Computers in Industry 171 (2025) 104336

Contents lists available at ScienceDirect

Computers in Industry

journal homepage: www.sciencedirect.com/journal/computers-in-industry

SAMAC-R^3-MED: Semantic alignment and multi-agent collaboration of retriever-reranker-responder models for multimodal engineering documents

Fei Li, Xinyu Li, Sijie Wen, Haoyang Huang, Jinsong Bao *

College of Mechanical Engineering, Donghua University, Shanghai 201620, China

ARTICLE INFO

Keywords:
Multimodal engineering documents
Multimodal semantic alignment
Retriever-Reranker-Responder
Multi-agent collaboration
Retrieval-augmented generation

ABSTRACT

In the manufacturing industry's lifecycle, a vast amount of engineering documents in text, table, and image formats is generated. Retrieval-augmented generation (RAG) models can enhance retrieval efficiency and adapt to evolving document knowledge. However, challenges in understanding multimodal semantic associations and the absence of engineering-semantic-aligned RAG models result in suboptimal accuracy. This paper introduces a novel approach, namely SAMAC-R^3-MED, to tackle these challenges. First, a fine-grained context enhancement strategy is applied to multimodal large language models (MLLMs), bridging multimodal semantic understanding by constructing multi-modal semantic trees (MMST) and multi-modal knowledge graphs (MMKG), forming a hybrid retrieval base. Second, to bridge the semantic gap in RAG models, a new training framework, retriever-reranker-responder (R^3), is proposed, utilizing supervised and reinforcement learning with ranking feedback to enhance alignment. Third, a multi-channel hybrid retrieval strategy is implemented for the multi-agent collaboration R^3 models, integrating expert feedback, semantic trees, and graphs to optimize the RAG pipeline and improve the accuracy of retrieving multimodal associative semantic contexts. An engineering documents chat (eDoChat) system is implemented, in the case of wind turbine assembly, validating the effectiveness in retrieving and generating accurate multimodal answers. Ablation experiments show R^3 models outperform traditional RAG models, and SAMAC-R^3-MED achieves state-of-the-art results in multimodal retrieval and generation tasks.

1. Introduction

In the design and manufacturing process, engineers must generate and consult numerous engineering documents containing multimodal elements such as text, tables, and images. To improve efficiency, an effective information retrieval system is essential for supporting decision-making (Park et al., 2023). With the expansion of large language models (LLMs), these models have become transformative tools for knowledge storage and question-answering, and can be fine-tuned for downstream tasks (Ma et al., 2023; Susnjak et al., 2024). However, the continuously evolving domain-specific knowledge in engineering documents renders the factual information encoded in the parameters of LLMs obsolete. Frequent fine-tuning to update these outdated factual representations in LLM parameters poses significant challenges (Mitchell et al., 2022; Nguyen et al., 2024). Moreover, engineering documents typically contain domain-specific language, multiple data formats, and unique multimodal contextual relationships that general purpose-trained LLMs do not handle well.

At present, researchers employ a variety of RAG techniques to address the continual evolution of domain-specific knowledge in engineering documents (Lewis et al., 2020). These approaches construct and dynamically update external knowledge bases, thereby enabling the effective retrieval of factual, domain-specific information and enhancing the ability of LLMs to comprehend and generate content that is both time-sensitive and context-aware. VectorRAG supports LLMs in generative tasks by retrieving similar texts, excelling in generating coherent responses when relevant document context is required (Izacard and Grave, 2020; Guu et al., 2020a). However, traditional RAG methods often rely on paragraph chunking for engineering documents, overlooking the hierarchical structure. Moreover, due to the domain-specific language and symbols in engineering documents, semantic gaps frequently arise between components of RAG models, such as retrieval,

* Corresponding author.
E-mail addresses: 1229077@mail.dhu.edu.cn (F. Li), lixinyu@dhu.edu.cn (X. Li), 2221212@mail.dhu.edu.cn (S. Wen), huanghaoyang0521@163.com (H. Huang), bao@dhu.edu.cn (J. Bao).

https://doi.org/10.1016/j.compind.2025.104336
Received 16 October 2024; Received in revised form 5 June 2025; Accepted 2 July 2025
Available online 9 July 2025

rerank, and response models. Given the complexity of semantic relationships in engineering documents, context retrieved from large heterogeneous corpora may lead to inaccurate or incomplete LLMs analyses. GraphRAG, natural language processing (NLP) tasks using knowledge graphs (KGs) (Edge et al., 2024a; Hu et al., 2024), can extract structured knowledge of hidden entities and relationships from engineering documents, improving the accuracy and context-sensitivity of responses. However, GraphRAG typically performs poorly in abstract question-answering tasks or when the questions lack explicit entities. Moreover, the above RAG techniques do not support expert intervention to directly edit answers and provide real-time experiential knowledge feedback during the question-answering process. This limitation results in a lack of flexibility in retrieval systems.

Engineering documents encompass various forms of data, including text, charts, and images. It is crucial to effectively integrate and process these different modalities to ensure the consistency and completeness of information. The current mainstream multimodal retrieval technologies mainly include multimodal embedding and text embedding retrieval methods, as illustrated in Fig. 1. On one hand, multimodal embedding utilizes Vision-and-Language Pre-training (VLP) models like Contrastive Language-Image Pre-training (CLIP) (Radford et al., 2021) and Vision-and-Language Transformer (ViLT) (Kim et al., 2021) to map image-text pairs into a unified vector space for multimodal similarity feature matching. On the other hand, the multimodal large-scale model BLIP (Li et al., 2023) and LLaVA (Liu et al., 2023) are employed for conducting multimodal summarization with multimodal output. Subsequently, the summarization is transformed into text embedding using a text model to facilitate the alignment of multimodal information. However, engineering documents contain rich structured semantic information, and the semantic mappings between different modalities are complex, with the significance of multimodal content often being highly context-dependent. The two aforementioned methods may struggle with maintaining accuracy in identifying and interpreting cross-modal semantics due to their lack of context-awareness, potentially leading to omissions or the inclusion of noisy images in multimodal responses.

To address the challenges in multimodal engineering document retrieval, this paper proposes the SAMAC-R^3-MED method, which integrates structured embedding with multi-agent collaboration R^3 model retrieval techniques, as shown in Fig. 1. The main contributions of this study are as follows:

(1) First, clustering techniques are employed for fine-grained cross-modal context-aware computation, enhancing MLLMs for semantic alignment. The method analyzes structured semantic information in multimodal engineering documents and constructs the bottom-up MMST for VectorRAG, improving abstract question-answering performance. It also extracts structured knowledge of entities and relationships to build the MMKG for GraphRAG. By combining the strengths of VectorRAG and GraphRAG, HybridRAG is implemented to provide more accurate multimodal answers.

(2) Second, to bridge the semantic gap in RAG models concerning engineering terminology and symbols, a novel R^3 models training framework is introduced, integrating supervised and reinforcement learning with prior ranking feedback to enhance semantic alignment and precision.

(3) Third, to increase system flexibility, an expert feedback agent (Agent-EF) is designed for real-time answer editing, collaborating with the MMST agent (Agent-MMST), MMKG agent (Agent-MMKG), and R^3 model to establish a multi-agent collaborative retrieval strategy. This optimizes the multimodal RAG pipeline, significantly enhancing the professionalism and accuracy of the eDoChat system's responses.

The paper is organized as follows. Section 2 reviews the previous works relevant to this research. Section 3 describes the methodology of the paper. Section 4 presents the ablation experiments and the evaluation of results based on engineering case studies. Section 5 discusses the findings, implications, and limitations of the proposed method. Finally, the paper is concluded in Section 6.

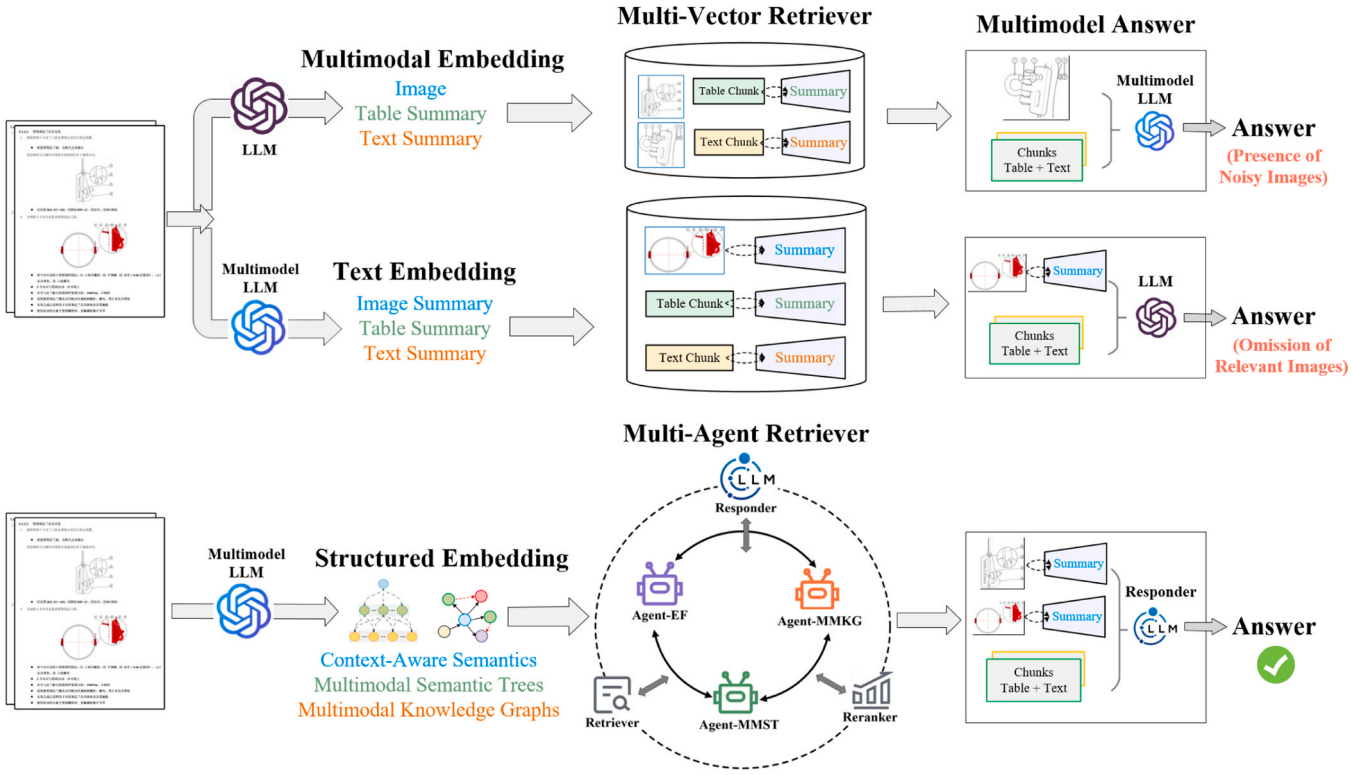


Fig. 1. Improvement of multimodal information retrieval methods for engineering documents.

2. Related work

2.1. Documents question & answering system

Documents question & answering system (DQAS) is a complex system integrating various technologies, including document retrieval, KGs, and neural networks (Yang et al., 2024). The DQAS framework operates in two primary modalities: text-based and multimodal-based. Its knowledge storage architectures include vector index databases (Lewis et al., 2020), KGs (Edge et al., 2024b), and LLMs (Mitchell et al., 2022). Text-based DQAS is widely applied. Wang et al. (2022) retrieved similar sample instance vectors and fed them into the model to generate answers. Wang et al. (2024) constructed a semantic graph of document structure relationships and a graph traversal module to form a QA system. Pereira et al. (2023) explored a multi-document QA system called Visconde, combining LLMs with neural re-ranking. However, engineering documents typically contain not only dense text but also visually rich multimodal elements such as titles, paragraphs, tables, and charts.

Multimodal-based DQAS (MDQAS) aims to return fused results of charts and text relevant to the query, posing challenges in understanding visually rich documents. Multimodal Transformers have advanced multimodal information retrieval (Liu et al., 2024a; Tao et al., 2023; Li et al., 2025; Wang et al., 2025). Existing multimodal QA frameworks—such as CLIP-DPR (Liu et al., 2022), VinVL-DPR (Liu et al., 2022), and UniVL-DR (Liu et al., 2022), as well as MARVEL (Zhou et al., 2023) and VISTA (Zhou et al., 2024)—leverage contrastive learning to map images and text into a unified vector space. These methods employ generative image-caption data to enhance visual semantics, enabling knowledge-intensive retrieval over visually rich documents. However, they often overlook the logical structures embedded in predefined document layouts and the context-dependent cross-modal entity-relation semantics. With the recent advances in MLLMs (Li et al., 2023; Liu et al., 2023), Li et al. (2024b) explore various contextual configurations of MLLMs through visual question answering (VQA) tasks to optimize their performance in image-caption alignment. Ding et al. (2024) propose PDF-MVQA, which constructs a VQA dataset from PDFs to enable answer retrieval from both paragraphs and charts. Nevertheless, engineering documents often contain structured charts alongside terminology-dense textual descriptions, forming complex and context-sensitive cross-modal semantic dependencies. Existing approaches struggle to ensure both retrieval accuracy and traceability under such conditions. To address this challenge, this paper proposes a context-aware cross-modal structured semantic alignment strategy. By performing hierarchical clustering and summarization of metadata within documents, the proposed method enhances MLLMs' capability to align fine-grained semantics between images and paragraphs, as well as between tables and paragraphs. A unified MMST vector space is constructed to bridge semantic disparities across modalities. Furthermore, MLLMs are utilized for metadata-driven entity-relationship extraction to build MMKG. By integrating vector and graph structures in a HybridRAG framework, the proposed approach significantly improves the accuracy and traceability of MDQAS for engineering documents.

