2025 Asia-Europe Conference on Cybersecurity, Internet of Things and Soft Computing (CITSC) | 979-8-3315-0420-5/25/$31.00 ©2025 IEEE | DOI: 10.1109/CITSC64390.2025.00056

Knowledge-based question answering system for hydraulic hoist using large language models

Shufeng Zheng
School of Mechanical and Power Engineering
Zhengzhou University
Zhengzhou, China
zhengsf68@163.com

Helin Li
School of Mechanical and Power Engineering
Zhengzhou University
Zhengzhou, China
lihelin@gs.zzu.edu.cn

Wenjie Wang
School of Mechanical and Power Engineering
Zhengzhou University
Zhengzhou, China
wenjiewang@zzu.edu.cn

Fan Zhang
Henan Xinhua Wuyue Pumped Storage Power Generation Co. Ltd
Xinyang, China
Zhangfan_xhwy@163.com

Huadong Zhao
School of Mechanical and Power Engineering
Zhengzhou University
Zhengzhou, China
zhaohuadong1978@163.com

***Abstract*—The hydraulic hoist is an essential electromechanical equipment widely used in water conservancy projects. Due to its system complexity and the diversity of failure modes, traditional knowledge management is inefficient, and it is difficult to support the demand for timely and accurate knowledge application during the operation and maintenance process of the equipment. This paper develops a knowledge-based question answering (QA) system for hydraulic hoists using the Large Language Model (LLM) combined with the Retrieval Augmented Generation (RAG) and LangChain methods. Firstly, a high-quality knowledge base is constructed with unstructured text data such as hydraulic hoist design data, failure cases, and maintenance manuals. Secondly, database retrieval technology is utilized to achieve graph retrieval and vector retrieval. Finally, combining the LLM and RAG methods, the answers are generated by reasoning between the retrieved relevant text segments and the user's questions. To verify the application effect of the QA system, this study employed a dataset comprising common failure types of hydraulic hoists and conducted a systematic QA evaluation and knowledge reasoning process test based on typical failure scenarios. The results show that the system can effectively deal with complex problem scenarios, provide fast and accurate solutions, and enhance the interpretability of the results through additional information, providing a valuable reference for the intelligent maintenance of electromechanical equipment in water conservancy projects.**

***Keywords*—retrieval augmented generation, question answering system, large language modeling, hydraulic hoist, knowledge base**

I. Introduction

As a key equipment in water conservancy projects, hydraulic hoists, with high power density and precise manipulation characteristics, are widely used in the opening and closing control of gates in sluice gates, locks and other water conservancy facilities [1]. However, due to the complex working conditions and harsh environment, hydraulic hoists are commonly found to have a variety of safety hazards such as leakage, clogging and cavitation during the operation process, which leads to a decline in the performance of the system and even shutdown accidents, posing a serious threat to the normal operation of water conservancy projects. The regular operation of the water conservancy project constitutes a serious threat. In addition, the hydraulic hoist consists of a complex hydraulic system, mechanical components and electrical control system, and the traditional manual diagnosis method faces enormous challenges in real-time and accurate fault detection and treatment [2]. Therefore, building an efficient and intelligent hydraulic hoist knowledge base QA system is particularly important, which integrates and analyzes a large amount of equipment operation data, fault cases and technical documents, and provides fast and accurate operation guidance and fault diagnosis suggestions for on-site operation and maintenance personnel.

