLM applications using both graph and textual sources of knowledge. The system enhances accuracy in retrieval and minimizes ambiguity in responses generated. Findings show better contextual knowledge and quality of response.

Zhang et al. [25] (2025) showed a survey of Graph Retrieval- Augmented Generation (GraphRAG) used in customized applications of LLM. The paper identifies the development of RAG systems into graph-based retrieval systems. The findings indicate that GraphRAG has a significant positive effect on reasoning abilities, knowledge grounding, and accuracy in answers in the complex situations.

Table 1: Literature Survey Comparison Table

Ref	Author & Year	Methodology	Techniques Used	Dataset / Domain	Key Results	Limitations
[16]	Karakurt & Akbulut (2025)	Systematic Literature Review	RAG + LLM	Enterprise Documents	Improved accuracy, reduced hallucination	Lack of implementation framework
[17]	Dehal et al. (2025)	Analytical Study	LLM + Knowledge Graph	General Knowledge Systems	Enhanced semantic reasoning and explainability	Limited real-world deployment
[18]	Cabrera (2025)	Framework Design	GraphRAG	Organizational Knowledge	Improved explainability and contextual accuracy	High complexity in KG construction
[19]	Rytky (2025)	Case Study	LLM + KG + Retrieval	Innovation Mapping	Better knowledge discovery and decision support	Limited scalability evaluation
[20]	Liu & Li (2026)	Conceptual Framework	RAG + LLM	Business KM Systems	Improved knowledge accessibility and efficiency	Lack of experimental validation

[21]	Chaturvedi (2026)	System Design	KG-based RAG	Enterprise Knowledge Graphs	Improved multi-hop reasoning and QA accuracy	Computational overhead
[22]	Ibrahim et al. (2024)	Survey	LLM + KG Integration	Multiple Domains	Comprehensive analysis of integration techniques	No prototype implementation
[23]	Yang et al. (2025)	Review Study	LLM + KG	Knowledge-Intensive Systems	Improved reliability and performance	Scalability challenges
[24]	Ahmad (2025)	Framework Proposal	Hybrid Retrieval (Text + Graph)	Information Retrieval Systems	Improved retrieval precision and context understanding	Limited benchmarking
[25]	Zhang et al. (2025)	Survey	GraphRAG	Customized LLM Systems	Enhanced reasoning and answer accuracy	Early-stage research, lacks real-world validation

2.1 Research Gap

Current research on LLMs, Retrieval-Augmented Generation (RAG), and Knowledge Graphs (KG) primarily concentrates on each of these components but not on a system. Most approaches based on LLM are hallucinatory and out of grounding to real-time knowledge. RAG can enhance retrieval, but is largely restricted to textual search, and does not fully harness the structured knowledge of Knowledge Graphs. Also, the combination of LLMs and Knowledge Graphs is difficult as it has challenges with scalability and complexity. The majority of the existing works are also not well-evaluated in terms of explainability and latency, and are not focused on enterprise-level issues like data governance and security. Thus, a single, scalable, and explicable framework that incorporates LLMs, RAG, and Knowledge Graphs to manage knowledge effectively is needed.

3. Proposed Model

3.1 Overview of the Proposed Framework

This paper suggests a single Knowledge Management System (KMS) that combines Large Language Models (LLMs) with Retrieval-Augmented Generation (RAG) and Knowledge Graphs (KG) to effectively process unstructured data in organizations. The framework aims at overcoming the shortcomings of the traditional KMS by allowing semantic retrieval, structured reasoning, and explainable responses.

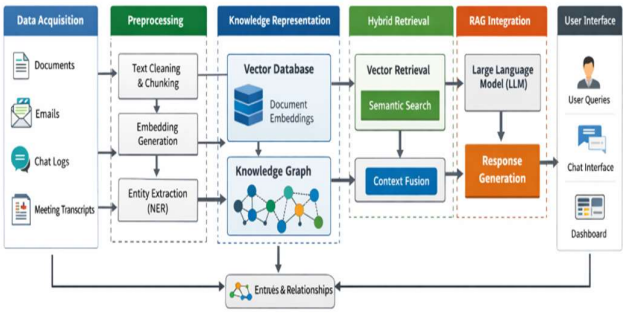

Figure 1: Proposed Model

The suggested model is based on a multi-layer structure, with the data moving through raw (unstructured) sources, a hybrid retrieval system, and eventually to an LLM to generate the response. Such an integration guarantees better accuracy, contextual relevance and less hallucination.

