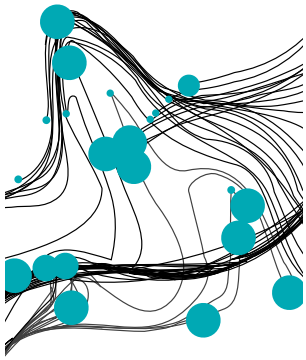
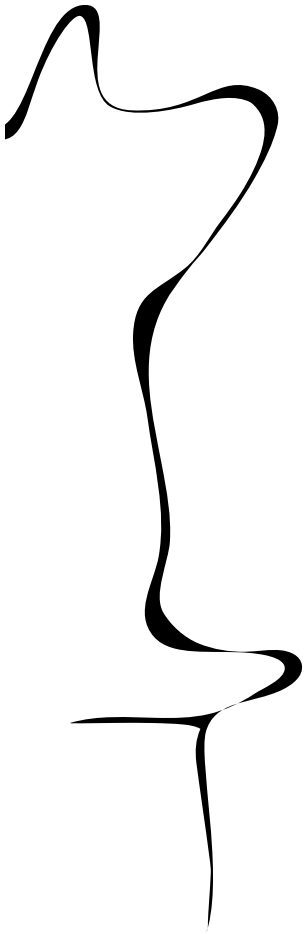


.9298

DMB

DATA MANAGEMENT AND BIOMETRICS



KNOWLEDGE GRAPH FOR QUERY ENRICHMENT IN RETRIEVAL AUGMENTED GENERATION IN DOMAIN SPECIFIC APPLICATION

Massimo Perna

MASTER'S ASSIGNMENT

Committee:
dr. E. Talavera Martínez, MSc (1st supervisor)
dr.ing. G. Englebienne

May, 2025

2025DMB0003
Data Management and Biometrics
EEMathCS
University of Twente
P.O. Box 217
7500 AE Enschede
The Netherlands

UNIVERSITY OF TWENTE. | DIGITAL SOCIETY INSTITUTE

Knowledge Graph for Query Enhancement in Retrieval Augmented Generation Domain-Specific Applications

Massimo Perna
Faculty of Electrical Engineering, Mathematics and Computer Science
University of Twente
m.perna@sudent.utwente.nl

***Abstract*—We propose and evaluate a hybrid approach to enhance Retrieval-Augmented Generation (RAG) systems by leveraging query enrichment through knowledge graphs. RAG systems, which combine retrieval mechanisms with generative models, are powerful tools for answering complex queries by incorporating external knowledge. However, these systems often face challenges in domain-specific contexts where embedding models may lack the precision required to retrieve relevant information. This limitation is particularly significant in specialized domains, such as finance and biomedicine, where nuanced understanding is essential.**

Our approach addresses this gap by employing a Large Language Model (LLM) not only as the engine powering the RAG system but also as a tool for extracting structured triplets during both the ingestion and querying phases. These triplets, stored in a knowledge graph, are injected into queries during inference to generate enriched and contextually aware inputs, improving the precision of the retrieval process.

We evaluate the proposed method on three datasets: a general-domain dataset with question-answer pairs from Wikipedia and two domain-specific datasets in finance and biomedicine, each comprising approximately 8,000 document chunks. Experimental results demonstrate significant improvements in retrieval precision and recall with an improvement of up 5% for the precision metric, 19% for the recall and an overall increase of on average 7% in the generative quality of the output, as well as enhanced relevance and coherence in the system-generated answers. These findings highlight the potential of knowledge graphs to bridge gaps in embedding precision and improve overall performance in both open- and closed-domain settings.

I. Introduction

The rise of Large Language Models (LLMs) has revolutionized natural language processing, enabling systems to handle complex queries by integrating external knowledge into their outputs. Retrieval-Augmented Generation (RAG) systems, which combine a retrieval mechanism with generative models, have emerged as a critical innovation in this domain [1]. These systems excel in tasks such as question answering, document summarization, and conversational AI by retrieving relevant external documents to complement their generative components. However, their effectiveness heavily depends on the precision of the retrieval process.

One of the key challenges in RAG systems lies in retrieving the most relevant documents from the database. Retrieval errors often result in incomplete or inaccurate answers, especially in domain-specific scenarios where general-purpose embedding models fail to capture nuanced relationships in the text [2]. This problem is exacerbated by the increasing complexity of queries and the limitations of large context windows, which, while capable of processing large text chunks, may still return irrelevant results [3].

Several approaches have been proposed to enhance retrieval. For instance, HyDE (Hypothetical Document Embeddings) [4] improves retrieval by generating synthetic documents based on a query. While effective in improving retrieval precision, this method is prone to hallucinations [1], as the generated documents may not align with real-world content. Additionally, it requires computationally expensive, large-scale LLMs with broad general knowledge to function effectively, limiting its applicability in closed-domain settings.

To address these challenges, we propose a hybrid approach that incorporates query enrichment through knowledge graphs to improve the retrieval accuracy of RAG systems. Our method involves extracting structured triplets (`Entity → Relation → Entity`) from documents during the ingestion phase and storing them in a knowledge graph. These triplets are later injected into queries during inference, enriching the query with additional context and enabling the retriever to return more relevant documents.

In our implementation, documents are divided into smaller chunks, which are embedded using pre-trained models such as RoBERTa [5]. These embeddings are stored in vector databases and retrieved using Approximate Nearest Neighbors (ANN) algorithms to ensure semantic relevance. Simultaneously, a knowledge graph is constructed using Named Entity Recognition (NER) to extract meaningful triplets, which are used to enhance query retrieval during inference.

This hybrid approach addresses key limitations in RAG systems by bridging the gap between general-purpose embeddings and domain-specific requirements. We evaluate our method on three datasets: a general-domain dataset based on Wikipedia and two domain-specific datasets in finance and biomedicine. Experimental results demonstrate substantial improvements in retrieval precision, recall, and the coherence of generated answers, highlighting the efficacy of query enrichment through knowledge graphs. These findings underscore the potential of this approach to improve both open- and closed-domain RAG systems.

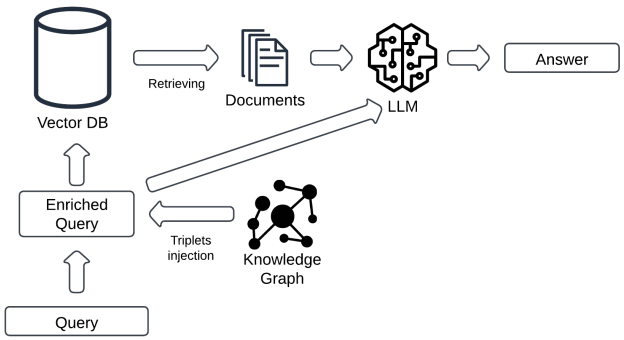


Fig. 1: Solution Pipeline Representation: When a query is made, it is first enriched using knowledge in the form of triplets injected by querying a knowledge graph. The enriched query is then passed to a vector database to retrieve relevant documents, which are subsequently provided to the LLM along with the query to generate a relevant answer.

A. Research questions

The research aims to address some of the fundamental challenges faced by RAG systems in domain-specific contexts, where retrieval precision and generative quality are critical. Through the following research questions, we aim to tackle these issues by evaluating the overall performance improvements introduced by a hybrid DPR approach.

- How RAG's retrieval power can be enhanced using a hybrid DPR model?
- How does the integration of a hybrid DPR improve RAG output quality on domain-specific knowledge without need to fine tune model on specific downstream task?
- How effective is the hybrid approach in improving retrieval and generation, and what challenges or limitations does it present?

B. Contributions and Significance

The main contributions of this work are as follows:

- We propose a novel enhancement for RAG systems using a knowledge graph to improve query enrichment and document retrieval in closed-domain scenarios.
- Our approach reduces hallucinations and improves the precision of document retrieval without requiring costly fine-tuning or retraining.

C. Structure of the report

This report is organized into 5 further sections:

1) **Related Work**: This section reviews existing research and approaches in the field, highlighting the state of the art and identifying gaps that this study aims to address.
2) **Methodology**: This section describes the methodologies employed to address the different aspects of the research problem, detailing the proposed approach and its components.
3) **Experimental Setup**: This section outlines the technical implementation of the proposed methods, including the datasets, tools, and configurations used for experimentation.
4) **Results**: This section presents the outcomes of the experiments, including performance metrics and key findings.
5) **Discussion**: This section provides an analysis of the results, discusses the strengths and limitations of the proposed approach, addresses the research questions, and suggests directions for future work.

II. Related Works

Retrieval-Augmented Generation (RAG) systems represent a critical advancement in natural language processing by integrating information retrieval with generative models. This dual approach enables the retrieval of relevant external knowledge to augment generated answers, addressing the limitations of purely generative models. Current implementations of RAG typically rely on either standalone knowledge graphs or vector databases to perform the retrieval step. Both methodologies require embedding models to encode text into vector representations that can be used by retrieval and generation modules.

Prominent embedding models used in RAG include OpenAI's GPT series [6], Google's T5 [7], and variants of BERT [8]. These models offer varying levels of adaptability and domain specificity, making them versatile tools for RAG systems across different applications.

