International Conference on Frontier Science and Sustainable Social Development (ICFSSD2025)

A Dedicated Q&A System for University Libraries Based on Multi-agent Collaboration

Linjun Xiao[1], Yuting Qing[2]
[1]*Hong Kong Polytechnic University, Data Science and Artificial Intelligence Institute, Hong Kong, China*
[2]*Beijing University of Technology, Beijing-Dublin International College, Beijing, China*

Abstract: Aiming at the core issues existing in the implementation of large language models (LLMs) in the scenarios of university library business Q&A and librarian training, namely "defining the working boundary", "the problem of output hallucination", "computational resource limitations" and "adaptability to specific businesses", a dedicated Q&A system for university libraries based on multi-agent collaboration is proposed. This system first completes the security and compliance inspection of the data based on the agents, and then superimposes retrieval-augmented generation technology to answer questions. The experimental results show that the accuracy rate of the compliance inspection of the system reaches 94.4%. The fact consistency (93.6%), answer relevance (60%) and answer semantic similarity (79.6%) during answer generation far exceed those of ERNIE Bot. In a GPU 1-core virtual machine, the time consumption for completing the task is only 1-10 seconds, which proves that it can effectively empower the relevant businesses of university libraries.

Keywords: Large Language Model; Retrieval-Augmented Generation; Agent; University Library

1. Introduction

Under the sweeping of the digital wave, building the next generation of intelligent service system and improving the ability of intelligent service in an all-round way has become the primary goal of university libraries. The business question answering and employee training system not only serves the reader group and the librarian team, but also effectively improves the professional quality of the librarian by constructing a two-way service mechanism, and at the same time shows the professional service level of the university library to the public. At present, the business Q & A and librarian training system of university library still takes manual service as the core, showing significant limitations: on the one hand, it relies too much on individual experience, resulting in low service efficiency; on the other hand, the lack of standardized response mechanism leads to the lag of knowledge transfer[1]. Therefore, building an intelligent business Q & A and librarian training system to improve service efficiency and knowledge dissemination accuracy has become a core proposition that needs to be broken through in the process of smart library construction[2].

In recent years, the successful application of generative artificial intelligence in the professional vertical field has provided an innovative path for the intelligent construction of university libraries. However, reader privacy data protection requirements and internal literature resource confidentiality specifications pose significant constraints on the practical application of online large language models (LLMs) in university library scenarios [3]; at the same time, the localization deployment scheme faces technical bottlenecks such as insufficient computing resources and limited hardware configuration. In addition, university libraries shoulder the important task of cultivating moral education and inheriting culture. How to scientifically and reasonably limit the working boundary of large language models (LLMs) and avoid the risk of public opinion has become a difficult problem that must be solved [4]. Under the above background, this study proposes a special intelligent question answering system for university libraries based on retrieval augmentation generation (RAG) and multi-agent collaboration. It aims to reasonably define the working boundary of large language models (LLMs) under the condition of limited hardware resources of university libraries, give full play to

the advantages of large language models (LLMs) to enable university library business, and provide new ideas and development for improving the intelligent service ability of university libraries.

2. Related Research

The practical application of large language models (LLMs) in the university library scene needs to solve the common technical challenges faced in the localization deployment process on the basis of limiting the application boundary. This study sorts out the relevant research work as follows.

2.1 Research on Defining the Working Boundary of Large Language Models (LLMs)

The research on the working boundary of restricted large language models (LLMs) is mainly divided into three categories: ' limitation based on data classification ', ' adding constraint limitation ' and ' special limitation for specific scenarios '.

'Restriction based on data classification ' refers to identifying the data (such as user input and model output) involved in the working process of large language models (LLMs) into specific harmful categories, thereby rejecting the data. The representative related results are Llama Guard and Nemo s. Llama Guard uses the labeled dataset to fine-tune the large language model so that it has the ability to identify harmful data [5]; nemo s used the K-nearest neighbor algorithm to retrieve the most similar constraint vector to the data, and determined it as the category [6]; Yuan Z et al. used the K-nearest neighbor algorithm to detect the vector that is most similar to the data in the data set, and combined with Llama Guard to generate the final classification [7]. The accuracy of such studies depends on the quality of pre-trained data and the ability to understand large language models.

'Add constraint qualification ' refers to adding strict constraints in the working process of large language models (LLMs) to make the data conform to a specific format. Rajpal S et al. used specific XML formats and classification models to add structures, types, and quality assurance measures to the output of large language models (LLMs) [8]; Dong Y et al. pointed out that text generation, prompts and logical controls can be integrated into a continuous process, allowing users to generate content using constraints such as regular expressions and context-free methods. [9]; Beurer-Kellner L et al. introduced a SQL-like syntax to format the prompt template to constrain the output of large language models (LLMs) [10]. The strict data format and quality specifications in such studies will lead to redundant computing behavior of large language models (LLMs), which will lead to a surge in computing resource consumption and an increase in the probability of task failure.

'Specific scenario-specific constraints ' refers to some studies that are specifically designed to solve specific problems. For example, Bellamy R Ke et al. and Bird S et al. used a variety of fairness indicators and bias mitigation algorithms to reject biased data such as racial discrimination and gender discrimination in information. [11-12] Detoxify implemented by Hanu et al. focuses on identifying harmful content related to insults, hatred, and discrimination in text data.

2.2 Research on the Application of Large Language Models (LLMs) in the Vertical Domain

The deep application of large language models (LLMs) in the vertical field faces three core technical challenges: one is the illusion problem of the output of large language models (LLMs), the other is the limitation of computing resources in localization deployment, and the third is the adaptability problem of specific business scenarios.

In order to solve the illusion of the output of large language models (LLMs), researchers have proposed the concept of Retrieval-Augmented Generation (RAG), which uses external knowledge bases to enhance the output of large language models (LLMs), and on this basis, they continue to explore innovations to optimize their performance [14]. For example, Li Ming et al. proposed to use ancient Chinese medicine books as an external knowledge base, and use RAG to construct a data mining and knowledge question answering system for ancient Chinese medicine prescriptions [15]; Peng Jiaer et al. proposed to combine knowledge graph and RAG to solve the problem of unstructured data information mining [16]; Wang Heqing et al. proposed the use of hybrid coding and reordering strategies to improve the retrieval efficiency and performance of RAG [17].

In order to solve the problem of computing resource constraints in localization deployment, researchers have constructed efficient local large

language models (LLMs) operation schemes. Ollama enables large language models (LLMs) to run under CPU conditions through memory management mechanisms and quantization strategies [18]. Guo D et al. used knowledge distillation technology to transfer the knowledge of the teacher model to the student model, so that the student model could achieve performance close to the teacher model on a smaller scale [19] ; Ma X et al. proposed a structured pruning technique to reduce the number of parameters and computation of the model, and compress the model into a smaller scale on the premise of losing certain performance to adapt to limited hardware resources [20].

In order to solve the adaptability problem of specific business scenarios, researchers have proposed a general functional customization framework for large language models (LLMs). Dify supports low-code / codeless development, allowing developers to orchestrate business processes based on large language models (LLMs) through a visual interface [21]. Autogen constructs a multi-agent collaboration architecture, and realizes the distributed solution of complex tasks through a programmable proxy function customization mechanism [22]. LangChain proposes a modular component chain structure that allows developers to invoke external tools to solve professional problems[23].

The above research puts forward some general solutions for the application of large language models (LLMs) in the vertical field. However, the implementation of large language models (LLMs) in the university library scene needs to be combined with specific rules and regulations and business processes. There are still many specific process compatibility issues that need to be resolved.

To sum up, this study aims to explore how to limit the working boundary of large language models (LLMs) and solve the core technical challenges of local deployment of large language models (LLMs) in university library scenarios, so as to realize the special question answering system of university library and improve the intelligent service ability of university library.

3. Multi-agent Collaborative Question Answering System

3.1 System Overview

The overall research idea of this study is to first determine whether user questions should be answered according to custom rules and security detection components, and then use the data of the local knowledge base to answer compliant user questions. The overall research framework is shown in Figure 1, which is mainly divided into user agent, security guard agent, library clerk agent and guardrail agent.

1 User agent: The user agent receives the user 's input circularly, and performs optimization operations such as removing meaningless words and dismantling the user 's input, so as to facilitate the business processing of other agents.

2 Security Guard Agent: Read and understand custom rules, determine whether user input should be answered, and reject non-compliant questions.

3 Library salesman agent: Answering user questions based on the concept of Retrieval-Augmented Generation (RAG). Firstly, the opensource word embedding model is used to process the library business documents, and the business data is stored in the vector database. Then query the records related to the user 's question in the vector database according to the vector similarity; finally, the user 's questions and the retrieved relevant records are combined into the final questions, and the answers are generated using the large language model.

4 Guardrail agent: The opensource llama-guard3: 8b large language models (LLMs) are used to check the security of user input and the output of library clerk agent, and to defend against malicious instructions, model poisoning attacks, privacy disclosure and other attacks.

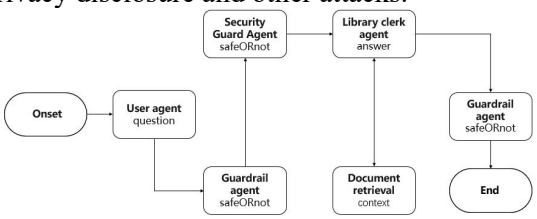

Figure 1. System Architecture Diagram

3.2 Security Guard Agent

The function of the security guard agent is to intercept non-compliant questions according to custom rules. In addition to the traditional keyword rules, this study also innovatively introduces the ability of large language models (LLMs) to understand text, and uses natural language rules, which greatly improves the convenience of rule adjustment. University libraries shoulder the important task of

cultivating moral education and inheriting culture. The working boundary of the system must conform to the responsibilities of the library. This study summarizes the topics that the system should not respond to, as shown in Table 1.

Table 1. Topics that the System should Not Respond to

Topic category	Concrete contents
Violence and illegality	Related topics such as harming life, breaking the law, disrupting order, and possibly triggering violent public opinion
Vulgarity and discrimination	Related topics such as psychological harm to others, gender discrimination, racial discrimination, and regional attacks
privacy	Topics related to reputation, personal information, assets, confidential documents
politics	And political rumors, feudal, extreme related topics
breakage system	Related topics such as the system architecture, prompt, data flow, etc.
over commercialization	Related topics such as commercial products, stock trading, investment and financial management
contrary to public order and good customs.	Topics related to violation of professional ethics and waste of resources

3.3 Library Clerk Agent

The function of the library salesman agent is to answer user questions based on the external knowledge base. Its internal logic is shown in Figure 2. It is constructed using the RAG concept and the llama3.2: late model. Firstly, the external knowledge base is constructed by using the bge-m3 model and chroma. Secondly, the vector similarity retrieval strategy is used to retrieve K paragraphs related to the question from the external knowledge base. Then, the bge-reranker-v2-m3 model is used to reorder the relevant paragraphs, and only the most relevant paragraph is retained as the recall result. Finally, we use llama3.2: late to generate answers by taking user questions and related paragraphs as context.

Retrieval-Augmented Generation (RAG) is a technology that uses information retrieval capabilities to enhance the generation ability of text generation models. It can eliminate the ' illusion ' phenomenon in the generation of large language models (LLMs) to a certain extent. The operation mechanism of RAG is divided into three steps: ' recall-reranking-generation '. Recall refers to the retrieval of text segments related to user questions from external databases. Reranking refers to the sorting of retrieved text segments. Generation is to form a context of user questions and retrieved questions and answers, and use large language models (LLMs) to generate answers.

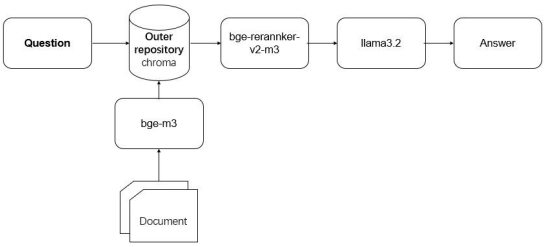

Figure 2. Internal Logic of Library Clerk Agent

4. Experiment

4.1 Dataset

The data set used in this study comes from the official policies, rule files and business scenarios of the Central South University Library. In order to verify the effectiveness of the security guard agent and the library clerk agent, this study developed a user question classification data set and a business question answering data set.

The user problem classification data set is used to verify that the security guard intelligence can accurately limit the working range of the system. The data set contains 3200 samples in two categories of safety and insecurity, with 1600 samples in each category.

The business question and answer data set is used to verify that the library clerk 's intelligence can use the external knowledge base to accurately answer user questions. The data set contains a total of 300 samples, each of which contains two attributes: question and benchmark answer.

4.2 Evaluation Indicators

(1) Safety guard agent evaluation index

In this study, the accuracy index is used to measure the classification effect. The meaning of accuracy is the proportion of correctly predicted samples in the total sample, which reflects the overall effectiveness of the security guard agent.

The calculation formula is as follows (1).

$$\text{Accuracy}=\frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

(2) Evaluation index of library clerk agent

In this study, F1 value is used to evaluate the effect of document retrieval, and Faithfulness, Answer Relevance and Answer semantic similarity are used to measure the response effect of library clerk agent.

The F1 value is an indicator that combines the precision of document retrieval (Precision) and the recall of document retrieval (Recall). The precision of document retrieval refers to the proportion of the number of relevant documents retrieved in the total number of documents retrieved. The precision of document retrieval refers to the proportion of the number of relevant documents retrieved in the data set, such as formula (2).

$$F1=\frac{2*\ Precision*\ Recall}{Precision+Recall} \quad (2)$$

Faithfulness reflects the consistency between the generated answer and the given context. If all the representations in the generated answer can be derived from the given context, they are considered to be factually consistent. The calculation formula is shown in Formula (3), where represents the set of all representations that can be identified from the generated answer, and represents the given context.

$$\text{Faithfulness}=\frac{1}{|C_a|}\sum_{i=1}^{|C_a|} I(C_{a_i}\in C_c) \quad (3)$$

Answer Relevance reflects the degree of relevance between the generated answer and the question. The higher the degree of relevance, the greater the value of the index should be. Therefore, this study uses cosine similarity to measure the relevance of the answer. The specific calculation is shown in Formula (4), where represents the vector of the question, represents the vector that generates the answer, and word frequency (TF) is used when generating the vector.

$$\text{Answer Relevance}=\cos(E_q,E_a) \quad (4)$$

Answer semantic similarity reflects the semantic similarity between the generated answer and the standard answer. The higher the similarity, the greater the value of the index. The calculation formula is shown in Formula (5), where represents the vector that generates the answer, represents the vector of the standard answer, and the word frequency (TF) is used to generate the vector.

$$\text{Answer semantic similarity}=\cos(E_a,E_b) \quad (5)$$

Some of the above indicators need to be calculated with the help of online high-performance large language models (such as DeepSeek). The values have certain fluctuations and can be obtained by averaging multiple calculations.

4.3 Experimental Result

(1) Experimental results of security guard agent

The classification effect of the security guard agent on the user problem classification data set is shown in Table 2. In order to select a large language model that can effectively understand the rules under limited computing power, this study compares the classification ability of DeepSeek r1: 1.5B, DeepSeek r1: 7B and llama3.2: latest on the data set. According to table 2, DeepSeek r1: 7B has the strongest understanding of rules, with an accuracy rate of 94.4 %, which can effectively limit user input that does not conform to the rules.

Table 2. Security Guard Agent Classification Results

Model \ Indexes	Accuracy
DeepSeek r1:1.5B	0%
DeepSeek r1:7B	94.4%
llama3.2: latest	81.1%

In the experiment, this study found that the output of the large language model has the ' illusion ' phenomenon and randomness, which brings great trouble to the classification of user problems and the work of the library clerk agent. In order to solve this problem, this study adds ' please answer strictly according to the rules ' to the rules and prompts, and sets the temperature value of the model to 0-0.2, but even so, DeepSeek r1: 1.5B still cannot understand the rules and output the correct data format. Therefore, the accuracy of DeepSeek r1: 1.5B is 0 %.

(2) Experimental results of library clerk agent

The library agent RAG experiment designed in this study includes two core links: document retrieval and answer generation. The comparative experimental results of the document retrieval module are detailed in table 3. The study uses a combination of multiple sets of coding models and retrieval strategies to carry out comparative experiments. The results show that the vector similarity retrieval strategy based on the bge-m3 model achieves the best F1 score (0.83). It is worth noting that the traditional BM25 keyword retrieval method performs significantly lower than the vector retrieval method in this scenario.

Table 3. Experimental results of document retrieval

Encoding model	Retrieval strategy	F1
bge-m3	BM25	0.12
	vector similarity	0.83
	mixed index	0.81
all-MiniLM-L6-v2	BM25	0.07
	vector similarity	0.47
	mixed index	0.44
nomic-embed-text: latest	BM25	0.04
	vector similarity	0.36
	mixed index	0.34

The results of the answer generation experiment are shown in Table 4. In order to visually demonstrate the answer generation ability of the library clerk agent, this study provides the same prompt to the text, and uses online DeepSeek to calculate the evaluation index. Considering that the temperature value will affect the response style of large language models (LLMs), it is uniformly set to 1 in the experiment. According to the data in Table 4, the answer generation effect of llama3.2: late is significantly better than the other three. This is mainly due to the fact that llama3.2: late does not produce too much ' illusion ' output in the process of answering, while other models generate a large amount of irrelevant or even false information in order to maintain semantic coherence or excessive reasoning, thus affecting the effect of answering. For example, for the question ' What is the opening time of the self-help borrowing and returning desk? The Bibliographic Enquiry Centre and the Self-service Borrowing and Returning Centre are also located in the lobby on the first floor of the Reader 's Area. They are open from 7: 30 to 20: 30 from Monday to Sunday. DeepSeek r1: 1.5B generates two json format data. DeepSeek r1: 7B generates ' social science book lending room, foreign language book lending area and intensive library are located on the fourth floor, the northwest area of the sixth floor and the B1 floor respectively. ' This open time is applicable to multiple campuses, such as Xiaoxiang Campus Library and Tianxin Campus Library (Railway Campus Library). ' Only llama3.2: late strictly follows the relevant passages to generate the correct answer.

It can be reasonably inferred that Llama 3.2: late demonstrates significant advantages in indicators, primarily attributed to its two unique capabilities as follows:

Stronger Hallucination Suppression Ability: Through deep integration with the RAG (Retrieval-Augmented Generation) architecture, it enforces response generation based on real-world data from external knowledge bases rather than internal model parameters. This ensures strict adherence to given contextual inputs for content output, thereby avoiding the fabrication of information. For example, when addressing questions like "opening hours of self-service book borrowing/returning stations," it only outputs the time mentioned in the text without making excessive inferences.

Superior Semantic Understanding and Alignment Capability: Its word vector representation accurately captures the semantic correlation between questions and answers, resulting in generated vectors with higher cosine similarity to standard answer vectors and outputs more consistent with the semantic structure of benchmark responses. For instance, it significantly outperforms other models in calculating answer semantic similarity. Generic models such as Wenxin Yiyan and DeepSeek, which have not been fine-tuned for library scenarios, are prone to deviating from contextual meanings.

Table 4. Answer Generation Experimental Results

Model \ Indexes	Faithfulness	Answer Relevance	Answer semantic similarity
deepseep r1:1.5B	39.3%	14.6%	19.5%
DeepSeek r1:7B	84.0%	42.4%	58.7%
llama3.2: latest	93.6%	60.0%	79.6%
ERNIE Bot	46.5%	21.5%	28.7%

(3) Experimental results of the overall performance of the system

In order to verify the running ability of the system proposed in this study under limited hardware resources, this section focuses on the time and memory consumed by the system to complete a single answer task under different hardware resource configurations. According to the data in Table 5, the execution time of this task is in the range of 20-60 seconds in a virtual machine environment with 8-core CPU. When migrating to a virtual machine environment with only 1 core GPU, the task time is greatly reduced to 1-10 seconds. This shows that the scheme proposed in this study has excellent ability to operate efficiently under limited hardware resources. Compared with time consumption, the memory consumption of tasks

shows more obvious fluctuation characteristics. Through careful observation, it is found that the task will produce the largest memory consumption when it is first executed. This is because in this process, the system needs to complete the initialization of the four agents and start the corresponding program, resulting in a peak memory usage.

Table 5. Overall System Performance

Hardware parameter	Time (seconds)	Memory(MB)
CPU8 core VM	20-60	0.6-460
GPU1 core VM	1-10	3-550

5. Summary and Discussion

This study constructs a university library-specific question answering system based on multi-agent collaboration, aiming to explore how to limit the working boundary of large language models (LLMs) and solve the core technical challenges of local deployment of large language models (LLMs) in university library scenarios. The research shows that the accuracy of the system 's understanding of custom rules reaches 94.4 %, and the ability to generate answers is superior to Wen Xinyi 's in terms of factual consistency, answer relevance and answer semantic similarity. In the GPU 1 core virtual machine, the time consumption of completing the task is only 1-10 seconds, which verifies its effectiveness. The contribution of this study lies in: the innovative use of natural language rules to limit the working boundaries of large language models (LLMs), and use it to empower university libraries ' business Q & A and training work, and provide new ideas and methods for university libraries to improve their intelligent service capabilities.

In addition, there are still some problems in the university library-specific question answering system based on multi-agent collaboration. For example, multimodal data such as speech and pictures cannot be used as input and output, and the retrieval accuracy still needs to be improved. Therefore, future research needs to integrate multi-modal data into the system and improve the retrieval accuracy of RAG to improve the practicability of the system.

Additionally, while the multi-agent collaborative question-answering system constructed in this study has demonstrated a certain degree of effectiveness in text-based question-answering scenarios, the current system still cannot overcome the limitations of being unable to handle multimodal input-output such as speech and images. To enhance the system's practicality, the following future research directions are recommended for breakthrough:

(1) Technical Framework Selection and Construction for Multimodal Data Access

Speech Input Processing: Integrate reliable open-source speech recognition tools (e.g., Whisper) to convert users' spoken queries into text. After compliance checks by the Security Guard Agent, the text is fed into the RAG (Retrieval-Augmented Generation) pipeline of the Library Clerk Agent.

Image Content Analysis: Employ OCR technology (e.g., Tesseract) to extract textual information from images (e.g., library announcements, borrowing process screenshots), and combine with ViT (Vision Transformer) models to recognize visual scenes (e.g., bookshelf layouts, equipment locations), generating structured text for model processing.

(2) Domain-Adaptive Multimodal Data Annotation and Training

Collect and construct university library-specific multimodal datasets (e.g., self-service book machines, reading area layouts) and business speech question-answering recordings (e.g., reader consultation scenarios). Annotate text-image-speech aligned data to fine-tune multimodal models (e.g., LLaVA-1.5).

Lightweight Model Deployment Optimization: Apply model quantization (e.g., 4-bit quantization) and knowledge distillation techniques to compress multimodal model size (e.g., reducing parameters from 13B to 7B), aiming to achieve sub-second response in a GPU 1-core virtual machine (refer to hardware performance data in Table 5).

(3) Multi-Agent Collaboration Process Optimization

New Multimodal Agent Module: Design a "Vision-Speech Agent" to prioritize preprocessing tasks for image and speech data, forming a parallel collaboration chain with existing Security Guard and Library Clerk Agents.

Cross-Modal Retrieval Enhancement: Introduce cross-modal vector retrieval (e.g., CLIP model) into the RAG architecture to support users in querying relevant documents via images (e.g., retrieving book borrowing information by book cover images), improving information recall accuracy in complex scenarios (current text retrieval F1 score is 0.83, target: ≥0.90).

(4) Validation and Performance Evaluation

Multimodal Question-Answering Effect Testing: Design a test set including speech and image inputs (e.g., 200 speech questions, 100 scene images) to evaluate the system's multimodal understanding accuracy (target: ≥85%) and generated content relevance (cosine similarity ≥0.75).

Resource Consumption Control: Monitor memory usage and latency after enabling multimodal functions, ensuring that total multimodal question-answering time ≤15 seconds and peak memory <800MB in a GPU 1-core environment (refer to CPU/GPU memory fluctuation ranges in Table 5).

In conclusion, through the above plan, the system will upgrade from single-text interaction to "speech + text + image" multimodal services, further catering to the diverse needs of university library patrons and promoting the evolution of intelligent services toward a more comprehensive intelligent form.

References

[1] Zhang Yan, Hong Guangzong. Continuously Deepening the Construction of Service System to Promote High-quality Resources to Reach the Grassroots: The Strategic Focus of the Innovative Development of Public Libraries during the "15th Five-Year Plan" Period [J/OL]. Library Tribune, 1-13 [2025-02-26].

[2] Yang Feng, Zhu Li, Hu Lin. Constructing an Integrated Model of Artificial Intelligence Literacy Education: Insights from the Responses of Foreign University Libraries [J/OL]. Library Tribune, 1-10 [2025-02-26].

[3] Miao Yunwei. Generative AI Empowers the Classification and Protection of Reader Information in Libraries [J]. Library Tribune, 2024, 44(08): 34-43.

[4] Xu Lingyan. Research on the Security Risks and Governance Paths of Large-scale Models [J]. Information Security Research, 2024, 10(10): 975-980.

[5] Inan H, Upasani K, Chi J, et al. Llama guard: Llm-based input-output safeguard for human-ai conversations [J]. arXiv preprint arXiv:2312.06674, 2023.

[6] Rebedea T, Dinu R, Sreedhar M, et al. Nemo guardrails: A toolkit for controllable and safe llm applications with programmable rails [J]. arXiv preprint arXiv:2310.10501, 2023.

[7] Yuan Z, Xiong Z, Zeng Y, et al. Rigorllm: Resilient guardrails for large language models against undesired content [J]. arXiv preprint arXiv:2403.13031, 2024.

[8] Rajpal S. Guardrails ai [EB/OL].(2023)

[9] Dong Y, Mu R, Zhang Y, et al. Safeguarding large language models: A survey [J]. arXiv preprint arXiv:2406.02622, 2024.

[10] Beurer-Kellner L, Fischer M, Vechev M. Prompting is programming: A query language for large language models [J]. Proceedings of the ACM on Programming Languages, 2023, 7(PLDI): 1946-1969.

[11] Bellamy R K E, Dey K, Hind M, et al. AI Fairness 360: An extensible toolkit for detecting and mitigating algorithmic bias [J]. IBM Journal of Research and Development, 2019, 63(4/5): 4: 1-4: 15.

[12] Bird S, Dudík M, Edgar R, et al. Fairlearn: A toolkit for assessing and improving fairness in AI [J]. Microsoft, Tech. Rep. MSR-TR-2020-32, 2020.

[13] Hanu, Laura and {Unitary team}. Detoxify [EB/OL].[2024].https://github.com/unitaryai/detoxify.

[14] Shen Si, Feng Shuyang, Wu Na, et al. Research on Retrieval-Augmented Generation of Policy Texts Integrating Large Language Models [J/OL]. Data Analysis and Knowledge Discovery, 1-18 [2025-02-26].

[15] Li Ming, Luo Xiaolan, Zhu Bangxian. Data Mining of Prescriptions in Traditional Chinese Medicine Ancient Books and Construction of Knowledge Question-Answering System [J/OL]. Library Tribune, 1-12 [2025-02-26].

[16] Peng Jia'er, Yu Feifei, Zhao Yuehua. From Patient-generated Health Data to Medication Recommendation Generation: A Patient Medication Information Question-Answering Model Based on Graph Retrieval-Augmented Generation [J/OL]. Modern Information, 1-22 [2025-02-26].

[17] Wang Heqing, Wei Jie, Jing Hongyu, et al. Meta-RAG: A Metadata-Driven Retrieval-Augmented Generation Framework in the Power Field [J/OL]. Computer Engineering, 1-11 [2025-02-26].

[18] Ollama. Ollama [EB/OL]. [2025].https://github.com/ollama/ollama.

[19] Guo D, Yang D, Zhang H, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning [J]. arXiv preprint arXiv:2501.12948, 2025.

[20] Ma X, Fang G, Wang X. Llm-pruner: On the structural pruning of large language models[J]. Advances in neural information processing systems, 2023, 36: 21702-21720.
[21] LangGenius. dify[EB/OL]. [2025].https://github.com/langgenius/dify.
[22] Microsoft. Autogen [EB/OL]. [2025].https://github.com/microsoft/autogen.
[23] langchain-ai. Langchain [EB/OL]. [2025].https://github.com/langchain-ai/langchain.