2.2. Retrieval-augmented LLMs generation

RAG enhances the updateability and interpretability of LLMs by integrating external knowledge bases. However, both VectorRAG and GraphRAG exhibit inherent limitations (Sarmah et al., 2024). RAPTOR, proposed by Sarthi et al. (2024), employs tree-based recursive summarization to retrieve contextual vectors from different leaf layers, while PDFTriage, introduced by Saad-Falcon et al. (2023), leverages layout features to enable LLMs to locate answers based on document structure or content. These approaches significantly improve VectorRAG's capability in perceiving hierarchical semantics. Nevertheless, they still struggle to capture deep associations among multimodal metadata—such as text, images, and tables—in engineering documents, and offer limited support for entity-relation retrieval involving extensive domain-specific terminology. Leveraging the advantages of graph structures in entity-relation reasoning, GraphRAG has been widely explored. For instance, Li et al. (2024a) integrate case-based reasoning into knowledge base question answering (GS-CBR-KBQA), Sarica et al. (2023) automatically map design documents into entity-relation graphs using semantic networks to construct retrievable design knowledge bases, and Edge et al. (2024a) utilize LLM-generated global summaries to derive knowledge graphs that drive GraphRAG. Although these methods enhance entity-relation retrieval and reasoning, they remain less effective for abstract question answering tasks where queries lack explicit entities or require fine-grained contextual understanding. To address the structured charts, dense technical terminology, and intertwined semantics typical in engineering documents, this paper proposes a fine-grained, context-aware cross-modal semantic alignment strategy and constructs a HybridRAG system composed of MMST and MMKG. The system enables joint retrieval and reasoning over vector and graph structures, thereby combining the advantages of hierarchical multimodal semantic capture and entity-relation inference.

Retrieval-augmented language models (RALMs) have seen improvements in various components: retrievers, rerankers, and LLM responders (Chen et al., 2024; Yu et al., 2024a; Touvron et al., 2023). To mitigate the semantic gap between retrievers and responders, UniGen (Li et al., 2024b) introduces a unified generation framework for both retrieval and question answering, employing LLM-generated connectors to bridge the gap between query inputs and response targets, as well as between document identifiers and answers. End-to-end system training efforts include Atlas (Izacard et al., 2023), which fine-tunes an encoder-decoder model during the retrieval stage. REALM (Guu et al., 2020b) is a bidirectional masked language model fine-tuned for open-domain question answering. Yu et al. (2024b) propose the PopALM architecture, which pre-trains popular language models in the social media domain and leverages reinforcement learning to guide LLMs in generating responses favored by the target audience. Although these methods effectively narrow the semantic gap between retrieval and generation, they still lack an integrated "retrieval-reranking--response" semantic alignment framework tailored for engineering documents. To address this limitation, this paper proposes a semantic alignment R^3 training architecture, which systematically bridges the semantic gap in engineering-domain RALMs by jointly optimizing the three-stage model using supervised learning combined with reinforcement learning guided by prior ranking feedback.

2.3. Optimizing the pipeline of RAG

The optimization of the RAG system pipeline aims to improve document retrieval efficiency and quality by refining query and retrieval strategies, balancing retrieval speed with response context depth (Chan et al., 2024). Advanced RAG, such as long-context reorder (Liu et al., 2024b), hybrid fusion retriever (Sawarkar et al., 2024), and sentence window retrieval (Sun et al., 2023), adapt to diverse query needs, retrieving relevant and content-rich information. Techniques like HyDE (Gao et al., 2022), which uses hypothetical document embeddings, and T-RAG (Fatehkia et al., 2024), which integrates entity tree and vector query strategies, further enhance retrieval. However, when dealing with diverse types of QA tasks in engineering documents—such as abstract questions, entity-based queries, and multimodal problems—the aforementioned methods face challenges in simultaneously achieving recall coverage and answer accuracy within a unified RAG pipeline. Inspired by AutoGen (Wu et al., 2023) and ChatDev (Qian et al., 2023), this work introduces the Agent-EF, Agent-MMST, and Agent-MMKG to balance retrieval efficiency and accuracy. A multi-channel multi-agent collaboration R^3 retrieval algorithm is developed, implementing a HybridRAG that incorporates expert feedback. This approach optimizes the RAG pipeline for multimodal information, significantly improving system flexibility and answer quality.

3. Methodology

3.1. Overall framework

For the retrieval and generation of multimodal semantics in industrial documents, the framework of SAMAC-R^3-MED method is designed as shown in Fig. 2. It primarily includes two offline modules, namely, multimodal semantic trees and graphs and semantic alignment of R^3 models. Additionally, there are two other online modules in the framework, namely, the RAG task input and the multi-agent collaboration R^3 models for multimodal answer generation.

For the one offline module, optical character recognition (OCR) is first employed to extract explicit or implicit layer element instances from engineering documents. A context-aware semantic alignment strategy is applied to enhance MLLMs for engineering image summarization tasks. Recursive clustering and summarization techniques are then used for the semantic computation of multimodal metadata, forming hierarchical MMST to implement VectorRAG. Second, to construct MMKG within GraphRAG, GPT-4 extraction prompts are used to extract entities and relationships from metadata and summarize descriptions, generating hierarchical graphs. Engineering images and tables undergo entity/relationship extraction based on these summarized descriptions, linking related text entities to form comprehensive multimodal graphs for the engineering documents.

To bridge the semantic gap between engineering terminology and symbol semantics in RAG models, an offline module featuring a novel R^3 model training framework is designed, using supervised fine-tuning (SFT) and reinforcement learning (RL) with prior ranking feedback for semantic alignment. A two-stage retrieval approach is implemented: in the first stage, the retriever model is fine-tuned using supervised learning to obtain the top-k results. In the second stage, the reranker model learns a prior semantic similarity ranking to reorder these top-k results. In the responder model stage, a relevance measure based on prior ranking is designed to train the reward model (RM). Through SFT and the proximal policy optimization (PPO) algorithm, reward signals are updated in the LLMs to improve relevance scores between {query, passage, answer}.

In the online multimodal information retrieval phase, a hybrid approach combining VectorRAG and GraphRAG with expert feedback is employed. An indexing pipeline with a multi-agent system is established, incorporating the Agent-EF, Agent-MMST, and Agent-MMKG. The multi-agent collaboration R^3 algorithm is designed to optimize the RAG pipeline and facilitate the retrieval of multimodal associative semantic contexts. Based on the user's query, the RAG task for engineering documents is executed, supporting multimodal answer generation, scoring, and re-editing as needed.

3.2. Multimodal semantic trees and graphs

As shown in Fig. 3, a set of engineering documents $D = \{D_1, D_2, \ldots, D_d\}$ is collected, for constructing MMST. In this process, a single document D is divided into different leaf layers through n levels of section headings. Among them, section headings form the set $H = \{H_1, H_2, \ldots, H_h\}$, each leaf layer comprises multiple sections, forming the set $\{Se\}$. In the formation of one tree layer metadata, multi-scale multimodal metadata is configured to capture the document's hierarchical features, which includes a set $P = \{p_1, p_2, \ldots, p_a\}$ formed by a paragraphs p, paragraph p is composed of b sentences s from set $S = \{s_1, s_2, \ldots, s_b\}$. The paragraphs are associated with m tables T, jointly forming set $\{P, T\} = \{(p_1, t_1), (p_2, t_2), \ldots, (p_m, t_m)\}$ with context paragraphs. Additionally, n images V are integrated with context paragraphs to form the set $\{P, V\} = \{(p_1, v_1), (p_2, v_2), \ldots, (p_n, v_n)\}$. Minimal chunks (sentences, images, and tables) are used to assemble continuous sentences into paragraph sets, while the contextual semantics of images and tables are aggregated to form assembly chunks. The gaussian mixture model (GMM) is applied to cluster the paragraph vectors of $\{P, T\}$, $\{P\}$, and $\{P, V\}$ data nodes, enabling context-aware semantics computation.

Assuming the data is generated from a mixture of multiple gaussian distributions, for N text segments with Z-dimensional dense vector embeddings, given the membership of a text vector x in the k^{th} gaussian distributions, the likelihood of the text vector x is expressed as:

$$P(\mathrm{x}|k) = N(\mathrm{x}; \mu_k, \Sigma_k) \tag{1}$$

In Eq. (1), the μ_k and Σ_k are the mean and covariance matrix of the k^{th} gaussian distribution, respectively. The overall probability distribution is a weighted combination of k gaussian distributions:

$$P(\mathrm{x}) = \sum_{k=1}^{K} \pi_k N(\mathrm{x}; \mu_k, \Sigma_k) \tag{2}$$

In Eq. (2), π_k represents the mixture weight of the k^{th} gaussian distribution. To determine the optimal number of clusters, the bayesian information criterion is used for model selection (Sarthi et al., 2024).

Context-aware multimodal semantic alignment is achieved through GMM clustering and summarization by MLLMs, forming image-summary cluster nodes $\{V, S_u\}$ and table-summary cluster nodes $\{T, S_u\}$, which are combined with the summaries of similar paragraph nodes

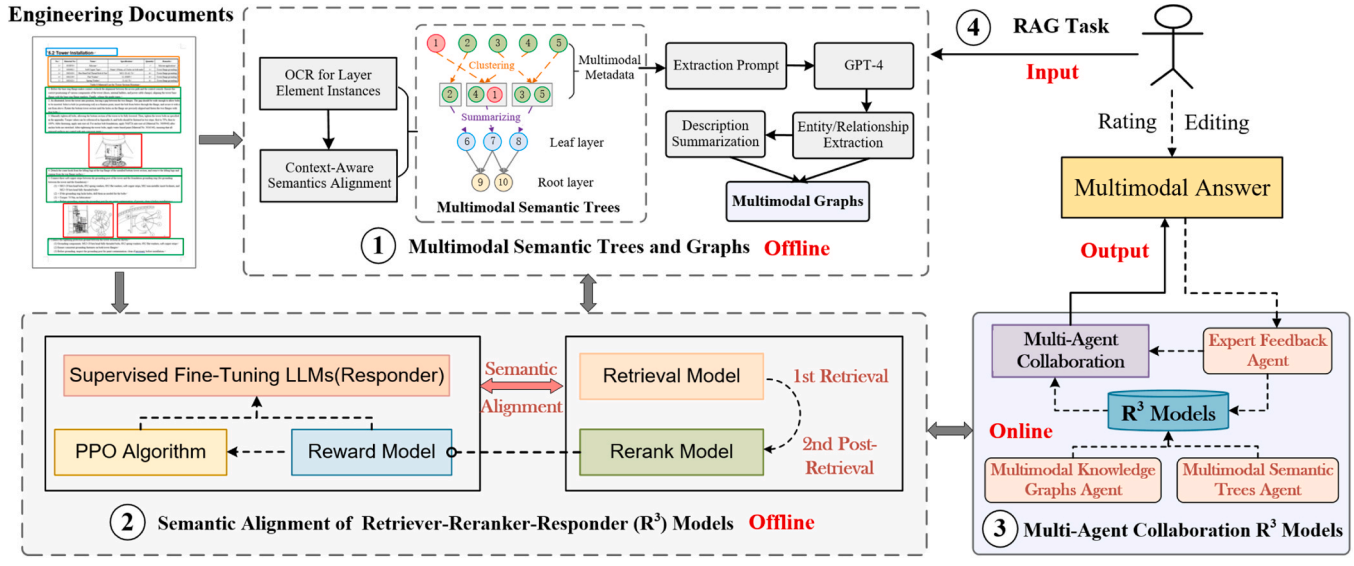


Fig. 2. Framework of the SAMAC-R^3-MED method.

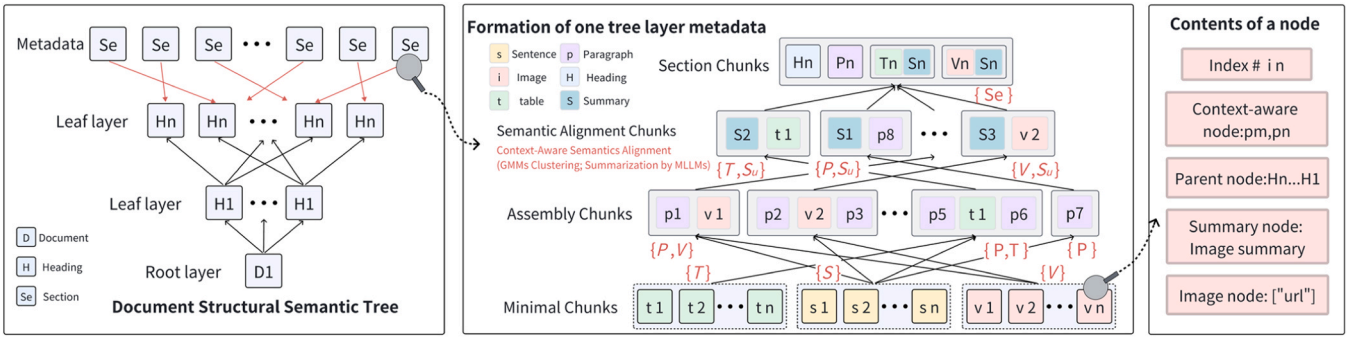


Fig. 3. Construction process of MMTS.

