MULTIMODAL RETRIEVAL-AUGMENTED GENERATION IN KNOWLEDGE SYSTEMS: A FRAMEWORK FOR ENHANCED SEMANTIC SEARCH AND RESPONSE ACCURACY

Marko Mihajlović*
[0009-0006-7979-0341]

Singidunum University,
Belgrade, Serbia

Abstract:
Maintaining structured and up-to-date documentation remains a critical challenge in knowledge management, especially as organizational data becomes increasingly unstructured and multimodal. This paper presents a framework for a Multimodal Retrieval-Augmented Generation (RAG) Knowledge Database Assistant, designed to enhance semantic search and improve the accuracy of generated responses. By combining retrieval-augmented generation techniques with support for diverse data modalities (e.g., text, images, and structured metadata), the proposed system mitigates hallucination risks and increases the reliability of information access. The framework enables precise, context-aware question answering, even when underlying knowledge repositories are incomplete or inconsistently maintained. Our approach demonstrates how multimodal integration and RAG pipelines can form a robust foundation for next-generation knowledge systems.

Keywords:
Multimodal Retrieval, Retrieval-Augmented Generation, Knowledge Management, Semantic Search, Question Answering.

INTRODUCTION

In modern knowledge management, organizations increasingly face the challenge of maintaining structured, accurate, and up-to-date documentation. While unstructured knowledge bases often contain a wealth of valuable information, their lack of formal organization poses significant obstacles to effective search, retrieval, and usage [1]. As the scale and complexity of digital knowledge assets continue to grow, traditional keyword-based search methods and manually curated documentation are no longer sufficient to meet users' expectations for precision and relevance [2].

To address these challenges, this paper introduces a Multimodal Retrieval-Augmented Generation (RAG) Knowledge Database Assistant, a framework designed to enhance the usability and searchability of unstructured and semi-structured information repositories. The system integrates advanced retrieval-augmented generation techniques with multimodal data processing to support both text and image-based queries, enabling accurate, context-aware answer generation with traceable references [3].

Correspondence:
Marko Mihajlović

e-mail:
marko.mihajlovic.23@singimail.rs

DOI: 10.15308/Sinteza-2025-544-549

The proposed framework begins with exporting unstructured data into a defined format, which is then parsed using a custom-built tool to produce a structured XML representation of articles. Each article is automatically enriched with a synthetic question, generated by a language model (e.g., ChatGPT), and paired with a corresponding answer extracted from its content. These question–answer pairs, along with article titles and content, are embedded using OpenAI's Ada model, with weighted emphasis placed on the title and generated questions to optimize semantic representation.

The resulting embeddings are stored in a Redis database and indexed via RediSearch for efficient similarity-based retrieval [4]. During inference, a user's query is embedded and compared against the stored vectors to retrieve the top-matching articles. These are then combined with the original query and fed into a generative language model, which produces a coherent, accurate response enriched with references to the source materials. In addition to text-based interactions, the system also supports visual search. Image embeddings are created using the Vision Transformer (ViT) model and stored alongside textual metadata in the same vector database. Users can upload images or screenshots to retrieve related content, enabling a robust multimodal search experience that bridges textual and visual information.

By unifying structured parsing, semantic embedding, generative language modeling, and multimodal retrieval, the proposed framework delivers a scalable and extensible solution for navigating large, unstructured knowledge systems. It ensures high retrieval accuracy, minimizes hallucinated outputs, and supports a wide range of use cases where traditional search systems fall short.

2. METHODS

The following section outlines the design and implementation details of the proposed multimodal RAG-based framework.

2.1. DATA PREPARATION

The data preparation phase is responsible for transforming unstructured content into structured, semantically rich representations suitable for vector-based retrieval. This includes processing both textual and image-based data.

