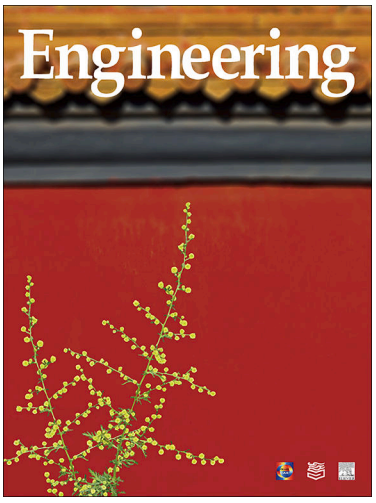
Journal Pre-proofs



Knowledge Enhanced Industrial Question-Answering Using Large Language Models

Ronghui Liu, Hao Ren, Haojie Ren, Wu Rui, Wei Cui, Xiaojun Liang, Chunhua Yang, Weihua Gui

PII: S2095-8099(25)00452-7
DOI: https://doi.org/10.1016/j.eng.2025.07.035
Reference: ENG 2012

To appear in: *Engineering*

Received Date: 3 September 2024
Revised Date: 8 July 2025
Accepted Date: 18 July 2025

Please cite this article as: R. Liu, H. Ren, H. Ren, W. Rui, W. Cui, X. Liang, C. Yang, W. Gui, Knowledge Enhanced Industrial Question-Answering Using Large Language Models, *Engineering* (2025), doi: https://doi.org/10.1016/j.eng.2025.07.035

Research

Intelligent Manufacturing—Article

Knowledge Enhanced Industrial Question-Answering Using Large Language Models

Ronghui Liu [a,c], Hao Ren [b,c,g,*], Haojie Ren [d,f], Wu Rui [c], Wei Cui [a,c,*], Xiaojun Liang [c],

Chunhua Yang [c,e] Weihua Gui [c,e]

[a] *School of Future Technology, South China University of Technology, Guangzhou 510641, China*

[b] *School of Information Science and Engineering, East China University of Science and Technology, Shanghai 200237, China*

[c] *Department of Network Intelligence, Peng Cheng Laboratory, Shenzhen 518055, China*

[d] *State Key Laboratory of Ocean Engineering, Shanghai Jiao Tong University, Shanghai 200240, China*

[e] *School of Automation, Central South University, Changsha 410083, China*

[f] *Shenzhen Research Institute of Shanghai Jiao Tong University, Shenzhen 518063, China*

[g] *Wuhu Overseas Students Pioneer Park, Wuhu 241006, China*

* Corresponding authors.

E-mail addresses: renhao@ecust.edu.cn (H. Ren), aucuiwei@scut.edu.cn (W. Cui).

Abstract: Modern industrial systems have grown increasingly extensive, complex, and hierarchical, with operations relying on numerous knowledge-based queries. These queries necessitate considerable human resources while also requiring high levels of accuracy, subjectivity, and consistency, all of which critically influence operational efficiency. To overcome these challenges, this study proposes an industrial retrieval-augmented generation (RAG) method designed to enhance large language models (LLMs) using domain-specific knowledge, thereby improving the precision of question answering. A comprehensive industrial knowledge base was constructed from diverse sources, including journal articles, theses, books, and patents. A Text classification model based on bidirectional encoder representations from transformers (BERTs) was trained to accurately classify incoming queries. Furthermore, the general text embedding–dense passage retrieval (GTE–DPR) model was employed to perform word embedding and vector similarity retrieval, facilitating the alignment of query vectors with relevant entries in the knowledge base to obtain initial responses. These initial results were subsequently refined by LLMs to produce accurate final answers. Experimental evaluations confirm the effectiveness of the proposed approach. In particular, when applied to ChatGLM2-6B, the RAG method increased the ROUGE-L score from 32.52% to 55.04% and improved accuracy from 50.52% to 73.92%. Comparable improvements were also observed with LLaMA2-7B, underscoring the RAG framework's capability to significantly enhance the accuracy and relevance of industrial question-answering (QA) systems.

Keywords: Retrieval augmented generation; Knowledge enhancement; Question answering; Large language models; Industrial knowledge automation

Smart manufacturing has been established as a strategic national priority aimed at strengthening the global competitiveness of the manufacturing sector. The rapid advancement of artificial intelligence technologies, driven by the proliferation of big data, along with changes in research paradigms, such as the emergence of cyber-physical systems and interdisciplinary knowledge integration, has accelerated the transformation of manufacturing toward digitalization, interconnectivity, and intelligent automation. The fourth industrial revolution is anticipated to facilitate the automation and intelligent execution of knowledge-intensive tasks in manufacturing [1,2]. Nevertheless, current technologies remain inadequate in fully and efficiently exploiting industrial knowledge, thereby hindering the realization of timely intelligent analysis, accurate control, and autonomous decision-making capabilities [3,4]. Consequently, the effective integration and application of industrial knowledge have become essential to improving the performance and competitiveness of industrial systems.

Industrial question-answering (QA) systems are designed to retrieve targeted industrial knowledge and deliver precise answers, rather than merely presenting a ranked list of documents. These systems play a vital role in the integration and practical utilization of industrial knowledge [5]. Since the 1960s, a variety of QA methodologies have developed, which can generally be grouped into three main categories: corpus-based, knowledge graph-based, and large language model (LLM)-based approaches [6]. Corpus-based methods represent the earliest form of QA systems and typically rely on rule-based techniques or similarity vector computations to extract information from large text corpora. For instance, Masood et al. [7] developed an expert system for the selection of rapid prototyping technologies, incorporating data from 39 commercially available systems produced by 21 different manufacturers. Similarly, Ruiz et al. [8] introduced a system that enables hand-free information retrieval for manufacturing personnel, utilizing processed and annotated PDF documents along with similarity vector matching to ensure high classification accuracy and reliable responses. In another example, Liu et al. [9] applied natural language processing (NLP) methods to construct a decision-support system capable of accurately addressing technical queries in pressure vessel design. Although such traditional approaches have demonstrated conceptual effectiveness, they limit efficiency in semantic understanding, conceptual association, and retrieval efficiency [10,11].

Recently, knowledge graph-based methods have emerged as a structured and interpretable approach for industrial QA, facilitating reasoning over entities and their interrelationships [12]. For instance, Han et al. [13] proposed a multi-hop fault diagnosis framework that employs knowledge graph reasoning and decision-making to support fault detection in hot rolling line equipment. In the field of aviation assembly, Liu et al. [14] developed a joint reasoning framework that integrates a named entity recognition model with subgraph embedding techniques, and their findings include a comparative analysis between this approach and LLM-based QA systems. In the chemistry domain, Zhou et al. [15] introduced a novel knowledge graph QA system that utilizes hybrid embeddings to support fact-based information retrieval for chemistry-related research and industrial use cases. Similarly, Lee et al. [16] presented a general-purpose QA model that combines a knowledge graph and QA methods to ensure consistency in time and quality during the review of purchase order documents. Zhou et al. [17] also applied an injection molding knowledge graph in combination with fine-tuned bidirectional encoder representations from Transformers (BERT) model to interpret user intent and retrieve semantically relevant knowledge. Over the past several decades, knowledge graph-based methods have attracted considerable interest due to their interpretability and well-structured representation. Nonetheless, challenges persist, particularly in managing large-scale datasets, adapting to dynamic environments, handling incomplete or inaccurate information, and establishing precise relationships among semantic entities [18,19].

LLMs have been trained on extensive text corpora comprising billions of parameters to enhance natural and interactive communication between humans and machines [20–22]. Hostetter et al. [23] employed advanced chatbot technologies to perform QA tasks in the field of fire engineering, demonstrating the transformative potential of LLMs within this domain. Similarly, Rivera et al. [24] applied LLMs in the coal mining sector, achieving accurate and contextually appropriate answers through the use of tailored prompting strategies. Addressing QA tasks involving tabular data, Mo et al. [25] introduced a novel few-shot table prompting method to mitigate the generation of invalid Structured Query Language (SQL) or Not only SQL (NoSQL) queries by LLMs, particularly in cases involving complex questions and tables with numerous columns. However, LLMs often exhibit limited comprehension of highly specialized industrial knowledge, which can result in responses that appear fluent yet contain factual inaccuracies or fabricated content [26]. To address these limitations, retrieval-augmented generation (RAG) has emerged as a promising approach that combines the generalization ability of pre-trained LLMs with the retrieval of relevant domain-specific knowledge [27–32]. For instance, Wang and Li [33] integrated LLMs with industrial knowledge bases to overcome technical challenges arising from gaps in domain expertise in operations and maintenance. Despite such progress, RAG methods based on LLMs continue to encounter two major challenges: insufficient incorporation of specialized domain knowledge and the persistence of hallucinated outputs in generated responses.