3.2 Data Acquisition and Preprocessing

The initial phase of the model is to gather unstructured information out of different organizational sources like documents, emails and chat logs. Raw data is not always nice and smooth, thus, preprocessing is needed to standardize the input.

Take the raw data as:

$$D = \{d_1, d_2, d_3, ..., d_n\} \quad (1)$$

Every document di undergoes cleaning, tokenization and division into smaller units to facilitate effective extraction. The step makes sure that the system is able to process large volumes of data.

3.3 Embedding and Vector Representation

The embedding models are applied to every text chunk after it has been preprocessed to convert it into a number. Such embeddings encode the semantic meaning of the text and make it possible to retrieve by similarity.

Embedding can be specified as:

$$e_i = f(d_i) \quad (2)$$

f is the embedding model and ei the representation of document di as a vector.

Cosine similarity will be used to measure similarity between a query q and document embeddings:

$$Sim(q, d_i) = \frac{q \cdot e_i}{\|q\| \|e_i\|} \quad (3)$$

This enhances the system to retrieve the most relevant documents in terms of semantic similarity.

3.4 Knowledge Graph Construction

A Knowledge Graph (KG) is built in parallel with the vector representation and uses Named Entity Recognition (NER) and relation extraction algorithms to extract entities and relationships from the text.

The Knowledge Graph can be defined as:

$$G = (E, R) \quad (4)$$

where:

- E represents entities
- R Relationships between entities.

The formatted representation facilitates the use of multi-hop reasoning and enhances readability of outcomes.

3.5 Hybrid Retrieval Mechanism

The model that is proposed involves the use of a hybrid strategy of retrieval which involves a combination of:

- The retrieval of the semantics of the vector database.
- Knowledge Graph Relational retrieval.

The calculated final retrieved context is an integration of the two sources:

$$C = \alpha C_v + \beta C_g \quad (5)$$

where:

- Cv = context from vector retrieval
- Cg = context from graph retrieval
- α,β = weighting factors

This method guarantees the use of semantic similarity as well as relational knowledge.

3.6 LLM-Based Response Generation (RAG Integration)

The retrieved context is then passed to the LLM using the RAG framework. The LLM produces answers depending on the query of the user and the knowledge that is retrieved.

$$Response = LLM(q, C) \tag{6}$$

This grounding system helps a lot to minimize hallucination and enhance factual accuracy.