Articles are first extracted from unstructured sources (e.g., documentation, wikis) and structured into fields such as title, content, and a generated question. The questions are automatically created using a Large Language Model (LLM), such as ChatGPT, to semantically enrich the document and support better query matching. To embed the textual data, OpenAI's text-embedding-ada-002 model is used to generate 1536-dimensional vector representations for each article [5]. Fields are assigned weights to improve search relevance, titles and questions are prioritized (weight = 3.0), while content receives a lower weight (weight = 1.0) to minimize noise. These weights were empirically selected based on testing, where prioritizing the title and generated question over the full content yielded more accurate retrieval performance.

For visual data, the system extracts image references from structured XML. Using the OpenCLIP implementation of the ViT-B/32-quickgelu model, each image is preprocessed and encoded into a 512-dimensional normalized embedding vector [6]. Embeddings are cached locally to avoid redundant computation and are associated with metadata such as the image filename and post ID for later retrieval.

The framework utilizes two separate embedding pipelines, one for text and one for image data, each indexed in its own Redis vector database using RediSearch. This preparation pipeline ensures that both modalities, text and image, are encoded into high-quality embeddings, stored alongside useful metadata, and ready for indexing in the vector database. Both embedding pipelines leverage HNSW (Hierarchical Navigable Small World) indexing for efficient Approximate Nearest Neighbor (ANN) search, and cosine similarity is used to measure semantic proximity.

2.2. VECTOR DATABASE CONSTRUCTION

To enable scalable and efficient semantic retrieval, the system employs Redis as the underlying in-memory data store and RediSearch as the indexing and query engine. Redis is a high-performance, open-source, in-memory key-value store widely used for real-time applications due to its low-latency read/write operations. RediSearch is a Redis module that adds full-text search and secondary indexing capabilities, enabling complex queries, ranked retrieval, and schema definitions over Redis hashes [7].

In this framework, RediSearch is extended to support vector similarity search, transforming Redis into a vector database, a specialized type of database designed to store and retrieve high-dimensional vector representations (embeddings). Unlike traditional databases that rely on exact matching, vector databases enable approximate nearest neighbor (ANN) search, which is essential for retrieving semantically similar documents based on learned representations from models like OpenAI's Ada or OpenCLIP.

The system maintains two independent vector indexes, one for textual embeddings and another for image embeddings. Each index is tuned to the dimensionality and structure of its respective modality: 1536 dimensions for text (from Ada) and 512 for images (from ViT-B/32). These indexes are defined using RediSearch's schema configuration as shown in Listing 1 and leverage the HNSW (Hierarchical Navigable Small World) algorithm for fast and scalable ANN retrieval using cosine similarity as the distance metric. This setup allows the system to return semantically relevant results in real time, even across large datasets.

Text embeddings are stored with relevant metadata fields and configured weights to boost semantic precision during the search. Redis uses the HNSW indexing algorithm and cosine similarity for approximate nearest-neighbor retrieval.

Image embeddings are generated using OpenCLIP's ViT-B/32-quickgelu model, producing 512-dimensional normalized vectors. In addition to the embedding, each image post includes semantic metadata: title, content, and a permalink. These fields are stored in Redis using a hash and indexed with configurable weights as shown in Listing 2. This allows users to perform semantic searches using not only visual similarity, but also textual metadata tied to images (e.g., captions or context).

By separating these two vector spaces, the system supports modality-specific optimization and maintains retrieval precision across both text and image inputs. This modular structure is extensible and forms the foundation for building robust multimodal retrieval-augmented generation systems.

2.3. QUERY HANDLING AND ANSWER GENERATION

Once embeddings and metadata are stored in Redis, the system is equipped to handle real-time user queries and generate accurate, context-aware responses. The query processing pipeline supports both textual and visual inputs, enabling a flexible multimodal search experience.

