2024 4th International Conference on Artificial Intelligence, Robotics, and Communication (ICAIRC) | 979-8-3315-3122-5/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICAIRC64177.2024.10900039

A Retrieval-Augmented Generation-Based Method for Aviation Accident Data Analysis

Jianzhong Yang, Xinyu Xiang*, Xiyuan Chen
College of Safety Science and Engineering, Civil Aviation University of China
Tianjin, China
*Enzo_xxy@163.com

Abstract—**In response to the growing demand for more intelligent and efficient methods of analyzing aviation accidents, this paper explores an AI-based approach to data mining and utilization in aviation accident investigation reports. Based on the framework of Retrieval-Augmented Generation (RAG), we constructed a benchmark dataset of aviation accident reports, we also built an external knowledge base from the Air Transport Association Specification 100 (ATA100) and enhanced the content with components in Aircraft Manual Maintenance (AMM). Experiments of sensitive factors were conducted to find out the most influential factors of RAG. The results show that the LLM, the retrieved number of external knowledge segments (top-k), and especially the embedding model have the greatest influence on RAG. The optimal factors reached scores of 0.75 in recall coverage rate, 0.74 in two-digit codes accuracy and 0.53 in four-digit codes accuracy. Based on the optimal sensitive factors, we conducted experiment of external knowledge enhancement. The results indicate that a more comprehensive external knowledge base can benefit both LLM and embedding model as recall coverage rate improved 8%, two-digit codes accuracy improved 11%, and four-digit codes improved 16%. The outcome of our work indicates that the RAG-base analysis method can significantly enhance the efficiency and accuracy of aviation knowledge extraction task while being highly automated and self-iterative, demonstrating its potential for future applications in the aviation industry.**

Keywords-aviation accident analysis, Retrieval-Augmented Generation, Transformer, Air Transport Association Specification 100, Large Language Model, Prompt Engineering

I. Introduction

The safety of civil aviation is the lifeline of civil aviation transportation industry. Civil aviation accident is widely concerned by public for the reason that it usually causes great loss in both economy and life security of people. As an international issue, to effectively prevent civil aviation accident from happening is one of the most important works of aviation safety authority of every country in the world. Nongtian Chen et al. made a thorough analysis over an A320 maintenance accident adapting REASON model, and found out that the main cause of this accident lay in the organization level [1]. Lei Wang et al. made a statistical analysis over 626 pieces of accident investigation data collected from ASN and summarized the features of those accidents in aspects of operation phase, occurrence rate and the average time span of investigation [2]. Agus Pramono et al. identified the main contributing factors of 97 incident/accident investigation reports by listing and comprehensively analyzing all contributing factors follow specific rules [3]. Xin Wang et al. proposed a novel model based on fusion with a double dictionary, fusion character features and bilinear attention networks to quickly mine important information in aviation safety reports, helping safety personnel improve efficiency [4].

However, such research method put too much reliance on manual processing and statistical analysis which led to time-consuming efforts, susceptibility to human error, and inefficient analysis processes [14] and inability to big data. The aviation industry is in a rapid developing phase which means the aviation data has a high updating frequency, thus, there is a demand of a highly automated and integrated analysis approach with self-iterate capacity. With the development of machine learning, the methodology of accident analysis has changed as well. Jianping Bao et al. mined the main causes of aircraft runway related accident as well as the degree of relevance of the relationships between those causes by an improved LDA-Apriori algorithm [5]. Lijun Wei et al. provided a complex system theory to dig out the main contributing causes and to classify the main accident modes, by applying those data as knots, the author built an aviation accident cause net model in order to support the design of the aviation accident prediction system [6]. Zhitong Zhu et al. visualized flight data to support the graph analysis of aviation accident [7]. Wenlong Sun et al. adapted words frequency analysis to locate the key words in Chinese and America safety recommendation and did a cluster analysis using k-means algorithm over those key words, then they established a classification criterion and classified safety recommendation and presented them all in a mind map [8].

The success of ChatGPT brought large language models into the public eye. Large language model (LLM) is an artificial intelligence system that can understand an generate human language by processing enormous text data and has already become one of the core research topics of neural natural language processing [9]. The exciting performance of GPT2 indicated a promising future towards building language processing systems [18], and the next generation, GPT3, showed better performance with its massive amount of 175 billion parameters in the few-shot setting [19]. All the improvements of LLM are originated from a talented work, the Transformer structure. As the foundation of BERT [21] and GPT models, Transformer was introduced in 2019 [20], it proposed ‘self-attention’ mechanism that is constructed with multi-head attention using Scaled Dot-Product Attention algorithm. This structure actually became the fountain of future natural language processing (NLP) tasks.