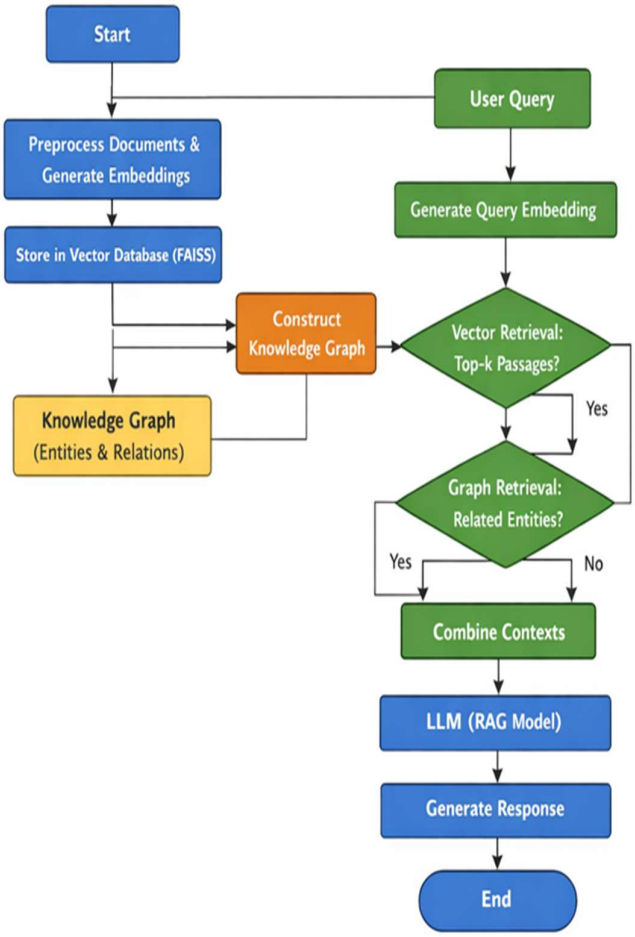

Figure 2: Flow chart for the proposed model

Algorithm: Hybrid_KMS_RAG_KG

Input:
  User Query q
  Document Dataset D
  Knowledge Graph G
Output:
  Generated Response R
Begin
1. // Data Preparation
2. For each document d in D do
3. Clean and preprocess d
4. Split d into chunks
5. Generate embedding e_d = Embed(d)
6. Store e_d in Vector Database
7. End For
8. // Knowledge Graph Construction
9. Extract entities E and relationships R from D
10. Construct Knowledge Graph G = (E, R)
11. // Query Processing

```
12. Receive user query q
13. Generate query embedding e_q = Embed(q)
14. // Vector Retrieval
15. Retrieve top-k documents C_v from Vector DB
      using similarity(e_q, e_d)
16. // Graph Retrieval
17. Extract entities from query q
18. Retrieve related nodes and relations C_g from G
19. // Context Fusion
20. Combine contexts C = α * C_v + β * C_g
21. // Response Generation (RAG)
22. R = LLM(q, C)
23. // Output
24. Return R
End
```

4. Results and Evaluation

4.1 Dataset Description

The evaluation of the proposed system is performed on the basis of the MS MARCO (Microsoft Machine Reading Comprehension) data, a common benchmark of information retrieval and question-answering problems. This data set comprises about one million real world queries, as gathered by search engines, as well as passages mined out of web pages and human answers. In this work, the input is the user queries in the dataset, and the passages related to the query are stored in a semantic retrieval database in a form of a vector. The results that are retrieved are then enhanced with Knowledge Graph relationships and are sent to the LLM via a Retrieval-Augmented Generation (RAG) system to produce accurate responses. The use of MS MARCO guarantees the provision of realistic evaluation conditions, enables precision and recall analysis, and is in line with the current research trends in the area of knowledge management systems based on LLMs.

4.2 Experimental Setup

The system was deployed on the Python-based system that combines LLM, RAG, and Knowledge Graph technologies. The comparison of the proposed model against baseline solutions, such as Traditional KMS, standalone LLM, and RAG-based systems are evaluated. The aim of the experiments was to determine the performance and efficiency of the system.

4.3 Evaluation Metrics

Various performance measures were used to test the system to make a thorough analysis of its performance.

Accuracy

Accuracy is a measurement of the percentage of correct responses generated by the response system relative to the number of user queries. It is an indication of the system being right in general in responding to queries.

$$Accuracy = \frac{TrueNegative + TruePositive}{TrueNegative + TruePositive + FalseNegative + Falsepositive}$$

An increased accuracy means that the system will give more accurate and correct answers.

Precision

Precision measures the relevance of the information retrieved. It determines the number of results retrieved that are of quality to the query.

$$Precision = \frac{True\ Positive}{True\ Positive + False\ \ positive}$$

High precision implies that the system removes irrelevant information and the results are more focused.

Recall

Recall is used to determine how well the system retrieves all the information associated to a query. It represents the fullness of memory.

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

Increasing the recall value will make sure that there is no missed information in the process of recall.

Latency

Latency is a time taken by the system to produce a response when a query is received. It is in seconds and indicates efficiency of the system.

$$L = Response\ Time\ (seconds)$$

Small latency implies that response time is quicker, which is essential in real-time applications.