$\{P, S_u\}$ to create semantic alignment chunks. These multi-scale chunks are aggregated within the same section to form section chunks. The contents of an image node in the tree structure include {index, context-aware node, parent node, summary node, image node}. Multiple section chunks form the leaf layer of the MMST, enabling semantic understanding of the entire hierarchical structure of the industrial document.

To construct the MMKG for engineering documents, assembly domain rules are designed to define the classes and relationships of multimodal entities. The multi-scale multimodal metadata of engineering documents is analyzed to construct the entity set E, which primarily includes text entities (E_W), image entities (E_V), and table entities (E_T). E_W typically includes assembly processes A_C, assembly procedures A_P, assembly steps A_S, assembly partsA_E, and assembly tools A_T, such that $E_W = \{A_C \cup A_P \cup A_S \cup A_E \cup A_T\}$. E_V usually contains assembly sequence diagrams V_S, and E_T usually includes assembly bill of materials tables T_L. Furthermore, the attributes of various multimodal entities are defined to provide additional information about node entities in the graph, enhancing semantic richness between entities and their relationships. The main entity categories, definitions, and attribute descriptions for assembly engineering documents are presented in Table 1.

To address the characteristics of assembly process data, a set of relationships R for entities is constructed, mainly including operational relationships R_{op}, temporal relationships R_{te}, and structural relationships R_{st}, i.e. $R = \{R_{op} \cup R_{te} \cup R_{st}\}$. The operational relationship R_{op} represents specific procedures, the temporal relationship R_{te} represents the implicit sequential order between different nodes, and the structural relationship R_{st} represents the inherent connections between nodes. The main relationships and their definitions are shown in Table 2, and the "HasImg" and "HasTab" relationships establish connections between multimodal entities in the graph. Based on the entity-relationship-attribute definitions for engineering documents in Tables 1 and 2, extraction prompts for GPT-4 are designed to extract entities, relationships, and summarize metadata descriptions, creating hierarchical MMKG. The ontology construction of multimodal graphs for assembly engineering documents is illustrated in Fig. 4.

Table 1
Definitions of multimodal entities in assembly engineering documents.

Multimodal Entity	Entity Category	Entity Definition	Attribute Name
Text Entity (E_W)	Assembly Process A_C	All assembly methods and operational procedures for converting raw materials or semi-finished products into the final product	Process Name; Process Specification; Summary Description
	Assembly Procedure A_P	An independent assembly step performed at the same location with the same equipment, involving multiple steps	Procedure Name; Duration; Associated Process; Summary Description
	Assembly Step A_S	A single assembly action performed at a fixed position using the same tool without changing the operating method	Step Name; Associated Procedure; Summary Description
	Assembly Part A_E	The smallest unit that forms the assembly machine or product	Part Name; Specification; ID; Summary Description
	Assembly Tool A_T	Manual or automated tools used during the assembly process to complete various steps and procedures	Tool Name; Specification; Model; Summary Description
Image Entity (E_V)	Assembly Sequence Diagram V_S	Images guiding the assembly process	URL; Image Summary
Table Entity (E_T)	Assembly Bill of Materials Table T_L	A table listing tools required for the assembly process	Markdown; Table Summary

3.3. Semantic alignment of Retriever-Reranker-Responder models

To bridge the semantic gap in RAG models within the engineering domain, a training framework for the semantic alignment of R^3 models is designed, as shown in Fig. 5. The retriever consists of two main stages, the retriever model and reranker model. The SFT strategy is applied to optimize the retriever model, enhancing its ability to discriminate between positive and negative samples in industrial semantics. The reranker model learns the prior semantic similarity ranking for post-retrieval. To reduce the semantic discrepancy between responder, retriever, and reranker model, composite tuples are constructed from the context data of the two-stage retrieval. A relevance measure is set for responder to train the RM, enhancing LLMs' learning of relevance scores among {query, context, answer} and reducing hallucinations. RL is used to minimize the PPO objective, updating reward signals in the SFT of LLMs to achieve industrial semantic alignment in R^3 models, further enhancing alignment between the retriever's capabilities and the generator's output preferences.

3.3.1. One-stage retriever model

To better support engineering semantic embeddings and dense knowledge retrieval tasks, BGE-M3 is used as the embedding model, encoding polluted text into embeddings and recovering clean text through a lightweight decoder:

$$\min \sum_{x \in X} - \log \mathrm{Dec}(x|\mathrm{e}_{\tilde{X}}), \mathrm{e}_{\tilde{X}} \leftarrow \mathrm{Enc}(\tilde{\mathrm{X}}) \tag{3}$$

In Eq. (3), Enc, Dec are the encoder and decoder, X, $\tilde{\mathrm{X}}$ indicate the clean and polluted texts of x, $\mathrm{e}_{\tilde{X}}$ represents the embedding. The dictionary for query and positive/negative samples is constructed based on the semantic analysis of engineering documents, {"query": str, "positive":

Table 2
Types and definitions of relationships between multimodal entities.

Relationship Type	Relationship Category	Relationship Definition	Connected Modalities
Operational Relationship R_{op}	Prepare	Indicates the parts and tools needed before an assembly task	Text-Text
	Has Tool	Indicates a tool required during the operation	Text-Text
	Check	Indicates a step to check during the operation process	Text-Text
	Operate	Indicates an action taken on a part	Text-Text
	On/Under/ Before/After/ Side of	Indicates the relative position relationships between different assembly entities	Text-Text
Temporal Relationship R_{te}	Step To	Indicates the sequence between different procedures within the same process	Text-Text
	Next	Indicates the sequence between different steps within the same procedure	Text-Text
Structural Relationship R_{st}	Has Image	Indicates the related image associated with a text entity	Text-Image
	Has Tabe	Indicates the related table associated with a text entity	Text-Table

List[str], "negative": List[str]}, forming a JSON dataset. Through contrastive learning, BGE-M3 learns domain-specific semantic knowledge, enhancing the model's embedding discriminative ability to distinguish paired texts from negative samples. For each pair of hard-labeled texts: $Q:(q,p)$ (query , passage), the contrastive loss function is as follows:

$$\min .\sum_{(q,p)} -\log \frac{\exp(\langle e_q, e_p\rangle/\tau)}{\exp(\langle e_q, e_p\rangle/\tau)+\sum_{Q'}\exp(\langle e_q, e_{p'}\rangle/\tau)} \tag{4}$$

In Eq. (4), $\langle\cdot\rangle$ denotes the inner product operator, $p' \in Q'$ represents negative samples that do not match with q, and τ is the temperature parameter. For fine-tuning specific retrieval tasks, for each text pair (q, p), an additional task-specific instruction prompt I_t needs to be appended to the query side $q' \leftarrow q + I_t$. For example, the instruction prompt "search for relevant context for the query" can be used.

3.3.2. *Two-stage reranking for post-retrieval*

To improve the relevance of retrieval results, a combination of embedding models and reranking models is employed. The BGE-reranker-large model is trained to learn prior semantic similarity ranking, focusing on reranking the top-k documents returned by the embedding model. Localization contrast estimation (LCE) is introduced to learn the negative sample distribution by sampling from the top results of the target retriever. The LCE loss for a batch of hard-labeled texts Q is defined in Eq. (5).

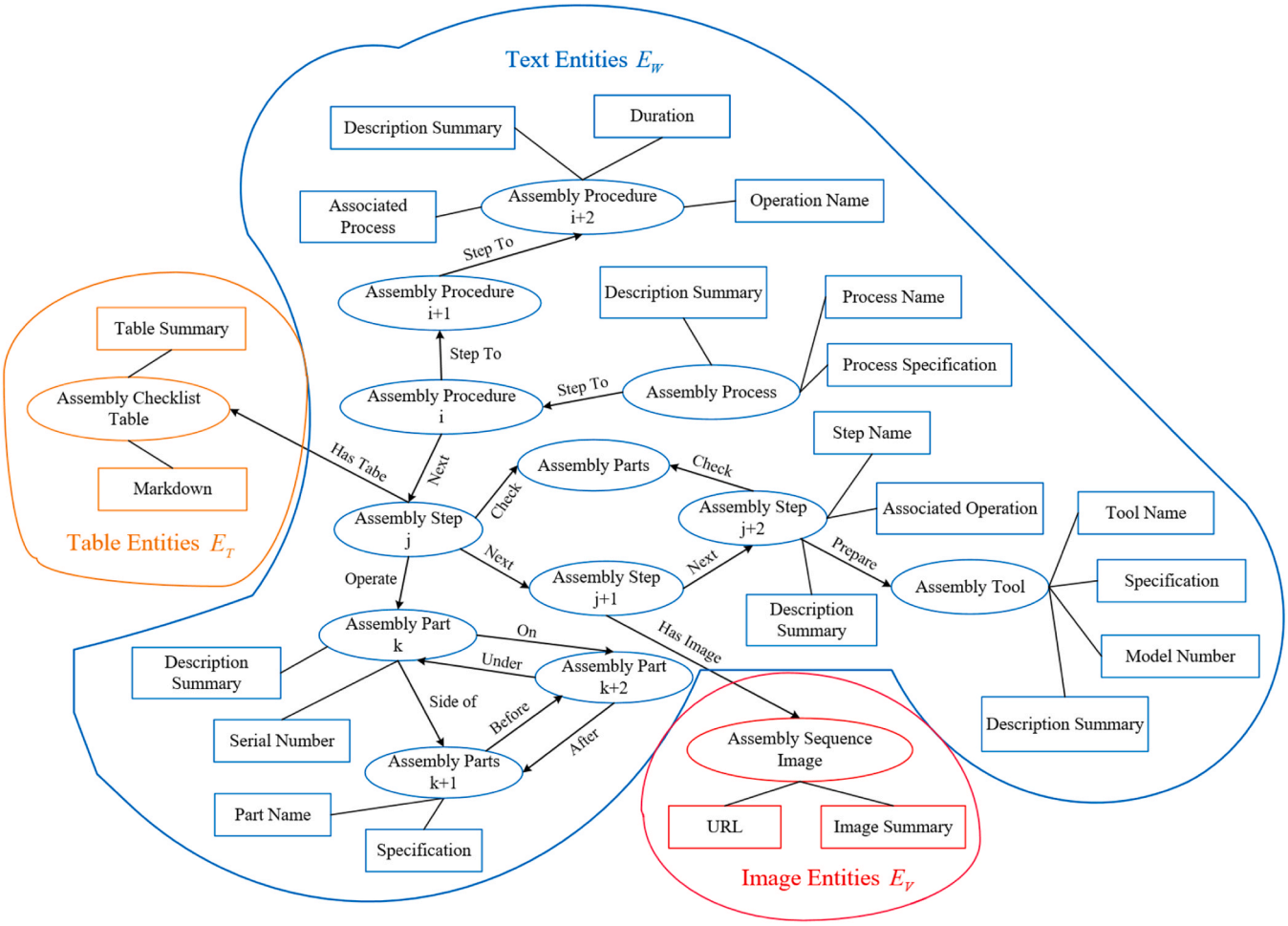


Fig. 4. Ontology of MMKG for assembly engineering documents.

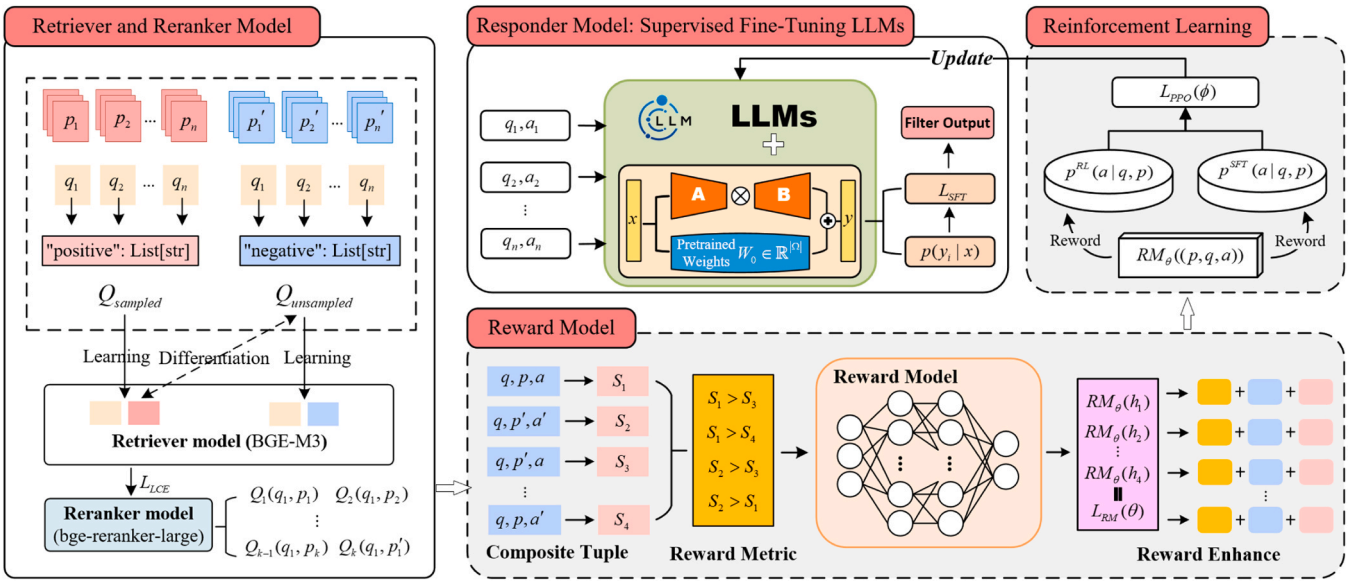


Fig. 5. Training framework for the semantic alignment of R^3 models.

$$L_{LCE} = \frac{1}{|Q|} \sum_{(p,q)\in Q, G_q \sim R_q^k} -log \frac{\exp(dist(q,p))}{\sum_{p' \in G_q} \exp(dist(q,p'))} \tag{5}$$

For each query q, samples are drawn from the set R_q^k comprising the top-k ranked documents. The group G_q is hence formed, containing positive samples p that match q and negative samples p' that do not match q sampled from R_q^k. The function $dist(\cdot)$ denotes the Euclidean distance. Compared to the standard noise contrastive estimation loss, LCE uses the target retriever to locate negative samples and focuses learning on the top-ranked results instead of randomly sampled noise negatives.