BERT (Bidirectional Encoder Representations from Transformers), in particular, has had a transformative impact on retrieval tasks. By jointly conditioning on both left and right contexts in all transformer layers, BERT learns nuanced word representations that capture semantic relationships across entire sentences or passages. This capability has allowed BERT to achieve state-of-the-art performance in tasks such as question answering, language inference, and semantic similarity [8]. Its evolution into specialized models, such as RoBERTa [5], further enhances its utility by extending context windows and improving its adaptability to domain-specific scenarios.

The concept of embeddings underpins many modern retrieval systems. Embeddings encode words, sentences, or texts as continuous vectors in a high-dimensional space, where semantically similar elements are positioned closer together. This approach enables efficient semantic searches using techniques like Approximate Nearest Neighbors (ANN) [9]. Unlike traditional retrieval methods such as BM25 [10], which rely on term frequency and inverse document frequency (TF-IDF), embeddings allow RAG systems to retrieve semantically related text even when the query and documents share limited lexical overlap. However, to achieve high-quality retrieval, documents must first be chunked into smaller pieces to preserve semantic granularity and improve relevance in the retrieval phase.

Knowledge Graphs (KGs) [11] present an alternative and complementary approach to enhancing retrieval. As structured representations of entities and their relationships, KGs model real-world knowledge through interconnected nodes (entities)

and edges (relationships). KGs enable systems to capture intricate interrelations and hierarchies, which can be leveraged to enrich queries with contextual information. This capability is particularly beneficial in complex or closed-domain tasks where precise understanding of relationships is crucial.

Several studies have demonstrated the effectiveness of Knowledge Graphs and Dense Passage Retrieval (DPR) in improving RAG systems. For example, Siriwardhana et al. [12] explored the application of DPR in open-domain question answering, while Wang et al. [13] and Xu et al. [14] examined the integration of KGs in real-world business scenarios, such as customer assistance at LinkedIn. These studies highlight the versatility of hybrid retrieval systems in addressing diverse use cases. Research on DPR [15], [16] has further emphasized the importance of improving access to stored knowledge by enhancing embedding models rather than merely increasing storage capacity.

Hybrid retrieval approaches, combining semantic (dense) and keyword-based (sparse) methods, have also been widely explored. For instance, Sawarkar et al. [17] introduced hybrid retrievers to improve retrieval performance on datasets like Google NQ and HotpotQA, which feature open-domain question answering tasks. While hybrid methods have shown promise, their application to more complex RAG systems remains limited, particularly in scenarios requiring advanced reasoning capabilities. Challenges such as high computational cost, increased latency, and difficulty in handling complex queries persist, as noted in studies by Cuconasu et al. [18] and Zeng et al. [19].

Despite these advancements, a significant gap exists in fully integrating Knowledge Graphs with dense retrieval systems for RAG. While embeddings excel at capturing semantic relationships, their performance can degrade in specialized domains where precise context is critical. Conversely, KGs offer structured context but are limited in scalability and generalization to broader datasets. The intersection of these methodologies presents a promising avenue for future research, particularly in enhancing RAG systems to bridge the gap between sparse and dense retrieval approaches.

In summary, the evolution of RAG systems has been propelled by innovations in embedding models, Knowledge Graphs, and hybrid retrieval techniques. However, challenges such as domain adaptation, computational efficiency, and query complexity highlight the need for further exploration. This work builds on these advancements by proposing a hybrid approach that leverages both dense retrieval and query enrichment via Knowledge Graphs to improve precision and relevance in RAG systems.

III. Methodology

In this section we present the methodology pipeline adopted to conduct the experiment:

- **Data Cleaning:** Raw datasets, often containing extraneous elements like HTML artifacts, are processed to remove irrelevant content while preserving semantic integrity. This ensures that the text remains meaningful and consistent for downstream tasks.
- **Data Chunking:** Cleaned data is divided into smaller, fixed-size chunks to comply with tokenization limits of embedding models. Overlaps and natural splits ensure semantic coherence across chunks.
- **Text Embedding:** Each chunk is embedded using the RoBERTa-Large model to generate high-dimensional text representations suitable for efficient semantic retrieval.
- **Knowledge Graph Creation:** Entity-relation triplets are extracted via Named Entity Recognition (NER) to construct a knowledge graph.
- **Storage and Retrieval:** A vector database stores the embeddings and links them to the knowledge graph. Queries are enriched with graph triplets, enabling precise retrieval of relevant chunks.
- **Answer Generation:** Retrieved chunks and enriched queries are fed into a language model to produce accurate, context-aware answers.

A. Data Cleaning

Before processing the datasets, only the first 81 QA pairs were used, as they provided a sufficiently large number of chunks to conduct the experiment while remaining computationally manageable, after which a rigorous data cleaning pipeline was applied to ensure that all inputs were consistent and free from irrelevant elements. For example, the Google Natural Questions dataset contained raw text embedded with HTML components as a result of web scraping from Wikipedia pages. These HTML artifacts were removed using the Large Language Model, which was tasked with preserving meaningful information while eliminating formatting tags and unnecessary metadata.

The cleaning pipeline was designed to maintain the semantic integrity of the text, ensuring that important details such as key facts, dates, names, and lists were preserved. Below is the specific prompt used for this cleaning process:

Data Cleaning Prompt

You are a helpful assistant. You will receive a text extracted from a Wikipedia HTML page. Your task is to clean the text by only removing HTML characters and formatting. Preserve all meaningful information and keep as many words as possible from the original page, including dates, names, lists, and other important details without cutting anything. DO NOT MODIFY TEXT. Also, remove the footer that contains Wikipedia's terms and conditions.
After cleaning, return a JSON object with the following fields: - `"document_text"` for the cleaned version of text
Text: {`document`}

This cleaning workflow was uniformly applied to all datasets, ensuring consistency in structure and semantics. By

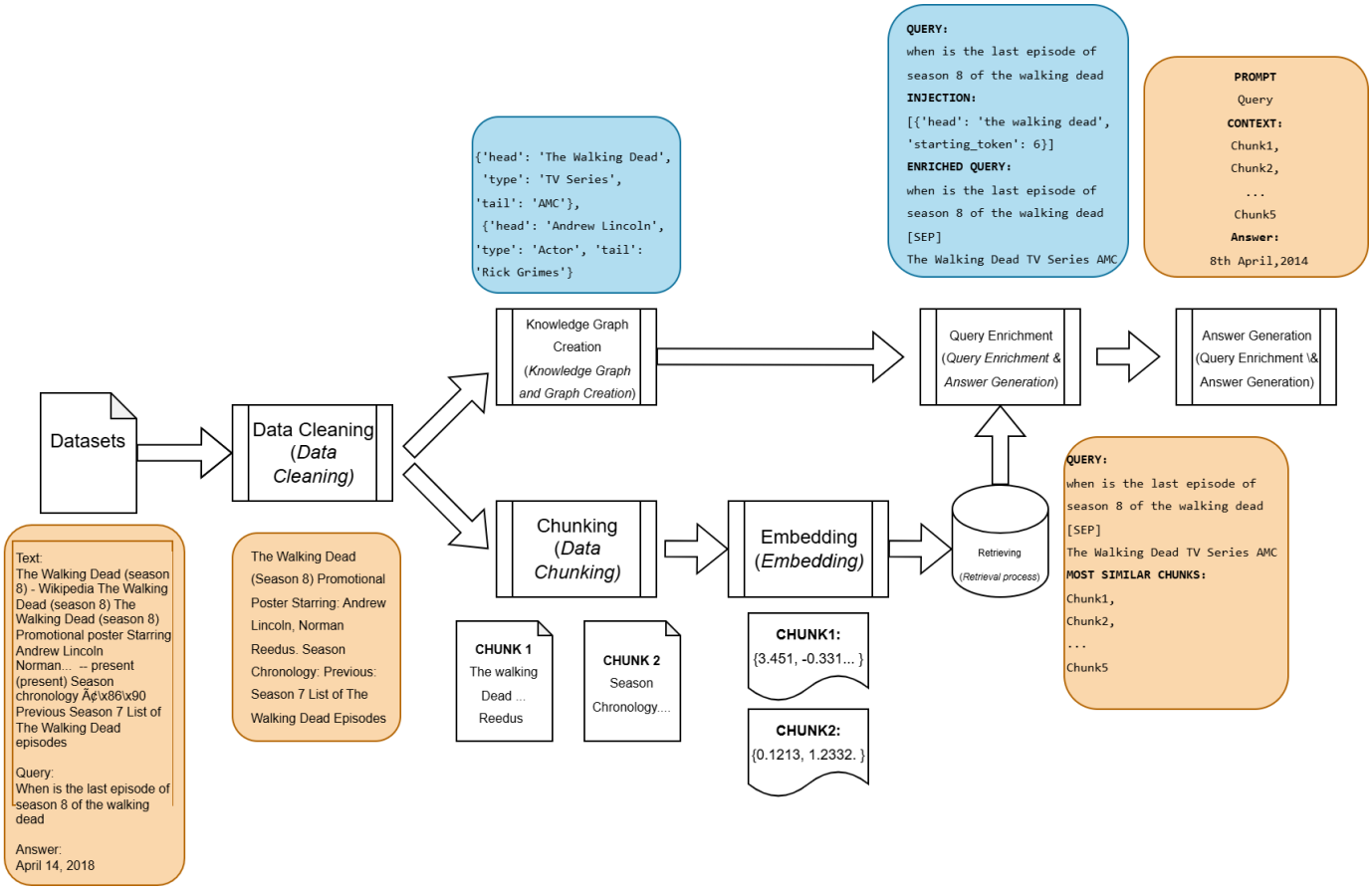


Fig. 2: Methodology Pipeline: Pipeline representing the various stages of the methodology. Each block represent a stage of the methodology with the relative subsection in methodology. Moreover for each step an example of the real results is given based on the first entry of the Google NQ Dataset (see experimental setup). The example box in blue represent the enhancement applied.

eliminating extraneous formatting, the datasets became more suitable for downstream processing, particularly for embedding and retrieval tasks.