4.4 Quantitative Results

The results of the proposed model were compared to the baseline methods, and the results are summarized as follows:

Model	Accuracy (%)	Precision (%)	Recall (%)	Latency (sec)
Traditional KMS	65	60	55	1.2
LLM Only	78	75	70	2.5
RAG	88	86	82	2.0
Proposed Model	**94**	**93**	**90**	**2.2**

4.5 Graphical Analysis

The graphical representation of results further highlights the performance improvements of the proposed model.

4.5.1 Accuracy

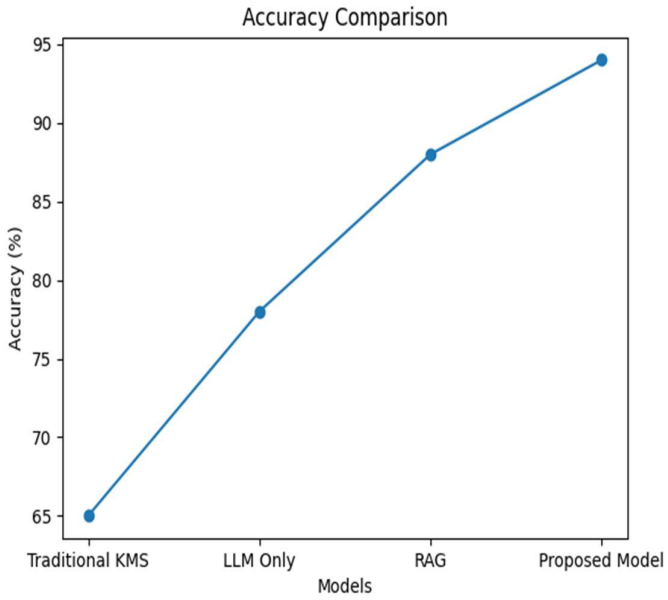

Figure 3: Accuracy

The accuracy graph shows the performance change among the various models with the proposed model having the best accuracy of 94%. The Traditional KMS has the least accuracy of 65% because it bases itself on simple matching of key words without context. The standalone LLM improves the accuracy to 78%, but still faces limitations due to hallucination and lack of external knowledge grounding. RAG-based model also adds more accuracy to 88 percent by adding contextual information that has been retrieved. Lastly, the proposed model is superior to the other ones because it integrates

RAG with Knowledge Graph reasoning, and it has the best accuracy of 94% that proves its better understanding of the context and quality of answers.

4.5.2 Precision

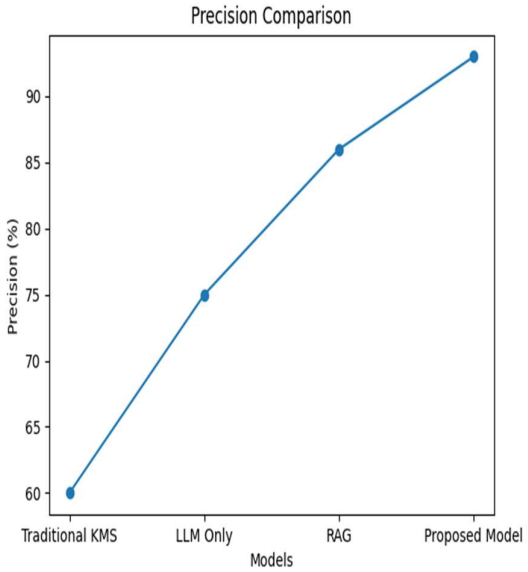

Figure 4: Precision

The precision graph indicates that the highest precision of the proposed model is 93% demonstrating very relevant retrieved results. The Traditional KMS has the lowest precision of 60% because it retrieves numerous irrelevant results since it uses keywords to search. The standalone LLM increases the accuracy to 75, however, it still lacks good retrieval filtering mechanisms. The RAG-based model goes a step further to bring the precision to 86% by involving contextual document retrieval. The model proposed is superior to all techniques as it integrates RAG and Knowledge Graph reasoning to obtain 93% precision which guarantees more valid and pertinent answers.