To address the limitations of conventional RAG frameworks, particularly issues related to hallucination and insufficient integration of domain-specific knowledge, this study proposes several enhancements tailored for industrial QA scenarios, such as manufacturing, equipment maintenance, and fault diagnosis. First, the industrial knowledge base is divided into specialized sub-knowledge bases to enable more accurate and efficient retrieval. Second, a BERT-based industrial question classifier is introduced to filter out non-industrial queries and direct relevant questions to the corresponding sub-knowledge base. These improvements

ai of the responses. The main contributions of this work are as follows:

(1) A dedicated industrial knowledge base is constructed by collecting five representative types of documents, including journal articles, dissertations, books, patents, and other relevant materials. These documents undergo preprocessing steps, such as recognition, cleansing, deduplication, and segmentation, to create a structure and focused knowledge base that supports efficient retrieval and utilization.

(2) A BERT-based text classifier is developed and trained to filter questions before retrieving information from LLMs. By screening queries in advance, the classifier helps to ensure that generated answers remain within the bounds of verifiable human knowledge, thereby reducing the likelihood of hallucination outputs.

(3) A knowledge-enhanced LLM is implemented, which retrieves relevant information by processing both user queries and knowledge base content. This approach leverages the general text embedding–dense passage retrieval (GTE–DPR) model for word embedding and uses Facebook artificial intelligence similarity search (FAISS) for vector similarity search, enabling the language model to integrate retrieved domain-specific knowledge into fluent and accurate answers.

(4) Extensive comparative experiments were conducted to evaluate the effectiveness of the proposed industrial RAG framework. The results demonstrate notable improvements across modules, including question classification, knowledge retrieval, and answer generation. In addition, ablation studies highlight the contributions of each component, confirming that their integration significantly enhances the quality and relevance of industrial question answering.

The remainder of this paper is organized as follows: Section 2 reviews relevant literature on LLM, knowledge retrieval techniques, and RAG. Section 3 introduces the proposed QA approach that incorporates industrial knowledge to enhance the performance of LLMs. Section 4 presents the experimental setup and results. The conclusion and potential directions for future research are provided in the final section.

2. Related work

This section reviews the existing literature relevant to the present study, focusing on three primary areas: pre-trained LLMs (Section 2.1), knowledge retrieval methods (Section 2.2), and RAG (Section 2.3). Each of these areas contributes critically to enhancing the performance and accuracy of QA systems within specialized industrial domains.

2.1. Pre-trained LLMs

Pre-trained LLMs refer to large-scale autoregressive models in NLP that are trained on extensive text corpora and contain billions or even trillions of parameters. These models are designed to predict the next token in a sequence based on preceding tokens, with representative examples including PanGu-α [34] and ChatGPT [35]. Given a word sequence $X = \{x_1, x_2, ..., x_n\}$ and its probability distribution $p(X)$, the training objective of LLMs involves maximizing the log-likelihood, as defined in Eq. (1). These models support a wide range of applications, such as machine translation, question answering, and text generation [2,34]:

$$L = \sum_{i=1}^{n} \log p(x_n | x_1, ..., x_{n-1}; \theta) \quad (1)$$

where $p(X) = p(x_n | x_1, x_2, ..., x_{n-1}; \theta)$ denotes the probability of the nth token x_n conditioned on the preceding tokens $x_{1:n-1}$, and θ represents the model parameters to be learned during training.

In 2022, OpenAI's release of ChatGPT drew widespread attention from the global research community and marked a significant shift toward the era of general-purpose generative artificial intelligence [35–38]. This development endowed LLMs with impressive capabilities in general-domain question answering, primarily enabled by the Transformer architecture, which comprises multi-head attention (MHA) mechanisms and fully connected (FC) feed-forward networks (FFN), as illustrated in Fig.

1. g" as an example, through the stacked Transformer layers of the PanGu-α model to predict the next word. Fig. 1(b) details a single Transformer layer, comprising two key sub-modules: multi-head self-attention and the FFN, both encapsulated with LayerNorm and residual connections.

MHA(·): Each self-attention module within a Transformer includes four projection matrices, as depicted in Fig. 1(a): W_h^k,W_h^q, W_h^v,$W_h^m \in \mathbb{R}^{d\times d/N_h}$ where d denotes the hidden dimension, h is the index of the attention head, and N_h is the total number of heads. Given the output from the previous layer $H_{l-1} \in \mathbb{R}^{N\times d}$(with N being the sequence length), the three key components (query Q_h, key K_h, and value V_h) are computed as shown in Eq. (2).

$$\begin{cases} Q_h = H_{l-1}W_h^q \\ K_h = H_{l-1}W_h^k \\ V_h = H_{l-1}W_h^v \end{cases} \quad (2)$$

The attention output is then computed accordingly using Eq. (3).

$$\begin{cases} A_h = Q_h K_h^{\mathrm{T}} = H_{l-1}W_h^q {W_h^k}^{\mathrm{T}} H_{l-1}^{\mathrm{T}} \\ \mathrm{Attention}_h(H_{l-1}) = \mathrm{softmax}(A_h/\sqrt{d})V_h \\ = \mathrm{softmax}(A_h/\sqrt{d})H_{l-1}W_h^v \end{cases} \quad (3)$$

where A_h is the attention score matrix for head h, $Attention_h(\cdot)$ denotes the weighted output for each token, and $\mathrm{softmax}(\cdot)$ normalizes scores across the sequence.

Finally, the output from all attention heads is aggregated and computed using Eq. (4).

$$\begin{cases} \mathrm{MHA}(H_{l-1}) = \sum_{h=1}^{N_h} \mathrm{A\,ttention}_h(H_{l-1})W_h^m \\ H_l^{\mathrm{MHA}} = H_{l-1} + \mathrm{MHA}(\mathrm{LayerNorm}(H_{l-1})) \end{cases} \quad (4)$$

where H_l^{MHA} represents the residual-enhanced output of the multi-head attention block at layer l, and $\mathrm{LayerNorm}(\cdot)$is the layer normalization function.

FFN(·): It consists of two linear layers, represented by $W^1 \in \mathbb{R}^{d\times d_{ff}}$,$b^1 \in \mathbb{R}^{d_{ff}}$,$W^2 \in \mathbb{R}^{d_{ff}\times d}$,$b^2 \in \mathbb{R}^d$, where d_{ff} denotes the dimension of the inner layer, as shown in Fig. 1(b). The output of MHA(·) is passed into FFN(·) to compute the corresponding output, as given by Eq. (5).

$$\begin{cases} \mathrm{FFN}(H_l^{\mathrm{MHA}}) = \mathrm{GeLU}(H_l^{\mathrm{MHA}}W^1 + b^1)W^2 + b^2 \\ H_l = H_l^{\mathrm{MHA}} + \mathrm{FFN}(\mathrm{LayerNorm}(H_l^{\mathrm{MHA}})) \end{cases} \quad (5)$$

where $\mathrm{GeLU}(\cdot)$denotes the Gaussian Error Linear Unit activation function.

Despite the remarkable advancements of large models in language understanding and generation, challenges persist, particularly in domain-specific applications. LLMs often encounter difficulties when handling specialized or knowledge-intensive queries. In scenarios involving topics beyond their training data or requiring current information, these models may generate inaccurate or fabricated outputs. Such limitations underscore the necessity for enhancing the accuracy and reliability of LLMs, especially within specialized and rapidly evolving fields [39,40].

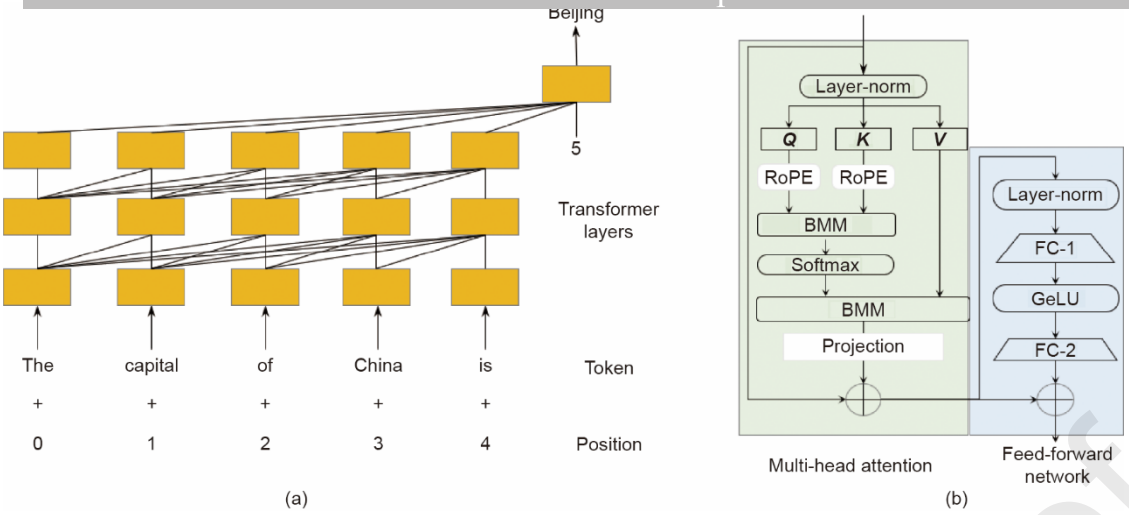


Fig. 1. Architecture overview of large-scale LLMs: (a) Token-level processing workflow of the autoregressive PanGu-PanGu-α model, where embeddings combined with positional encodings are passed through stacked decoder blocks; (b) Inner structure of a Transformer layer, consisting of multi-head self-attention and feed-forward sub-layers, each connected via residual pathways [34]. ***Q***, ***K***, and ***V*** represent the query, key, and value matrices, respectively; BMM denotes batch matrix multiplication used to compute attention scores efficiently; FC-1 and FC-2 refer to the first and second Fully Connected layers within the feed-forward network, respectively.