In recent years, researchers have proposed various methods based on signal analysis, model diagnosis, and machine learning to solve the problem of hydraulic system fault diagnosis. For example, Huang et al. [3] achieved remarkable results in detecting hydraulic system leakage and wear faults based on vibration signal and spectrum analysis. Wang et al. [4] established a spatio-temporal network fault diagnosis model combining convolutional neural networks and short-term and long-term memory, which realizes efficient diagnosis of faulty valves and improves the fault tolerance of digital hydraulic systems. Liu et al. [5] proposed a muti-branch neural network-based method, which recognizes the recognition of typical fault patterns by training a classification model. However, these methods rely on much sensing data, cumbersome feature extraction, and domain-specific expertise. They cannot support the rapid troubleshooting and problem-solving needs of field operation and maintenance managers for abnormal problems. In addition, the research and application of domain knowledge graphs and QA systems for hydraulic hoists are still in their infancy [6]. Knowledge management methods, such as traditional document query and database retrieval methods, are underperforming in terms of accuracy, relevance, transparency, and response time, and lacking in targeted intelligent diagnostic assistance support tools.

To address these challenges, this study proposes an

979-8-3315-0420-5/25/$31.00 ©2025 IEEE
DOI 10.1109/CITSC64390.2025.00056

innovative approach to construct a knowledge-based QA system for hydraulic hoists based on LLM. The system is designed first to build a knowledge base containing hydraulic hoist control principles, fault cases, and maintenance manuals. It then combines RAG technology with the structured characteristics of a Neo4j graph database to organize high-quality hydraulic hoist data in the form of nodes and relations, enhancing the depth and contextual relevance of information retrieval. Finally, the system generates credible responses to user input questions through efficient retrieval technology and semantic reasoning. The system's function and knowledge reasoning process are tested by constructing typical fault scenarios, and the results show that the system can quickly and accurately identify problems, demonstrate the reasoning process clearly, and provide targeted solutions. This approach offers clear advantages over traditional file query and database retrieval methods regarding accuracy and real-time performance. This study provides new ideas for the intelligent diagnosis of hydraulic hoists in water conservancy engineering and practical references for designing and optimizing intelligent QA systems for complex electromechanical equipment in the industry.

II. Methordlogy

A. LLMs

LLMs have excellent natural language understanding and generation capabilities. By training on large amounts of textual data, including books, articles, and technical manuals, LLMs can grasp the patterns and structures of language to perform complex tasks such as question answering and text summarization. However, since the performance of LLMs depends on the quality of the training data, they can produce inaccurate or spurious responses and suffer from a lack of transparency and interpretability. These problems are particularly significant in specialized vertical domains and require additional technical support to improve response reliability and accuracy. To this end, this study investigates the construction of a knowledge-based QA system for hydraulic hoists in the field of hydraulic engineering, based on an open-source large language model, combined with a Neo4j graph database and RAG technology.

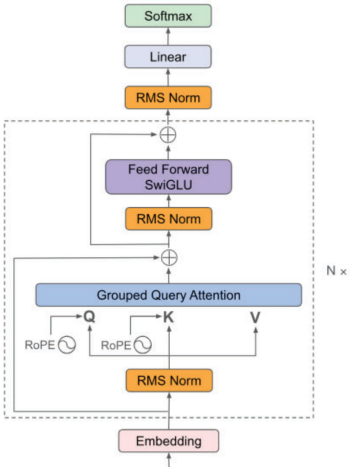


Fig. 1. Llama3 model architecture.

Llama3 [7], developed by Meta AI, is widely used as an efficient generative language model due to its features of allowing users to download it for free, supporting local model training and fine-tuning for specific tasks or datasets [8]. The architecture of Llama3 is shown in Fig. 1. The model adopts the *N*-stacked Transformer architecture [9] and introduces the Grouped Query Attention mechanism instead of the traditional multi-head attention mechanism, as well as the use of Root Mean Square Normalization (RMSNorm) instead of Layer Normalization (LayerNorm), which are improvements that significantly increase the computational speed and memory efficiency. Meanwhile, by applying the rotary position embeddings (RoPE), the model can incorporate positional information more efficiently.

B. Graph and Vector Databases

This study introduces Neo4j's graph database and vector database technologies as the knowledge base to support the QA system. Graph databases are used to store structured knowledge, representing information such as entities, relationships, attributes, components, failure types and maintenance history of hydraulic hoists. The graph database can store complex semantic relationships and improve the quality of LLM responses. Vector database stores the relevant information of hydraulic hoist in the form of high-dimensional vectors with semantics. After a user asks a question, the system will quickly match relevant content in the database based on semantic similarity, thus ensuring that the answer comes from authoritative and relevant information. The vector database not only improves the retrieval speed, but also provides high-quality contextual support for the model.