B. Data Chunking

To accommodate the token limits of modern embedding models and ensure efficient retrieval, the datasets were divided into smaller, manageable chunks. A fixed window of 196 words (approximately 256 tokens) was used, with a tolerance of up to 30 additional words to allow natural splits at punctuation marks, such as periods, question marks, and exclamation marks. Additionally, an overlap of 20 characters was introduced between contiguous chunks to maintain semantic continuity and prevent information loss at chunk boundaries.

Parameter	Value
Word Number	128
Overlap Value	32
End-of-Sentence Characters	. ? !
Tolerance Window	30

TABLE I: Chunking Parameters

Smaller chunks offer several advantages for Retrieval-Augmented Generation (RAG) systems. First, they improve the granularity of the retrieval process, ensuring that the most relevant portions of a document are retrieved in response to a query. Smaller chunks also align with the token limits of embedding models, preventing truncation and preserving the quality of the embeddings.

Moreover, this chunking strategy supports our model's limitations. Since the embedding model used in this study has less parametric knowledge compared to larger models, it benefits from processing smaller and more digestible input chunks. This approach ensures that the system maintains retrieval relevance without being overwhelmed by excessive context.

By maintaining semantic coherence and breaking down documents into smaller, self-contained units of meaning, this chunking strategy significantly enhances the retrieval and generative capabilities of the system, especially in domain-specific applications.

C. Embedding

Once chunked, the text segments were embedded using the RoBERTa-Large model [5]. This model was chosen for

its strong performance in generating robust, domain-agnostic text representations. Its ability to distinguish nuanced textual elements, such as case sensitivity, and its large context window make it well-suited for handling diverse datasets.

Parameter	Value
Embedding Dimension	1024
Context Window	512
Number of Parameters	355M
Tensor Type	F32, I64

TABLE II: RoBERTa-large [5] model parameters

D. Knowledge Graph and Graph Creation

The graph creation and embedding processes were carried out in parallel, leveraging a two-step approach. In the first step, the LLM was utilized to analyze the text and extract meaningful triplets acting as a Named Entity Recognition (NER) extractor. These triplets were structured as `(entity -> relation -> entity)`. This automated extraction ensured that the relationships and entities within the dataset were accurately identified and semantically meaningful.

In the second step, each identified entity was transformed into a corresponding node within the knowledge graph. Simultaneously, we generated embeddings for each entity using **all-MiniLM-L6-v2** [20], a compact model distilled from Microsoft's **MiniLM-L12-H384-uncased** [21]. These embeddings enabled efficient semantic search capabilities within the graph database.

This hybrid approach allowed the knowledge graph to serve as a dynamic context enrichment tool, seamlessly integrating structured knowledge with semantic representations. Below is an example of the prompt used to extract entities and relations from text:

NER Prompt Used for Extraction

Act as a Named Entity Recognition (NER) model to extract entities and relationships from the provided text.
Return the output as a list of relations in the JSON format:
`[ {'head': '', 'type': '', 'tail': ''} ]`
TEXT: {text}

The knowledge graph facilitated query enrichment by providing structured data in the form of triplets, which were incorporated into the query embedding process to enhance retrieval performance and ensure more contextually relevant results.

To illustrate the structure of the triplets extracted from the knowledge graph, consider the following example from an extraction from a BioASQ sample:

- `{'head': 'Pre-exposure prophylaxis (PrEP)', 'type': 'is effective against', 'tail': 'HIV infection'}`
- `{'head': 'tenofovir/emtricitabine', 'type': 'is a component of', 'tail': 'Pre-exposure prophylaxis (PrEP)'}`

These triplets represent structured relationships between entities, providing rich contextual data that enhances the query embedding process and improves retrieval performance.

KG	Entities	Relations
BioASQ	52,032	70,267
OpenQA	1,610	1,988
FinQA	1,452	1,010

TABLE III: Knowledge Graph statistics: number of entities and relations for each Knowledge Graph.

As we can see the biggest Knowledtge Graph is the one containing BioASQ, very likely because of the very high number of documents and high-specific terms.

E. Retrieval process

For data storage and retrieval we utilized a vector database. Each document was segmented into smaller chunks and indexed with unique IDs, linking each chunk to its position in the original dataset. This indexing scheme enabled the efficient tracing of embeddings back to their respective text segments, which were then used as input for the LLM.

As similarity measure we used the L2-distance (Euclidean distance), was deemed suitable for our requirements as it provided robust performance without the need for alternate measures. For every retrieval request, the top five most similar embeddings were returned, ensuring an optimal balance between precision and recall in the retrieval process. This mechanism improved the likelihood of retrieving the most contextually relevant chunks while minimizing computational overhead.

F. Query Enrichment & Answer Generation

The final step involved generating answers through the RAG system for a given query. This process began with query enrichment, where the initial query was augmented using the knowledge stored in the knowledge graph. Named Entity Recognition (NER) was performed on the query to identify key entities, which were then used to fetch a predefined number of relevant triplets (five in this case) from the knowledge graph. These triplets were appended to the original query, separated by the `[SEP]` token to maintain clarity and structure.

The inclusion of `[SEP]` tokens was particularly effective in distinguishing triplets within the enriched query, aligning with the segment-level training paradigm of models such as BERT. This structural separation enabled the RAG system to process enriched queries more effectively, improving the relevance of retrieved chunks and enhancing the quality of generated answers.

Below is an example of a query from the BioASQ dataset, illustrating the impact of enrichment:

Query Example

Original Query: Concizumab is used for which diseases?
Enriched Query: Concizumab is used for which diseases? [SEP] Concizumab instance of monoclonal antibody [SEP] Concizumab instance of monoclonal [SEP] Concizumab instance of antibody [SEP] Concizumab instance of monoclonal monoclonal

After enrichment, the query was embedded and compared with other embeddings in the vector database to retrieve the most relevant chunks. These retrieved chunks, along with the enriched query, were then passed to the LLM for answer generation.

The final output was generated by the LLM and evaluated for relevance and correctness. Below is the specific prompt used for the RAG-based question-answering task:

RAG Prompt for Question Answering

You are a helpful assistant. You will receive a question and relevant context from a document. If the answer to the question is present in the document, provide a direct and precise answer without adding extra details. If the information is not found in the document, respond only with "NO DOCUMENT."
Question: {user_question}
Document Context: {document_context}
Answer:

Example of Prompt

You are a helpful assistant. You will receive a question and relevant context from a document. If the answer to the question is present in the document, provide a direct and precise answer without adding extra details. If the information is not found in the document, respond only with "NO DOCUMENT."
Question: when did nsw last won a state of origin series
Document Context:
South Wales won the deciding match i...
by winning the 2007 series, as well as the 2008 series...
and 1080i DVB-T and PAL...
Queensland defeated New South...

IV. Experimental Setup

A. Datasets

The experiments conducted in this study utilized three distinct datasets: one open-domain dataset and two domain-specific datasets. The open-domain dataset, Google Natural

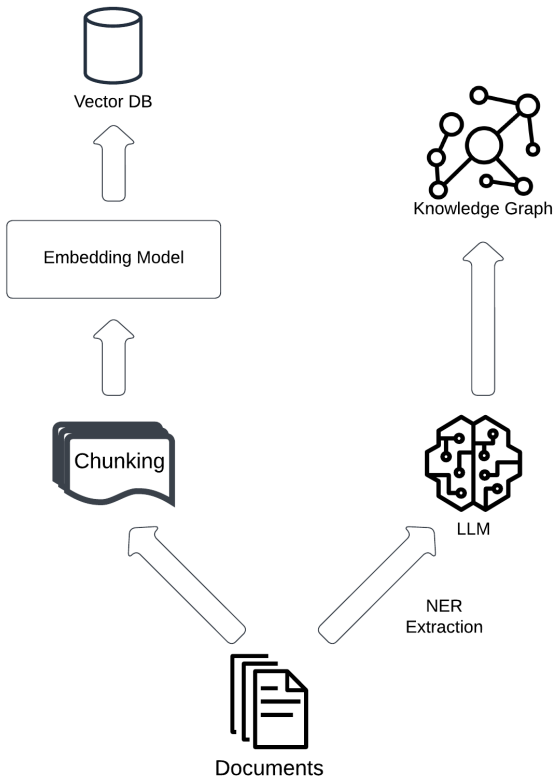


Fig. 3: Ingestion Pipeline Representation: When a document is ingested into the system, it follows two pathways: the classical approach, where the document is first chunked, embedded, and stored; and a new approach, where the document is processed by an LLM functioning as a Named Entity Recognizer (NER), extracting knowledge in the form of triplets and storing them in a knowledge graph.