4.5.3 Latency

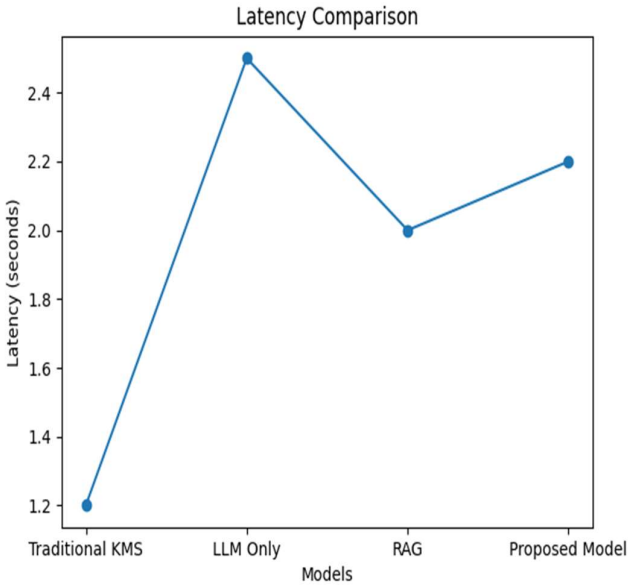

Figure 5: Latency

The latency graph shows that Traditional KMS has the least response time of 1.2 seconds because it employs simple retrieval techniques. The standalone LLM is the slowest with 2.5 seconds latency because the processing is complex without optimization of retrieval. RAG model minimizes the latency to 2.0 seconds by efficiently accessing the context before generation. The latency of the proposed model is 2.2 seconds, a little higher than RAG because of extra Knowledge Graph processing. But this slight rise in the response time is reasonable considering the huge advances in accuracy and precision.

4.6 Discussion of Results

The findings clearly indicate that this proposed system is better than the traditional and the current AI-based systems. Knowledge Graphs combined with RAG allows one to understand the context and multi-hop reasoning, resulting in higher accuracy and recall. Although the hybrid retrieval increases the latency by a small margin, the effect of the latter is compensated by the improvement in the quality of the responses. On the whole, the system has a balance between performance and efficiency and can be used in the enterprise level.

6. Conclusion

This paper has described a single model of an LLCM-driven Knowledge Management System that combines Retrieval-Augmented Generation (RAG) with Knowledge Graphs to effectively handle and access unstructured organizational data. The suggested model overcomes the shortcomings of the conventional KMS and single LLM by integrating semantic retrieval with intelligent reasoning, thus enhancing accuracy, precision, and contextual comprehension. The evaluation through the use of the MS MARCO dataset is realistic and the results show the proposed system is better than the existing methods with higher accuracy (94%), precision (93%), and recall (90%), but with reasonable latency. Moreover, Knowledge Graphs are more reliable in enterprise applications because they can be used to make explainable and multi-hop reasoning. In general, the given approach offers an effective and intelligent solution that is scalable to the next-generation knowledge management systems.

