# Improving the Fact-Centric Knowledge Web with GPT-Based Fact Re-Ranking

1st Murun Enkhtaivan
*Department of Computer Science*
*San Jose State University*
San Jose, CA
murun.enkhtaivan@sjsu.edu

***Abstract*—Large Language Models (LLMs) are robust but prone to hallucination. The Fact Centric Knowledge Web (FCWB), authored by Sinha and Shiramatsu, was proposed as a retrieval augmented generation (RAG) system with a knowledge base of a web of interconnected facts and keywords [1]. Their system is also composed of a Fact Finding Retrieval Agent, which finds relevant information from the Knowledge Web and iteratively generates a sub-query if necessary to retrieve more relevant information from the Knowledge Web and to essentially generate a better final answer to the user's query. While effective in many cases, the baseline FCKW system sometimes struggles with complex multi-hop questions that require more complex reasoning. In this paper, we propose enhancing the second component, Fact Finding Retrieval Agent: a GPT-based Fact Re-Ranker system that semantically ranks the retrieved facts before answer generation. Our method detects the user's question type and tailors the re-ranking of the facts accordingly, instead of just using the retrieved top-k facts of the baseline approach. With enhancement on the second component, our method improves the evaluation metric, Approximate Match accuracy, from 65.7% to 85% on test questions. Through case studies, we show how the re-ranking system helps avoid hallucinations, enables light reasoning, and improves the precision of the final generated answers from LLM.**

***Index Terms*—Fact-Centric Knowledge Web, Retrieval-Augmented Generation, LLMs, Semantic Re-ranking, Hallucination Mitigation, Information Retrieval**

## I. Introduction

LLMs have revolutionized information access, but they remain expensive to retrain on custom datasets and are vulnerable to hallucination, providing inaccurate facts. Also, LLMs have been trained up until a specific date, such as September 2023 [6]. Therefore, RAG is widely used as a source of knowledge for a company or a personal custom dataset. With RAG, we do not need to retrain the whole LLM, but we can still retrieve information from the new custom dataset. The Fact Centric Knowledge Web (FCKW) was introduced to mitigate hallucination and answer multi-hop questions by storing keywords and facts as a knowledge graph[1]. This baseline study is divided into two main parts: building a Knowledge Web and retrieving from the Knowledge Web to generate a final answer. Instead of using traditional ordered triplets to store the knowledge base, the first component uses a graph representation and links keywords with facts. While this baseline approach of building a knowledge web helps with grounding, we notice some issues with the second component of FCKW. The second component is a Fact Finding Retrieval agent. In implementing and evaluating the beeline system, we observed that the current baseline system sometimes answers incorrectly or can not answer directly when responding to a multi-hop user query that requires reasoning. When the system retrieves facts from the built Knowledge Web, it selects facts linked to keywords in the query. Then, it ranks those facts based on the cosine similarity of the query, the fact, and the degree of the fact. Using the fact ranking score from the previous step, the system selects the top-k facts from the list and uses them to generate a final answer to the query. Instead of using top-k facts, we propose a change: reranking the retrieved candidate facts using ChatGPT-3.5. We selectively choose the most answer-relevant facts by instructing the LLM to consider the question type, such as inference, numerical reasoning, or date logic. This module improves the factual consistency and answer precision, avoids hallucination, and supports reasoning without changing the already built knowledge web. This way, we are only altering the second component of the FCKW system, yet improving the final answer.

## II. Structure of Knowledge Web

Our approach builds on the architecture of the original study. The knowledge web is constructed by decomposing the training documents into atomic and independent facts [1]. Figure 1 from the original study[1] shows an example of a paragraph and decomposed facts via LLM.

The keywords from the facts are also extracted. They link the facts and related keywords together as nodes to use them later for querying purposes. This graph representation is stored in a graph database as part of the Knowledge Web. Facts are stored in a Neo4j graph, where edges connect them to corresponding keyword nodes[4]. Figure 2 shows this representation in the Graph database.

The next part of building the knowledge web is embedding the keywords using OpenAI's text-embedding-ada-002 [2] model to store as vectors in the Pinecone database, which is later used for semantic vector search [3]. Retrieval is done by first finding keywords that match the query embedding utilizing semantic vector search. Thus, we find keywords from our vector database that are semantically the same as the

| Original Documents | Generated Facts |
|---|---|
| Green is the fourth studio album by British progressive rock musician Steve Hillage. Written in spring 1977 at the same time as his previous album, the funk-inflected "Motivation Radio" (1977), "Green" was originally going to be released as "The Green Album" as a companion to "The Red Album" (the originally intended name for "Motivation Radio"). However, this plan was dropped and after a US tour in late 1977, "Green" was recorded alone, primarily in Dorking, Surrey, and in London. | 1. Green is a fourth studio album.<br>2. Green is by Steve Hillage.<br>3. Steve Hillage is a British musician.<br>4. Green is in the progressive rock genre.<br>5. Green was written in spring 1977.<br>6. Motivation Radio is a funk-inflected album.<br>7. Motivation Radio was released in 1977.<br>8. Green was originally going to be called The Green Album.<br>9. Motivation Radio was originally going to be called The Red Album.<br>10. The Green Album and The Red Album were intended to be companion albums.<br>11. Green was recorded after a US tour in late 1977.<br>12. Green was recorded in Dorking, Surrey, and London. |