Questions (Google NQ) [22], is a widely recognized benchmark designed to evaluate retrieval and reasoning capabilities across diverse general knowledge topics. For the domain-specific datasets, we selected FinQA [23], which emphasizes financial reasoning tasks, and BioASQ [24], a challenging biomedical dataset that focuses on expert-level question answering.

To further illustrate the nature of the datasets used in this study, we provide representative examples from Google NQ, FinQA, and BioASQ, showcasing their distinct characteristics and formats:

- **Google NQ:**
  - **Question:** Where does "jinx, you owe me a coke" come from?
  - **Short Answer:** Jinx is a children's game where penalties occur if two people say the same word simultaneously.
  - **Document (Excerpt):** Jinx is a children's game with varying rules and penalties. In America, one person

Task	Dataset Description
Google NQ [22]	**Domain:** General Knowledge **Purpose:** Evaluates the ability to answer questions based on long documents, including reasoning beyond sentence boundaries. **Data Source:** Wikipedia articles and Google search queries. **Question Types:** Fact-based questions, answers often found verbatim or paraphrased. **Number of Examples:** 307,373 questions. **Answer Format:** Short answers or spans from the document.
FinQA [23]	**Domain:** Finance **Purpose:** Evaluates the ability to answer complex questions requiring numerical reasoning and the integration of textual and tabular data. **Data Source:** Financial reports, tables, and documents. **Question Types:** Complex, multi-step reasoning questions involving calculations and financial data interpretation. **Number of Examples:** 8,000 questions. **Answer Format:** Numeric answers, sometimes with textual explanations.
BioASQ-QA [24]	**Domain:** Biomedical **Purpose:** Evaluates expert question answering in the biomedical field with challenging datasets. **Data Source:** Questions generated by biomedical experts, with reference answers and supporting material. **Question Types:** Expert-generated, requiring deep domain knowledge. **Number of Examples:** 4,721 questions. **Answer Format:** Golden standard reference answers and supporting material.

TABLE IV: Details of Google NQ, FinQA, and BioASQ-QA datasets

can say, "Jinx, you owe me a coke," to get a free coke from the other person.

- **FinQA:**
  - **Document:** Historical invoice from Tabacalera Cubana, S.A. (1941).
  - **Question:** What is the total amount charged for the shipment?
    Answer:$3.51 (including postage, registration, and war risk insurance).
  - **Details (Excerpt):** The shipment includes 960 cigarettes priced at $3.00, with additional charges such as postage ($0.50) and insurance ($0.01).
- **BioASQ:**
  - **Question:** Concizumab is used for which diseases?
  - **Answer:** Hemophilia A and B.
  - **Documents (Excerpts):**
    - PubMedID 37341887: Concizumab, a monoclonal antibody, prevents bleeding episodes in patients with hemophilia A and B.
    - PubMedID 35869698: Discusses non-factor products like concizumab targeting coagulation pathways.

For readability purposes, the full text of the ground truths has been truncated.

For the Google Natural Questions dataset, we utilized a subset of the training set, consisting of 158 unique Wikipedia pages. This approach allowed us to reduce computational requirements while retaining the diversity of the dataset's queries. Similarly, for the domain-specific datasets FinQA and BioASQ, only the most relevant document chunks were retained to focus on the evaluation of system performance in specialized contexts.

The selection of these datasets was designed to strike a balance between evaluating general-purpose retrieval capabilities and testing the precision and relevance of domain-specific performance. This combination ensures that our system can be assessed for its generalizability across open-domain tasks and its ability to deliver accurate and contextually appropriate results in closed-domain scenarios such as finance and medicine.

B. Baseline

The baseline is represented by the RAG system without any query enhancement techniques. It consists of the embedder, the vector database, and the DPR, which work together to retrieve the context to pass along with the query to the LLM for response generation. More specifically, the baseline skips the process of triplet injection from the knowledge graph, instead passing the retrieved context directly to the model to generate and evaluate the output.

C. Evaluation Process

The evaluation process involved the following steps:

1) Two versions of the experiment were conducted for each dataset: one using a simple RAG model and the other incorporating triplets as a context enhancer. Specifically, the retrieval process was divided into two scenarios: (1) embedding the question without adding triplets to enhance the context, and (2) embedding the question after adding context through triplets to improve retrieval.
2) For each version of the sub-experiment, the first phase focused on evaluating the retrieval process, and the second phase evaluated the generated answers.
3) Results were then compared metric-wise to assess whether the introduction of the Knowledge Graph improved the relevance of the retrieved context as well as the quality of the answers.

1) of K: information retrieval, k represents the number of top-ranked results considered during evaluation. Selecting $k = 3, 5, 10$ provides a balanced assessment of system performance, capturing both precision at lower ranks and overall retrieval effectiveness. These values were chosen based on the following considerations:

- **Consistency:** evaluation practices in information retrieval commonly use these k values, ensuring comparability with prior research.
- **Across Depths:** multiple k levels helps understand precision-recall trade-offs as more results are considered.

, evaluating at $k = 3, 5, 10$ offers a practical and well-rounded measure of retrieval quality.

D. Validation Metrics

This subsection describes the methodology used to evaluate the performance of our approach.

For the **retrieval module**, we evaluate performance using **Precision** and **Recall**, which measure the system's ability to extract relevant documents per query. For the **generative module**, we rely on the Ragas **LLM-Based Context Recall** metric, which assesses the relevance of generated answers by checking alignment with ground-truth answers. Given the brevity of most answers, this single metric was deemed sufficient, as it effectively captures the presence of correct answers within the generated text.

When evaluating a Retrieval-Augmented Generation (RAG) system, it is important to consider two distinct components. First, the **retrieval module**, which determines how effectively the system retrieves relevant documents that ground the generative component's answers. Second, the **generative module**, which assesses the quality of the answers generated based on the retrieved documents. The evaluation of these components provides a comprehensive understanding of the system's overall performance [25].

The evaluation employs three key metrics: **Precision**, **Recall**, and **LLM-Based Context Recall**, derived from the Regas [26] framework.

- **Precision**: Precision measures the proportion of true positive predictions among all positive predictions made by the system. It reflects the accuracy of the model in identifying relevant results.

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

- **Recall**: Recall evaluates the ability of the system to retrieve all relevant instances in the dataset. It is calculated as:

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

- **LLM-Based Context Recall**: This metric measures how well the retrieved contexts support the claims in the reference answer. It works by breaking down the reference into individual claims and checking if each claim can be attributed to the retrieved contexts. The metric simplifies the process by using the reference as a proxy for context, avoiding the need for manual annotation of reference contexts. Scores range from 0 to 1, with higher scores indicating better alignment between the retrieved contexts and reference claims. An LLM is used as the scorer (*LLM-as-a-Judge* [27]) .

$$\text{Context Recall} = \frac{|\text{GT Claims Attributable to Context}|}{|\text{Total Claims in GT}|}$$

For each iteration of the RAG system, the retrieved context, the answer, the question, and the ground-truth answer are passed to an LLM (in this case, the same LLM that powers the RAG system) to generate the evaluation result.

Example of High Context Recall:

- **Question:** What is the capital of France?
- **Response:** The capital of France is Paris.
- **Reference:** The capital of France is Paris.
- **Retrieved Contexts:**
  1) Paris is the capital city of France.
  2) France is a country in Europe.

In this case, all claims in the reference are fully supported by the retrieved contexts. Hence, the context recall score would be high (close to 1.0).

Example of Low Context Recall:

- **Question:** What is the capital of France?
- **Response:** The capital of France is Paris.
- **Reference:** The capital of France is Paris.
- **Retrieved Contexts:**
  1) The Eiffel Tower is in Paris.
  2) France is a popular tourist destination.

Here, the retrieved contexts fail to directly support the claim that Paris is the capital of France. Consequently, the context recall score would be low (close to 0.0).

E. Implementation Details

1) Language and Framework: We used *Python 3.12.8* as the programming language to implement the entire experiment.

2) Models Used: We used OpenAI's **GPT-4o** [28] for data cleaning tasks due to its superior ability to preserve the text's fidelity to the original content. Additionally, **GPT-4o-mini** [29] was used as the generative model for answering questions.

3) Knowledge Graph Framework: **Neo4j** [30], an open-source graph database, was employed to represent our knowledge graph. It autonomously handled the creation and structuring of relationships between nodes, streamlining the graph-building process.

4) Vector Database: For the vector database, we relied on the open-source database **Chroma-DB** [31], which natively implements the L2 distance as the similarity measure.

5) Embedding parameters: The embedding process was implemented locally using PyTorch [32] and the HuggingFace [33] library to deploy the models used, ensuring computational efficiency. Each chunk was tokenized with the following configuration:

Listing 1: Tokenization Parameters

```
encoded_input = tokenizer(input,
    return_tensors='pt',
    padding='max_length',
    truncation=True,
    max_length=512,
    pad_to_multiple_of=512
)
```

This configuration ensured compatibility with variable-length chunks and avoided the need for manual adjustments, making the process seamless and scalable.