3.3.3. Supervised fine-tuning LLMs

The parameters of LLMs are fine-tuned using supervised learning to acquire semantic knowledge in the industrial domain. The ChatGLM3 model is used for SFT as described. Specifically, given a labeled example $(q; a)$, the input text q is converted into a cloze question $c(q)$ using a pattern that contains a single mask token. The cloze question is a pattern-transformed version of the input text that includes a single masked token for predicting the answer. The candidate labels $a' \in \Omega$ are then mapped to the cloze question's answers, referred to as the verbalizer $u(a)$. The conditional probability of predicting a given q is expressed with Eq. (6):

$$p(a_i|q) = \frac{p(u(a_i)|c(q; a_{i-1}))}{\sum_{a' \in \Omega} p(u(a')|c(q; a_{i-1}))} \tag{6}$$

where, Ω denotes the token set of the ChatGLM3 vocabulary, u represents the Multi-Layer Perception (MLP), and c is the decoder of the GLM. The dimension of $p(a_i, q)$ is $\mathbb{R}^{|\Omega|}$. Defined with Eq. (7), the objective function of SFT for ChatGLM3 is to maximize the likelihood,

$$L_{SFT} = \frac{1}{L} \sum_{i=0}^{L} CE(a_i, p(a_i|q)) \tag{7}$$

where L represents the target sequence length, which decides how many of the most recently generated tokens in the prefix to consider for context augmentation, and CE is the cross-entropy loss.

3.3.4. Reward model training

Using ChatGLM3 as the backbone, the RM is trained to enable ChatGLM3 to predict the popularity of generated answers. Given a query and the top-k contexts retrieved by the reranker model based on prior semantic similarity, $S((q,p,a)) \in \mathbb{R}$ is designed as the relevance measure for the model's response, where (q,p,a) represents the tuple of (query, passage, answer). To construct the training dataset for the RM, for each pair of matching and non-matching tuples (q,p,a) and (q,p',a'), additional tuples (q,p',a) and (q,p,a') are assembled to form a composite tuple:

$$h = ((q,p,a)(q,p',a')(q,p',a)(q,p,a')) \tag{8}$$

Using the dataset $H_* = \{h_1, \ldots, h_N\}$ to train the RM. For a given h, if the answer is hallucinated, factually incorrect, or lacks relevant positive sample contexts, the relevance score should be low; otherwise, it should be higher, determining the final score order:

$$\begin{aligned} &S((q,p,a)) > S((q,p',a)), S((q,p,a)) > S((q,p,a')), S((q,p',a')) \\ &\quad > S((q,p',a)), S((q,p',a')) > S((q,p,a)) \end{aligned} \tag{9}$$

The RM is trained using a contrastive loss function to minimize discrepancies:

$$\begin{aligned} L_{RM}(\theta) = -\frac{1}{4} E_{h\sim H_*} [&\log(\sigma(RM_\theta(h_1) - RM_\theta(h_3))) + \log(\sigma(RM_\theta(h_1) - RM_\theta(h_4))) \\ &+ \log(\sigma(RM_\theta(h_2) - RM_\theta(h_3))) + \log(\sigma(RM_\theta(h_2) - RM_\theta(h_1)))] \end{aligned} \tag{10}$$

3.3.5. Reinforcement learning

PPO algorithm is applied to integrate the reward signals provided by the RM, which assesses the relevance of generated answers, into the SFT of ChatGLM3. This approach combines SFT with RL using prior ranking feedback and metrics to achieve industrial semantic alignment between the retriever and the generator. The parameters of ChatGLM3 are updated by minimizing the PPO objective:

$$L_{PPO}(\phi) = RM_\theta((q,p,a)) - \beta \log(\frac{p^{RL}(a|q,p)}{p^{SFT}(a|q,p)}) \tag{11}$$

The logits of the End-of-Sequence (EOS) token represent the scalar reward as $RM_\theta((q,p,a)) = v(p_{eos}|d(p; a|q))$. β is a scalar coefficient of the Kullback-Leibler divergence, $p^{RL}(a|q,p)$ denotes the logits of the active model, while $p^{SFT}(a|q,p)$ denotes the logits of the reference model.

3.4. Multi-agent collaboration R^3 models

To improve the accuracy of multimodal associative semantic context in RAG, Agent-EF, Agent-MMST, and Agent-MMKG are defined. Each agent focuses on different types of structured knowledge in engineering documents and collaborates with the R^3 models to assign and manage tasks, enabling efficient and adaptive multimodal retrieval. The multi-agent collaboration R^3 model's multi-channel retrieval strategy is illustrated in Fig. 6, with three channels: one-agent, two-agents, and three-agents. The one-agent channel involves either Agent-MMST or Agent-MMKG working with the R^3 models. The two-agents channel includes two combinations: Agent-EF with Agent-MMST, or Agent-EF with Agent-MMKG, collaborating with the R^3 models. The three-agents channel features Agent-EF, Agent-MMST, and Agent-MMKG jointly working with the R^3 models. The eDoChat system then generates multimodal answers relevant to the query based on this multi-agent collaboration.

The algorithm for three-agent collaboration R^3 models, presented in Table 3, processes a set of engineering documents $D = \{D_1, D_2, \ldots, D_n\}$ and a user query as input, generating multimodal answers as output. The process includes building MMST and MMKG for the engineering documents D. The retriever model embeds the user query to obtain its vector representation q_e. Agent-EF, Agent-MMST, and Agent-MMKG are invoked in parallel to retrieve relevant contexts. Agent-EF retrieves similar historical Q&A for q_e, passing it to the reranker model to rank the top-k historical Q&A based on prior semantic similarity. Notably, the historical Q&A set stores expert-rated and edited responses, enabling Agent-EF to provide real-time expert knowledge feedback. Agent-MMST and Agent-MMKG are simultaneously called to retrieve multimodal contexts. Agent-MMST performs hierarchical retrieval using MMST and the FAISS library to retrieve multimodal-aligned metadata chunks, which are then reranked to obtain the top-k results. Agent-MMKG generates cypher queries for graphs involving multimodal entities and relationships. If relevant multimodal contexts are retrieved by three agents, which are then forwarded to the responder for generating multimodal answers. Otherwise, the responder generates an answer directly through a prompt based on the user query.

4. Case study

4.1. Experimental data

To evaluate the effectiveness of the proposed method, a case study was conducted on the assembly guidance documents of a specific wind turbine model. A dataset of 235 engineering documents was collected, containing multimodal content such as titles, page numbers, paragraphs, images, and tables. Detailed statistics of the multimodal engineering documents and the semantic alignment training dataset for the R^3 models are presented in Table 4. These statistics include the number of tables, images, and tokens used in the training, validation, and test sets. The dataset comprises 200 documents for training, 16 for validation, and 19 for testing. "#ImgSumm" indicates the number of image-summary pairs, "#TabSumm" represents the number of table-summary pairs, "#AvgDocTokens" refers to the average tokens per document, "#AvgTabSummTokens" denotes the average tokens per table-summary pair, and "#AvgImgSummTokens" indicates the average tokens per image-summary pair.

The categorization of question types in multimodal engineering documents includes abstract query, entity-based query, and multimodal query. Each type is defined as follows: abstract queries involve abstract or conceptual information beyond concrete entities; entity-based queries focus on one or more specific entities such as assembly parts, procedures, steps, or tools; and multimodal queries require cross-modal reasoning across text, image, or tabular data. Example questions for each type are illustrated in Table 5. The number of verification questions per document for each question type is 20, 15, and 5, respectively.

Based on the collected engineering document dataset, a subset was constructed for training, validation, and testing of the R^3 models for semantic alignment, with data statistics provided in Table 4. First, a JSON dataset of positive and negative samples for each query was

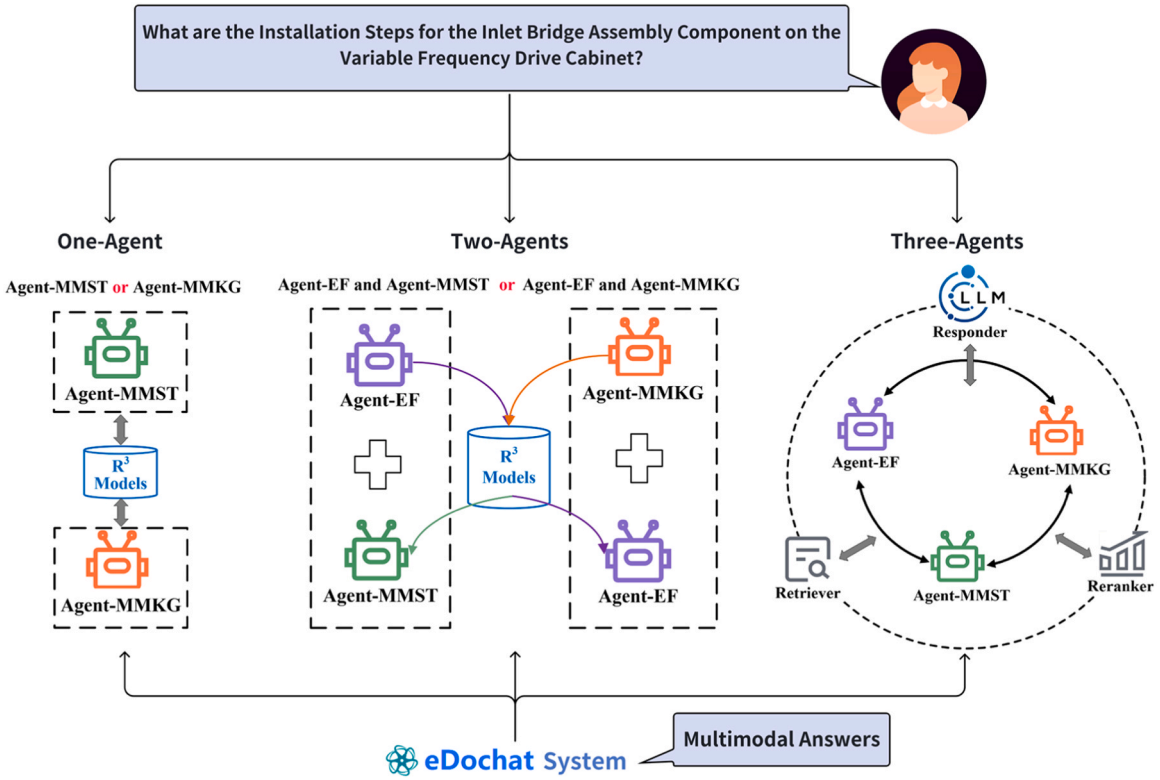


Fig. 6. Multi-channel strategy for multi-agent collaboration R^3 models.

Table 3
Algorithm of multi-agent collaboration R^3 models.

Algorithm 1 Three-Agent Collaboration R^3 Models

Definition: Agent-MMST; Agent-MMKG; Agent-EF; E (Retriever Model); R (Reranker Model); G (Responder Model);

Input: $D = \{D_1, D_2, ..., D_n\}$ (Engineering documents), q (User query)

Output: a (Answers with multi-modal response)

1: **while** q **do**
2: Vector of $q_e \leftarrow get.E(q)$
3: Agent-EF, Agent-MSST and Agent-MMKG. $search(q_e)$
4: where Agent-EF. $search(q_e)$
5: Similar Historical Q&A ← Agent-EF. $search(q_e)$
6: Top-k Historical Q&A ← $R.rerank$ (Similar Historical Q&A)
7: where Agent-MSST. $search(q_e)$
8: Similar Metadata Chunks ← MSST. $search(q_e)$
9: Top-k Metadata Chunks ← $R.rerank$ (Similar Metadata Chunks)
10: where Agent-MMKG. $search(q)$
11: Cypher ← $G.generate(q)$
12: Multi-modal Entities and Relations ← Cypher. $search$ (MMKG)
13: **if** Similar (Historical Q&A) or (Metadata Chunks) or (Entities and Relations) == **True**
14: Multi-modal Contexts = (Top-k Historical Q&A, Metadata, Entities and Relations)
15: $a \leftarrow G.generate$ (Multi-modal Contexts)
16: **else**
17: $a \leftarrow G.generate(q)$
18: **end if**
19: **end while**

Table 4
Dataset statistics for multimodal engineering documents and R^3 models.