2.2. Knowledge retrieval

Traditional knowledge retrieval methods primarily rely on sparse retrieval techniques, such as best match 25 (BM25) [41] and term frequency–inverse document frequency (TF–IDF) [42]. These approaches build inverted indexes based on word frequency statistics, enabling efficient retrieval and demonstrating strong recall performance in practical applications, as shown in Eq. (6).

$$\begin{cases} \mathrm{TF\text{–}IDF}(t,d_i,D) = \mathrm{TF}(t,d_i) \times \mathrm{IDF}(t,D) \\ \mathrm{TF}(t,d_i) = \dfrac{f_{tf}(t,d_i)}{\sum_{t' \in d_i} f_{tf}(t',d_i)} \\ \mathrm{IDF}(t,D) = \log \dfrac{n}{|d_i \in D : t \in d_i| + 1} \end{cases} \tag{6}$$

where $\mathrm{TF}(t,d_i)$ represents the frequency of a given term t within a specific document d_i, while $\mathrm{IDF}(t,D)$ captures the rarity of the term t across the entire document set D. The function $f_{tf}(t,d_i)$ counts the occurrences of term t in document d_i, and $\sum_{t' \in d_i} f_{tf}(t',d_i)$ gives the total number of terms (t') in d_i. The variable n indicates the total number of documents in D, and $|d_i \in D : t \in d_i|$ denotes the number of documents in which the term t appears. Despite their effectiveness, traditional sparse retrieval methods primarily rely on term frequency statistics and often overlook the semantic relationships between words [43,44].

Pre-trained language models, such as BERT [45], have significantly improved text understanding in recent years, effectively addressing the limitations of traditional methods that overlook semantic relationships between words. For instance, Karpukhin et al. [46] proposed the dense passage retriever, a BERT-based model that generates dense vector representations through separate query and passage encoders, enabling relevance scoring between query and document vectors. This approach demonstrated notable gains over traditional retrieval models. Furthermore, Wang et al. [47] introduced a hybrid retriever framework that integrates BM25 with dense retrieval models. By applying a convex linear combination (Convex-fusion) to merge sparse and dense retrieval scores, the hybrid retriever effectively compensates for the limitations of each individual method and achieves

2.3. RAG

RAG is a technique that enhances the performance of LLMs by integrating information retrieval with generative capabilities [48]. As described by Lewis et al. [49], this approach strengthens the reasoning ability of LLMs, particularly in domains requiring specialized knowledge. By combining indexing, retrieval, and generation modules, RAG significantly improves the accuracy and reliability of responses, making it especially effective for knowledge-intensive tasks [50,51].

With advancements in technology, RAG has developed into more complex systems that have improvements in indexing techniques, such as sliding windows, fine-grained text segmentation, and metadata fusion [48,50]. For instance, Eibich et al. [52] employed hypothesis document embeddings (HyDEs) combined with LLM-based re-ranking to enhance retrieval accuracy. Similarly, Zhou et al. [53] introduced meta-knowledge summaries to personalize queries and increase retrieval depth. Despite its effectiveness, RAG continues to face challenges, such as limited retrieval efficiency and a lack of diversity in generated content.

Recent studies have increasingly emphasized modular RAG architectures, which offer improved flexibility and scalability by refining individual components, such as integrating similarity search modules and fine-tuning retrievers [54]. Common strategies include reorganizing RAG modules [55] and reconfiguring retrieval pipelines [56]. The adoption of modular RAG methods is growing, enabling sequential processing and facilitating end-to-end integration. Representative examples include FlashRAG [57], Telco-RAG [58], and AutoRAG [59]. Nonetheless, current RAG-based systems continue to encounter limitations, primarily due to the inclusion of irrelevant or conflicting information and difficulties in handling complex query formulations [50].

To mitigate the impact of irrelevant or contradictory external information, which can result in inaccurate or hallucinated responses, Yan et al. [60] proposed the corrective RAG (CRAG) framework. This method incorporates lightweight retrieval evaluators to assess the quality of retrieved content and expands retrieval coverage through large-scale web searches, thereby improving the robustness and reliability of the generated responses. Despite these advancements, RAG-based systems still face challenges in addressing complex queries with high precision. To improve performance in multi-hop question answering tasks, Chan et al. [61] proposed refining queries through explicit rewriting, decomposition, and disambiguation. In addition, Hei et al. [62] tackle the issue of dynamic relevance by improving document retrieval recall and answer accuracy using a two-stage retrieval framework supported by compact classifiers.

In industrial QA systems, the objective is not to rely on LLMs for generating conversationally fluent responses. Instead, these systems prioritize the retrieval of highly accurate and reliable answers by leveraging human-validated external cognitive knowledge bases. This involves the comprehension, application, and transformation of factual data, along with structured summarization and interpretation through LLMs. Therefore, industrial applications necessitate access to specialized and domain-specific knowledge that may not be present in the pre-training data of LLMs. It is also crucial to prevent hallucinated or fabricated outputs commonly associated with generative models.

3. Methodology

This section first presents an overview of the proposed framework. It then provides a detailed description of the industrial knowledge base. Subsequently, query analysis is presented, followed by the design of the classification module. The final subsection describes the knowledge retrieval and ranking mechanism designed to support industrial QA.

3.1. Framework overview

Inspired by the interactive response capabilities of LLMs and the flexibility of integrating domain-specific external knowledge, this work proposes an industrial QA method. This method enhances LLMs by utilizing industrial knowledge to enable accurate, efficient, and automated knowledge retrieval. As illustrated in Fig. 2, the framework consists of four main components: industrial knowledge base construction; industrial question classifier and retrieval interface; industrial knowledge retrieval; and response generation using LLMs.

Construction of industrial knowledge base (Fig. 2(a)): Industrial queries often span domains, such as electrical engineering, materials science, chemistry, construction, and related fields. This type of knowledge reflects human understanding, utilization, and adaptation of the natural environment. To support efficient retrieval and QA, relevant information from these five core domains is systematically summarized and organized. The knowledge base is developed by cleaning, tokenizing, and segmenting content from journal articles, dissertations, and production-related data, thereby refining the scope of retrieval and improving

Industrial question classifier (Fig. 2(b)): This component classifies incoming industrial questions to define the response scope of the system and further refine the knowledge retrieval process. A BERT-based text filter is used to screen and categorize questions, evaluating their relevance to the industrial domain. It also aligns the questions with the appropriate sub-knowledge base.

Industrial knowledge retrieval (Fig. 2(c)): This module employs dense retrieval methods to identify the most relevant information fragments from the knowledge base, ensuring both accuracy and relevance. Initially, related texts are extracted from a broad corpus, which is then used to enhance the ranking performance during retrieval.

LLM prompted responses (Fig. 2(d)): This component converts retrieved knowledge chunks into coherent and accurate responses suited to human comprehension. The knowledge fragments are combined with user questions to form a prompt template, which is input into LLMs using prompt engineering techniques. The generated responses are then delivered to the user.

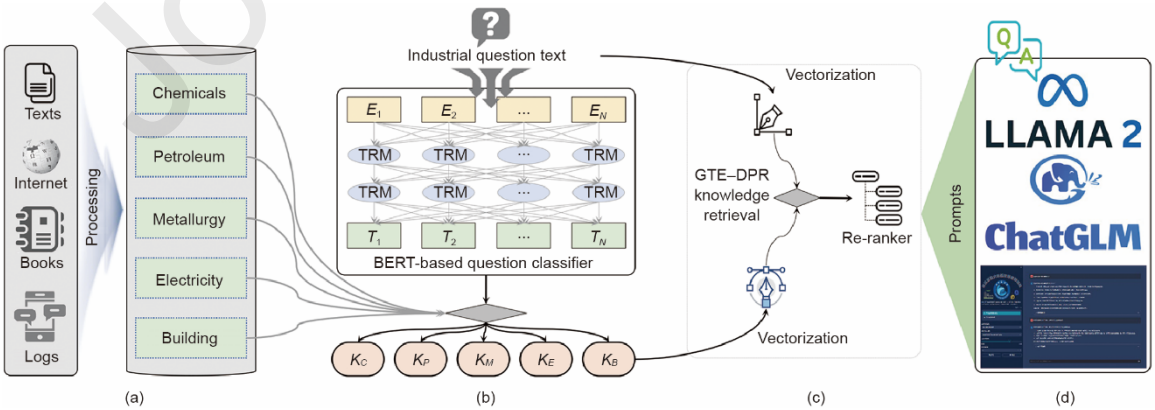
3.2. Industrial knowledge base construction

A precise and domain-specific knowledge base is crucial for effective industrial QA. The construction process begins by converting documents into editable text, followed by the use of regular expressions to remove elements, such as tables of contents, images, annotations, and blank spaces. Subsequent steps include data cleaning and the removal of redundant content to ensure the textual material is complete and coherent. A predefined stop-word list is employed to eliminate special symbols and non-informative words, such as “some” and “many.” After this, tokenization is applied to divide the text into individual words while maintaining sentence-level meaning. Text chunking is then carried out to group semantically related content, using newline characters, specifically "\n" and "\n\n", to segment the text accordingly.