When a user submits a text-based query, the system uses OpenAI's text-embedding-ada-002 model to generate a 1536-dimensional embedding of the query.

```
DEFINE TEXT_INDEX ON HASHES
  PREFIX: "article_text:"
  SCHEMA FIELDS:
    - title: TEXT, WEIGHT = 3.0
    - question: TEXT, WEIGHT = 3.0
    - content: TEXT, WEIGHT = 1.0
    - permalink: TEXT
    - embedding: VECTOR
        TYPE = FLOAT32
        DIM = 1536
        INDEX_METHOD = HNSW
        DISTANCE_METRIC = COSINE
```

Listing 1. Text Index Schema Definition

```
DEFINE IMAGE_INDEX ON HASHES
  PREFIX: "image_clip:"
  SCHEMA FIELDS:
    - title: TEXT, WEIGHT = 2.0
    - content: TEXT, WEIGHT = 1.0
    - permalink: TEXT
    - embedding: VECTOR
        TYPE = FLOAT32
        DIM = 512
        INDEX_METHOD = HNSW
        DISTANCE_METRIC = COSINE
```

Listing 2. Image Index Schema Definition

This embedding is then compared to stored vectors in the text index using cosine similarity leveraging RediSearch's vector capabilities. The top-k most relevant articles (typically k = 3) are retrieved based on similarity scores. If a user uploads an image or screenshot, the system processes it using the ViT-B/32-quickgelu model via OpenCLIP to obtain a normalized 512-dimensional embedding. This embedding is queried against the image index, and top-matching results are returned using the same ANN-based retrieval mechanism. For both query types, the system supports Retrieval-Augmented Generation (RAG) by combining the retrieved context with the original user query. The retrieved content is formatted into a structured prompt that is passed to a generative LLM (e.g., GPT-4). Predefined instructions ensure that the model produces accurate, grounded answers based strictly on the retrieved information.

The generated response includes:

- A concise and semantically accurate answer to the user's query
- Inline or endnote-style references linking to the relevant source articles
- Confidence and traceability through alignment with original data

This RAG pipeline significantly reduces hallucinations by anchoring the generative process to high-quality, contextually relevant embeddings, while also enhancing transparency by linking answers to verifiable sources [8]. At inference time, the system seamlessly integrates multimodal retrieval and generation by embedding the user query, retrieving the most relevant content from Redis, and synthesizing an accurate, reference-backed response using a large language model [9].

3. RESULTS AND DISCUSSION

The implementation of a comprehensive, multimodal knowledge management system demonstrates substantial benefits for enterprise environments. Digital knowledge management systems provide rich affordances for organizational knowledge work, such as improved organizational memory and information sharing [10]. These systems enable better collaboration, productivity gains, and enhanced safety in workplace environments [11]. Furthermore, the integration of multimodal archive resources, including text, images, audio, and video, has become increasingly important for effective knowledge management in the era of big data [12]. By leveraging Large Language Models (LLMs)

Table 1. Benchmark results

Supports Unstructured Data	**Traditional Search (Keyword-Based)**	**ML-Based Classifiers**	**Semantic Search (Text-Only)**	**Proposed Multimodal RAG Framework**
Supports Unstructured Data	Limited	Moderate	High	**High**
Handles Image Queries	No	No	No	**Yes**
Contextual Understanding	Low	Medium	High	**Very High**
Summarization & QA Capabilities	No	No	Limited	**Yes (via LLM)**
Real-Time Semantic Retrieval	No	Slow	Yes	**Yes**
Scalability	Medium	Medium	High	**High**
User Satisfaction	Low	Medium	High	**Very High**
Ease of Integration	High	Medium	Medium	**High**

Table 2. Qualitative Assessments of Indexing Solutions for Semantic and Vector Search

Feature / Capability	**Elasticsearch + kNN**	**FAISS**	**Pinecone**	**Redis + RediSearch**
In-Memory Performance	Moderate	High	High	**Very High**
Real-Time Updates	Limited	No	Yes	**Yes**
Full-Text Search Integration	No	No	Limited	**Yes**
Vector Search Support	Plugin-Based	Native	Native	**Native (via Module)**
Horizontal Scalability	Medium	Requires Custom	Built-In	**Built-In**
Ease of Integration	Moderate	Complex	Easy	**Easy**
Deployment Flexibility	Self-Hosted Only	Self-Hosted Only	Cloud Only	**Cloud or Self-Hosted**
Multi-Modal Support	Requires Custom Code	Requires Custom	Moderate	**Yes**

and Vision Transformers (ViT), the system successfully converts unstructured data into a structured and semantically searchable format. This transformation is key to enabling more accurate information retrieval and enhancing decision-making processes across departments.