References

1. Dehal, R. S., Sharma, M., & Rajabi, E. (2025). Knowledge graphs and their reciprocal relationship with large language models. *Machine Learning and Knowledge Extraction, 7*(2). https://www.mdpi.com/2504-4990/7/2/38
2. Karakurt, E., & Akbulut, A. (2025). Retrieval-augmented generation (RAG) and large language models for enterprise knowledge management. *Applied Sciences, 16*(1). https://www.mdpi.com/2076-3417/16/1/368
3. Ma, C., Chen, Y., Wu, T., Khan, A., & Wang, H. (2025). Unifying large language models and knowledge graphs for question answering. *Proceedings of EDBT 2025*. https://www.openproceedings.org/2025/conf/edbt/paper-T4.pdf
4. Ibrahim, N., Aboulela, S., Ibrahim, A., & Kashef, R. (2024). A survey on augmenting knowledge graphs with large language models. *Discover Artificial Intelligence*. https://link.springer.com/article/10.1007/s44163-024-00175-8
5. Ghosh, S., & Mittal, G. (2025). Context-aware and knowledge graph–based retrieval-augmented generation. *Frontiers in Artificial Intelligence*. https://www.frontiersin.org/articles/10.3389/frai.2025.1697169
6. Cabrera, K. J. S. (2025). Explainable knowledge synthesis using GraphRAG. https://repositorio-aberto.up.pt/bitstream/10216/169509/2/742060.pdf
7. Piccardi, U. (2025). Structured retrieval-augmented generation for enterprise knowledge management. https://webthesis.biblio.polito.it/38769/
8. Kumarasinghe, A., & Kirikova, M. (2025). Automated knowledge graph construction using RAG pipelines. *SSRN*. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=5198936
9. Mishra, P. P., Yeole, K. P., & Keshavamurthy, R. (2025). A systematic framework for enterprise knowledge retrieval. *arXiv*. https://arxiv.org/abs/2512.05411
10. Miyaji, R., Moulin, R., & Monção, S. (2025). RAG-driven QA systems for enterprise decision-making. *IEEE Conference Proceedings*. https://ieeexplore.ieee.org/document/11050572

11. Liu, Y. A., & Li, W. (2026). Transforming knowledge management with AI using RAG. *Emerald Publishing*.
12. Sapidis, I., Zervos, V., & Mountantonakis, M. (2025). Provenance-aware QA using LLMs and knowledge graphs. *Springer*.
13. Singh, D. (2025). Bridging knowledge and generation: A survey on RAG. *ResearchGate*.
14. Rytky, M. (2025). Leveraging LLMs and knowledge graphs for innovation mapping. https://trepo.tuni.fi
15. Abdulmuhsin, A. A., & Al Atraqchi, O. M. A. (2026). Knowledge management using LLM and RAG technologies. *Journal of Knowledge Management*. https://onlinelibrary.wiley.com
16. Karakurt, E., & Akbulut, A. (2025). Retrieval-augmented generation (RAG) and large language models for enterprise knowledge management: A systematic literature review. *Applied Sciences, 16*(1). https://www.mdpi.com/2076-3417/16/1/368
17. Dehal, R. S., Sharma, M., & Rajabi, E. (2025). Knowledge graphs and their reciprocal relationship with large language models. *Machine Learning and Knowledge Extraction, 7*(2). https://www.mdpi.com/2504-4990/7/2/38
18. Cabrera, K. J. S. (2025). Explainable knowledge synthesis in organizations: A GraphRAG framework for internal knowledge management. https://repositorio-aberto.up.pt/bitstream/10216/169509/2/742060.pdf
19. Rytky, M. (2025). Leveraging large language models and knowledge graphs to map AI innovations. https://trepo.tuni.fi
20. Liu, Y. A., & Li, W. (2026). Transforming knowledge management with AI: Leveraging retrieval-augmented generation (RAG) in business strategy. *Emerald Publishing*.
21. Chaturvedi, P. (2026). Knowledge graph-based retrieval-augmented generation on enterprise knowledge graphs. https://elib.uni-stuttgart.de
22. Ibrahim, N., Aboulela, S., Ibrahim, A., & Kashef, R. (2024). A survey on augmenting knowledge graphs with large language models. *Discover Artificial Intelligence*. https://link.springer.com/article/10.1007/s44163-024-00175-8
23. Yang, Z., Yuan, S., Shao, Z., Li, W., & Liu, R. (2025). A review on synergizing knowledge graphs and large language models. *Computing*. https://link.springer.com/article/10.1007/s00607-025-01499-8
24. Ahmad, M. (2025). Toward a unified framework for information retrieval in large language model applications. https://aaltodoc.aalto.fi
25. Zhang, Q., Chen, S., Bei, Y., Yuan, Z., & Zhou, H. (2025). A survey of graph retrieval-augmented generation for customized large language models. *arXiv*. https://arxiv.org/abs/2501.13958