Given that industrial texts are rich in technical terminology and generally composed of short sentences, the maximum chunk length is limited to 256 characters, with a maximum overlap of 32 characters between chunks. This overlapping technique ensures semantic continuity between segments. The final knowledge base is stored in an editable format using the structure “Title: Chunk.” Furthermore, based on the source content, the knowledge base is divided into five sub-domains corresponding to major industrial sectors: chemical engineering (K_{C}), petroleum (K_{P}), metallurgy (K_{M}), electricity (K_{E}), and building materials (K_{B}), as shown in Eq. (7).

$$K = f_p((K_{\mathrm{C}},K_{\mathrm{P}},K_{\mathrm{M}},K_{\mathrm{E}},K_{\mathrm{B}}),L,O) \quad (7)$$

where K represents the industrial knowledge base, L is the chunk length, O is the character overlap between chunks, and $f_p(\cdot)$ denotes the processing function applied to the industrial corpus. The parameter $L = 256$ specifies the maximum chunk length, while $O = 32$ indicates the maximum character overlap between chunks.



Fi four main components: (a) industrial knowledge base construction, (b) industrial question classifier and knowledge retrieval interface, (c) industrial knowledge retrieval, and (d) response generation using improved LLMs. E_1,E_2,...,E_N denote the token embeddings; T_1,T_2,...,T_N are the corresponding token representations; and TRM refers to a transformer block used for contextual encoding.

3.3. Question classifier for narrowing retrieval scope

LLMs, such as ChatGPT [35], often generate inaccurate or misleading responses when addressing queries from specialized or interdisciplinary domains. Let Q_a denote the complete set of input questions, R represent the subset of questions that fall within a specific domain's scope, and C indicate the subset of questions for which the model can provide professional and accurate answers. It follows that $C \leq R \leq Q_a$. To constrain queries within the domain-relevant subset R, a question text filter is employed to map the broader set Q_a into R, thereby increasing the proportion of relevant and answerable queries. This process is formalized in Eq. (8).

$$R = \aleph_t(Q_a) \quad (8)$$

where $\aleph_t(\cdot)$ denotes the question text filter, which determines whether a given user query is relevant to the industrial domain. Questions identified as irrelevant are excluded from further processing to prevent ambiguity and hallucinations commonly associated with LLMs. If a query is deemed relevant, it is further classified to determine its corresponding industrial subdomain, such as chemical, petroleum, metallurgy, electricity, or building materials, and is then mapped to the appropriate sub-knowledge base K_T. The process of industrial question classification is further defined by Eq. (9).

$$\begin{cases} K_T = f_{\text{cls}}(f_{\text{flt}}(Q)) = \{k \in K | \text{Ctg}(k) = f_{\text{cls}}(R)\} \\ R = \{Q \in Q_a | f_{\text{flt}}(Q) = 1\} \end{cases} \quad (9)$$

where K_T denotes the sub-knowledge base associated with the relevant query category, $k \in K$ represents an individual knowledge item within the industrial knowledge base, and $\text{Ctg}(k)$ denotes its assigned category label, and the function f_{flt} is responsible for filtering questions based on industrial relevance, while f_{cls} classifies the filtered questions into their corresponding industrial subdomains. Let Q represent a user-submitted query, and the transition $Q_a \rightarrow R$ indicates the mapping of all input questions to the subset that pertains to the industrial domain. The label T identifies the specific category to which the question belongs, namely, metallurgy, petroleum, chemical, electricity, or building materials.

Fig. 3 illustrates the process involving the industrial question filter f_{flt} and the classifier f_{cls}, which determine whether a question is relevant to the industrial domain and identify its corresponding category. The industrial text filter f_{flt} is built on a BERT-based model, which assesses the necessity of a response and assists the language model in generating suitable answers. Within this framework, the BERT network functions either as a filter (upper path) or as a classifier (lower path), enabling text vectorization of the input questions. The primary distinction between the two lies in their classification objectives: the filter performs binary classification, whereas the classifier addresses a multi-class classification task. Final classification outcomes are derived by feeding the question embeddings into an FC layer, as defined in Eq. (10).

$$P(G|H) = \text{softmax}(W \cdot H) = \text{FC}(H) \quad (10)$$

where W denote the parameter matrix of the classification network (an FC network), which is fine-tuned by maximizing the log-likelihood of the correct label. G is the predicted category label, H denotes the hidden representation from the "[CLS]" token, and "[CLS]" is a special token in BERT used to summarize the input sequence for classification. The classification output is derived from the "[CLS]" token representation, based on two contributing elements that enhance the model's interpretability. The classifier distinguishes among six categories: non-relevant, metallurgy, petroleum, chemistry, building materials, and electricity. During training, only the parameters of the FC layer are updated. Although the filter and classifier can be unified into a single multi-class classification model, they are discussed separately in this work to improve clarity and interpretability.

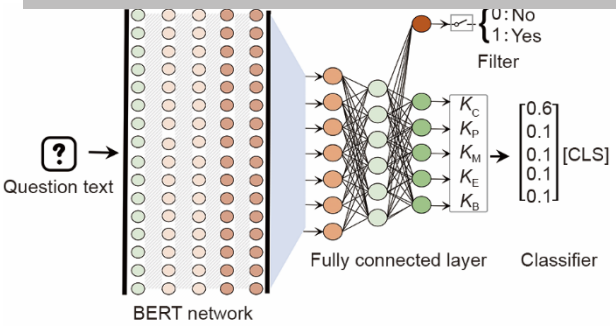


Fig. 3. Process of filtering and classifying industrial questions to enable precise knowledge retrieval.

In this study, the dimension of the BERT-based question text embedding is set to 768. The FC network comprises two hidden layers with dimensions of 384 and 768. The output layer has a dimension of 6, corresponding to the industrial categories, along with an additional class for non-relevance. The non-relevance class serves to constrain responses within the boundaries of human-validated cognitive knowledge bases, focused on the understanding, utilization, and transformation of nature, rather than enabling unrestricted, human-like interactive responses.

3.4. Industrial knowledge retrieval

Knowledge retrieval is a key component of the proposed industrial QA framework. This process involves mapping both user queries and knowledge base content into a continuous vector space using a pre-trained LLM. The similarity between the query and individual knowledge fragments is then computed to identify initial candidate segments. These candidates are subsequently ranked based on their similarity scores, as defined in Eq.(11).

$$K_Q = \{K_T^j, j := \arg_j \max(\mathrm{sim}(q, K_j))\} \quad (11)$$

where sim() denotes the similarity function, such as FAISS or cosine similarity. The variable $q \in Q_a$ represents a question within the set of answerable queries Q_a, and $K_j \in K$ refers to the jth knowledge chunk identified as relevant to q, where j is the index of the chunk. $\arg_j \max()$ denotes the index j at which the maximum value is attained. K_Q denotes the final set of knowledge fragments retrieved from the corresponding sub-knowledge base K_T that matches the query domain. This retrieval process aims to maximize recall of relevant information and extract the top-K most informative context segments.

DPR is employed using a dual-encoder architecture to support industrial knowledge retrieval [46], as shown in Fig.4. This architecture enables efficient retrieval of large volumes of text chunks by replacing the traditional BERT model with the general text embedding (GTE-large-zh) model, which generates 1024-dimensional word embeddings [63]. GTE-large-zh demonstrates enhanced capability in capturing complex semantic information and accurately representing word relationships. To improve similarity matching across high-dimensional text vectors, a similarity matrix $s \in \mathbb{R}^m$ is computed between the DPR-based query embeddings and the embeddings of knowledge chunks using the FAISS library, based on L_2 distance [64]. A smaller L_2 distance corresponds to a higher degree of similarity.

Given the varying precision levels of different retrieval engines, the quality of the retrieved results can be inconsistent. Therefore, as expressed in Eq.(12), it is essential to perform an additional ranking step to accurately filter and organize the final set of relevant contextual information.

$$K_Q^r = \{K_Q^i, i := \arg_i \mathrm{sort}(\mathrm{Scores}(q, K_i))\} \quad (12)$$

where K_Q^r denotes the rankedd set of knowledge chunks K_Q^i that are most relevant to the query q, obtained through a sorting function Sort(). K_i denotes the ith knowledge fragment in the retrieved set, and $\arg_i \mathrm{sort}()$ denotes the index i corresponding to the sorted order of values. In this study, the BGE-reranker-large model is employed for this purpose. It is trained on a multilingual

da uage scenarios.