	Datasets		Train	Valid	Test
Multi-modal Engineering Documents (Texts、Tables、Images)	#Documents		200	16	19
	#ImgSumm		2097	159	184
	#TabSumm		832	52	59
	#AvgDocTokens		11695	11574	12036
	#AvgTabSummTokens		83.27	81.69	86.45
	#AvgImgSummTokens		75.35	73.21	76.30
R^3 Models (Semantic Alignment Training)	Retriever		12980	1035	1273
	Reranker		7964	679	725
	Responder	SFT	23140	1845	2270
		RM	8765	1560	1746
		RL	6580	1373	1285

constructed to enhance the retriever model's vector representation for engineering semantics. Second, samples were extracted from the top-k documents relevant to each query, including both matching and non-matching positive and negative examples, to train the reranker model for learning prior semantic similarity ranking. Finally, based on positive and negative context samples from the top-k documents, datasets for SFT, RM, and RL were constructed within the responder model to perform semantic alignment training between the generator and retriever. The SFT subset was designed to optimize and adjust LLM parameters, equipping the responder model with domain-specific knowledge and engineering symbols. A training set of 8765 composite tuples $\{(q,p,a)(q,p',a')(q,p',a)(q,p,a')\}$ was created for RM, with relevance metrics established to enhance the scoring of {query, passage, answer}. Additionally, 6580 tuples were sampled for the RL training set. By minimizing the PPO objective, reward signals from RM were updated into the SFT of LLMs, achieving a combination of supervised learning and reinforcement learning with prior ranking feedback and metrics.

Based on the wind turbine assembly guidance engineering document dataset presented in Table 4, MMST and MMKG were constructed. A comprehensive statistical summary of entity and relationship types, along with their quantities in MMKG, and the types and quantities of nodes and chunks in MMST, is provided in Table 6. MMKG primarily includes three types of entities: text, image, and table $\{E_W, E_V, E_T\}$, with quantities of 28572, 2440, and 943, respectively. According to the operational, temporal, and structural relationships $\{R_{op}, R_{te}, R_{st}\}$ defined in Table 2, the quantities in MMKG are 13675, 9752, and 3383, respectively. The visualization of MMKG for the wind turbine assembly process is shown in Fig. 7. Each MMST consists of a root layer and multiple leaf layers, with each leaf layer containing multiple multi-scale multimodal metadata elements. The quantities of section metadata chunks, assembly metadata chunks, and minimal metadata chunks calculated for the 235 MMST are 3760, 4895, and 38,641, respectively.

4.2. Evaluation metrics

Evaluation Metrics: To assess the multimodal retrieval accuracy of engineering documents, the following metrics are used: NDCG@K (Normalized Discounted Cumulative Gain), a relevance-weighted ranking score that rewards relevant documents appearing earlier and is normalized against an ideal ranking; MRR@K (Mean Reciprocal Rank), the average reciprocal rank of the first relevant document, highlighting how early the first correct answer is returned; Recall@K, the share of all relevant documents that are retrieved within the top-K results, indicating coverage; Precision@K, the fraction of the top-K results that are relevant, indicating list purity; and Accuracy@K, a binary top-K metric that records 1 if the unique ground-truth document is present within the first K positions and 0 otherwise. K ranges from 1 to 100, with K = 10 adopted as the primary evaluation benchmark.

For table-summary, context-enhanced text generation, and

Table 5
Definitions, examples, and number of verification questions per document for each question type.

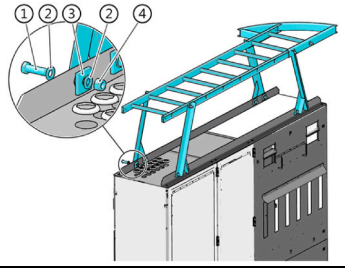
Question Type	Description	Number of Verification Questions per Document	Example Question
Abstract query	Queries involving abstract or conceptual information beyond concrete entities.	20	What are the installation steps for the inlet bridge assembly component on the variable frequency drive cabinet?
Entity-based query	Queries focused on one or more specific entities such as assembly parts, procedures, steps, or tools.	15	What are the assembly procedures for installing the motor's main shaft, commutator, bearing, and sealing ring, and what are the corresponding assembly steps for each component?
Multimodal query	Queries requiring cross-modal reasoning across text, image, or tabular data.	5	Here is the schematic diagram of the variable frequency drive cabinet assembly. What are the detailed installation steps for the fasteners? 

Table 6
Statistics of MMST and MMKG nodes and relationships in engineering documents.

	Entities/Nodes Categories	Quantities		Relations/Metadata Chunks Categories	Quantities
MMKG	Text Entity E_W	28572	**MMKG**	Operational Relationship R_{op}	13675
	Image Entity E_V	2440		Temporal Relationship R_{te}	9752
	Table Entity E_T	943		Structural Relationship R_{st}	3383
MMST	Text Node	48511	**MMST**	Metadata of Sections Chunks	3760
	Image Node	2440		Metadata of Assemble Chunks	4895
	Table Node	943		Metadata of Minimal Chunks	38641

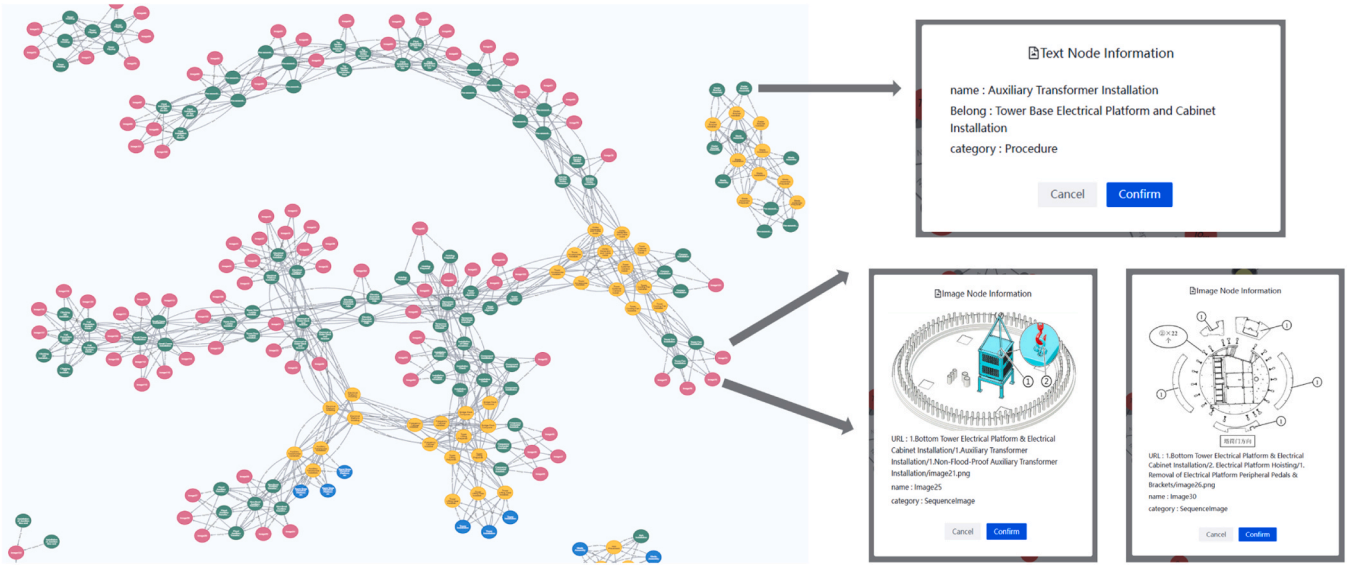


Fig. 7. Visualization of MMKG for assembly processes.

image-summary tasks, the evaluation metrics include: ROUGE-{1, 2, L}, which measures content coverage by computing unigram, bigram and longest-common-subsequence overlap between generated and reference texts; BLEU-N, a precision-oriented n-gram metric (typically up to 4-gram) with a brevity penalty that captures fluency and exactness; METEOR, which balances precision and recall while accounting for stemming, synonyms and word order to provide a more robust sentence-level quality score; F-1 Match, combining the harmonic mean of precision and recall to evaluate lexical matching between the generated and reference text; and Accuracy, which determines whether the generated text exactly matches the expected answer.

By emphasizing early hits, coverage, and ranking quality during the retrieval stage, and balancing information completeness, terminology accuracy, and strict consistency during the generation stage, the proposed evaluation framework comprehensively assesses practical performance in multimodal document retrieval-based question answering

scenarios. It ensures that the results are not only retrievable but also precisely generated.

4.3. Results and analysis

4.3.1. Overall performance

To comprehensively evaluate the effectiveness of SAMAC-R^3-MED, systematic experiments were conducted under an environment configured with Ubuntu 20.04, PyTorch 2.3, Python 3.11.9, and CUDA 11.8/cuDNN 8.9.2, utilizing 8 ×NVIDIA A100 GPUs (40 GB) with BF16 mixed precision and gradient checkpointing. The dataset subset used for training is listed in Table 4. The entire R^3 fine-tuning pipeline was built using HuggingFace-TRL and LangChain-FAISS libraries. (1) Retriever: BGE-M3 was employed and trained using in-batch hard negative contrastive learning with a learning rate of 3×10^{-5}, effective batch size of 512, and temperature of 0.05. Training was conducted for 3 epochs with early stopping based on validation performance. (2) Reranker: A cross-encoder, BGE-reranker-large, was adopted with local contrastive estimation (train_group_size=16). The training configuration included a learning rate of 6×10^{-5}, effective batch size of 32, maximum input length of 512, and weight decay of 0.01. The model was fine-tuned for 5 epochs to capture fine-grained relevance. (3) Responder: ChatGLM3–6B with LoRA injection (rank=8) was used for instruction SFT with a learning rate of 3×10^{-5}, batch size of 128, and R-Drop regularization, trained for 3 epochs. For reward modeling, 80 % of ChatGLM3–6B's lower layers were frozen, and a single-layer MLP head was attached. The reward model was trained using a quadruplet contrastive loss with margin= 1.0, τ= 0.07, learning rate= 1×10^{-5}, and batch size= 64 for 2 epochs. During the reinforcement learning phase, KL-PPO was employed with the following settings: clip= 0.2, β linearly increased from 0.05 to 0.15, value coefficient= 0.5, entropy coefficient= 0.01, horizon= 128, GAE λ= 0.95, γ= 0.99, and policy learning rate= 5×10^{-6}, accumulating approximately 2–3 k policy updates. In each iteration, the top-10 retrieved documents were used, retaining the top-5 passages to generate responses within a maximum of 512 tokens. The R^3 training process was designed to balance convergence speed, memory efficiency, and inference stability. MMST and MMKG were constructed for engineering documents. A hybrid retrieval strategy was implemented using multi-agent collaboration via the R^3 model. Retrieval evaluations were conducted by executing test queries over the entire corpus. Baseline models used in the experiment included VinVL-DPR (Liu et al., 2022), CLIP-DPR (Liu et al., 2022), UniVL-DR (Liu et al., 2022), Marvel-DPR (Zhou et al., 2023), Marvel-ANCE (Zhou et al., 2023), VISTA (Zhou et al., 2024), and SAMAC-R^3-MED. Evaluation results for NDCG@10, MRR@10, Recall@5, Precision@1, and Accuracy@10 are presented in Table 7.

For the multimodal retrieval task involving texts, tables, and images in the wind turbine assembly guidance documents, tables were summarized using the responder model. Table 7 shows that UniVL-DR outperforms baseline models such as VinVL-DPR, CLIP-DPR, and Marvel-DPR, but lags behind Marvel-ANCE and VISTA. VISTA improves retrieval performance by integrating visual embeddings, enhancing a strong text encoder with image understanding, and using a multi-stage training algorithm. However, SAMAC-R^3-MED achieves the highest multimodal retrieval performance, with NDCG@10, MRR@10, Recall@5, Precision@1, and Accuracy@10 scores of 69.2, 77.5, 75.9, 80.6, and 63.1, respectively, surpassing VISTA. This is primarily due to SAMAC-R^3-MED constructing MMST, MMKG, and the R^3 models' training framework for semantic alignment within engineering documents. Combined with the multi-agent collaboration strategy, this approach significantly improves multimodal retrieval. Additionally, context-aware enhanced MLLMs for image and table alignment within MMST and MMKG effectively bridge modality gaps.

4.3.2. Ablation studies

To further analyze the multimodal retrieval performance of the proposed method in engineering documents, ablation experiments were conducted to evaluate the effects of semantic alignment and multi-agent collaboration R^3 model retrieval strategy. First, the effectiveness of the proposed context-aware image-summary semantics alignment (CAI-SSA) strategy was validated. Different MLLM baseline models, BLIP-2 and LLaVA-Llama-3–8B, were fine-tuned based on multimodal datasets from Table 4, performing image-summary tasks in both with and without CAI-SSA modes. Evaluation results for ROUGE-L, BLEU-1, F-1 Match, and Accuracy, presented in Table 8, show that CAI-SSA significantly enhances performance for both models. BLIP-2 with CAI-SSA shows an average improvement of 2.64 %, while LLaVA-Llama-3–8B achieves the best performance with improvements of 1.87 %, 2.84 %, 2.63 %, and 3.58 % in ROUGE-L, BLEU-1, Accuracy, and F-1 Match, respectively. These results confirm the performance gains provided by CAI-SSA, which improves understanding of abstract image semantics in industrial documents through rich contextual paragraph descriptions.

To evaluate the effectiveness of constructing MMST and MMKG for engineering documents and the performance of semantic alignment in R^3 models during the retrieval phase, an image-summary task was conducted using the LLaVA-Llama-3–8B with the CAI-SSA strategy. Baseline models for comparison included BM25, mDPR, mE5$_{large}$, and R^3 models. The R^3 combines BGE-M3-retriever with BGE-reranker and SFT+RM+PPO LLM-responder for semantic alignment. Three modes were configured: without MMST, with MMST, and with both MMST and MMKG, utilizing FAISS for similarity retrieval. The evaluation metrics in Fig. 8 show higher performance in the "with MMST" mode compared to "without MMST," with the highest performance in the "with MMST and MMKG" mode, indicating that MMST and MMKG consistently outperform their respective retrievers. The mE5$_{large}$ baseline significantly improves upon BM25 and mDPR, while the R^3 achieves the highest scores across ROUGE-{1,2,L}, BLEU-1, and METEOR, with values of 55.14 %, 35.08 %, 51.39 %, 34.48 %, and 41.42 %, respectively, confirming the effectiveness of semantic alignment in the R^3 models. With MMST, R^3

Table 8
Performance of different MLLMs with CAI-SSA strategy.

MLLMs	Strategy	ROUGE-L	BLEU-1	Accuracy	F-1 Match
BLIP−2	w/o CEI-SA	35.23	20.71	43.81	54.28
	w/ CEI-SA	**37.16**	**23.43**	**46.39**	**57.61**
LLaVA-Llama−3–8B	w/o CEI-SA	40.28	25.02	50.78	59.94
	w/ CEI-SA	**42.15**	**27.86**	**53.41**	**63.52**

Table 7
Multimodal retrieval results for engineering documents using diverse baseline models.

Methods	NDCG@10	MRR@10	Recall@5	Precision@1	Accuracy@10
VinVL-DPR	29.9	38.6	35.6	39.7	20.1
CLIP-DPR	37.7	48.7	45.1	49.3	26.2
UniVL-DR	48.9	59.4	57.5	62.2	39.4
Marvel-DPR	40.1	51.2	49.8	55.6	32.1
Marvel-ANCE	54.5	67.3	62.5	69.5	45.6
VISTA	61.7	73.8	70.2	75.3	53.4
SAMAC-R^3-MED (Ours)	**69.2**	**77.5**	**75.9**	**80.6**	**63.1**

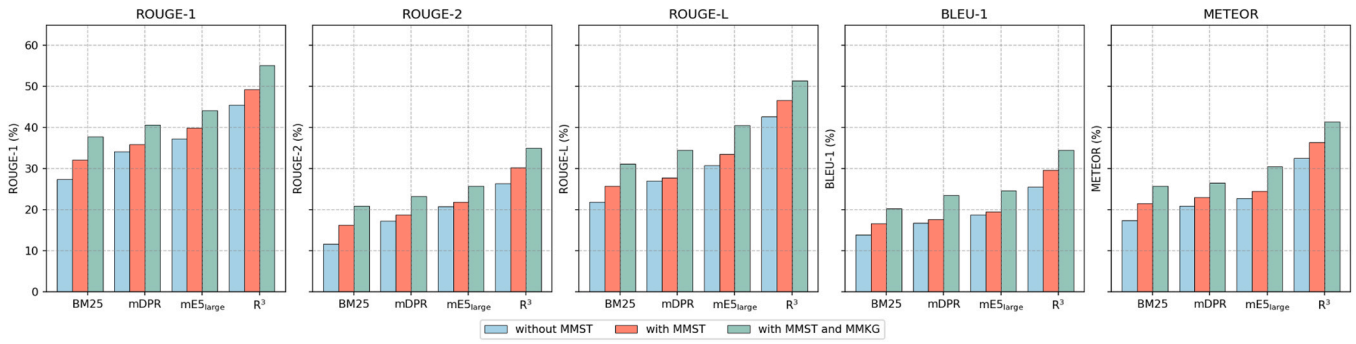


Fig. 8. Results of different retrieval methods with and without MMST and MMKG.

shows improvements in ROUGE-1, BLEU-1, and METEOR by 3.74 %, 4.08 %, and 3.87 %, respectively, highlighting MMST's advantages. Furthermore, with MMST and MMKG, ROUGE-1, BLEU-1, and METEOR scores increased by 5.82%, 4.86 %, and 5.06 %. This demonstrates the retrieval benefits of HybridRAG composed of MMST and MMKG. MMST's performance is attributed to its multi-scale metadata, which captures hierarchical semantics in engineering documents for better similarity matching, while MMKG excels in establishing relationships between entities, enabling relational retrieval to compensate the limitations of similarity matching.

To quantify the individual contributions of the MMST, MMKG, and R^3 within SAMAC-R^3-MED, an ablation study was performed on the full model (R^3 + MMST + MMKG), the results are summarized in Table 9. The full configuration achieved the best retrieval effectiveness and answer accuracy across the abstract, entity-based, and multimodal query scenarios, yielding NDCG@10 and F-1 match of 70.2 and 68.7, respectively, for abstract queries. Removing MMST caused the NDCG@10 and F-1 match for abstract queries to drop by 19.3 and 22.3 percentage points, indicating that MMST is essential for context-aware cross-modal metadata abstraction and structured semantic alignment. Omitting MMKG predominantly affected entity-based queries, reducing NDCG@10 and F-1 match by 14.7 and 14.5 percentage points, thereby underscoring the role of MMKG in precise entity-level relation localization. Eliminating R^3 degraded performance for all three query types, with average reductions of 7.3 points in NDCG@10 and 7.5 points in F-1 match, confirming the necessity of R^3 's semantic-alignment architecture that integrates supervised learning with reinforcement learning guided by prior ranking feedback. Collectively, MMST, MMKG, and R^3 act synergistically to underpin the state-of-the-art performance of SAMAC-R^3-MED in multimodal engineering document retrieval and question answering tasks.

To validate and enhance the responder performance within the R^3 baseline, various SFT methods were applied to adjust LLM parameters under the hybrid retrieval mode, combined with RL using RM and the PPO algorithm for semantic alignment and answer generation in the engineering domain. The datasets for training, validation, and testing, as presented in Table 4, were used to evaluate various SFT methods for ChatGLM3–6B as the responder. Test results in Table 10 show that various SFT models achieve higher metrics compared to the original model, demonstrating the effectiveness of SFT strategies in enhancing semantic understanding. LoRA (r = 8) delivers the best results and is used with various LLM backbone networks to train the RM, with reward signals updated through the PPO algorithm. Performance outcomes under the SFT+RM+PPO strategy show that Baichuan2–7B scores the lowest, while LLaMA3–8B outperforms ChatGLM3–6B. Qwen2–7B achieves the best performance, with BLEU-1, ROUGE-{1,2,L}, and METEOR scores of 36.65 %, 57.34 %, 37.54 %, 53.68 %, and 43.82 %, respectively. Notably, ChatGLM3–6B using the SFT+RM+PPO strategy shows significant improvement over SFT strategy alone, validating the effectiveness of relevance measurement in RM and confirming the LLMs' ability to enhance learning, relevance scores, and answer accuracy.

To assess the performance of multi-agent collaboration R^3 models, Qwen2–7B, trained under the SFT+RM+PPO strategy, was used as the responder. Following the algorithm in Table 3, Three agent combination modes were tested: one-agent (either Agent-MMST or Agent-MMKG alone), two-agents (Agent-MMST or Agent-MMKG with Agent-EF), and three-agents (Agent-MMST, Agent-MMKG, and Agent-EF). Performance was evaluated using BLEU-1, ROUGE-L, and METEOR metrics, as shown in Fig. 9. The results indicate that three-agents configuration achieved the highest scores across all metrics. Specifically, the METEOR score increased by 6.4 % and 5.71 %, and the ROUGE-L score improved by 3.5 % and 2.66 %, compared to using Agent-MMST or Agent-MMKG alone and the two-agents mode, respectively. The superior performance in the three-agent mode is attributed to Agent-MMST's multimodal semantic retrieval, Agent-MMKG's semantic association capabilities, and Agent-EF's real-time expert feedback. Combined with the R^3 model's semantic alignment, these elements significantly enhance performance metrics.

To further evaluate the performance of different responders in the multi-agent collaboration R^3 strategy, experiments were conducted to compare retrieval performance using Agent-MMST as the primary agent across various agent and responder combination modes. Answer generation results using three LLMs, namely Qwen2–7B, GPT4o, and GPT4, are presented in Table 11. Notably, Qwen2–7B was trained using the SFT+RM+PPO strategy for semantic alignment as shown in Fig. 5. Ablation experiments were evaluated using F-1 Match, Accuracy, and Average Time, where average time represents the response time for the entire retrieval-augmented generation process. The results show that GPT-4 sets a new benchmark on the Table 4 dataset, achieving the highest accuracy and F-1 match scores across all agent modes, with maximum values of 57.5 % and 74.8 %. The performance of Qwen2–7B, despite having only 7B parameters, is comparable to GPT4o. However,

Table 9
Ablation results for the key components (MMST, MMKG, and R^3) of SAMAC-R^3-MED.

Question Type	Abstract query		Entity-based query		Multimodal query	
Different Components	NDCG@10	F-1 Match	NDCG@10	F-1 Match	NDCG@10	F-1 Match
w/o MMST	50.9	46.4	65.8	63.9	56.4	54.7
w/o MMKG	67.1	66.1	53.6	52.6	61.8	60.2
w/o R^3	61.8	60.5	60.4	59.7	63.5	62.4
w/ R^3+MMST+MMKG	**70.2**	**68.7**	**68.3**	**67.1**	**69.1**	**69.2**

Table 10
Test results of responders using various SFT, RM, and RL strategies.

Methods	SFT				SFT+RM+PPO			
LLMs	ChatGLM3-6B				ChatGLM3-6B	LLaMA3-8B	Qwen2-7B	Baichuan2-7B
Evaluation	Original	Freeze (l=2)	P-Tuning v2 (p = 16)	LoRA (r = 8)	LoRA (r = 8)			
BLEU−1	24.98	28.25	27.35	**28.91**	34.48	35.78	**36.65**	30.36
ROUGE−1	43.15	47.06	45.12	**47.39**	55.14	56.49	**57.34**	49.89
ROUGE−2	23.79	**28.21**	25.93	27.83	35.08	36.36	**37.54**	31.24
ROUGE-L	38.63	43.14	41.27	**43.65**	51.39	52.67	**53.68**	46.43
METEOR	30.86	34.89	33.46	**35.24**	41.42	42.79	**43.82**	37.62

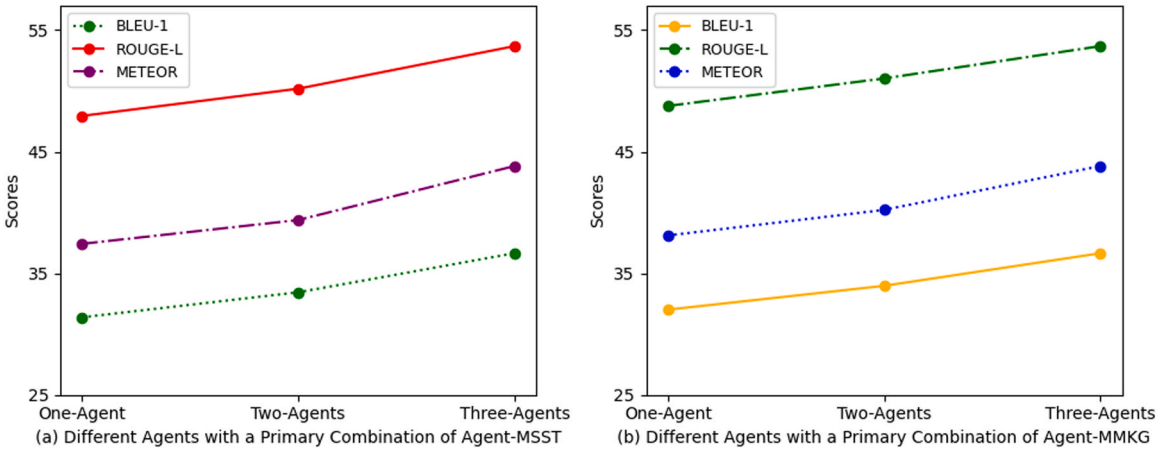


Fig. 9. Results of multi-agent collaboration R^3 models.

Table 11
Performance of various generators in R^3 models with multi-agent collaboration.

Retriever	Qwen2-7B			GPT4o			GPT4		
Configuration	Accuracy	F-1 Match	Average Time	Accuracy	F-1 Match	Average Time	Accuracy	F-1 Match	Average Time
One-Agent	50.8 %	64.5 %	7.3 s	51.5 %	65.9 %	5.8 s	53.6 %	70.0 %	6.1 s
Two-Agents	52.1 %	67.0 %	7.3 s	53.2 %	68.1 %	5.8 s	55.8 %	72.3 %	6.1 s
Three-Agents	54.3 %	69.4 %	8.5 s	55.7 %	70.5 %	6.7 s	57.5 %	74.8 %	7.0 s

Qwen2–7B's response time is approximately 1.3 s longer than GPT4o and GPT4. In the one-agent and two-agent modes, Qwen2–7B's average time is consistent at 7.3 s, while in the three-agents mode, the time increases to 8.5 s. This is primarily due to the addition of Agent-MMKG, which increases processing time for cypher generation, compared to Agent-MMST's calculations.

4.4. Results of multimodal retrieval

SAMAC-R^3-MED method has been applied to develop the eDoChat system, as shown in Fig. 10, illustrating its multimodal information RAG functionality. Based on the wind turbine assembly engineering documents detailed in Table 4, MMST and MMKG were established to form an external knowledge base. The system employs a multi-agent collaboration R^3 strategy to retrieve query-relevant context, including text, image summaries, and table summaries. Qwen2–7B is utilized as the responder for the R^3 model, generating comprehensive textual descriptions and producing multimodal responses with text, tables, and images, as shown on the right side of Fig. 10. The eDoChat system supports various multimodal question-answering functionalities, including joint answers from multiple documents, expert scoring and re-editing of multimodal answers, source document indexing, and viewing chat histories. Expert re-edited answers are fed back in real time through Agent-EF, enabling the retrieval of expert knowledge for subsequent interactions. Additionally, the expert insights gained through the interactive system contribute to the continuous learning and semantic alignment optimization of the R^3 models.

The RAG performance of engineering documents was evaluated across platforms including ChatPDF, AMiner, GPT-4, and eDoChat, with partial results shown in Table 12. Based on the engineering document titled "Tower base electrical platform and cabinet installation.pdf," the query was:" What are the installation steps for the inlet bridge assembly component on the variable frequency drive cabinet?" Expert ratings were provided for the answers from different platforms. Table 12 results show that ChatPDF, AMiner, and GPT-4 deliver text and table-based answers without including image-related information. GPT-4 scores 7.5, providing a complete table and relevant description of the assembly steps. AMiner and ChatPDF offer text-based answers reflecting the table content, but ChatPDF receives the lowest score of 6 due to incomplete table parsing, particularly missing material quantity details. The eDoChat system, built using the SAMAC-R^3-MED method, supports multimodal answer output and delivered the best results, receiving a perfect score of 10 from experts. This validates the accuracy and effectiveness of the SAMAC-R^3-MED, largely due to the construction of MMST and MMKG for engineering documents, enhanced by context-aware multimodal metadata alignment. The multi-agent collaboration strategy and semantic alignment of the R^3 models for multimodal indexing significantly improved the performance of multimodal RAG.

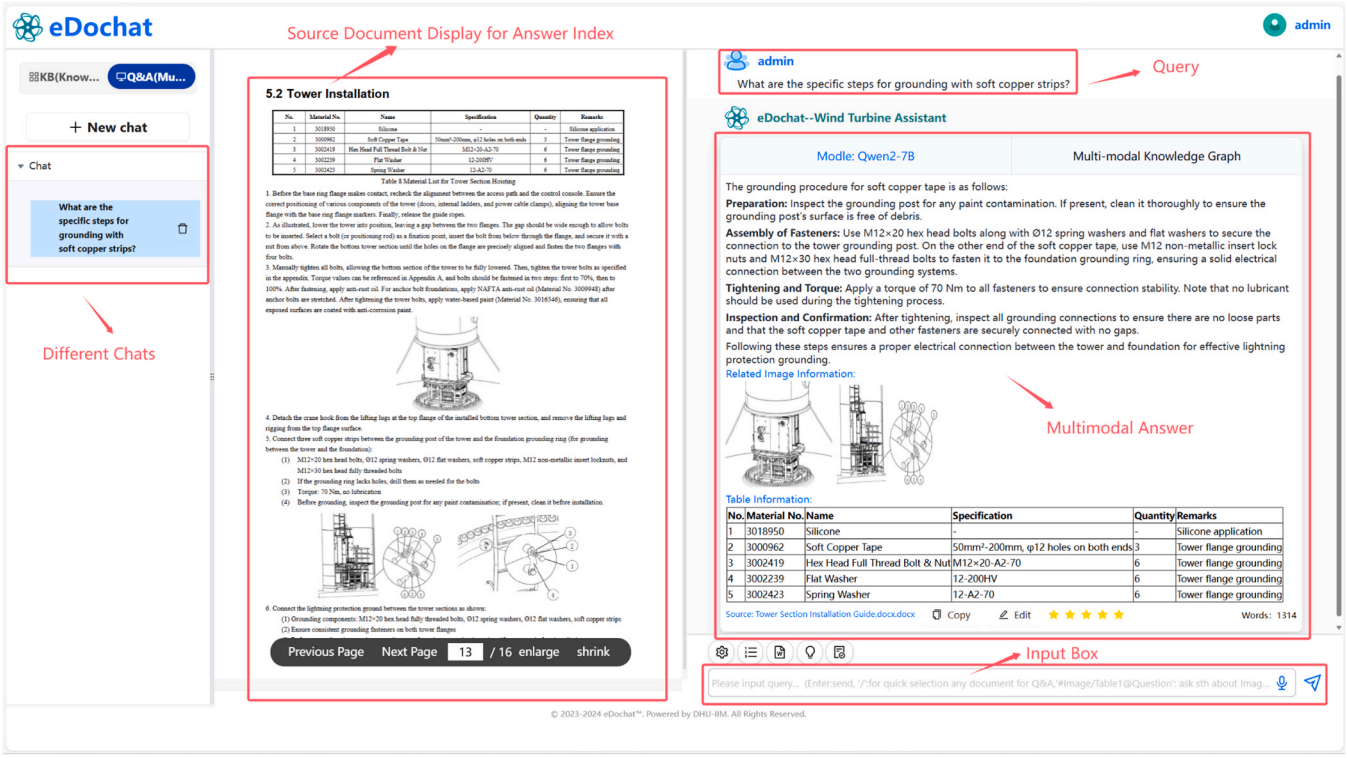


Fig. 10. Multimodal information RAG functionality of the eDoChat system.

5. Discussion and limitations

Engineering documents exhibit greater complexity in structural and multimodal semantic associations compared to general documents. Existing methods struggle with multimodal industrial semantic understanding and alignment, often creating a semantic gap between retriever and generator in RAG models, reducing accuracy. This paper presents the SAMAC-R^3-MED method, which employs a context-aware cross-modal semantic alignment strategy to construct MMST and MMKG. It employs a novel semantic alignment RAG models training framework and integrates the multi-agent collaboration R^3 strategy to effectively implement HybridRAG, enabling multimodal answer generation. Evaluation on real-world industrial retrieval tasks shows that SAMAC-R^3-MED achieves NDCG@10 and MRR@10 scores of 69.2 and 77.5. Table 9 further quantifies the contributions of MMST, MMKG, and R^3 to the performance across abstract, entity based, and multimodal query types through ablation experiments. Extensive comparative and ablation studies demonstrate that the proposed method significantly outperforms existing multimodal RAG approaches in terms of cross modal semantic alignment, bridging semantic gaps in RAG models, and optimizing HybridRAG indexing structures and pipelines.

Context-aware cross-modal structural semantic alignment strategy. By leveraging GMMs clustering and summarization techniques, MLLMs are enhanced to achieve fine-grained, context-aware cross-modal semantic alignment, creating a unified semantic space to bridge cross-modal differences. Additionally, MMST and MMKG are developed for the cross-modal structural alignment of engineering documents, forming a hybrid indexing structure to improve the retrieval of multimodal semantic contexts. Comparative experiments demonstrate that fine-grained image context awareness enriches image semantics, significantly improving F-1 match and accuracy in image-summary tasks. Among the tested RAG modes, including with MMST, with MMKG, and a hybrid of both, HybridRAG performs best, achieving ROUGE-L and METEOR scores of 51.39 % and 41.42 %. These results represent improvements of 4.68 % and 5.06 % over the with MMST mode, confirming the effectiveness of context-aware structured semantic alignment in HybridRAG.

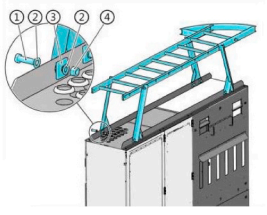
Semantic alignment training framework for RAG models. A training framework for industrial semantic alignment, called R^3, is proposed. This framework employs supervised learning and reinforcement learning strategies with prior ranking feedback to enhance semantic consistency in retrieval and generation tasks. Various SFT methods were tested to adjust LLMs parameters, with the LoRA ($r = 8$) method showing significant improvements in ROUGE-{1,2,L}, BLEU-1, and METEOR metrics compared to original, freeze, and p-tuning v2 methods. In the R^3 framework, a reward model with prior ranking is used to measure relevance. Under the SFT+RM+PPO strategy, LLMs demonstrated significantly better performance compared to using SFT alone. Tests with multiple baseline models and R^3 models in hybrid mode with MMST and MMKG confirmed the superior performance of the R^3 models.

Multi-agent collaboration R^3 models for multimodal RAG. A multi-agent collaboration mechanism is introduced to optimize the RAG pipeline, enabling efficient and adaptive hybrid multimodal retrieval tasks. Three-channel strategies for different agent combinations in R^3 models are configured, involving Agent-EF, Agent-MMST, and Agent-MMKG, utilizing various LLMs trained under the SFT+RM+PPO strategy as responders for answer generation. Test results show that the three-agents mode delivers the best performance, with accuracy and F-1 match scores of 54.3 % and 69.4 %. The GPT-4 responder in the three-agent mode achieves the highest accuracy and F-1 match scores of 57.5 % and 74.8 %, with an average response time of 7.0 s. Ablation experiments confirm that multi-agent collaboration R^3 strategy significantly enhances the flexibility and answer quality of the eDoChat system.

Besides the above advances, the limitations of the proposed method are also discovered in two aspects.

Finer-grained context awareness: It is consistently observed that the context-aware cross-modal structural semantic alignment strategy employed in HybridRAG achieves the most favorable results in multi-modal retrieval tasks for engineering documents compared to other methods. However, semantic associations in vertical-domain

Table 12
RAG results of different platforms for wind turbine assembly guidance engineering documents.

<table>
<tr><th>Question:</th><th>Instance (translated to English with Chinese)</th><th>Expert rating</th></tr>
<tr><td></td><td>What are the installation steps for the inlet bridge assembly component on the variable frequency drive cabinet?</td><td>(1~10)</td></tr>
<tr><td>Chat PDF</td><td>Installation Steps for the Bridge Frame Components on the Variable Frequency Cabinet:
Material Preparation:
Prepare hexagonal head bolts M16×50, flat washers Ø16, hexagonal nuts M16, and other required fasteners.
Bridge Frame Installation:
(1) Fix the bridge frame securely on the platform using four supports.
(2) Secure the bridge frame with eight sets of fasteners, each consisting of a hexagonal head bolt, flat washer, square pad, and hexagonal lock nut.
(3) Tighten the fasteners using a semi-lubricated torque setting of 170 Nm.</td><td>6</td></tr>
<tr><td>AMiner</td><td>Installation Steps for the Bridge Frame Components on the Variable Frequency Cabinet:
Material List: Hexagonal head bolts M16×50 (Qty: 12), Flat washers Ø16 (Qty: 24), Hexagonal nuts M16 (Qty: 12).
Support Installation: Use four supports to secure the bridge frame, fixing it with eight sets of fasteners.
Each fastener set includes: Hexagonal head bolt M16×50-8.8, Flat washer 16-200HV, Square pad ϕ16, and Hexagonal lock nut M16-8. Tighten with a torque of 170 Nm, semi-lubricated.
Ensure adherence to safety standards and operational guidelines for a safe and stable installation.</td><td>7</td></tr>
<tr><td>GPT-4</td><td>Installation of Supports and Bridge Frame:
1. Securely fix the bridge frame in place using four supports, ensuring their stability.
2. Use eight sets of fasteners to secure the bridge frame, with each set including: one M16×50-8.8 hexagonal head bolt, one 16-200HV flat washer, one ϕ16 square pad, and one M16-8 hexagonal lock nut.
Tighten each fastener to 170 Nm using a torque wrench, applying a semi-lubricated state for effective tightening.
Following these steps ensures the stable and safe installation of the bridge frame components on the variable frequency cabinet.
Table Information:
<table>
<tr><th>S/N</th><th>Name</th><th>Specification</th><th>Code</th><th>Quantity</th></tr>
<tr><td>1</td><td>Hex Head Bolt</td><td>M16×50</td><td>GB-T5783</td><td>12</td></tr>
<tr><td>2</td><td>Flat Washer</td><td>Ø16</td><td>GB-T97</td><td>24</td></tr>
<tr><td>3</td><td>Hex Nut</td><td>M16</td><td>GB-T6184</td><td>12</td></tr>
</table></td><td>7.5</td></tr>
<tr><td>eDoChat</td><td>Installation Process for the Inlet Bridge Assembly on the Variable Frequency Drive Cabinet:
1. Secure the bridge at the designated location using four supports.
2. Use eight sets of fasteners for added security. The installation sequence for each set is as follows:
First, place a 16-200HV flat washer on the M16×50-8.8 hexagonal head bolt.
Then, add a ϕ16 square pad and another 16-200HV flat washer.
Finally, secure with an M16-8 hexagonal lock nut. Tighten each fastener with a torque wrench to 170 Nm in a semi-lubricated condition.
Verify that all components are properly installed and tightened to ensure the stability and safety of the assembly.
Related Image Information:
 
Table Information:
<table>
<tr><th>S/N</th><th>Name</th><th>Specification</th><th>Code</th><th>Quantity</th></tr>
<tr><td>1</td><td>Hex Head Bolt</td><td>M16×50</td><td>GB-T5783</td><td>12</td></tr>
<tr><td>2</td><td>Flat Washer</td><td>Ø16</td><td>GB-T97</td><td>24</td></tr>
<tr><td>3</td><td>Hex Nut</td><td>M16</td><td>GB-T6184</td><td>12</td></tr>
</table></td><td>10</td></tr>
</table>

engineering documents are complex and dynamic. The proposed CAI-SSA strategy enhances the understanding of engineering images by sensing and clustering similar semantic context paragraphs. To further improve the quality of multimodal information responses, finer-grained context awareness is necessary to bridge the semantic gaps between modalities more effectively. For example, sentence-level semantic sensing can filter out noisy contexts and identify more precise information, continuously improving image-to-summary task quality.

Enhancing accuracy and explainability: Although Table 9 demonstrates that the full configuration (R^3 + MMST + MMKG) of SAMAC-R^3-MED attains the best results on abstract, entity-based, and multimodal queries, its performance on multi-entity entity-based queries remains modest, with NDCG@10 and F-1 match of only 68.3 and 67.1, respectively. These queries typically involve multiple assembly entities—e.g., motor's main shaft, commutator, bearing, and sealing ring in the example of Table 5, which can introduce noisy nodes during MMKG retrieval, lowering precision and interpretability. Given the chained assembly dependencies among these entities, introducing Chain-of-Thought (CoT) prompting to explicitly reveal the LLM's intermediate reasoning can refine the MMKG retrieval path and deepen the Responder's inference, thereby further improving SAMAC-R^3-MED in multi-entity query scenarios.

6. Conclusion

To address the frequent and large-scale multimodal information retrieval tasks in the manufacturing lifecycle, a HybridRAG method integrating multi-agent collaborative R^3 models with a context-aware cross-modal structural semantic alignment strategy is proposed. The method constructs MMST and MMKG for engineering documents using fine-grained cross-modal context-aware strategies, forming a structured hybrid index. Leveraging a novel R^3 models training framework for semantic alignment, it employs supervised and reinforcement learning with prior ranking feedback to bridge semantic gaps in RAG models for engineering. The approach integrates a multi-agent collaboration R^3 retrieval algorithm, incorporating Agent-EF, Agent-MMST, and Agent-MMKG, enabling HybridRAG with expert feedback support. This optimizes the RAG pipeline, significantly improving the flexibility and answer quality of eDoChat system.

Evaluation based on wind turbine assembly engineering documents shows that SAMAC-R^3-MED significantly improves the comprehensiveness and accuracy of multimodal answers compared to baselines such as VinVL-DPR, UniVL-DR, Marvel-DPR, and VISTA. Ablation experiments demonstrate that applying the CAI-SSA strategy for structured semantic alignment of MMST and MMKG greatly enhances ROUGE-L, BLEU-1, F-1 Match, and accuracy metrics. The R^3 models trained with semantic alignment outperform traditional mDPR and $mE5_{large}$ models. The multi-agent collaboration R^3 retrieval mechanism performs best in the three-agent mode, enhancing multimodal context retrieval via multi-scale metadata in MMST and entity-relationship retrieval in MMKG, compensating for limitations in similarity matching. Expert feedback integration further boosts F-1 match and accuracy. Overall, SAMAC-R^3-MED demonstrates state-of-the-art performance in multimodal retrieval and answer generation tasks.

CRediT authorship contribution statement

Fei Li: Visualization, Software, Methodology, Writing – original draft, Validation, Project administration, Data curation. **Xinyu Li:** Validation, Resources, Formal analysis, Investigation. **Sijie Wen:** Software, Validation, Formal analysis. **Haoyang Huang:** Visualization, Formal analysis, Validation. **Jinsong Bao:** Project administration, Funding acquisition, Supervision, Methodology, Conceptualization.

Declaration of Competing Interest

The authors whose names are listed immediately below certify that they have NO affiliations with or involvement in any organization or entity with any financial interest (such as honoraria; educational grants; participation in speakers' bureaus; membership, employment, consultancies, stock ownership, or other equity interest; and expert testimony or patent-licensing arrangements), or non-financial interest (such as personal or professional relationships, affiliations, knowledge or beliefs) in the subject matter or materials discussed in this manuscript.

Acknowledgements

The authors acknowledge the funding supports from the National Natural Science Foundation of China (No. 52405256), and the Shanghai Rising-Star Plan (Yangfan Program) from the Science and Technology Commission of Shanghai Municipality (No. 22YF1400200).

Data availability

The data that has been used is confidential.

References

Chan, C.M., Xu, C., Yuan, R., et al., 2024. RQ-RAG: learning to refine queries for retrieval augmented generation[J]. arXiv:2404.00610, arXiv Prepr.
Chen, J., Xiao, S., Zhang, P., et al., 2024. Bge m3-embedding: multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation [J]. arXiv:2402.03216, arXiv Prepr.
Ding, Y., Ren, K., Huang, J., et al., 2024. MVQA: a dataset for multimodal information retrieval in PDF-based visual question answering[J]. arXiv:2404.12720, arXiv Prepr.
Edge, D., Trinh, H., Cheng, N., et al., 2024b. From local to global: A graph rag approach to query-focused summarization[J]. arXiv:2404.16130, arXiv Prepr.
Edge, D., Trinh, H., Cheng, N., et al., 2024a. From local to global: a graph rag approach to query-focused summarization[J]. arXiv:2404.16130, arXiv Prepr.
Fatehkia, M., Lucas, J.K., Chawla, S., 2024. T-RAG:lessons from the LLM trenches[J] arXiv:2402.07483, arXiv Prepr.
Gao, L., Ma, X., Lin, J., et al., 2022. Precise zero-shot dense retrieval without relevance labels[J]. arXiv:2212.10496, arXiv Prepr.
Guu, K., Lee, K., Tung, Z., et al., 2020a. Retrieval augmented language model pre-training[C]//International conference on machine learning. PMLR 3929–3938.
Guu, K., Lee, K., Tung, Z., et al., 2020b. Retrieval augmented language model pre-training[C]. Int. Conf. Mach. Learn. PMLR 3929–3938.
Hu, Y., Lei, Z., Zhang, Z., et al., 2024. GRAG: graph retrieval-augmented generation[J]. arXiv:2405.16506, arXiv Prepr.
Izacard, G., Grave, E., 2020. Leveraging passage retrieval with generative models for open domain question answering[J]. arXiv:2007.01282, arXiv Prepr.
Izacard, G., Lewis, P., Lomeli, M., et al., 2023. Atlas: Few-shot learning with retrieval augmented language models[J]. J. Mach. Learn. Res. 24 (251), 1–43.
Kim, W., Son, B., Kim, I., 2021. Vilt: Vision-and-language transformer without convolution or region supervision[C]//International conference on machine learning. PMLR 5583–5594.
Lewis, P., Perez, E., Piktus, A., et al., 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks[J]. Adv. Neural Inf. Process. Syst. 33, 9459–9474.
Li, J., Li, D., Savarese, S., et al., 2023. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. Int. Conf. Mach. Learn. PMLR 19730–19742.
Li, J., Luo, X., Lu, G., 2024a. GS-CBR-KBQA: graph-structured case-based reasoning for knowledge base question answering[J]. Expert Syst. Appl. 257, 125090.
Li, L., Peng, J., Chen, H., et al., 2024b. How to configure good in-context sequence for visual question answering[C]. Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. 26710–26720.
Li, P., Tao, H., Zhou, H., et al., 2025. Enhanced Multiview attention network with random interpolation resize for few-shot surface defect detection[J]. Multimed. Syst. 31 (1), 36.
Li, X., Zhou, Y., Dou, Z., 2024b. Unigen: a unified generative framework for retrieval and question answering with large language models[C]. Proc. AAAI Conf. Artif. Intell. 38 (8), 8688–8696.
Liu, H., Li, C., Li, Y., et al., 2023. Improved baselines with visual instruction tuning[J]. arXiv:2310.03744, arXiv Prepr.
Liu, N.F., Lin, K., Hewitt, J., et al., 2024b. Lost in the middle: how language models use long contexts[J]. Trans. Assoc. Comput. Linguist. 12, 157–173.
Liu, T., Hu, Y., Gao, J., et al., 2024a. Hierarchical multi-modal prompting transformer for multi-modal long document classification[J]. IEEE Trans. Circuits Syst. Video Technol. 34 (7), 6376–6390.
Liu, Z., Xiong, C., Lv, Y., et al., 2022. Universal vision-language dense retrieval: Learning a unified representation space for multi-modal retrieval[J]. arxiv:2209.00179, arxiv Prepr.

Ma, X., Wang, L., Yang, N., et al., 2023. Fine-tuning llama for multi-stage text retrieval [J]. arXiv:2310.08319, arXiv Prepr.
Mitchell, E., Lin, C., Bosselut, A., et al., 2022. Memory-based model editing at scale. PInt. Conf. Mach. Learn. MLR 15817–15831.
Nguyen, Z., Annunziata, A., Luong, V., et al., 2024. Enhancing Q&A with domain-specific fine-tuning and iterative reasoning: a comparative study[J]. arXiv:2404.11792, arXiv Prepr.
Park, S., Joung, J., Kim, H., 2023. Spec guidance for engineering design based on data mining and neural networks[J]. Comput. Ind. 144, 103790.
Pereira, J., Fidalgo, R., Lotufo, R., et al., 2023. Visconde: Multi-document qa with gpt-3 and neural reranking[C]. European Conference on Information Retrieval. Cham. Springer Nature, Switzerland, pp. 534–543.
Qian, C., Cong, X., Yang, C., et al., 2023. Communicative agents for software development[J]. arXiv:2307.07924, arXiv Prepr.
Radford, A., Kim, J.W., Hallacy, C., et al., 2021. Learning transferable visual models from natural language supervision[C]// International conference on machine learning. PMLR 8748–8763.
Saad-Falcon, J., Barrow, J., Siu, A., et al., 2023. Pdftriage: Question answering over long, structured documents[J]. arXiv:2309.08872, arXiv Prepr.
Sarica, S., Han, J., Luo, J., 2023. Design representation as semantic networks[J]. Comput. Ind. 144, 103791.
Sarmah, B., Hall, B., Rao, R., et al., 2024. HybridRAG: integrating knowledge graphs and vector retrieval augmented generation for efficient information extraction[J]. arXiv: 2408.04948, arXiv Prepr.
Sarthi, P., Abdullah, S., Tuli, A., et al., 2024. Raptor: recursive abstractive processing for tree-organized retrieval[J]. arXiv:2401.18059, arXiv Prepr.
Sawarkar, K., Mangal, A., Solanki, S.R., 2024. Blended RAG: improving RAG (Retriever-Augmented Generation) accuracy with semantic search and hybrid query-based retrievers[J]. arXiv:2404.07220, arXiv Prepr.
Sun, W., Yan, L., Ma, X., et al., 2023. Is ChatGPT good at search? investigating large language models as re-ranking agents[J]. arXiv:2304.09542, arXiv Prepr.
Susnjak, T., Hwang, P., Reyes, N.H., et al., 2024. Automating research synthesis with domain-specific large language model fine-tuning[J]. arXiv:2404.08680, arXiv Prepr.
Tao, H., Duan, Q., Lu, M., et al., 2023. Learning discriminative feature representation with pixel-level supervision for forest smoke recognition[J]. Pattern Recognit. 143, 109761.
Touvron, H., Lavril, T., Izacard, G., et al., 2023. Llama: open and efficient foundation language models[J]. arXiv:2302.13971, arXiv Prepr.
Wang, S., Xu, Y., Fang, Y., et al., 2022. Training data is more valuable than you think: A simple and effective method by retrieving from training data[J]. arXiv:2203.08773, arXiv Prepr.
Wang, Y., Lipka, N., Rossi, R.A., et al., 2024. Knowledge graph prompting for multi-document question answering[C]. Proc. AAAI Conf. Artif. Intell. 38 (17), 19206–19214.
Wang, Z., Tao, H., Zhou, H., et al., 2025. A content-style control network with style contrastive learning for underwater image enhancement[J]. Multimed. Syst. 31 (1), 1–13.
Wu, Q., Bansal, G., Zhang, J., et al., 2023. Autogen: enabling next-gen llm applications via multi-agent conversation framework[J]. arXiv:2308.08155, arXiv Prepr.
Yang, T., Mei, Y., Xu, L., et al., 2024. Application of question answering systems for intelligent agriculture production and sustainable management: a review[J]. Resour. Conserv. Recycl. 204, 107497.
Yu, E., Li, J., Xu, C., 2024b. PopALM: popularity-aligned language models for social media trendy response prediction[J]. arXiv:2402.18950, arXiv Prepr.
Yu, Y., Ping, W., Liu, Z., et al., 2024a. RankRAG: unifying context ranking with retrieval-augmented generation in LLMs[J]. arXiv:2407.02485, arXiv Prepr.
Zhou, J., Liu, Z., Xiao, S., et al., 2024. VISTA: visualized text embedding for universal multi-modal retrieval[J]. arXiv:2406.04292, arXiv Prepr.
Zhou, T., Mei, S., Li, X., et al., 2023. MARVEL: unlocking multi-modal capability of dense retrieval via visual module plugin[J]. arxiv:2310.14037, arxiv Prepr.