To evaluate the effectiveness of the proposed framework, a comparison was conducted against common retrieval methods across several key dimensions relevant to enterprise knowledge management.

As shown in Table 1, the proposed multimodal RAG framework consistently outperforms keyword-based search and single-modality systems across multiple dimensions, including contextual understanding, real-time retrieval, user satisfaction, and the ability to process both text and image inputs. The integration of question generation and article summarization using LLMs significantly enhances the accessibility of information and the speed with which users can make informed decisions [7].

In addition to evaluating retrieval methods, it is also important to consider the underlying infrastructure supporting semantic search. Table 2 compares Redis with other commonly used indexing and vector storage solutions in terms of speed, scalability, flexibility, and ease of integration.

As shown in Table 2, Redis with RediSearch provides several advantages over alternative indexing and vector database solutions, including in-memory speed, support for real-time updates, full-text search capabilities, and seamless integration of vector and scalar data. These characteristics are essential for enterprise applications that require low-latency access to large and evolving knowledge bases [5].

The combination of these two elements, a robust multimodal retrieval pipeline and a high-performance indexing backend, positions this system as a competitive and future-proof solution for enterprise knowledge management. It effectively bridges the gap between unstructured data and actionable insight, supporting scalable, cost-efficient operations while improving user satisfaction and organizational intelligence.

From a practical standpoint, the system has demonstrated strong scalability, with the ability to handle increasing volumes of data and user traffic without compromising performance. This makes it well-suited for use in large and growing enterprise environments. Moreover, the system's ability to reduce time spent manually searching for documents results in measurable cost savings, enhances user satisfaction, and reduces the risk of knowledge loss. Compared to traditional or single-modality retrieval systems, the proposed multimodal RAG assistant provides a holistic and future-ready solution to enterprise knowledge management challenges.

4. CONCLUSION AND FUTURE IMPROVEMENTS

This work presented a robust and scalable multimodal knowledge management framework that leverages advanced machine learning models, specifically Large Language Models (LLMs) and Vision Transformers (ViT), in conjunction with a high-performance vector database to transform unstructured data into an accessible, searchable, and actionable resource. The system supports both text and image-based queries, enabling diverse information retrieval scenarios within enterprise environments. By integrating semantic embeddings, question generation, and article summarization, the framework significantly enhances decision-making, improves knowledge accessibility, and reduces the time spent searching for critical information. Furthermore, the modular design allows for horizontal scalability, cost efficiency, and seamless adaptation to evolving enterprise needs.

The proposed Multimodal Retrieval-Augmented Generation (RAG) Knowledge Assistant offers significant value for enterprise environments by transforming unstructured data into accessible, searchable knowledge [13]. It enhances decision-making by delivering accurate, context-rich information quickly and supports both text and image queries for comprehensive retrieval. Integration with large language models enables summarization and question-answering, reducing information overload and boosting productivity. By recovering lost or overlooked content, improving user satisfaction, and scaling with organizational growth, the system drives cost efficiency and provides a strategic advantage in managing enterprise knowledge.

Future enhancements aim to further increase the system's versatility and effectiveness by expanding support for additional file types such as PDFs and spreadsheets, enabling more comprehensive content search. Image retrieval will be improved through segmentation-based models that identify and focus on key visual regions, enhancing result relevance. The system will also support interactive file delivery within the chat interface, allowing users to download referenced content directly. Additionally, dynamic directory structuring will organize articles into semantic categories for better navigation. Ongoing model evaluation and exploring alternatives like BERT and DeepSearch, will optimize performance across various retrieval tasks and domains.