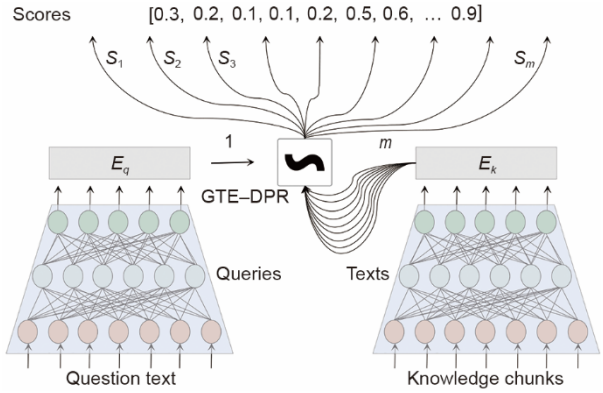


Fig. 4. Architecture of the DPR-based retrieval framework for computing relevance scores between questions and knowledge chunks. m is the total number of knowledge chunks; S_1,...,S_m denote the similarity scores between the question and each chunk; E_q and E_k are the embeddings of the question and knowledge chunks, respectively.

3.5. Industrial QA and evaluation

After retrieving the relevant contextual knowledge, each knowledge chunk is combined with the user's original question and sequentially input into the LLM. The model integrates the provided knowledge to generate accurate and controlled responses aligned with the query. Consequently, the answers are grounded in validated cognitive knowledge bases rather than relying on generic, conversational outputs that imitate human expression. This process is formally described in Eq.(13).

$$A_k = \mathrm{LLMs}\left(M \cdot K_Q^r\right), M \overrightarrow{\propto} \mathrm{Scores} \quad (13)$$

where A_k denotes the final answer generated by the LLM. The term M represents virtual weight matrices used to emphasize shared or overlapping knowledge across the ranked knowledge chunks K_Q^r. The variable Scores refers to the sorted relevance scores between the query and the retrieved knowledge chunks. $\overrightarrow{\propto}$ denotes element-wise proportionality between vectors.

As shown in Table1, the effectiveness and reliability of the proposed method are evaluated using several standard metrics. For the question classification module, precision (P), recall (R), and F_1 score are adopted as evaluation indicators [8,11,16,26]. The performance of the knowledge retrieval module is assessed using mean reciprocal rank (MRR) [65]. Finally, for the answer generation module, recall-oriented understudy for gisting evaluation-longest common subsequence (ROUGE-L) and accuracy (ACC) are used to evaluate the quality and correctness of the generated responses [66].

Table 1

Evaluation metrics employed to measure the performance of the proposed industrial QA approach.

Items	Metrics	Description	Equation
Question classification	Precision (P)	Proportion of correctly predicted for all predicted positives.	$\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}}$
	Recall (R)	Proportion of actual positive samples correctly identified.	$\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}$
	F_1 score	Harmonic mean of precision and recall for balance.	$\frac{2PR}{P+R}$
Knowledge retrieval	MRR	Correct answer position in ranked retrieval results.	$\frac{1}{\lvert Q\rvert}\sum_{i=1}^{\lvert Q\rvert}\frac{1}{\mathrm{rank}_i}$
QA	ROUGE-L	Matching similarity between the generated and ground-truth.	$\frac{(1+\beta^2)R_A P_A}{R_A+\beta^2 P_A}$
	ACC	Experts-based ACC of the LLMs-generated answers.	$\frac{\mathrm{CR}}{\mathrm{Num}}$

where TP and FP represent the numbers of true positives and false positives, respectively, while TN and FN denote true negatives and false negatives, respectively. $|Q|$ indicates the total number of queries, and rank_i refers to the rank position of the first correct answer for the ith query. The ROUGE-L R and P are defined as $R_A = \frac{\mathrm{LCS}(X,Y)}{L_m}$ and $P_A = \frac{\mathrm{LCS}(X,Y)}{L_n}$, respectively, where X and Y denote the ground-truth and generated texts, respectively; L_m and L_n are their corresponding lengths; and $\mathrm{LCS}(X,Y)$ is the longest common subsequence between X and Y. The parameter β is a hyperparameter that adjusts the balance between P and R. CR indicates the number of correct responses, while Num represents the total number of evaluated instances.

4. Experimental study

Industrial QA emphasizes the retrieval of validated cognitive knowledge, as opposed to generating conversational, human-like responses. Accordingly, the experimental study is designed to evaluate key components of the proposed method, including knowledge base construction, question classification, knowledge retrieval, and answer generation performance. An ablation study is also conducted to verify the effectiveness of each module within the framework.

4.1. Description and setup

The industrial knowledge base comprises approximately 12 410 documents accumulated over the past two decades. These include 3462 journal articles, 1689 theses, 210 books, 5477 patents, and various other materials related to industrial production. All documents were converted into editable text, with elements, such as tables of contents, images, annotations, blank spaces, and redundant content removed. Special characters and non-informative words were processed using a stopword list. To preserve sentence integrity, the texts were tokenized into individual words, and newline characters ("\n" and "\n\n") were identified to ensure continuity of content. The knowledge base was structured using a chunking method with a maximum chunk length of 256 characters and an overlap of 32 characters between chunks. Texts were stored in a "Title: Chunk" format, resulting in a total of 107 045 retrievable segments.

Industrial QA datasets were derived from the constructed knowledge base. In this process, questions were generated by LLMs based on corresponding answers and subsequently reviewed and refined by domain experts to ensure their ACC. To support the

ev eum, chemicals, electricity, and building materials.

In the question classification experiment, the design adhered to the principle of controlling computational complexity without compromising the validity of the experimental conclusions. The classification model was configured with a maximum chunk length of 128, a batch size of 16, a dropout rate of 0.1 to mitigate overfitting, cross-entropy as the loss function, and the Adam optimizer with an initial learning rate of 1×10^{-5}. The model was trained for 3 epochs using a batch size of 16. For performance comparison and validation, four baseline models were employed: FastText [67], TextCNN [68], TextRNN [69], and TextRCNN [70], using the parameter configurations specified in their respective original studies.

For knowledge retrieval and re-ranking, the parameter k is set to five or ten, and three models are used for the corresponding experiments: General DPR [46], Domain DPR [63], and Domain + GTE–DPR.

For the industrial QA task, two publicly available LLMs, ChatGLM2-6B and LLaMA2-7B, are selected as the base models. The generation parameters are configured with a temperature of 0.7 and a maximum output length of 500 tokens. All experiments are conducted on the same hardware setup, which includes two NVIDIA GeForce RTX 3090 GPUs (Peng Cheng Laboratory, China) and a 104-core Intel(R) Xeon(R) Gold 6230R CPU ((Peng Cheng Laboratory) running at 2.10 GHz.

4.2. Results of question classification

Question classification is an essential function in industrial QA systems, as it determines the appropriate handling of user queries. As previously noted, the classification model is implemented using BERT. A classification dataset was constructed by randomly selecting 7800 question samples from the full dataset, with equal representation across six categories: metallurgy, petroleum, chemicals, electricity, building materials, and non-industrial. The datasets were divided into three parts: 6000 samples for training, 1200 for testing, and 600 for validation.

Fig. 5 presents the performance of the proposed industrial question classification system. In Fig. 5(a), the classification results on the validation set are shown. The confusion matrix reveals three misclassifications of petroleum as chemical and three misclassifications of chemicals as petroleum. Overall, the validation set achieves over 95 correct predictions out of 100 samples for each category. Fig. 5(b) illustrates the results on the test set, where confusion is again observed between petroleum and chemical categories, with 11 petroleum questions misclassified as chemicals and 8 chemicals questions misclassified as petroleum. This misclassification suggests a degree of similarity between content in these two domains. On the test set, the classification model achieves more than 185 correct predictions out of 200 samples per category, with particularly strong performance in the metallurgy and electricity categories. These results confirm that the BERT-based classification model performs effectively for industrial classification.

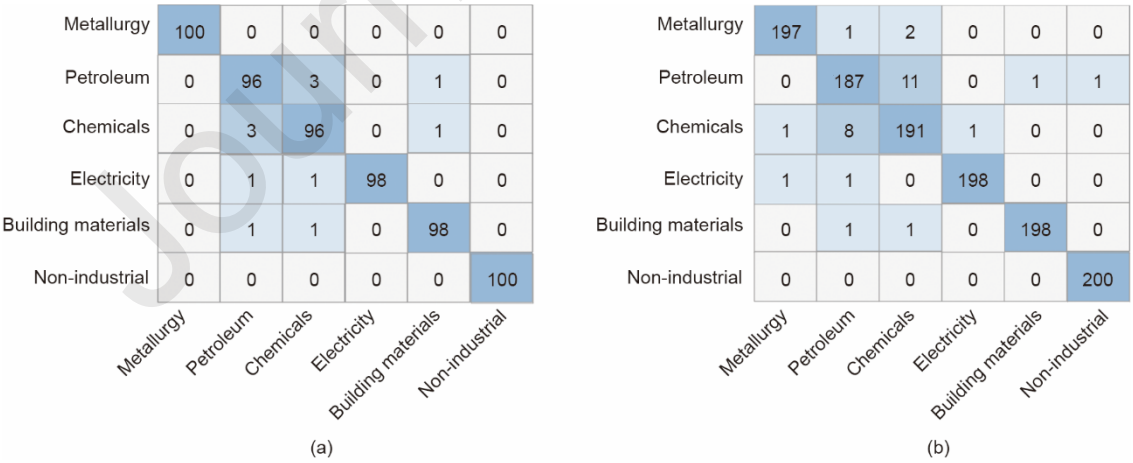


Fig. 5. Confusion matrix of classification accuracy for (a) the validation set with 600 samples and (b) the test set with 1200 samples from a single experiment.