V. Results

This section presents the evaluation results of the Retrieval-Augmented Generation (RAG) system, comparing the baseline model with the enhanced version incorporating triplets as context. The results are grouped by dataset—Google Natural Questions, BioASQ, and FinanceQA—and analyzed across three metrics: Precision, Recall, and LLM-Based Context Recall.

For each dataset, the performance of the retrieval module is evaluated using Precision and Recall, while the generative module is assessed using the LLM-Based Context Recall metric. Visualizations and comparisons are provided to highlight the impact of the triplet-enhanced approach on both the retrieval and generative components of the system. The results demonstrate how the inclusion of triplets as a context enhancer affects the relevance of retrieved documents and the quality of generated answers.

A. BioASQ

The evaluation of our proposed hybrid RAG system on the BioASQ dataset highlights significant improvements in both retrieval and generative performance when compared to the baseline system. The enhancements achieved through query enrichment with knowledge graphs are evident in the retrieval metrics precision, recall, and F1 score, as well as in the quality of the system-generated answers.

As observed in Figure 4 (a,b,c), we achieved an improvement across all three key metrics at all values of K, indicating that the number of relevant documents retrieved increased. This demonstrates that, on average, a larger quantity of relevant documents was retrieved across all runs.

Furthermore, Table V demonstrates that the *Enhanced* system retrieves more relevant context on average, as shown by the increase in LLM context recall in BioASQ (Table V first column). This improvement directly translates to better generative performance and a notable reduction in the number of queries with missing context, where no documents are retrieved (TAble V secondo column). Specifically, the number of queries with missing context is reduced by over 30%, confirming the *Enhanced* system's improved ability to provide sufficient context for generative tasks. These results emphasize the system's capability to address gaps in information retrieval, leading to more reliable and accurate responses.

To highlight the differences in question answering between the baseline and enhanced systems, we provide three illustrative examples. These examples demonstrate how the enhanced system successfully retrieves and utilizes relevant context to generate accurate and complete answers, whereas the baseline system struggles in certain scenarios.

Example 1: Response vs. No Response

Query: What type of extracolonic tumors does the PMS2 germline mutation cause?

textbfEnhanced Query: What type of extracolonic tumors does the PMS2 germline mutation cause?

Ground Truth: Heterozygous PMS2 mutation carriers were at small increased risk for colorectal and endometrial cancer

Enhanced System Response: Endometrial cancer.

Baseline System Response: NO DOCUMENT

Example 2: Precise and direct asnwer

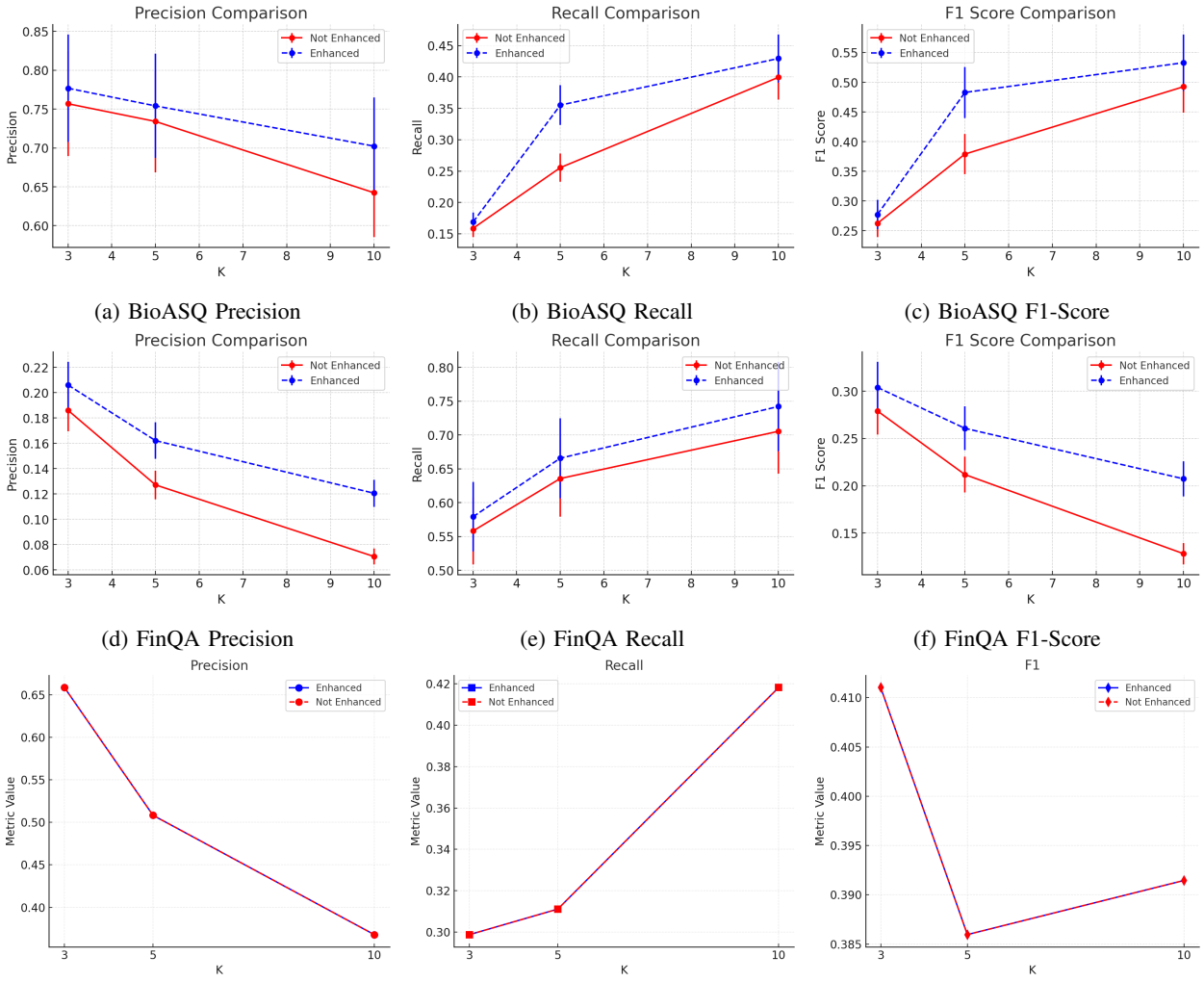
Query: Can valproate be used during pregnancy?

Enhanced Query: Can valproate be used during pregnancy? [SEP] valproate causes neurocognitive delay and deficits [SEP] valproate causes teratogenicity [SEP] valproate absence from list of drugs [SEP] valproate prescribed to treat migraine [SEP] pregnancy followed by in utero [SEP] pregnancy followed by postpartum [SEP] pregnancy followed by childbearing [SEP] pregnancy followed by childbearing age

Ground truth: No. Valproate is associated with high risk of malformations and should no be used during pregnancy.

Enhanced System Response: No, valproate is contraindicated during pregnancy due to its teratogenicity and potential for neurocognitive delays and deficits.

Baseline System Response: Valproate is a teratogen and should be approached with caution during pregnancy. Women of childbearing age taking valproate should be warned of its teratogenicity and advised to discuss treatment options with their physician.



(a) BioASQ Precision (b) BioASQ Recall (c) BioASQ F1-Score

(d) FinQA Precision (e) FinQA Recall (f) FinQA F1-Score

(g) Google NQ Precision (h) Google NQ Recall (i) Google NQ F1-Score

Fig. 4: Retrieval metrics across datasets: BioASQ, FinQA, and Google NQ. Rows correspond to datasets; columns represent Precision, Recall, and F1-Score, respectively. y-axis is scaled to highlight detailed variations. Error bars denote the standard error of the mean, calculated as $\text{SE} = \frac{\sigma}{\sqrt{n}}$, where σ is the standard deviation and n is the sample size. Notably, F1 consistently improves at $K = 5$, the parameter used for passing documents to the RAG model for answer generation.

Example 3: Failed Expansion - Irrelevant context injected

Query: What should you do if you find an individual that you suspect has overdosed?

Enhanced Query: What should you do if you find an individual that you suspect has overdosed? [SEP] individuals at risk of witnessing or experiencing opioid overdose [SEP] individuals carry mutant (GCN)(13) allele [SEP] individuals with varying levels of disease activity

Ground truth: Call emergency services. Try to wake the person up. Administer naloxone if you have it. Cardiopulmonary resuscitation.

Enhanced System Response: NO DOCUMENT

Discussion: These examples clearly illustrate the improvements achieved by the enhanced system. In Examples 1, the baseline system failed to retrieve any relevant documents, leading to no response. In contrast, the enhanced system successfully retrieved and leveraged the necessary context to provide concise and accurate answers. In Example 2, both systems provided responses, but the enhanced system demonstrated superior clarity and relevance by focusing directly on the core of the question, while the baseline system included excessive and less focused details. These results underline the enhanced system's ability to retrieve and utilize more relevant context, improving the overall quality of responses.

B. FinQA

The evaluation of the proposed *Enhanced* system demonstrates substantial improvements across precision, recall, and

TABLE V: Generative Metrics Across All Datasets. For each measure we can observe the difference in scores between the enhanced system and the baseline sysmte (Not Enhanced)

Dataset	LLM Context Recall		n. of Missing Context	
	Not Enhanced	Enhanced	Not Enhanced	Enhanced
BioASQ	0.71	0.79	12	5
FinQA	0.66	0.73	42	33
Google NQ	0.39	0.40	60	61

F1-score metrics compared to the *Not Enhanced* baseline. These improvements are particularly evident in the retrieval performance as K increases, where the *Enhanced* system consistently outperforms the baseline. The enhanced performance highlights the effectiveness of the proposed method in achieving better precision, ensuring more accurate retrieval of relevant items, while simultaneously improving recall, thereby increasing coverage. Furthermore, the higher F1-scores underscore the system's ability to maintain a balanced trade-off between precision and recall, showcasing the robustness of the enhancement approach across varying retrieval complexities. The evaluation has been performed over 130 question-answer pairs.

As observed in Figure 4 (d,c,f), we achieved an improvement across all three key metrics at all values of K, indicating that the number of relevant documents retrieved increased. This demonstrates that, on average, a larger quantity of relevant documents was retrieved across all runs.

To highlight the differences in question answering between the baseline and enhanced systems, we provide three illustrative examples. These examples demonstrate how the enhanced system successfully retrieves and utilizes relevant context to generate accurate and complete answers, whereas the baseline system struggles in certain scenarios.

Example 1: Response vs. No Response

Query: How does the insurance value of the shipment compare to its total value?

textbfEnhanced Query: How does the insurance value of the shipment compare to its total value? [SEP] Shipment needed_by June 28 [SEP] Total Transaction Value Labeled Under $926,888.00 marked under 'Cash Receipts Income 1001-0102' [SEP] Total Transaction Value Amount $926,888.00

Ground truth: The insurance value of $100.00 is significantly higher than the total value of the shipment, which is $3.51.

Enhanced System Response: The insurance value of the shipment is $100.00, which is significantly higher than its total value of $3.51.

Baseline System Response: NO DOCUMENT

Example 2: Answer v no Answer

Query: What is the address of the payee?

textbfEnhanced Query: What is the address of the payee? [SEP] Payee based_in New York, NY [SEP] Payee receives_payment_from R.J. Reynolds Tobacco Company [SEP] Payee receives_payment Hugh Benson Carter [SEP] Payee location New York, NY

Ground truth: The address of the payee is 930 Dakota Ave South, Huron, SD 57350.

Enhanced System Response: 930 Dakota Ave South, Huron, SD 57350

Baseline System Response: NO DOCUMENT

Example 3: Failed Expansion - No Triplet Injected

Query: What is the percentage of VAT applied to each charge on the invoice?

Ground truth: 20
Enhanced System Response: NO DOCUMENT

Discussion: In this case, due to the nature of the dataset, which contains responses in the form of precise numbers and/or pieces of text, the metrics that we primarily improved pertain to the ability to retrieve a piece of text, rather than enhancing the quality of the question itself, as in the case of the dataset above (*BioASQ*). This indicates that, overall, the number of hits in retrieving significant pieces of documents has increased.

C. Google NQ

The evaluation of our proposed hybrid RAG system on the Google Natural Questions dataset shows no improvement, except for a single question. The performance remains identical, likely because the provided context is shallow and lacks specificity, rendering further context enhancement ineffective as it is already saturated

As we can observe from the figures (Figure 4 g,h,i), there is a perfectly overlapping performance between the two techniques used.

Similarly, as shown in Table V, the generative metrics are nearly identical, with only one notable difference observed in a specific question.

To highlight the differences in question-answering performance between the baseline and enhanced systems, we provide three illustrative examples. These examples demonstrate how the enhanced system successfully retrieves and utilizes relevant context to generate accurate and complete answers, whereas the baseline system struggles in certain scenarios. However, when the context is already saturated, further enrichment can introduce noise, potentially degrading the performance of the retrieval system, as shown in [34] and [35]. general, when query expansion does not have a strong similarity with a chunk containing the correct response (as in Example 1, where an element of the answer is directly present in the expansion), the augmented context does not provide any advantage, as it fails to add meaningful information that would help the retriever prioritize one chunk over another. Since the chunks already share a common topic, the additional context does not improve retrieval. Another factor that may have degraded the performance of the enhanced system is the triplet injection step, which failed to provide additional useful context to the query. This issue likely arises due to the poor Named Entity Recognition (NER) performance of the LLM on the dataset corpus, which consists of much longer texts compared to the shorter documents in other datasets. The degradation in LLM performance over long contexts has been noted in several studies [36] [37]. Additionally, many missing-context questions showed no signs of enhancement, indicating that no entity extraction or entity matching occurred within the knowledge graph, a problem also confirmed in [38] paper presenting evaluation of GraphRAG system.

Example 1: Response vs. No Response

Query: "Where did The Beatles' final live performance take place?"

textbfEnhanced Query:where did the beatles final live performance take place [SEP] The Beatles recorded one more album Abbey Road [SEP] The Beatles historical context marked the end of an era [SEP] The Beatles lampshade on the roof of Apple Corps [SEP] The Beatles last song performed 'Get Back'

Ground truth: the roof of the headquarters of the band 's multimedia corporation Apple Corps at 3 Savile Row

Enhanced System Response: "The Beatles' final live performance took place on the rooftop of 3 Savile Row, London."

Baseline System Response: NO DOCUMENT

Example 2: Correct v Uncorrect answer

Query: "who holds the most women's wimbledon titles"

textbfEnhanced Query:who holds the most women's wimbledon titles [SEP] Wimbledon Ladies' Singles prize money £2,000,000 (2016) [SEP] Wimbledon Ladies' Singles surface Grass [SEP] Wimbledon Ladies' Singles established 1884 [SEP] Wimbledon Ladies' Singles venue AELTC

Ground truth: Martina Navratilova

Enhanced System Response: "Martina Navratilova holds the record."

Baseline System Response: "Helen Wills Moody holds the most titles."

Example 3: Failed expansion - Irrelevant context

Query: "when does life is strange before the storm part 2"

Enhanced Query:when does life is strange before the storm part 2 [SEP] Life Is Strange: Before the Storm platform Microsoft Windows, PlayStation 4, Xbox One [SEP] Life Is Strange: Before the Storm engine Unity [SEP] Life Is Strange: Before the Storm composed_by Daughter [SEP] Life Is Strange: Before the Storm written_by Zak Garriss [SEP] Episode 2 released_on 19 October 2017

Ground truth: October 2017

Enhanced System Response: NO DOCUMENT

We can still observe two examples where the document chunks retrieved by the enhanced query were more effective in locating the correct part of the text compared to a generic chunk. This justifies the slightly higher metrics observed in the generative component.

D. State Of the Art Comparison

this section, we will compare the results we obtained with those from comparable SOTA systems that are publicly available. For each dataset, we will compare a different system, highlighting the pros and cons of the system proposed in this work with the other available approaches. Results are presentend in table VI

1) Natural Questions: the Google NQ dataset, we will analyze the paper from [38], which proposes a comparison between RAG and GraphRAG using several techniques. In particular, we will focus on the results they obtained with the

Dataset	Google NQ			BioASQ			FinQA		
	Recall	Precision	F1	Recall	Precision	F1	Recall	Precision	F1
Our Model	0.34	0.51	0.40	0.31	0.74	0.45	0.66	0.16	0.25
GraphRAG (triplet)	0.31	0.22	0.37						
Medical RAG				0.78	0.82	0.80			
FinQA							0.60	0.59	0.60

TABLE VI: Performance comparison of different models across datasets.

GraphRAG method using triplets, which relies solely on the triplets context for the LLM to respond. We will compare our model (GPT-4o-mini) with their experiment using LLaMA-3.1 70B, as it is a model with similar performance (benchmarks from [39] and [40]). The main difference lies in the fact that they use only triplets as the basis of their context, exposing them to issues such as missing relations and the absence of textual context. In contrast, we rely on triplets to enhance text retrieval, thereby reducing this risk. In our evaluation, our model achieves an F1-score of 0.40, outperforming GraphRAG (0.37). A notable improvement is observed in precision, where our model achieves 0.51 compared to GraphRAG's 0.22. This confirms that our approach retrieves more relevant documents while reducing noise, mitigating the primary weakness of GraphRAG, which lacks contextual textual information. However, recall remains comparable (0.34 vs. 0.31), indicating that both methods retrieve a similar breadth of information. These results suggest that our triplet-enhanced retrieval strategy provides a more accurate yet similarly comprehensive contextualization for the LLM.

2) : the BioASQ dataset, we will compare another RAG system from [41], which proposes a specific model called Medical RAG. This model is tailored for medical databases to enhance performance on medical QA tasks, including BioASQ. Their method, compared to ours, is highly domain-specific and utilizes extensive knowledge from a large medical corpus that extends far beyond the BioASQ document collection, seamlessly integrating different sources. This model outperforms ours, suggesting that access to a larger amount of information can effectively improve results in domain-related tasks. This insight indicates that, to enhance performance, a viable strategy could be integrating more information into the knowledge graph or enabling online research to further enrich the context. Our model achieves an F1-score of 0.45, significantly lower than Medical RAG's 0.80. The primary gap lies in recall (0.31 vs. 0.78), indicating that our retrieval system does not capture as much relevant information as Medical RAG, which benefits from extensive external medical knowledge. However, our model demonstrates a strong precision of 0.74, suggesting that while it retrieves fewer documents, they are highly relevant. These results indicate that retrieval in domain-specific settings requires broader external knowledge integration, as evidenced by Medical RAG's superior recall.

3) : FinQA, there is no publicly available paper that specifically evaluates retrieval performance; rather, most papers focus on the accuracy of the dataset's answers. However, we can consider the findings presented by [42], which evaluate the effectiveness of general-purpose LLMs, such as GPT-4, in analyzing financial documents and providing valid responses. Their study does not focus on a RAG system but instead examines a set of different tasks, including response generation, Named Entity Recognition (NER), Information Extraction (IE), and Relation Extraction (RE). Our model achieves an F1-score of 0.25, with a high recall (0.66) but low precision (0.16). This indicates that our system retrieves a broad set of financial documents but struggles with filtering out irrelevant information. In contrast, FinQA reports an F1-score of approximately 0.60, suggesting that models fine-tuned specifically for financial document retrieval are more effective at narrowing down relevant contexts. The observed high recall in our model suggests that it has strong retrieval breadth, but the lack of precision implies that improvements in relevance filtering, such as entity linking and relation extraction, could significantly enhance performance.

VI. Discussion

The results of our study underscore the effectiveness of integrating knowledge graphs into RAG systems to address challenges inherent in domain-specific retrieval tasks as we can see in table VI. For instance, in the BioASQ and FinQA datasets, the incorporation of enriched context significantly increased retrieval metrics and reduced the number of queries with missing context. Furthermore, the enhanced system consistently generated more accurate and contextually relevant answers, as illustrated by the examples provided. These improvements highlight the potential of query enrichment using knowledge graphs to tackle limitations in embedding precision, especially in scenarios requiring nuanced understanding.

However, the results on the Google NQ dataset indicate that the benefits of query enrichment may diminish when the provided context is already saturated, as further enrichment could introduce noise. This highlights an important limitation of the proposed approach and suggests that its applicability may vary depending on the dataset characteristics and the depth of the initial context.

The strengths and limitations of the hybrid approach provide a roadmap for future optimizations. While query enrichment effectively addresses gaps in domain-specific tasks, balancing its trade-offs and understanding the scope of its application remain crucial for further development.

Hereafter a summarizing table, showing the average improvements in the considered metrics for each dataset

A. Answer to Research Questions

1) ***How can RAG's retrieval power be enhanced using a hybrid DPR model?:*** In this work, we proposed a hybrid

TABLE VII: Summary of metrics across all datasets. For ease of comparison, the Mean Precision and Mean Recall at k considered (3,5,10)

have been reported. across all k provides a single representative value, smoothing out variations and offering a more stable measure of retrieval effectiveness.

Dataset	LLM Context Recall		n. of Missing Context		Mean Recall@K (MR@K)		Mean Precision@K (MP@K)	
	Baseline	Enhanced	Baseline	Enhanced	Baseline	Enhanced	Baseline	Enhanced
BioASQ	0.71	0.79	12	5	0.27	0.31	0.71	0.74
FinQA	0.66	0.73	42	33	0.63	0.66	0.13	0.16
Google NQ	0.39	0.40	60	61	0.34	0.34	0.51	0.51

approach to enhance Retrieval-Augmented Generation (RAG) systems by leveraging query enrichment through knowledge graphs. By integrating structured knowledge in the form of triplets, the system bridges the gap between general-purpose embeddings and the specific requirements of domain-specific contexts. The approach improved retrieval metrics such as precision and recall, as demonstrated across datasets like BioASQ and FinQA, with significant increases in retrieval performance and the relevance of retrieved context. This hybrid model effectively enhances the retrieval power of RAG by enriching queries with domain-specific knowledge, enabling better alignment with the intended tasks.

*2) **How does the integration of a hybrid DPR improve RAG output quality on domain-specific knowledge without needing to fine-tune the model on specific downstream tasks?**:* The integration of a hybrid DPR improves RAG output quality by enhancing context relevance without the need for extensive fine-tuning. The enriched context provided by knowledge graphs ensures more accurate and domain-relevant retrievals, which directly impacts the quality of generated answers. This is particularly evident in domain-specific datasets like BioASQ and FinQA, where the enhanced system demonstrated a reduction in queries with missing context and generated more precise and contextually aligned answers. By avoiding the need for model fine-tuning on specific downstream tasks, this approach provides a scalable and adaptable solution for improving RAG systems in diverse scenarios.

*3) **How can we identify the strengths and limitations of this hybrid approach?**:* The strengths and limitations of this hybrid approach were identified through evaluation across datasets with varying domain specificity. For datasets like BioASQ and FinQA, the method demonstrated clear improvements in retrieval and generative performance, as reflected in metrics like precision, recall, and LLM Context Recall. However, the Google NQ dataset highlighted a key limitation: when the initial context is already saturated, further enrichment can introduce noise, potentially degrading performance. This emphasizes the importance of balancing query enrichment and retrieval noise and suggests that the approach's effectiveness depends on the characteristics of the dataset and the depth of the initial context. These insights provide a roadmap for optimizing the methodology in future research.

B. Limitations

As we can see, the method is fundamentally robust, but its practicality can sometimes be questioned due to its expense and complexity, both in monetary and computational terms. This includes the increased cost of leveraging a large language model (LLM) for Named Entity Recognition (NER) extraction and the effort required to set up and maintain a knowledge graph. These factors become especially problematic when dealing with corpora that lack deep domain specificity. In such cases, the process of query enrichment may fail to improve performance and could even degrade it by introducing additional noise into the context, thereby reducing the overall effectiveness of the system.

Another notable limitation of this methodology lies in the selection of the LLM itself. More advanced or powerful LLMs could potentially yield better results in both data cleaning and NER extraction, enabling the generation of clearer and more consistent outputs. The choice of a less capable model may hinder the process, limiting its ability to fully capture and utilize the nuances of the data. As a result, the overall effectiveness and reliability of the approach could suffer, especially in scenarios where high precision and domain-specific expertise are required.

Also, only a small yet significant fraction of the available datasets has been used, simplifying the evaluation of performance. While this ensures a manageable scope for testing, it limits the generalizability of the results across a broader range of datasets and real-world scenarios.

C. Future Work

Future research could explore replacing transformer-based models with newer technologies like *modern BERT* [43], which enhances text and code embeddings, expands context windows, and reduces the need for excessive text chunking. Compact, high-performing LLMs could also be leveraged to improve scalability and cost efficiency, particularly for resource-constrained applications.

Additionally, further investigation is needed to optimize query enrichment techniques to minimize noise and improve retrieval performance, especially in contexts where excessive enrichment leads to saturation. Exploring advanced model architectures, such as more efficient text embedding methods and adaptable language models, could further enhance the robustness of hybrid RAG systems.

By addressing these challenges, future research can improve the balance between enrichment and retrieval quality, extend applicability to diverse datasets, and continue advancing the state of the art in information retrieval and generation.

VII. Conclusion

In this work, we proposed a hybrid approach to enhance Retrieval-Augmented Generation (RAG) systems by leveraging query enrichment through knowledge graphs. By integrating structured knowledge in the form of triplets, our system bridges the gap between general-purpose embeddings and the specific requirements of domain-specific contexts. The approach was evaluated across three datasets—Google NQ, FinQA, and BioASQ—spanning open- and closed-domain tasks, and demonstrated significant improvements in retrieval precision, recall, and the coherence of system-generated answers, particularly in specialized domains like finance and biomedicine.

The results underscore the scalability of this approach to other domain-specific tasks, such as legal and scientific document retrieval, and highlight its adaptability without requiring fine-tuning on specific downstream tasks. However, we also identified that query enrichment may introduce noise in scenarios where the context is already saturated, limiting the applicability of this approach in certain cases.

References

[1] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel *et al.*, "Retrieval-augmented generation for knowledge-intensive nlp tasks," *Advances in Neural Information Processing Systems*, vol. 33, pp. 9459–9474, 2020.

[2] S. Barnett, S. Kurniawan, S. Thudumu, Z. Brannelly, and M. Abdelrazek, "Seven failure points when engineering a retrieval augmented generation system," in *Proceedings of the IEEE/ACM 3rd International Conference on AI Engineering-Software Engineering for AI*, 2024, pp. 194–199.

[3] B. Jin, J. Yoon, J. Han, and S. O. Arik, "Long-context llms meet rag: Overcoming challenges for long inputs in rag," *arXiv preprint arXiv:2410.05983*, 2024.

[4] L. Gao, X. Ma, J. Lin, and J. Callan, "Precise zero-shot dense retrieval without relevance labels," *arXiv preprint arXiv:2212.10496*, 2022.

[5] Y. Liu, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.