In summary, the continued development of this multimodal knowledge management framework will not only enhance its technical capabilities but also solidify its role as a strategic enabler of enterprise intelligence. By bridging unstructured data with accessible, context-rich insights, the system lays a strong foundation for future innovations in information retrieval, organizational learning, and scalable decision support.

REFERENCES

[1] I. Aviv, I. Hadar, and M. Levy, "Knowledge Management Infrastructure Framework for Enhancing Knowledge-Intensive Business Processes," *Sustainability*, vol. 13, no. 20, p. 11387, Oct. 2021, doi: 10.3390/su132011387.

[2] D. Li and Z. Zhang, "MetaQA: Enhancing human-centered data search using Generative Pre-trained Transformer (GPT) language model and artificial intelligence.," *PLoS One*, vol. 18, no. 11, p. e0293034, Nov. 2023, doi: 10.1371/journal.pone.0293034.

[3] S. Sarto, L. Baraldi, A. Nicolosi, R. Cucchiara, and M. Cornia, "Towards Retrieval-Augmented Architectures for Image Captioning," ACM Transactions on Multimedia Computing, *Communications, and Applications*, vol. 20, no. 8, pp. 1–22, Jun. 2024, doi: 10.1145/3663667.

[4] O. Frieder, C. I. Muntean, I. Mele, R. Perego, F. M. Nardini, and N. Tonellotto, "Caching Historical Embeddings in Conversational Search," *ACM Transactions on the Web*, vol. 18, no. 4, pp. 1–19, Oct. 2024, doi: 10.1145/3578519.

[5] D. S. Asudani, P. Singh, and N. K. Nagwani, "Impact of word embedding models on text analytics in deep learning environment: a review.," *Artif Intell Rev*, vol. 56, no. 9, pp. 10345–10425, Feb. 2023, doi: 10.1007/s10462-023-10419-1.

[6] Y. Yang, Y. Zhou, W. Shi, H. Fu, and S. Gao, "Intrusion detection: A model based on the improved vision transformer," *Transactions on Emerging Telecommunications Technologies*, vol. 33, no. 9, Apr. 2022, doi: 10.1002/ett.4522.

[7] Χ. Αποστολακη, "Extensive performance evaluation of popular relational and non-relational data stores for full-text search," 2025.

[8] C. Yao and S. Fujita, "Adaptive Control of Retrieval-Augmented Generation for Large Language Models Through Reflective Tags," *Electronics (Basel)*, vol. 13, no. 23, p. 4643, Nov. 2024, doi: 10.3390/electronics13234643.

[9] K. Rangan and Y. Yin, "A fine-tuning enhanced RAG system with quantized influence measure as AI judge," *Sci Rep*, vol. 14, no. 1, Nov. 2024, doi: 10.1038/s41598-024-79110-x.

[10] H. Safadi, "Balancing Affordances and Constraints: Designing Enterprise Social Media for Organizational Knowledge Work," *MIS Quarterly*, vol. 48, no. 1, pp. 347–374, Mar. 2024, doi: 10.25300/misq/2023/16499.

[11] J. Schuir and F. Teuteberg, "Understanding augmented reality adoption trade-offs in production environments from the perspective of future employees: A choice-based conjoint study," *Information Systems and e-Business Management*, vol. 19, no. 3, pp. 1039–1085, May 2021, doi: 10.1007/s10257-021-00529-0.

[12] Y. Zhou, X. Wang, R. Zhao, Z. Zhang, and Q. Sheng, "Multimodal archive resources organization based on deep learning: a prospective framework," *Aslib Journal of Information Management*, Jan. 2024, doi: 10.1108/ajim-07-2023-0239.

[13] S. Siriwardhana, S. Nanayakkara, R. Weerasekera, R. Rana, T. Kaluarachchi, and E. Wen, "Improving the Domain Adaptation of Retrieval Augmented Generation (RAG) Models for Open Domain Question Answering," *Trans Assoc Comput Linguist*, vol. 11, pp. 1–17, Jan. 2023, doi: 10.1162/tacl_a_00530.