To further assess the performance of the BERT-based question classification model, comparative experiments were conducted against four baseline models: FastText [67], TextCNN [68], TextRNN [69], and TextRCNN [70]. Each experiment was executed ten times to ensure the consistency and reliability of the classification outcomes. Table 2 presents the comparison results in terms of precision, recall, and *F*1-score. On the test set, the proposed model consistently outperformed all baseline models. In particular, it achieved a mean *F*1-score of 97.79%, which is approximately 20.17% higher than the next best-performing model, TextRCNN. Furthermore, across ten independent runs, the BERT classifier exhibited a standard deviation of only 0.23% in *F*1-score, whereas all neural baseline models showed standard deviations of at least 0.8%. This low variance demonstrates the model's robustness to random initialization and data shuffling, an important attribute for industrial environments where frequent retraining may be required.

Table 2

Comparison of the proposed model and four baseline models for question classification on the test set over ten experimental runs.

Models	Precision (%)				Recall (%)				*F*1-score (%)			
	Mean	Max	Min	Std	Mean	Max	Min	Std	Mean	Max	Min	Std
FastText [67]	65.12	66.61	63.88	0.75	65.13	66.65	64.2	0.71	64.87	66.52	63.67	0.80
TextCNN [68]	77.17	78.67	75.63	0.83	77.44	78.83	75.92	0.84	77.23	78.71	75.74	0.83
TextRNN [69]	75.73	79.08	74.50	1.34	76.33	78.75	75.50	0.98	75.74	78.49	74.76	1.13
TextRCNN [70]	75.45	76.96	72.81	1.10	76.27	77.58	74.17	0.92	77.62	78.11	74.92	1.13
BERT-based	**97.81**	**98.16**	**97.42**	**0.23**	**97.79**	**98.16**	**97.41**	**0.23**	**97.79**	**98.16**	**97.41**	**0.23**

4.3. Results of knowledge retrieval

The knowledge retrieval dataset was divided into 100 000 queries for training and 7045 queries for testing. To evaluate the effectiveness of the proposed retrieval strategy, performance was assessed using top-k retrieval metrics with $k = 5$ and $k = 10$. Three models were considered for comparison: General DPR [46], Domain DPR [63], and the proposed Domain + GTE–DPR model. Each experiment was conducted five times to ensure the consistency and reliability of the results.

Table3 summarizes the comparative performance of the models based on MRR@5 and MRR@10. The proposed Domain + GTE–DPR model outperformed both baselines across all metrics. Specifically, it achieved a mean MRR@5 of 90.12% and a mean MRR@10 of 90.28%, surpassing the retrieval accuracy of the other approaches. Moreover, the model exhibited the lowest deviations among all methods, indicating excellent stability and robustness. The findings underscore the effectiveness of domain-specific training, as evident from the performance improvements of the domain DPR model over the Domain DPR baseline.

Table 3

Comparison of the proposed model and three baseline models for knowledge retrieval on the test set across five experimental runs.

Models	MRR@5 (%)				MRR@10 (%)			
	Mean	Max	Min	Std	Mean	Max	Min	Std
General DPR [46]	4.66	4.66	4.65	0.01	4.99	4.99	4.98	0.00
Domain DPR [63]	88.13	88.17	88.10	0.03	88.39	88.39	88.38	0.00
Domain + GTE–DPR	**90.12**	**90.13**	**90.12**	**0.00**	**90.28**	**90.28**	**90.28**	**0.00**

Further enhancement was achieved by replacing the BERT-base-Chinese encoder with the GTE-base-zh embedding model, which improved both the accuracy and consistency of retrieval in the Domain + GTE–DPR configuration. This result highlights the importance of domain adaptation and the use of semantic embeddings in optimizing industrial knowledge retrieval. In addition, the near-zero variance (standard deviation = 0.00) observed for the proposed model reflects its exceptional consistency across repeated trials. By contrast, although the Domain DPR model achieved relatively strong results, it displayed minor performance variability, whereas the General DPR baseline showed lower retrieval accuracy and higher consistency. These observations confirm that general-purpose models are inadequate for handling domain-specific knowledge tasks in industrial settings without appropriate domain adaptation.

4.4. Industrial QA generation performance

Due to limitations in computational resources, the QA generation experiments were performed using two LLMs: ChatGLM2-6B and LLaMA2-7B. The dataset used for these experiments was consistent with that of the knowledge retrieval task and comprised the same set of queries and documents. From this dataset, a subset of 1000 QA pairs was randomly selected for evaluation. To ensure the reliability of the results, each experiment was conducted five times.

ings: without and with the integration of the RAG framework. The first approach, referred to as the LLM-based method, involved directly inputting questions into the LLM without any supporting retrieved content, thereby serving as the baseline. The second approach, termed the RAG-based method, provided the LLMs with additional contextual information in the form of relevant documents, which were obtained through the proposed classification and retrieval modules. This setup was designed to evaluate the extent to which the incorporation of retrieved knowledge improves the accuracy and quality of generated answers in industrial domains.

Table4 presents a comparative analysis of the industrial QA performance of two LLMs, ChatGLM2-6B and LLaMA2-7B, under LLM-based and RAG-based configurations. The results clearly demonstrate that the integration of the RAG framework significantly enhances both the quality and accuracy of the generated responses. Specifically, for ChatGLM2-6B, the average ROUGE-L score increased from 32.52% to 55.04%, while the average ACC improved from 50.52% to 73.92%. Similarly, for LLaMA2-7B, the ROUGE-L score rose from 29.54% to 54.00%, and average ACC increased from 42.92% to 70.60%. These improvements indicate that incorporating relevant retrieved documents into the prompt context enables the models to generate more accurate and contextually appropriate responses. Furthermore, the reduction in standard deviation across metrics suggests enhanced stability and consistency in output quality.

Table 4

Comparison of industrial QA performance of two LLMs with and without RAG using 1000 QA pairs.

Models	Frameworks	ROUGE-L (%)				ACC (%)			
		Mean	Max	Min	Std	Mean	Max	Min	Std
ChatGLM2-6B	LLM-based	32.52	35.12	31.75	1.45	50.52	51.3	49.7	0.58
	RAG-based	**55.04**	**55.32**	**54.32**	**0.41**	**73.92**	**74.7**	**73.0**	**0.72**
LLaMA2-7B	LLM-based	29.54	29.69	29.38	0.18	42.92	44	42.3	0.72
	RAG-based	**54.00**	**54.71**	**53.09**	**0.69**	**70.60**	**71.50**	**69.10**	**0.98**

The RAG framework yields considerable absolute gains, exceeding 22% in ROUGE-L and between 23% and 28% in ACC across both models, underscoring its effectiveness in improving factual ACC. The marked reduction in standard deviation, particularly in ROUGE-L (from 1.45% to 0.41% for ChatGLM2-6B), further validates that RAG not only enhances average performance but also ensures more consistent and dependable answer generation.

To assess the individual contributions of the question classification and knowledge retrieval modules within the proposed RAG framework for industrial knowledge integration, an ablation study was conducted. Three configurations were evaluated: the baseline ChatGLM2-6B model without any auxiliary modules; ChatGLM2-6B augmented with the question classification module; ChatGLM2-6B enhanced with both the question classification and knowledge retrieval modules.

The results of this study are summarized in Table5. The baseline configuration achieved a mean ROUGE-L score of 32.52% and a mean ACC of 50.52%. Introducing the question classification module led to marked improvements, raising the ROUGE-L and ACC scores to 46.40% and 66.06%, respectively. These gains are attributed to the classifier's ability to filter out irrelevant queries and direct appropriate inputs to the retrieval process. When both the question classification and knowledge retrieval modules were applied, the model achieved a further increase in performance, with the ROUGE-L mean score reaching 55.04% and ACC improving to 73.92%.

Table 5

Comparison of industrial QA performance of two LLMs with and without RAG using 1000 QA pairs.

Configurations	ROUGE-L (%)				ACC (%)			
	Mean	Max	Min	Std	Mean	Max	Min	Std
ChatGLM2-6B	32.52	35.12	31.75	1.45	50.52	51.3	49.7	0.58
ChatGLM2-6B + Classifier	46.40	46.97	45.81	0.49	66.06	66.7	65.2	0.59
ChatGLM2-6B + Classifier + Retrieval	**55.04**	**55.32**	**54.32**	**0.41**	**73.92**	**74.7**	**73.0**	**0.72**

These results demonstrate that each module contributes incrementally and significantly to overall system performance. The classifier alone enhanced ACC by over 15 percentage points, highlighting its role in refining input relevance. The addition of the retrieval module further improved factual consistency and completeness. Moreover, the standard deviation of the ROUGE-L score decreased from 1.45% in the baseline model to 0.41% in the full configuration, indicating increased output stability. This progressive improvement underscores the value of modular architecture in developing robust industrial and accurate industrial QA systems.