To ensure efficient processing of documents, this paper adopts the document loader of LangChain [10]. The LangChain framework is shown in Fig. 2. After the document is loaded, the data is first preprocessed and text segmented, including removing noisy data, formatting the text, and extracting key information. These steps divide all document content into structured text paragraphs, which ensures that each paragraph contains independent and complete information. The text passages are then converted into semanticized word vectors using the Embedding model, and the system also generates an embedding of the user's question. This embedding helps LLM to understand the meaning and context of the query for subsequent retrieval and QA generation. Finally, all converted word vectors are stored in the Neo4j database.

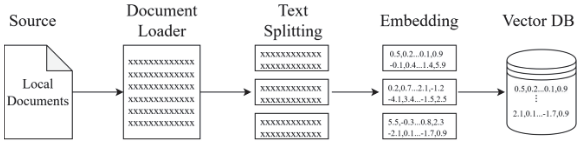


Fig. 2. LangChain framework.

LangChain provides all building blocks for RAG from simple to complex. As a framework for developing language model-driven applications, LangChain supports the loading and pre-processing of multi-format data (e.g., PDF files) and the embedding of data into vector databases for efficient retrieval.

C. RAG

The RAG approach tightly integrates LLMs with vector databases. RAG [11] enables the model to use specific datasets without additional training, improving response accuracy and reducing illusions [12]. RAG's web crawling capability makes it possible to obtain a large amount of hydraulic hoist data from reliable Internet sources. It retrieves content relevant to the user's problem from the vector database and puts this content as contextual inputs into the LLM, making the generated responses more accurate and reliable. In addition, the RAG approach provides traceability of the data source. It allows users to check the provenance of the retrieved content, thus enhancing the transparency and credibility of the QA system.

The RAG's architecture is shown in Fig. 3 and usually consists of two main components: a retriever and a generator. The role of the retriever is to retrieve the most relevant document fragments from an extensive database or knowledge base. These document fragments will be passed as context to the generator to help it generate more accurate and informative responses. The generator is usually a large-scale pre-trained language model such as GPT-4 [13], Llama3, etc. It generates the final answer or textual content based on the retrieved relevant documents.

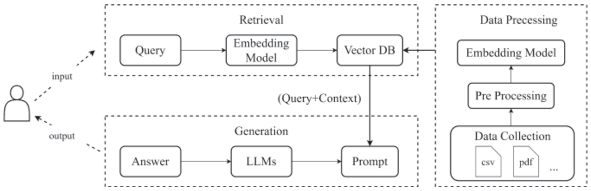


Fig. 3. RAG model architecture.

III. Construction of the Question Answering system

This section investigates the QA system construction method based on LLM (Llama3), Neo4j graph database and RAG framework. The approach is evaluated within a vertical domain by developing a QA system integrated with a knowledge base tailored to hydraulic hoists. The research is divided into three main parts: knowledge base construction, knowledge base retrieval and QA system construction, ensuring the completeness and relevance from data preparation to system execution. The overall framework of the QA system is shown in Fig. 4.

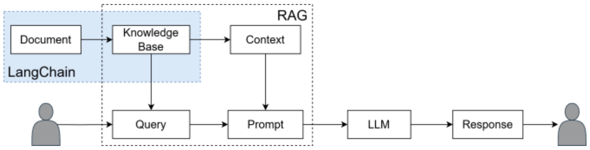


Fig. 4. Overall framework of the QA system.

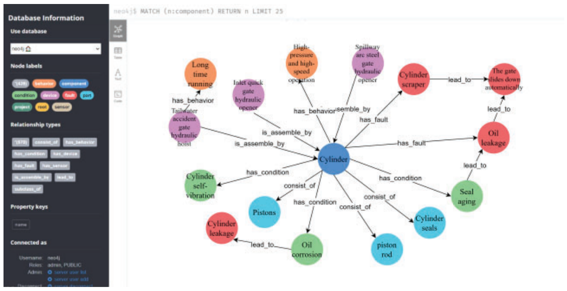
A. Knowledge Base Construction

The knowledge base construction method is divided into the following three main steps:

1) Text pre-processing: The first step involves acquiring technical documents such as hydraulic hoist maintenance manuals, standard specifications, and failure case records (e.g., PDF, HTML) from publicly available sources, databases, and other relevant channels. These documents are then ingested using LangChain's document loader. Subsequently, the data undergoes preprocessing, which includes text segmentation, noise removal, text formatting, and key content extraction to create structured data chunks.

2) Domain knowledge extraction, including named entity recognition and relationship extraction: This paper uses the pre-trained large language model Llama3 for knowledge extraction of each text passage. The entity types include the structure, behavior, state and fault of hydraulic hoist. The relationship types describe the interactions between these entities, such as the consist_of relationship. These entities and their relationships are stored into the Neo4j graph database, where each node represents an entity, and the edges connecting the nodes represent the relationships between them.

3) Visualization of the Hydraulic hoist knowledge base: We use Neo4j Browser to visualize the hydraulic hoist knowledge base. Due to the excessive number of nodes and relationships, only a partial of the knowledge graph is displayed in Fig. 5 for convenience.



```
neo4j$ MATCH (n:component) RETURN n LIMIT 25
```

Fig. 5. Partial knowledge base of the hydraulic hoist.

B. Graph Search and Vector Search

Neo4j graph databases retrieve graph databases using specialized query languages such as Cypher or SPARQL. Besides, the Graph Data Science (GDS) library integration in Neo4j also provides powerful similarity computation capabilities. GDS can be used for similarity analysis based on graphs and vectors, e.g., identifying related or similar nodes by calculating the vector similarity of different nodes, passing the embedded vectors to the clustering algorithms of GDS (e.g., Louvain, KMeans, etc.) to perform node clustering. In the GDS library, the cosine similarity of two nodes (or vectors) A and B can be computed by the following equation:

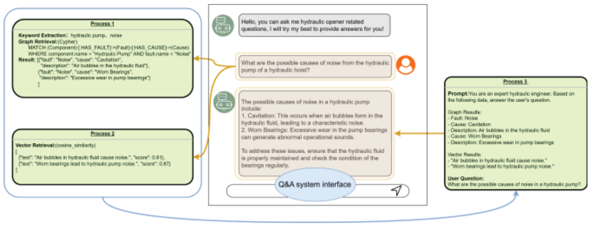
$$\text{Cosine Similarity}(A,B) = \frac{A \cdot B}{\sqrt{(A \cdot A)(B \cdot B)}} \quad (1)$$

Text chunks in a vector database must be divided into fixed-length segments to realize vector retrieval. In the retrieval process of RAG, the core task is to calculate the similarity between questions and chunks. First, the user's questions are vectorized using the embedding model. Then, the question vectors are matched with the existing segments in the vector database to get the most similar n-text blocks through similarity calculation. Finally, the most relevant content is added to the cue word template to form the final text sent to the LLM. By integrating the embedding model with the vector database, the RAG framework employs the vector searcher to retrieve the most pertinent content from the stored documents based on the user's query.

In this study, the above two hybrid methods are used for retrieval, taking full advantage of the technical merits of graph and vector databases to obtain the most relevant passages to the query. Finally, all the relevant texts are merged to generate the final dataset.

C. Development and Validation of the QA system

In this study, a text input interface is designed to enable users to ask questions about hydraulic hoists in a natural language format. Secondly, based on guaranteeing the accuracy and relevance of the system, attention is also paid to the realization of the QA process focusing on transparency, i.e., to ensure that the users can check the source of each answer and access information about the provenance of the retrieved content, thereby increasing the credibility of the system. The operation interface and reasoning process of the QA system is shown in Fig. 6.



```
Keyword Extraction: hydraulic pump, noise
Graph Retrieval:(Cypher)
    MATCH (Component)-[:HAS_FAULT]->(Fault)-[:HAS_CAUSE]->(Cause)
    WHERE component.name = "Hydraulic Pump" AND fault.name = "Noise"
Result: [{"fault": "Noise", "cause": "Cavitation",
          "description": "Air bubbles in the hydraulic fluid"},
         {"fault": "Noise", "cause": "Worn Bearings",
          "description": "Excessive wear in pump bearings"}
         ]
```

```
Vector Retrieval:(cosine_similarity)
[
 {"text": "Air bubbles in hydraulic fluid cause noise.", "score": 0.91},
 {"text": "Worn bearings lead to hydraulic pump noise.", "score": 0.87}
]
```

Hello, you can ask me hydraulic opener related questions, I will try my best to provide answers for you!

What are the possible causes of noise from the hydraulic pump of a hydraulic hoist?

The possible causes of noise in a hydraulic pump include:
1. Cavitation: This occurs when air bubbles form in the hydraulic fluid, leading to a characteristic noise.
2. Worn Bearings: Excessive wear in the pump bearings can generate abnormal operational sounds.

To address these issues, ensure that the hydraulic fluid is properly maintained and check the condition of the bearings regularly.

Prompt:You are an expert hydraulic engineer. Based on the following data, answer the user's question.

Graph Results:
- Fault: Noise
- Cause: Cavitation
- Description: Air bubbles in the hydraulic fluid
- Cause: Worn Bearings
- Description: Excessive wear in pump bearings

Vector Results:
- "Air bubbles in hydraulic fluid cause noise."
- "Worn bearings lead to hydraulic pump noise."

User Question:
What are the possible causes of noise in a hydraulic pump?

Fig. 6. Demonstration of the QA system interface and reasoning process.

The specific realization process is divided into three steps as follows:

1) Keyword extraction and graph database query: The keywords of the problem are extracted using LangChain, or by crafting a prompt for keyword extraction using the model. The keywords in the user's problem are then used to query the graph database using Cypher search statements to obtain relevant entities and descriptions.

2) Vector database retrieval: The RAG module retrieves the most relevant text passages to the problem from the vector database using a cosine similarity calculation. This step identifies the top *n* document passages that best match the user's query.

3) Answer generation and output: The information retrieved from the graph database and the vector database is combined with the user's original query. This combined input is structured into a prompt using LangChain's vector retrieval module and passed to LangChain's internal LLM for inference. The model generates an answer by integrating the user's query and retrieved content. Finally, the generated answer is formatted into a string and returned to the user.

To verify the effectiveness of the QA system, a collection of questions covering common types of hydraulic hoist failures was designed during the experiment, such as "What are the causes of noise in hydraulic pumps?" "How to detect the leakage of the hydraulic system?" etc. Fig. 6 illustrates an example wherein the query "What are the possible causes of noise from the hydraulic pump of a hydraulic hoist?" is used. The top two most similar text segments are retrieved and subsequently input into the large language model along with the query as prompt material through vector search and graph search. The questions are integrated as cue words into the model to generate the corresponding answers. The case validates the performance of the hydraulic hoist QA system in terms of accuracy, relevance, transparency and response time. The test results show that the QA system can quickly and accurately identify the possible causes of hydraulic pump noise, provide clear provenance information, and have a short response time, with obvious advantages in accuracy and real-time over traditional methods such as document query and database retrieval methods. Overall, the QA system can efficiently and accurately answer hydraulic hoist-related questions and provide a highly credible reasoning process, showing good application prospects in opener anomaly analysis, fault diagnosis, and maintenance guidance.

IV. Conclusion

In this paper, we employ the open-source Llama3 as the foundational large language model, utilize LangChain as the development framework, and integrate advanced technologies, including knowledge base construction, RAG retrieval, and generative models to develop a hydraulic hoist QA knowledge base. This work culminates in realizing a comprehensive knowledge-based QA system using LLM for hydraulic hoists. To evaluate system functionality and knowledge reasoning, we constructed a typical fault scenario, which initially verified the feasibility of the proposed method under local LLM deployment conditions. The integration of LLM and RAG empowers the QA system to accurately identify problems, clearly demonstrate the reasoning process, and provide targeted solutions. This approach offers significant advantages over traditional file query and database retrieval methods in terms of accuracy and real-time performance. The system proves to be highly valuable in the operation and maintenance of hydraulic hoists, while also offering valuable insights for the digitization and intellectualization of the operation and maintenance processes of complex electromechanical equipment in water conservancy projects. Moving forward, we plan to expand the system's application scenarios by enhancing its adaptive learning and multimodal comprehension capabilities. This will allow us to offer more prosperous, more efficient, and personalized services, as well as ensuring that the system remains aligned with ongoing technological advancements and evolving industry demands over the long term.

Acknowledgment

Key research project of Xinhua Hydropower under China National Nuclear Corporation (CNNC)] under Grant [number XHKY2022-33].

Postdoctoral Fellowship Program of China Postdoctoral Science Foundation (Grant No. GZC20232394)

References

[1] Akoz M S, Kirkgoz M S, Oner A A. Experimental and numerical modeling of a sluice gate flow. Journal of Hydraulic Research, 2009, 47(2), 167-176.

[2] Su Lei, Hua Song, and Hong Wang. "Fault diagnosis for hydraulic hoisting system based on the probabilistic SDG model." IEEE 10th International Conference on Industrial Informatics. IEEE, 2012, 627-630.

[3] Keke Huang, Shujie Wu, Fanbiao Li, and Hong Wang. Fault diagnosis of hydraulic systems based on deep learning model with multirate data samples. IEEE Transactions on Neural Networks and Learning Systems, 2021, 33(11), 6789–6801.

[4] Pei Wang, Yuxin Zhang, Matti Linjama, Liying Zhao, and Jing Yao. Fault identification and localization for the fast switching valve in the equal-coded digital hydraulic system based on hybrid CNN-LSTM model. Mechanical Systems and Signal Processing, 2025, 224, 112201.

[5] Huizhou Liu, Shibo Yan, Mengxing Huang, and Huang Zhong. A fault diagnosis method for hydraulic system based on multi-branch neural networks. Engineering Applications of Artificial Intelligence, 2024, 137, 109188.

[6] Li Zhe, Yi Wang, and Kesheng Wang. Intelligent predictive maintenance for fault diagnosis and prognosis in machine centers: Industry 4.0 scenario. Advances in Manufacturing, 2017, 5, 377–387.

[7] Dubey A. The llama 3 herd of models. arXiv preprint, 2024, arXiv:2407.21783.

[8] Chiyun Liu, Juisheng Chou. Automated legal consulting in construction procurement using metaheuristically optimized large language models. Automation in Construction, 2025, 170, 105891.

[9] Khan M. A unified transformer with memory encoder and graph attention networks for multidomain dialogue state tracking. Applied Intelligence, 2024, 54(17), 8347–8366.

[10] Mavroudis V. LangChain. University College of London, 2024. hal-04817573.

[11] Hlewis P. Retrieval-augmented generation for knowledge-intensive NLP tasks. Proceedings of the 34th International Conference on Neural Information Processing Systems, 2020, 33, 9459–9474.

[12] Guu K, Lee K, Tung Z, Pasupat P, and Chang M. Retrieval-Augmented Language Model Pre-Training. International Conference on Machine Learning, 2020, 3929–3938.

[13] OpenAI. GPT-4 technical report. arXiv preprint, 2023, arXiv:2303.08774.