Although LLMs have did a great job in multiple natural language generation tasks, they all suffered from response hallucination and lack of specific domain knowledge. As one of the advanced utilization paradigms of LLM, which are prompt engineering (PM) [22], retrieval-augmented generation (RAG) [10] and fine-tuning (FT) [23], RAG accelerated the process of knowledge extraction. It combined pre-trained parametric and non-parametric memory for language generation [10], using a LLM to generate text based on commands and integrates information from a separate retrieval system to improve output quality and contextual relevance [11]. RAG offers advantages over FT by using an external knowledge base to automatically update information without retraining the model, reducing computational resources and enhancing data security [25]. Compared to PM, RAG more effectively handles specific tasks by incorporating up-to-date, relevant information, achieving better performance while maintaining flexibility and efficiency [24]. RAG has been proved with the ability of addressing LLM hallucinations [12] and showed a great performance in domain-specific Question Answering (QA) task [13]. There are researches showing that RAG method has a brilliant capability of natural language processing. Kurnia Muludi et al. applied the Retrieval Augmented Generation (RAG) method to improve Question-Answering (QA) systems by addressing document processing in Natural Language Processing problems [14]. J Miao et al. created a chronic kidney disease specialized ChatGPT model integrated with a RAG system, revealing its potential in providing specialized, accurate medical advice [15]. A. Golatkar et al. provided two RAG based algorithms for copyright protection, CPR-KL and CPR-Choose which can be applied to any pre-trained conditional diffusion model [16]. J. Y. Antonio et al. proposed an expanded approach to chunk documents by moving beyond mere paragraph-level chunking to chunk primary by structural element components of documents, and proved that element type-based chunking can largely improve RAG results on financial reporting [17]. The successful application of RAG in financing, medication and education indicates its potential in aviation accident analysis.

The accuracy of RAG is mostly based on the ability of base model, which is LLM, and external knowledge base. However, most LLMs lack domain knowledge in the aviation industry, leading to shortcomings in aviation-specific tasks. Therefore, establishing an external knowledge base related to aviation knowledge becomes particularly important.

Building on this line of research, this paper proposes a Retrieval-Augmented Generation-based method for cause analysis and classification of aviation accidents, compliant with ATA100 regulations. On one hand, this method connected accident analysis with industry standards by merging ATA100 and Aircraft Maintenance Manual (AMM) as the external knowledge reference. On the other hand, it successfully implemented large-scale automated comprehensive analysis of aviation data through our workflow. Last but not least, the implement of RAG facilitated the method with a strong self-iterative capability. The main contributions can be summarized as follows:

- A benchmark dataset compliant with ATA100 regulations and an external knowledge base applying ATA100 were built.
- A thorough workflow was established to enable the utilization of the multi-turn conversational capability of LLMs, as well as to perform automatic batch analysis of aviation accident data.
- Experiments on the sensitive factors of RAG identified the LLM, embedding model, and the number of retrieved external knowledge segments as having the greatest impact on the performance of RAG.
- Experiment of external knowledge enhancement was conducted to evaluate the influence of knowledge base richness on the performance of RAG.

II. Methodology

A. Dataset

Provided by official agency, civil aviation accident investigation reports contained abundant information and domain knowledge as well as experience, technics and relationships. Available data is highly unstructured and text data holds a large scale among it. A data sample is shown in Table 1. Each sample is composed with a brief description of the process along with several simple classification and analysis of the accident. Much of the important information and knowledge still remains hidden in brief descriptions, waiting to be mined.

TABLE I. A sample of aviation accident data

Brief Summary	**Abstract (Table of Contents)**	**Abstract (Information)**
A plane was performing a flight at an altitude of 8,100 meters during the cruise phase, guided by air traffic control radar. At the tangent point SHX, an IAS DISAGREE fault was displayed, and the crew determined that the airspeed was unreliable. The crew executed the checklist and informed air traffic control. During the fault-handling process, they discovered that the left airspeed indicator was 20 knots lower than the right and standby airspeed indicators, leading the crew to conclude that the left airspeed indicator had failed. At XX , the aircraft experienced an EEC ALTN fault; after the status stabilized, the crew analyzed that the fault might have been caused by icing and requested to descend. During the descent to 5,700 meters, the EEC ALTN light went out, and the left airspeed gradually returned to normal, with the IAS DISAGREE fault display disappearing. The crew maintained good communication with air traffic control and established a contingency plan. Later, they executed an ILS approach to runway 12L at XX airport and landed normally. After the flight, maintenance replaced the captain's side pitot tube, flushed the captain's side pitot line, completed the pitot tube leak test, and conducted a low-speed check, all of which were normal, allowing the aircraft to be released.	***Aircraft Type***	XXXX
	Aircraft Incident Phase	Cruise
	Event Level	Minor Incident
	Event Type (Primary)	System Failure / Malfunction / Blockage
	Event Cause (Primary)	Mechanical
	Cause analysis	Inconsistent airspeed, left airspeed indicator failure.

Air Transport Association Specification 100 (ATA100) is a national regulation for the creation and maintenance of technical documentation in the aviation industry. The specification aims to standardize technical communication between airlines, maintenance personnel, and manufacturers, ensuring a uniform structure, format, and content for aircraft maintenance and operational documents. ATA100 includes various chapters and a numbering system that provides detailed descriptions of different aircraft systems and components, allowing technicians to quickly locate and use the relevant information. The content of ATA100 is highly structural as shown in fig.1. Each system is annotated with two-digit codes and is composed of several subsystems, which are annotated with four-digit codes. Each subsystem contains several components that differ only in the last character of the code. For example, 22 represents the Auto Flight system, which includes the 2210 Autopilot subsystem, containing the 2211 Flight Control Computer and other components.

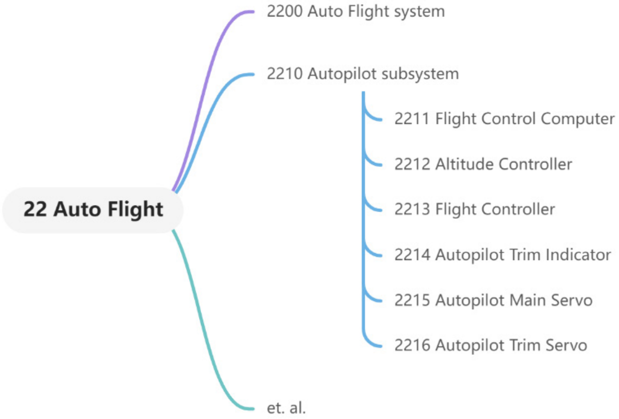


Figure 1. Partial View of the Auto Flight System in ATA100

We manually analyzed the defective components and systems from100 domestic accidents and annotated them using the four-digit ATA100 codes to construct the benchmark dataset for our research. The statistical results show that our benchmark covers 72 percent of common defective systems in ATA100, indicating that the outcomes of our work have a confidence level of no less than 0.72.

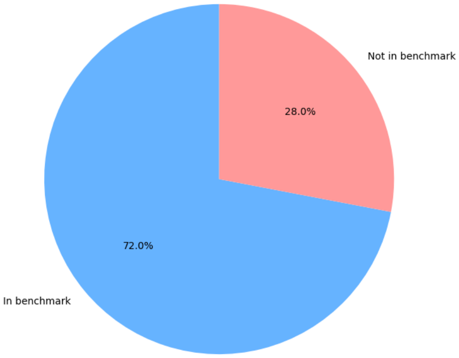


Figure 2. Proportion of common defective systems in ATA100 covered by our benchmark

B. Framework of RAG

Structurally analyzing, RAG method is composed with two main parts, retriever and generator. Retriever is an embedding model equipped with some similarity search methods which is used to vectorize the external knowledge documents and the input information form user or LLM. Generator is a generative language model composed of transformer which is usually either a BERT or an LLM. RAG framework is composed of two main parts, the first of which can be summarized as “query-summary-retrieval-answer”, while the second part is about the generation of vector store.

In the first part, “query” is the process of inputting the accident description and task requirements to the LLM. Considering that LLM has a max limitation of input token number, while the final input to LLM will be a combination of description and references, it is necessary to generate a brief summary before retrieval and final answer. Besides, due to the variety of expression, LLM tends to output some redundant explanations in its responses, diminishing the value of its answers. Prompt Engineering is considered as an effective method to guide the attention of LLM to key information [26], thereby, it is essential to construct a specified prompt with a reasonable structure for each turn of conversation. Followed summary is the retrieval process. To get a high-quality retrieval result, a comprehensive external knowledge base is needed which is the main content of the second part. Embedding model is critical in the construction of external knowledge vector base. It has the ability to mapping high dimensional data to low dimensional space to reduce the complexity of calculation, it can also understand the semantic relationships of words, and is thus capable of identifying synonyms and near-synonyms. The quality of vectorization will directly affect the quality of vector store, and will ultimately impact the results of retrieval. Thereby, an outstanding embedding model is vital to the method. However, relying solely on the embedding model is not sufficient to build a high-quality vector base. For the reason that every embedding model has a max limitation of input tokens, it is necessary to operate a pre-process, known as chunking, for the external knowledge documents. Chunking divides the external knowledge documents into small text pieces to meet the requirements of embedding model, which, however, also raises the possibility of semantic discrepancy. To diminish the influence of inappropriate chunking, reorganization of unstructured external knowledge document is necessary to preserve the consistency of its semantic structure. Once the vector store is constructed, the retrieval process can begin. Semantic similarity retrieval is a vector-based retrieval method, it firstly transfers texts or queries into vectors and then calculates the similarity between those vectors to perform the indexing. In RAG, the similarity between external knowledge vectors and summary vectors will be calculated to get the retrieval results. Generally, the more the retrieval results are, the better the response will be. However, it is not true for LLMs. As mentioned above, the attention of LLM is prone to be influenced by the content of Prompt, which means excessive information, especially domain-specific information, can distract the LLM and lead to incorrect answers. Therefore, an appropriate amount of retrieval results is also of great importance in our method. Finally, the summary and retrieval results are sent to LLM to generate the final answer.

To achieve the research goal, which is identifying the ATA100 codes of defective components and systems of each accident, a workflow was built as shown in figure 1. We firstly concatenate an accident description and a task description as the input for LLM, then LLM will summarize the causes and defective systems or components of each accident. Before the similarity retrieval, external knowledge is divided and vectorized by the embedding model to create a vector store. The vectorized accident summary, generated by the same embedding model, is then used to produce retrieval results which is a list of a pre-set number of external knowledge segments. Subsequently, both these external knowledge segments and the summary of descriptions will be submitted to the LLM for the generation of the final outcome. Besides, another workflow implemented in Python was established to enable the automatic batch analysis with our method.

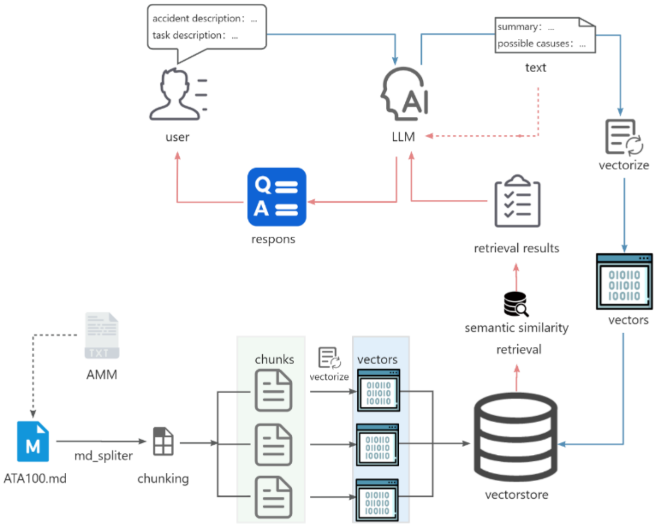


Figure 3. The workflow of RAG.

C. *Research of sensitive factors*

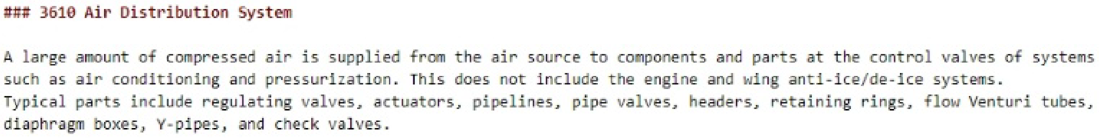
Learnt from the framework of RAG, the performance of RAG is greatly influenced by several key factors, which are LLM, embedding model, retrieved number of external knowledge segments (as known as top-k).

LLM is the core of RAG, its capability determines the quality of summary and answer. Different LLM was trained on different corpora and has various data scale, resulting various performance in different domain. Since our task is aviation domain-specific, a comparison experiment across different open-source LLMs is necessary.

Embedding model is responsible for the vectorization of LLM's output and external knowledge, which will directly affect the results of retrieval process and subsequently impact the final outcome of LLM, thus, an experiment across different embedding models is of vital importance.

Since too much retrieved information can cause the model's attention to be distracted, while recalling too little can result in insufficient reference knowledge, the exploration of an appropriate number of retrieved external knowledge segments is essential to assist the LLM in generating the best answer.

Besides those sensitive factors, the chunking process also need extra consideration. Chunking would divide the external knowledge document into segments to satisfy the demand of input token limitation of embedding model. Since the Markdown format can preserve the text structure, and there are mature tools for splitting Markdown documents, we reorganized our external document from PDF to markdown format, aiming to maintain structural integrity as well as avoid semantic discrepancies. After the conversion, the external knowledge document can be divided into structural pieces as shown in figure 2.



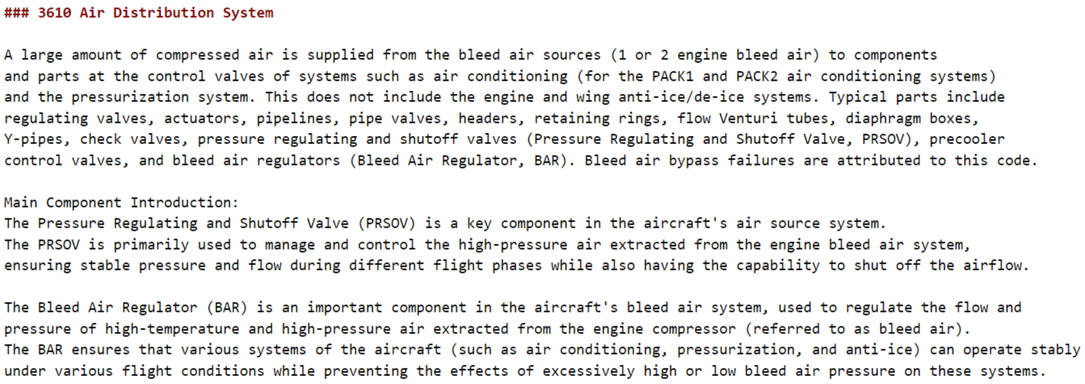
```
### 3610 Air Distribution System

A large amount of compressed air is supplied from the air source to components and parts at the control valves of systems
such as air conditioning and pressurization. This does not include the engine and wing anti-ice/de-ice systems.
Typical parts include regulating valves, actuators, pipelines, pipe valves, headers, retaining rings, flow Venturi tubes,
diaphragm boxes, Y-pipes, and check valves.
```

Figure 4. Example of a chunking piece of ATA100

D. *Research of the richness of the external knowledge base*

For the reason that ATA100 is a regulation document with comprehensive system information but few sub-system and component information, there are worries about the sufficient of retrieved results. To optimize the performance of RAG, we merged some information in AMM to ATA100, as shown in figure 3, for a more comprehensive external knowledge base. Comparison experiments were conducted to quantify the improvements.



```
### 3610 Air Distribution System

A large amount of compressed air is supplied from the bleed air sources (1 or 2 engine bleed air) to components
and parts at the control valves of systems such as air conditioning (for the PACK1 and PACK2 air conditioning systems)
and the pressurization system. This does not include the engine and wing anti-ice/de-ice systems. Typical parts include
regulating valves, actuators, pipelines, pipe valves, headers, retaining rings, flow Venturi tubes, diaphragm boxes,
Y-pipes, check valves, pressure regulating and shutoff valves (Pressure Regulating and Shutoff Valve, PRSOV), precooler
control valves, and bleed air regulators (Bleed Air Regulator, BAR). Bleed air bypass failures are attributed to this code.

Main Component Introduction:
The Pressure Regulating and Shutoff Valve (PRSOV) is a key component in the aircraft's air source system.
The PRSOV is primarily used to manage and control the high-pressure air extracted from the engine bleed air system,
ensuring stable pressure and flow during different flight phases while also having the capability to shut off the airflow.

The Bleed Air Regulator (BAR) is an important component in the aircraft's bleed air system, used to regulate the flow and
pressure of high-temperature and high-pressure air extracted from the engine compressor (referred to as bleed air).
The BAR ensures that various systems of the aircraft (such as air conditioning, pressurization, and anti-ice) can operate stably
under various flight conditions while preventing the effects of excessively high or low bleed air pressure on these systems.
```

Figure 5. Extended external knowledge (compared to fig. 2)

III. Experiments and Analysis

A. Experiment platform

All the experiments were conducted on an Intel(R) Xeon(R) Gold 6148 CPU at 2.40GHz with a 256 GB RAM and two Nvidia GeForce RTX 4090 GPU with a 24 GB memory. The programming language is Python 3.10 and several open-source libraries, including re, psycopg2 and FaissVector, were also used to support the data management and model inferring.

B. Experiments on Sensitive factors

Considering the available experiment platform, we chose GLM4 and Qwen series, including Qwen1.5-13B, Qwen1.5-32B and Qwen2-72B(int4), as candidate LLM models. We also selected several initial embedding models, which are BGE-m3, All-minilm-l6-v2, and gte-multilingual-base, from the huggingface hub website as candidate embedding models. The 'top-k' parameter is set to the values 5, 15, 25, 35, and 50.

To assess the test outcomes, three parameters were selected, which are accuracy of two-digit and four-digit codes and coverage of correctly recall of four-digit codes. A Python script was written to facilitated the calculation of the accuracy rates for both two-digit and four-digit codes within a dataset comprising 209 incidents, as well as to calculated the coverage rate of accurate four-digit codes within the recall of the external knowledge base. The resultant accuracy metrics are presented as below.

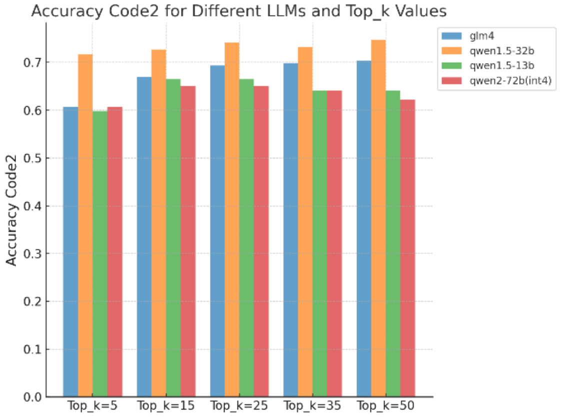


Figure 6. Accuracy code2 for different LLMs

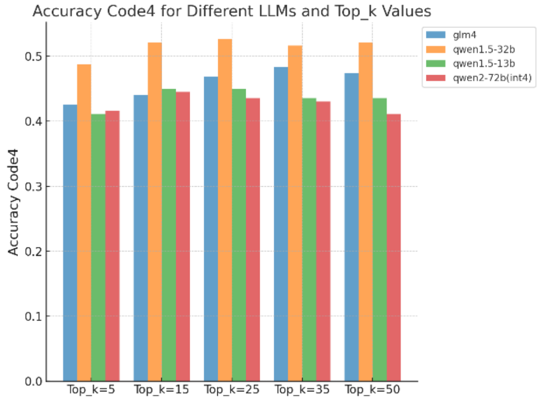


Figure 7. Accuracy code4 for different LLMs

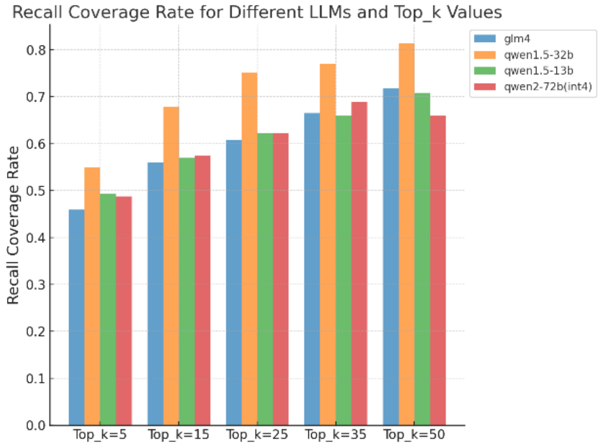


Figure 8. Recall coverage rate for different LLMs and top-k values

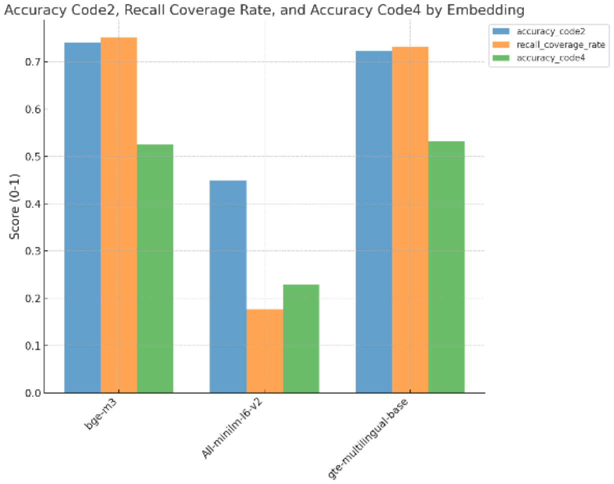


Figure 9. Accuracy code2, accuracy code4 and recall coverage rate for different embeddings (Top-k = 25)

As shown in the figures, Qwen1.5-32b showed a dominant performance over other LLMs, indicating its strong understanding of aviation knowledge. The outstanding performances of recall coverage rate in every top-k column represent that Qwen1.5-32b generates the best accident summary among other LLMs, providing the relatively most concise input for semantic similarity retrieval. The leading position in accuracy code2 and accuracy code4 experiments shows that Qwen1.5-32b has accumulated a richer body of aviation knowledge and has a deeper understanding of aviation knowledge. It is believed that Qwen1.5-32b is more suitable for aviation-related tasks.

From figures 6 and 7, most LLMs achieved the best performance around where top-k is equal to 25. Considering that recall coverage rate rises along with the increase in top-k, as shown in figure 8, there are two main explanations for the performance in figures 6 and 7. The explanation for the rise before where top-k equals 25 is that a low recall coverage rate results in the loss of correct external knowledge references. Although such situation has improved with the increase of top-k, redundant information has also increased, introducing additional interference to LLMs.

Learning from figure 9, BGE-m3 has the highest score in recall coverage rate, which means it has a better understanding of aviation knowledge, and is able to offer a relatively more comprehensive external knowledge reference for LLMs.

In conclusion, the optimal factors in our method are Qwen1.5-32b for the LLM, BGE-m3 for the embedding model and 25 for top-k.

C. *Experiments on external knowledge enhancement*

To explore the impact of the completeness of external knowledge base on the RAG, we conducted an external knowledge enhancement experiment on our benchmark dataset. The results are shown in figure 10.

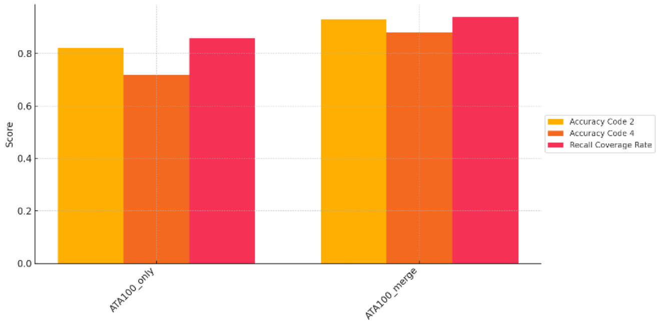


Figure 10. Result of external knowledge enhancement

The improvement brought by external knowledge enhancement is obvious on each parameter. By enhancing the external knowledge base, the retrieval of embedding model is more effective, improving the recall coverage rate from 0.86 to 0.94. And as a result of the improvement in recall coverage rate, two-digit codes accuracy has improved from 0.82 to 0.93, and four-digit codes has improved from 0.72 to 0.88 as well. One thing should be noticed is that the improvements in accuracy code2 and accuracy code4 are greater than the improvement made in recall coverage rate. The exceeded improvement is mainly because the quality of retrieved materials has improved with the enhancement of external knowledge, thus improved the performance of LLM.

IV. Conclusions

Aiming at locating the defective component or system in each accident description and classifying them according to ATA100 regulation, we built a workflow based on RAG method, as well as an external knowledge base merging ATA100 and some component information in AMM. The research on sensitive parameters reveals that the LLM, embedding model, and the number of retrieved external knowledge segments are all critical to RAG's performance. Among them, the embedding model has the most significant impact, as using an unsuitable model can reduce recall coverage by nearly 60%, leading to a drop in code2 and code4 accuracy by around 30%. The external knowledge enhancing experiment proved that enhancing the richness of the external knowledge can significantly improve the performance of RAG. The recall coverage rate improved 8% while the accuracy of two-digit and four-digit codes increased by 11% and 16%, respectively. The results of our work demonstrate that our RAG-based method can effectively identify defective components and systems in aviation accident reports, classify them using ATA100 codes, and provide technical guidance for accident investigations. These features make RAG feasible for industrial applications and highlight its potential for further research.

However, there are still few obstacles stopping the RAG from industry applications. Firstly, semantic discrepancies of external knowledge base exist due to the vectorization of text, highlighting the need for a more effective method of knowledge storing and accessing. Knowledge graph is known for its structured representation of information which can improve the accuracy of semantic understanding. The utilization of knowledge graph as external knowledge base would be a promising way to improve the performance of RAG. Secondly, the lacking of parameterized aviation knowledge in LLM also diminished the capability of our method, thus, it is necessary to train an aviation industry-specific LLM. Furthermore, our method currently employs only a vector retrieval strategy, but in the future, keyword-based retrieval and hybrid retrieval strategies can be considered based on task complexity.

References

[1] Chen Nongtian, TAN Xin, LI Rui. Application of REASON Model to investigation of the Aviation Maintenance Accident[J]. Journal of Transport Information and Safty,2012,30(02):96-98+126.

[2] Wang Lei, LIANG Yan. Statistical Analysis on Global Civil Aviation Accident Investigation Data [J]. China Transportation Review,2021,43(03):7-12.

[3] A. Pramono, J. H. Middleton and C. Caponecchia, Civil Aviation Occurrences in Indonesia. J. Adv. Transport., vol. 2020, 2020.

[4] X. Wang et al, Extracting Domain-Specific Chinese Named Entities for Aviation Safety Reports: A Case Study. Applied Sciences, vol. 13, (19), pp. 11003, 2023.

[5] Jianping Bao. Research on the topic of accident investigation report of runway based on improved LDA-Apriori algorithm [D]. Civil aviation flight university of China, 2024.DOI:10.27722/d.cnki.gzgmh.2024.000117.

[6] LI Junwei. Research on Cause Analysis and Control Technology of Aviation Accident Based on Complex System Theory [D]. Civil Aviation Flight University of China,2021.DOI:10.27627/D.cnki.gzmhy.2021.000474

[7] Zhu Zhitong. Analysis and Research of General Aviation Accident Based on Flight Data Visualization[D].Civil Aviation Flight University of China,2019.

[8] Sun Wenlong. Research on Safety Recommendations in Aviation Accident Investigation from the Perspective of Big Data [D].Civil Aviation University of China,2019.DOI:10.27627/d.cnki.gzmhy.2019.000339.

[9] Zhang Huaping et al, ChatGPT Performance Evaluation on Chinese Language and Risk Measures[J]. Data Analysis and Knowledge Discovery, 2023, 7(3): 16-25.

[10] P. Lewis et al, Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. ArXiv.Org, 2021.

[11] Yu, Retrieval-augmented Generation across Heterogeneous Knowledge. In Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Student Research Workshop, Seattle, Washington: Association for Computational Linguistics, Jul. 2022, pp. 52–58. doi: 10.18653/v1/2022.naacl-srw.7.

[12] R. Pradhan, Addressing AI hallucinations with retrieval-augmented generation. InfoWorld.Com, 2023.

[13] Y. Gao et al, Retrieval-Augmented Generation for Large Language Models: A Survey. ArXiv.Org, 2024.

[14] Kurnia Muludi et al, Retrieval-Augmented Generation Approach: Document Question Answering using Large Language Model. International Journal of Advanced Computer Science and Applications, vol. 15, (3), 2024.

[15] J. Miao et al, Integrating Retrieval-Augmented Generation with Large Language Models in Nephrology: Advancing Practical Applications. Medicina, vol. 60, (3), pp. 445, 2024.

[16] A. Golatkar et al, CPR: Retrieval Augmented Generation for Copyright Protection. ArXiv.Org, 2024.

[17] J. Y. Antonio et al, Financial Report Chunking for Effective Retrieval Augmented Generation. ArXiv.Org, 2024.

[18] Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. Language Models are Unsupervised Multitask Learners. OpenAI, 2019.

[19] Brown, T. B. et al, Language Models are Few-Shot Learners. In Advances in Neural Information Processing Systems (NeurIPS) 33, pp. 1877-1901. arXiv preprint arXiv:2005.14165, 2020.

[20] Vaswani, Ashish et al. Attention is All you Need. Neural Information Processing Systems (2017).

[21] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv preprint arXiv:1810.04805, 2019.

[22] Q. Ye et al, Prompt Engineering a Prompt Engineer. ArXiv.Org, 2023.

[23] J. Zheng et al, Fine-tuning Large Language Models for Domain-specific Machine Translation. ArXiv.Org

[24] F. Trad and A. Chehab, Prompt Engineering or Fine-Tuning? A Case Study on Phishing Detection with Large Language Models. Machine Learning and Knowledge Extraction, vol. 6, (1), pp. 367, 2024.

[25] S. Alghisi et al, Should We Fine-Tune or RAG? Evaluating Different Techniques to Adapt LLMs for Dialogue. ArXiv.Org, 2024.

[26] Liu, Pengfei et al. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. ACM Computing Surveys 55 (2021): 1 - 35.