6. Conclusion

This study presents a comprehensive approach to integrating domain-specific industrial knowledge into a robust and accurate QA system, with the goal of reducing hallucination and enhancing answer reliability in industrial contexts. The proposed framework combines a structured domain knowledge base with modules for question classification, knowledge retrieval, and response generation powered by LLMs. A BERT-based classifier was developed to accurately identify industrial queries, achieving a high F_1-score of 97.79%, thereby demonstrating its effectiveness in filtering relevant questions. For knowledge retrieval, the system incorporates the Domain + GTE–DPR model, which achieved strong performance metrics, with MRR@5 and MRR@10 values of 90.12% and 90.28%, respectively, underscoring the benefit of domain-specific adaptation.

Furthermore, the integration of the RAG method significantly improved the quality and relevance of the generated answers. Notably, the ChatGLM2-6B model exhibited a rise in the mean ROUGE-L score from 32.52% to 55.04%, and in ACC from

5(nd in ACC from 42.92% to 70.60%. The effectiveness of the proposed framework was further validated through ablation studies, which confirmed the complementary contributions of the question classification and knowledge retrieval components. Overall, the results indicate that the proposed method offers a reliable and efficient solution for intelligent question answering in industrial applications, establishing a strong foundation for further advancements in this field.

Despite its effectiveness, the current framework has certain limitations. Although the industrial knowledge base covers multiple domains, the processes of segmentation and preprocessing may not capture all domain-specific nuances. In addition, conventional evaluation metrics may fall short in accurately reflecting the consistency between AI-generated answers and expert assessments. Future work will focus on expanding and regularly updating the knowledge base to improve its comprehensiveness. It will also explore the integration of evaluation tools, such as quality assessment of machine-generated answers (QAMAI) [70], to better assess the alignment of model outputs with expert-level responses.

Acknowledgment

This work was supported in part by The Major Key Project of PCL under Grant PCL2023A09. It was also partially funded by the National Natural Science Foundation of China under Grants 52471291, Guangdong Basic and Applied Basic Research Foundation (No. 2023A1515240044), Shanghai Science and Technology Program under Grants 22ZR1432300, Chenguang Program of Shanghai Education Development Foundation and Shanghai Municipal Education Commission under Grants 22CGA10, and the National Key R\&D Program of China under Grants 2023YFC2811600, and the Funded by Anhui Postdoctoral Scientific Research Program Foundation (No. 2025B1084).

References

[1] Lu J, Wang X, Cheng X, Yang J, Kwan O, Wang X. Parallel factories for smart industrial operations: from big AI models to field foundational models and scenarios engineering. IEEE/CAA J Autom Sinica 2022;9(12):2079–86.

[2] Wu T, He S, Liu J, Sun S, Liu K, Han Q-L, Tang Y. A brief overview of ChatGPT: the history, status quo and potential future development. IEEE/CAA J Autom Sinica 2023;10(5):1122–36.

[3] Yuan X, Xu W, Wang Y, Yang C, Gui W. A deep residual PLS for data-driven quality prediction modeling in industrial process. IEEE/CAA J Autom Sinica 2024;11(8):1777–85.

[4] Sun PZ, Bao Y, Ming X, Zhou T. Knowledge-driven industrial intelligent system: concept, reference model, and application direction. IEEE Trans Comput Soc Syst 2023;10(4):1465–78.

[5] Ojokoh B, Adebisi E. A review of question answering systems. J Web Eng 2018;17(8):717–58.

[6] Yang T, Mei Y, Xu L, Yu H, Chen Y. Application of question answering systems for intelligent agriculture production and sustainable management: a review. Resour Conserv Recycling 2024;204:107497.

[7]Masood S H, Soo A. A rule based expert system for rapid prototyping system selection. Robot Comput-Integr Manuf 2002; 18(3–4): 267–274.

[8] Ruiz E, Torres MI, del Pozo A. Question answering models for human–machine interaction in the manufacturing industry. Comput Ind 2023;151:103988.

[9] Liu X, Cheng Z, Shen Z, Zhang H, Meng H, Xu X, et al. Building a question answering system for the manufacturing domain. Ieee Access 2022;10:75816–24.

[10] Calijorne Soares MA, Parreiras FS. A literature review on question answering techniques, paradigms and systems. J King Saud Univ Comput Inf Sci 2020;32(6):635–46.

[1] ep semantic mining. Autom Construct 2023;145:104670.

[12] Xiong H, Wang S, Tang M, Wang L, Lin X. Knowledge graph question answering with semantic oriented fusion model. Knowl Base Syst 2021;221:106954.

[13] Han H, Wang J, Wang X. Leveraging knowledge graph reasoning in a multihop question answering system for hot rolling line fault diagnosis. IEEE Trans Instrum Meas 2023;73:3505014.

[14] Liu P, Qian L, Lu H, Xue L, Zhao X, Tao B. The joint knowledge reasoning model based on knowledge representation learning for aviation assembly domain. Sci China Technol Sci 2024;67(1):143–56.

[15] Zhou X, Zhang S, Agarwal M, Akroyd J, Mosbach S, Kraft M. Marie and BERT - a knowledge graph embedding based question answering system for chemistry. ACS Omega 2023;8(36):33039–57.

[16] Lee SH, Choi SW, Lee EB. A question-answering model based on knowledge graphs for the general provisions of equipment purchase orders for steel plants maintenance. Electronics 2023;12(11):2504.

[17] Zhou ZW, Jong WR, Ting YH, Chen SC, Chiu MC. Retrieval of injection molding industrial knowledge graph based on transformer and BERT. Appl Sci 2023;13(11):6687.

[18] Wen P, Ma Y, Wang R. Systematic knowledge modeling and extraction methods for manufacturing process planning based on knowledge graph. Adv Eng Inform 2023;58:102172.

[19] Yu P, Gong W, Bai Z, Zhao H, Deng W. Knowledge graph civil aviation question answering based on deep learning. In: Proceedings of the 2022 China Automation Congress (CAC); 2022 Nov 25–27; Xiamen, China. Piscataway: IEEE; 2022. p. 600–4.

[20] von Rueden L, Mayer S, Beckh K, Georgiev B, Giesselbach S, Heese R, et al. Informed machine learning–a taxonomy and survey of integrating prior knowledge into learning systems. IEEE Trans Knowl Data Eng 2021;35(1):614–33.

[21] Topsakal O, Akinci TC. Creating large language model applications utilizing langchain: a primer on developing LLM apps fast. Int Conf Appl Eng Nat Sci 2023;1(1):1050–6.

[22] Tallat R, Hawbani A, Wang X, Al-Dubai A, Zhao L, Liu Z, et al. Navigating industry 5.0: a survey of key enabling technologies, trends, challenges, and opportunities. IEEE Commun Surv Tutor 2023;26(2):1080–26.

[23] Hostetter H, Naser M, Huang X, Gales J. The role of large language models (AI chatbots) in fire engineering: An examination of technical questions against domain knowledge. Nat Hazards Res 2024;4(4):669–88.

[24] Rivera AC, Moore A, Robinson S. Coal mining question answering with LLMs. 2024. arXiv:2410.02959.

[25] Mo T, Xiao Q, Zhang H, Li R, Wu Y. Domain-specific few-shot table prompt question answering via contrastive exemplar selection. Algorithms 2024;17(7):278.

[26] Zheng S, Pan K, Liu J, Chen Y. Empirical study on fine-tuning pre-trained large language models for fault diagnosis of complex systems. Reliab Eng Syst Saf 2024;252:110382.

[27] Ji Z, Yu T, Xu Y, Lee N, Ishii E, Fung P. Towards mitigating LLM hallucination via self reflection. Findings of the association for computational linguistics. In: Bouamor H, Pino J, Bali K, editors. Findings of the Association for Computational Linguistics: EMNLP 2023. Kerrville: Association for Computational Linguistics; 2023. p. 1827–43.

[28] Wang C, Long Q, Xiao M, Cai X, Wu C, Meng Z, et al. Biorag: a rag-llm framework for biological question reasoning. 2024. arXiv:2408.01107.

[2] ngs of the 63rd Annual Meeting of the Association for Computational Linguistics; 2025 Jul 27–Aug 1; Vienna, Austria. Kerrville: Association for Computational Linguistics; 2025. p. 11400–26.

30] Li J, Yuan Y, Zhang Z. Enhancing llm factual accuracy with rag to counter hallucinations: a case study on domain-specific queries in private knowledge-bases. 2024. arXiv:2403.10446.

[31] Mansurova A, Mansurova A, Nugumanova A. QA-RAG: exploring LLM reliance on external knowledge. Big Data Cog Comput 2024;8(9):115.

[32] Liu J, Lin J, Liu Y. How much can rag help the reasoning of LLM? 2024. arXiv:2410.02338.