[6] A. Neelakantan, T. Xu, R. Puri, A. Radford, J. M. Han, J. Tworek, Q. Yuan, N. Tezak, J. W. Kim, C. Hallacy *et al.*, "Text and code embeddings by contrastive pre-training," *arXiv preprint arXiv:2201.10005*, 2022.

[7] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *Journal of machine learning research*, vol. 21, no. 140, pp. 1–67, 2020.

[8] J. Devlin, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[9] S. Arya, D. M. Mount, N. S. Netanyahu, R. Silverman, and A. Y. Wu, "An optimal algorithm for approximate nearest neighbor searching fixed dimensions," *Journal of the ACM (JACM)*, vol. 45, no. 6, pp. 891–923, 1998.

[10] S. Robertson and H. Zaragoza, "The probabilistic relevance framework: Bm25 and beyond," *Found. Trends Inf. Retr.*, vol. 3, no. 4, p. 333–389, apr 2009. [Online]. Available: https://doi.org/10.1561/1500000019

[11] A. Hogan, E. Blomqvist, M. Cochez, C. d'Amato, G. D. Melo, C. Gutierrez, S. Kirrane, J. E. L. Gayo, R. Navigli, S. Neumaier *et al.*, "Knowledge graphs," *ACM Computing Surveys (Csur)*, vol. 54, no. 4, pp. 1–37, 2021.

[12] S. Siriwardhana, R. Weerasekera, E. Wen, and S. Nanayakkara, "Fine-tune the entire rag architecture (including dpr retriever) for question-answering," *arXiv preprint arXiv:2106.11517*, 2021.

[13] Y. Wang, N. Lipka, R. A. Rossi, A. Siu, R. Zhang, and T. Derr, "Knowledge graph prompting for multi-document question answering," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 17, 2024, pp. 19 206–19 214.

[14] Z. Xu, M. J. Cruz, M. Guevara, T. Wang, M. Deshpande, X. Wang, and Z. Li, "Retrieval-augmented generation with knowledge graphs for customer service question answering," *arXiv preprint arXiv:2404.17723*, 2024.

[15] I. D. P. R. Retrieving, "Retrieval-augmented generation: Is dense passage retrieval retrieving?"

[16] Y. Qu, Y. Ding, J. Liu, K. Liu, R. Ren, W. X. Zhao, D. Dong, H. Wu, and H. Wang, "Rocketqa: An optimized training approach to dense passage retrieval for open-domain question answering," *arXiv preprint arXiv:2010.08191*, 2020.

[17] K. Sawarkar, A. Mangal, and S. R. Solanki, "Blended rag: Improving rag (retriever-augmented generation) accuracy with semantic search and hybrid query-based retrievers," *arXiv preprint arXiv:2404.07220*, 2024.

[18] F. Cuconasu, G. Trappolini, F. Siciliano, S. Filice, C. Campagnano, Y. Maarek, N. Tonellotto, and F. Silvestri, "The power of noise: Redefining retrieval for rag systems," *arXiv preprint arXiv:2401.14887*, 2024.

[19] H. Zeng, Z. Yue, Q. Jiang, and D. Wang, "Federated recommendation via hybrid retrieval augmented generation," *arXiv preprint arXiv:2403.04256*, 2024.

[20] H. Face, "all-minilm-l6-v2," https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2, 2024, accessed: 2024-01-31.

[21] W. Wang, F. Wei, L. Dong, H. Bao, N. Yang, and M. Zhou, "Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers," *Advances in Neural Information Processing Systems*, vol. 33, pp. 5776–5788, 2020.

[22] T. Kwiatkowski, J. Palomaki, O. Redfield, M. Collins, A. Parikh, C. Alberti, D. Epstein, I. Polosukhin, M. Kelcey, J. Devlin, K. Lee, K. N. Toutanova, L. Jones, M.-W. Chang, A. Dai, J. Uszkoreit, Q. Le, and S. Petrov, "Natural questions: a benchmark for question answering research," *Transactions of the Association of Computational Linguistics*, 2019.

[23] Z. Chen, W. Chen, C. Smiley, S. Shah, I. Borova, D. Langdon, R. Moussa, M. Beane, T. Huang, B. R. Routledge, and W. Y. Wang, "Finqa: A dataset of numerical reasoning over financial data," *CoRR*, vol. abs/2109.00122, 2021. [Online]. Available: https://arxiv.org/abs/2109.00122

[24] G. Tsatsaronis, G. Balikas, P. Malakasiotis, I. Partalas, M. Zschunke, M. R. Alvers, D. Weissenborn, A. Krithara, S. Petridis, D. Polychronopoulos *et al.*, "An overview of the bioasq large-scale biomedical semantic indexing and question answering competition," *BMC bioinformatics*, vol. 16, pp. 1–28, 2015.

[25] H. Yu, A. Gan, K. Zhang, S. Tong, Q. Liu, and Z. Liu, "Evaluation of retrieval-augmented generation: A survey," *arXiv preprint arXiv:2405.07437*, 2024.

[26] S. Es, J. James, L. Espinosa-Anke, and S. Schockaert, "Ragas: Automated evaluation of retrieval augmented generation," *arXiv preprint arXiv:2309.15217*, 2023.

[27] J. Gu, X. Jiang, Z. Shi, H. Tan, X. Zhai, C. Xu, W. Li, Y. Shen, S. Ma, H. Liu *et al.*, "A survey on llm-as-a-judge," *arXiv preprint arXiv:2411.15594*, 2024.

[28] A. Hurst, A. Lerer, A. P. Goucher, A. Perelman, A. Ramesh, A. Clark, A. Ostrow, A. Welihinda, A. Hayes, A. Radford *et al.*, "Gpt-4o system card," *arXiv preprint arXiv:2410.21276*, 2024.

[29] OpenAI, "Gpt-4o mini: Advancing cost-efficient intelligence," https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/, 2024, accessed: 2024-01-31.

[30] Neo4j, Inc., "Neo4j - the graph database platform," 2024, accessed: Jan 31, 2025. [Online]. Available: https://neo4j.com/

[31] Chroma. (2024) Chroma: The ai-native open-source vector database. [Online]. Available: https://www.trychroma.com/

[32] PyTorch. (n.d.) Pytorch documentation. Accessed: 2024-07-28. [Online]. Available: https://pytorch.org/docs/stable/index.html

[33] H. Face. (n.d.) Transformers documentation. Accessed: 2024-07-28. [Online]. Available: https://huggingface.co/docs/transformers/it/index

[34] H. K. Azad and A. Deepak, "Query expansion techniques for information retrieval: a survey," *Information Processing & Management*, vol. 56, no. 5, pp. 1698–1735, 2019.

[35] O. Weller, K. Lo, D. Wadden, D. Lawrie, B. Van Durme, A. Cohan, and L. Soldaini, "When do generative query and document expansions fail? a comprehensive study across methods, retrievers, and datasets," *arXiv preprint arXiv:2309.08541*, 2023.

[36] N. F. Liu, K. Lin, J. Hewitt, A. Paranjape, M. Bevilacqua, F. Petroni, and P. Liang, "Lost in the middle: How language models use long contexts,"

Transactions of the Association for Computational Linguistics, vol. 12, pp. 157–173, 2024.

[37] Y. Kuratov, A. Bulatov, P. Anokhin, I. Rodkin, D. I. Sorokin, A. Sorokin, and M. Burtsev, "Babilong: Testing the limits of llms with long context reasoning-in-a-haystack," in *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024.

[38] H. Han, H. Shomer, Y. Wang, Y. Lei, K. Guo, Z. Hua, B. Long, H. Liu, and J. Tang, "Rag vs. graphrag: A systematic evaluation and key insights," *arXiv preprint arXiv:2502.11371*, 2025.

[39] A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Yang, A. Fan *et al.*, "The llama 3 herd of models," *arXiv preprint arXiv:2407.21783*, 2024.

[40] OpenAI, "Gpt-4o mini: Advancing cost-efficient intelligence," 2024, accessed: 2025-02-25. [Online]. Available: https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/

[41] N. T. Ngo, C. Van Nguyen, F. Dernoncourt, and T. H. Nguyen, "Comprehensive and practical evaluation of retrieval-augmented generation systems for medical question answering," *arXiv preprint arXiv:2411.09213*, 2024.

[42] X. Li, S. Chan, X. Zhu, Y. Pei, Z. Ma, X. Liu, and S. Shah, "Are chatgpt and gpt-4 general-purpose solvers for financial text analytics? a study on several typical tasks," *arXiv preprint arXiv:2305.05862*, 2023.

[43] B. Warner, A. Chaffin, B. Clavié, O. Weller, O. Hallström, S. Taghadouini, A. Gallagher, R. Biswas, F. Ladhak, T. Aarsen *et al.*, "Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference," *arXiv preprint arXiv:2412.13663*, 2024.

Generative AI Statement

During the preparation of this work, I used ChatGPT (in particular Gpt-4o) to proofread and refactor text structure, help structuring the LaTeX markup of this document and generate graphs, starting **exclusively** from my own productions and data. I would like to explicitly emphasize that no content has been generated entirely from scratch using generative AI. After using this tool/service, I thoroughly reviewed and edited the content as needed, taking full responsibility for the final outcome.