Fig. 1. Document and Decomposed Facts

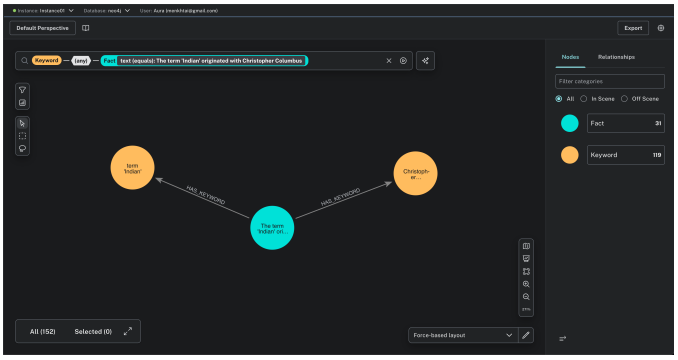

Fig. 2. Graph Representation of Keyword and Fact

query keywords. After identifying keywords from the vector database, we select related facts from our graph database.

We retained the exact structure of building a knowledge web of facts and keywords. Even though we attempted to change keyword indexing, it was expensive to change how we create the knowledge base and test it multiple times for improvements. Our main contribution was replacing the default ranking system with a semantic reranker. Regarding time and resources, focusing on improving the second component was more efficient. Changing the ranking system was faster and less expensive than changing the knowledge web structure.

## III. System Design

In the original study, the fact retrieval agent first finds candidate keywords from the database and works with facts relating to those keywords. The baseline system ranks these facts by assigning a Fact Ranking Score to select the top-k facts.

### A. Baseline Approach and Discussion

The fact ranking score is calculated with the function below:

$$\text{Score}(fact) = \text{cosSim}(\text{query}, fact) + \text{degree}(fact)$$

Essentially, the fact ranking score calculates the cosine similarity between the user query and the candidate fact and a number of extracted keywords from the query related to the fact. While this fact ranking system works, there are some limitations. For example, cosine similarity only compares the embeddings. Therefore, the cosine similarity may not reflect whether the fact is relevant to answer the user's question. The extracted facts relate to the query, but they might not be precise or pertinent to answer complex questions. Also, a degree(fact) might show some bias toward generic facts, as it gives more importance if the fact is more popular and includes more keywords that show in the query. We want to find facts highly relevant to the query or that could help LLM answer the question more accurately as a final answer generation.

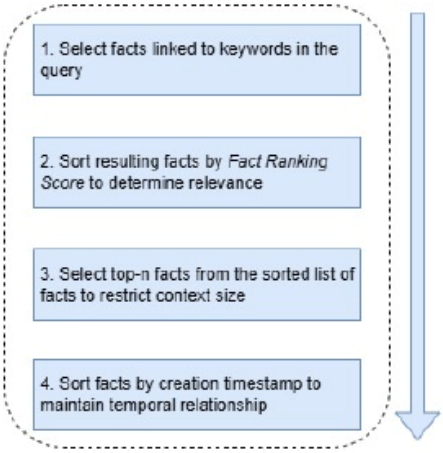


Fig. 3. Overview of Fact Retrieval of Current Baseline Method

### B. Examples with Baseline Method

Our evaluation identifies several questions that the system struggled to answer correctly. Below are three representative cases:

**1) Question requiring calculations**

*Question:* How old was Yale Law School when it celebrated its centennial?

*Correct Answer:* 100 years

*System Answer:* Yale Law School celebrated its centennial in 1943.

The baseline system correctly identified the year of the centennial celebration, but failed to infer that a centennial marks 100 years. This highlights a limitation in the system's ability to perform simple reasoning over numerical values.

*Facts retrieved:*

- In 1887, Yale College was renamed Yale University.
- Yale Law School was established in 1843.
- Yale School of Medicine was established in 1810.
- Yale Graduate School of Arts and Sciences was established in 1847.

- Yale School of Public Health was established in 1915.

**2) Question requiring specific information**

*Question:* Which Ivy League school educated the future Zappos investor Tony Hsieh?
*Correct Answer:* Harvard
*System Answer:* Yale University educated the future Zappos investor Tony Hsieh.

This example demonstrates a hallucination error. The system retrieved facts about Tony Hsieh and about Yale University, but incorrectly inferred a connection.

*Facts retrieved:*

- Zappos CEO Tony Hsieh co-founded the company in March 1996.
- Tony Hsieh bought Gold Spike from The Siegel Group.
- Gold Spike was owned by entrepreneur Tony Hsieh and his Downtown Project.
- Yale Graduate School of Arts and Sciences was established in 1847.
- Yale Law School was established in 1843.

**3) Question requiring date calculations**

*Question:* How old was Madonna when she moved to New York in 1977?
*Correct Answer:* 19 years old
*System Answer:* Based on the provided facts, Madonna moved to New York City in 1977 to pursue a career in modern dance. There is no specific information about Madonna's age at the time of her move, so we cannot determine her exact age in 1977.

Although the system retrieved partial information, it failed to identify her birth year to complete the date-based reasoning.

*Facts retrieved:*

- Madonna moved to New York City in 1977 to pursue a career in modern dance.
- Madonna was born in Bay City, Michigan.

Based on these examples, we observed that the system failed on queries requiring inference, calculation, or precise entity matching. The facts were present in the database but the baseline method failed to prioritize or combine them effectively.

## IV. System Design with Fact Re-Ranker Module

### A. Fact Re-Ranker Logic

To address these issues, we introduced the Fact Re-Ranker Module with enhancements, which identifies the query type and uses LLM to re-rank the facts after retrieving facts from the knowledge web. Our re-ranking module uses LLM to extract the most relevant facts from the retrieved facts we obtained from the Knowledge Web. The new re-ranking module works in these orders:

1) Input: A question from the user and the candidate facts from Knowledge Web.
2) Fact-Reranker detects the query type. Our fact reranker determines if the query needs special handling based on keywords of the query, such as "how old", "why", "reason", or "cause". If the question contains these kinds of keywords, the fact reranker classifies the question as an inference-type or calculation-type question.

```python
requires_calculation = any(keyword in query.lower() for keyword in
                           ["how old", "how many", "age", "years", "century", "centuries",
                            "percentage", "fraction", "ratio", "when", "date"])

requires_inference = any(keyword in query.lower() for keyword in
                         ["why", "how", "reason", "cause", "effect", "impact", "influence",
                          "relationship", "connection", "link", "association"])
```

Fig. 4. Question Type Detection Logic

3) Fact-Ranker creates special prompting based on the type of question we just detected. For example, for the calculation-type question, the prompt instructs that the "question might require calculating dates, ages, periods, or numerical values. Look for facts that provide dates, years, or periods".
4) We change the prompt for inference-type questions and say, "This question may require making inferences or connections between facts. Look for facts that directly address the relationship asked about". Therefore, now the system provides more explicit instructions to LLM when asked about ranking the facts.
5) Fact-Reranker has fallback options. If the LLM can not detect the most relevant facts to the system, we just use the top-3 retrieved initially facts.

The figure below represents the overall flow with the new fact re-ranker module.

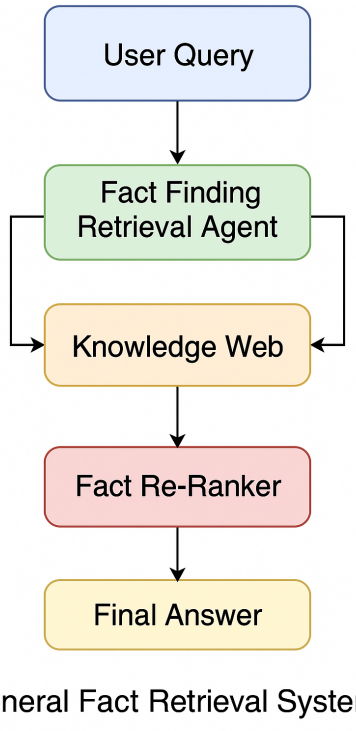


Fig. 5. An overview of RAG system with Fact Re-Ranking Module

### B. Answer Comparison

With a new Fact Re-Ranking approach, previously tested questions give these answers:

1. "How old was Yale Law School when it celebrated its centennial?"
Correct Answer: "100 years"
Baseline Answer: "Yale Law School celebrated its centennial in 1943."
New Answer: "Yale Law School celebrated its centennial in 1943, making it 100 years old at the time of the celebration."

2. "Which Ivy League school educated the future Zappos investor Tony Hsieh?"
Correct Answer: "Harvard"
Baseline Answer: "Yale University educated the future Zappos investor Tony Hsieh."
New Answer: "The facts provided do not contain information about which Ivy League school educated Tony Hsieh."

3. "How old was Madonna when she moved to New York in 1977?"
Correct Answer: "19 years old"
System Answer: "Based on the provided facts, Madonna moved to New York City in 1977 to pursue a career in modern dance. There is no specific information about Madonna's age at the time of her move, so we cannot determine her exact age in 1977 based on these facts."
New Answer: "Madonna was born in 1958, so she would have been 19 years old when she moved to New York City in 1977."

Based on test questions, we see that questions 1 and 3 are now an approximate match to the correct answer. Question 2 did not provide the correct answer, but with the new fact re-ranking module, the system no longer gives a hallucinated answer. Instead, it states that it lacks sufficient facts to respond — a safer behavior. Questions 1 and 3 were answered correctly due to our query type detection logic and enhanced fact re-ranking system with prompt customization.

## V. Experimental Setup and Results

We use a subset of the MuSiQue dataset, including 20 multi-hop factual questions and around 45 supporting passages[5]. The subset of the dataset covers various topics. We implemented the baseline approach following the original study. We also used the same databases as the original dataset to replicate the baseline approach. For keyword indexing, we used Pinecone to store the keywords and use it later for semantic search matching. To store facts and keywords, we use the Neo4J Graph database. Table 1 represents more detailed information about the training dataset.

We also use the same evaluation metric as the original study, Approximate Match. We manually judged the answer and ground truth to evaluate if the system's answer is an approximate match, following the same approach as the original study. Each question is manually labeled. With the baseline approach, our Approximate Match was around 60%. With the new fact re-ranking module, Approximate Match increased to around 80% as the system answered complex reasoning questions correctly.

TABLE I

| Component | Description |
|---|---|
| **Questions** | 20 multi-hop questions from the MuSiQue dataset. |
| **Passages** | 47 paragraphs associated with the questions. |
| **Keyword Nodes** | 783 keywords extracted from facts for indexing. |
| **Fact Nodes** | 300 context-independent facts from the passages. |

## VI. Discussion

There are some limitations to the current fact-reanking module. It relies on LLM. Thus, we are adding 1 component that needs LLM. This method adds cost to the total computation cost of LLM. There are already many components in the Fact Centric Knowledge Web that incorporate LLM.

However, the re-ranking module also offers pros, such as improved Approximate Match metric and a more precise answer to complex multi-hop questions. Our new approach improves safety by avoiding hallucinations and preventing possible incorrect answers. Also, the fact re-ranking system does single-shot learning and only sends one attempt to LLM, as we detect the question type before and utilize LLM with enhanced prompt instruction. This way, we will save computational cost by doing calculations before sending a prompt to LLM for the fact re-ranking operation. Also, fact re-ranking improves the answer by supporting simple inference and numerical calculations with enhanced prompt instructions.

## VII. Conclusion

We introduced an LLM-based fact re-ranking module into the Fact Centric Knowledge Web for Information Retrieval. This addition improves the factual accuracy and reasoning coverage, as demonstrated on complex multi-hop questions. Our implementation and enhancement of the Fact-Centric Knowledge Web demonstrates the significant impact that intelligent fact selection can have on answer quality. The Fact Re-Ranker addresses a critical limitation of the baseline system by ensuring that only the most relevant facts influence the final answer.

While this approach introduces additional computational costs and dependencies, the improvements in accuracy and the reduction in incorrect answers justify these trade-offs for applications where answer quality is paramount. The system's ability to acknowledge information gaps rather than providing incorrect answers is particularly valuable in real-world applications where trustworthiness is essential.

This work aligns with and extends the research presented in "Fact-Centric Knowledge Web for Information Retrieval," showing that the Knowledge Web structure, when combined with advanced fact selection techniques, offers a promising approach to reliable, multi-hop question answering. Future work may explore keyword indexing to enhance the Knowledge Web building process.

## REFERENCES


[1] A. Sinha and Y. Shiramatsu, "Fact-Centric Knowledge Web for Information Retrieval," in *Proc. of the 2023 ACM SIGIR Conf. on Research and Development in Information Retrieval*, 2023.

[2] OpenAI, "OpenAI Embeddings Documentation," 2022. [Online]. Available: https://platform.openai.com/docs/guides/embeddings

[3] Pinecone Systems Inc., "Pinecone: Vector Database for Machine Learning," 2023. [Online]. Available: https://www.pinecone.io/

[4] Neo4j Inc., "Neo4j Graph Database Platform." [Online]. Available: https://neo4j.com

[5] H. Trivedi, D. Yogatama, and A. Bosselut, "MuSiQue: A Multi-Hop Structured Question Dataset for Abstractive Answering," in *Proc. of the 2022 Conf. on Empirical Methods in Natural Language Processing (EMNLP)*, 2022.

[6] OpenAI, "GPT-3.5 API Documentation," 2023. [Online]. Available: https://platform.openai.com/docs/models/gpt-3-5