[33] Wang H, Li YF. Large language model empowered by domain-specific knowledge base for industrial equipment operation and maintenance. In: Proceedings of the 2023 5th International Conference on System Reliability and Safety Engineering (SRSE); 2023 Oct 20–23; Beijing, China. Piscataway: IEEE; 2023. p. 474–9.

[34] Zeng W, Ren X, Su T, Wang H, Liao Y, Wang Z, et al. Pangu-α: large-scale autoregressive pretrained chinese language models with auto-parallel computation. 2021. arXiv:2104.12369.

[35] Brown T, Mann B, Ryder N, Subbiah M, Kaplan J, Dhariwal P, et al. Language models are few-shot learners. In: Proceedings of the 34th International Conference on Neural Information Processing Systems; 2020 Dec 6–12; Vancouver, BC, Canada. New York: Curran Associates Inc.; 2020. p. 1877–901.

[36] Zhou B, Li X, Liu T, Xu K, Liu W, Bao J. CausalKGPT: industrial structure causal knowledge-enhanced large language model for cause analysis of quality problems in aerospace product manufacturing. Adv Eng Inform 2024;59:102333.

[37] Zhang X, Zhou Z, Ming C, Sun YY. GPT-assisted learning of structure–property relationships by graph neural networks: application to rare-earth-doped phosphors. J Phys Chem Lett 2023;14(50):11342–9.

[38] Sitapure N, Kwon JSI. CrystalGPT: enhancing system-to-system transferability in crystallization prediction and control using time-series-transformers. Comput Chem Eng 2023;177:108339.

[39] Kandpal N, Deng H, Roberts A, Wallace E, Raffel C. Large language models struggle to learn long-tail knowledge. In: Proceedings of the 40th International Conference on Machine Learning; 2023 Jul 23–19; Honolulu, HI, USA. Cambridge: MIT Press; 2023. p. 15696–707.

[40] Zhang Y, Li Y, Cui L, Cai D, Liu L, Fu T, et al. Siren's song in the AI ocean: a survey on hallucination in large language models. 2023. arXiv:2309.01219.

[41] Robertson S, Zaragoza H. The probabilistic relevance framework: BM25 and beyond. Found Trends Inf Retr 2009;3(4):333–89.

[42] Robertson SE, Walker S. On relevance weights with little relevance information. In: Proceedings of the 20th annual international ACM SIGIR conference on Research and development in information retrieval; 1997 Jul 27–31; Philadelphia, PA, USA. New York: Association for Computing Machinery; 1997. p. 16–24.

[43] Mariotti L, Guidetti V, Mandreoli F, Belli A, Lombardi P. Combining large language models with enterprise knowledge graphs: a perspective on enhanced natural language understanding. Front Artif Intell 2024;7:1460065.

[44] Siriwardhana S, Weerasekera R, Wen E, Kaluarachchi T, Rana R, Nanayakkara S. Improving the domain adaptation of retrieval augmented generation (RAG) models for open domain question answering. Trans Assoc Comput Linguist 2023;11:1–17.

[45] Kenton JDMWC, Toutanova LK. Bert: pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies; 2019 Jun 2–7; Minneapolis, MN, USA. Kerrville: Association for Computational Linguistics; 2019. p. 4171–86.

[4 s of the Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP); 2020 Nov 16–20; Online. Kerrville: Association for Computational Linguistics; 2020. p. 6769–81.

[47] Wang S, Zhuang S, Zuccon G. Bert-based dense retrievers require interpolation with bm25 for effective passage retrieval. In: Proceedings of the 2021 ACM SIGIR International Conference on Theory of Information Retrieval: Human Language Technologies; 2021 Jun 21; Canada. New York city: Association for Computing Machinery; 2021. p. 317–24.

[48] Fan W, Ding Y, Ning L, Wang S, Li H, Yin D, et al. A survey on rag meeting LLMs: towards retrieval-augmented large language models. In: Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining; 2024 Aug 25–29; Barcelona, Spain. New York city:

Association for Computing Machinery; 2024. p. 6491–501.

[49] Lewis P, Perez E, Piktus A, Petroni F, Karpukhin V, Goyal N, et al. Retrieval-augmented generation for knowledge-intensive NLP tasks. In: Proceedings of the 34th International Conference on Neural Information Processing Systems; 2020 Dec 6–12; Vancouver, BC, Canada. New York city: Curran Associates Inc.; 2020. p. 9459–74.

[50] Gao Y, Xiong Y, Gao X, Jia K, Pan J, Bi Y, et al. Retrieval-augmented generation for large language models: a survey. 2023. arXiv:2312.10997.

[51] Ma X, Gong Y, He P, Zhao H, Duan N. Query rewriting for retrieval-augmented large language models. In: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing; 2023 Dec 6–10; Singapore. Kerrville: Association for Computational Linguistics; 2023. p. 5303–15.

[52] Eibich M, Nagpal S, Fred-Ojala A. ARAGOG: advanced RAG output grading. 2024. arXiv:2404.01037.

[53] Zhou Y, Liu Z, Jin J, Nie JY, Dou Z. Metacognitive retrieval-augmented large language models. In: Proceedings of the ACM Web Conference 2024; 2024 May 13–17; Singapore. New York: Association for Computing Machinery; 2024. p. 1453–63.

[54] Muludi K, Fitria KM, Triloka J, Sutedi. Retrieval-augmented generation approach: document question answering using large language model. Int J Adv Comput Sci Appl 2024;15(3):776–85.

[55] Yu W, Iter D, Wang S, Xu Y, Ju M, Sanyal S, et al. Generate rather than retrieve: large language models are strong context generators. 2022. arXiv:2209.10063.

[56] Shao Z, Gong Y, Shen Y, Huang M, Duan N, Chen W. Enhancing retrieval-augmented large language models with iterative retrieval-generation synergy. 2023. arXiv:2305.15294.

[57] Jin J, Zhu Y, Dou Z, Dong G, Yang X, Zhang C, et al. FlashRAG: a modular toolkit for efficient retrieval-augmented generation research. In: Companion Proceedings of the ACM on Web Conference 2025; 2025 Apr 28–May 2; Sydney, NSW, Australia. New York city: Association for Computing Machinery; 2025. p. 737–40.

[58] Bornea AL, Ayed F, De Domenico A, Piovesan N, Maatouk A. Telco-RAG: navigating the challenges of retrieval augmented language models for telecommunications. In: Proceedings of the GLOBECOM 2024—2024 IEEE Global Communications Conference; 2024 Dec 8–12; Cape Town, South Africa. Piscataway: IEEE; 2024. p. 2359–64.

[59] Kim D, Kim B, Han D, Eibich M. AutoRAG: automated framework for optimization of retrieval augmented generation pipeline. 2024. arXiv:2410.20878.

[60] Yan SQ, Gu JC, Zhu Y, Ling ZH. Corrective retrieval augmented generation. 2024. arXiv:2401.15884.

[61] Chan CM, Xu C, Yuan R, Luo H, Xue W, Guo Y, et al. Rq-rag: learning to refine queries for retrieval augmented generation. 2024. arXiv:2404.00610.

[6 tion-answering. 2024. arXiv:2406.07348.

[63] Li Z, Zhang X, Zhang Y, Long D, Xie P, Zhang M. Towards general text embeddings with multi-stage contrastive learning. 2023. arXiv:2308.03281.

[64] Johnson J, Douze M, Jégou H. Billion-scale similarity search with GPUs. IEEE Trans Big Data 2021;7(3):535–47.

[65] Zhang W, Wang M, Han G, Feng Y, Tan X. A knowledge graph completion algorithm based on the fusion of neighborhood features and vBiLSTM encoding for network security. Electronics 2024;13(9):1661.

[66] Lin CY. Rouge: a package for automatic evaluation of summaries. In: Text summarization branches out. Kerrville: Association for Computational Linguistics; 2004. p. 74–81.

[67] Joulin A, Grave E, Bojanowski P, Mikolov T. Bag of tricks for efficient text classification. In: Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics; 2017 Apr 3–7; Valencia, Spain. Kerrville: Association for Computational Linguistics; 2017. p. 427–31.

[68] Kim Y. Convolutional neural networks for sentence classification. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP); 2014 Oct 25–29; Doha, Qatar. Kerrville: Association for Computational Linguistics; 2014. p. 1746–51.

[69] Liu P, Qiu X, Huang X. Recurrent neural network for text classification with multi-task learning. In: Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence; 2016 Jul 9–15; New York, NY, USA. Palo Alto: AAAI Press; 2016. p. 2873–9.

[70] Lai S, Xu L, Liu K, Zhao J. Recurrent convolutional neural networks for text classification. In: Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence; 2015 Jan 25–30; Austin, TX, USA. Palo Alto: AAAI Press; 2015. p. 2267–73.

[71] Vaira LA, Lechien JR, Abbate V, Allevi F, Audino G, Beltramini GA, et al. others. Validation of the quality analysis of medical artificial intelligence (QAMAI) tool: a new tool to assess the quality of health information provided by AI platforms. Eur Arch Otorhinolaryngol 2024;281:6123–31.

Declaration of interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐ nd was not involved in the editorial review or the decision to publish this article.

☐ The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: