PROCEEDINGS OF SCIENCE

OMAI: A Specialized Large Language Model for Operational Maintenance in Institute of High Energy Physics

Siyang Chen,[a,b,*] **Haibo Li,**[a] **Zhengde Zhang,**[a] **Zhihui Sun**[a] **and Yaodong Cheng**[a]

[a] *Computing Center, Institute of High Energy Physics, Chinese Academy of Sciences, Beijing 100049, P.R.China*

[b] *School of Nuclear Science and Technology, University of Chinese Academy of Sciences, Beijing 100049, P.R.China*

E-mail: lihaibo@ihep.ac.cn

This study presents the integration of Artificial Intelligence Operations (AIOps) at the Institute of High Energy Physics (IHEP) Computing Center to address the challenges of managing a complex IT infrastructure and providing efficient user support. We propose a question-answering (QA) system utilizing advancements in large language models and Natural Language Processing (NLP) to facilitate rapid problem resolution and document queries, aiming to reduce the operational staff's workload. The paper highlights the limitations of applying large language models in specialized domains due to insufficient exposure to domain-specific texts. It discusses using targeted fine-tuning and Retrieval-Augmented Generation (RAG) technology to enhance model performance. By compiling Helpdesk QA datasets for fine-tuning the Xiwu model, specializing in high-energy physics, and developing an RAG framework to integrate external knowledge, we significantly improve operational efficiency and support within the IHEP's computing environment.

International Symposium on Grids and Clouds (ISGC2024)
24 -29 March, 2024
Academia Sinica Computing Centre (ASGC), Institute of Physics, Academia Sinica Taipei, Taiwan

[*] Speaker

PoS(ISGC2024)034

1. Introduction

The convergence of Artificial Intelligence and Machine Learning technologies in IT operations management, known as Artificial Intelligence Operations (AIOps), marks a pivotal advancement designed to automate the intricate monitoring and analysis of the extensive data produced by IT systems. This innovative field aims to significantly reduce the need for human intervention, thereby enhancing operational efficiency. A prime example of the challenges AIOps seeks to address can be observed at the Computing Center of the Institute of High Energy Physics (IHEP CC), which grapples with the complexities of managing a hybrid storage solution. This setup combines disk and tape systems, specifically the Lustre and EOS file systems, boasting capacities of 50PB and 58PB, respectively. Supported by a formidable array of 87,552 CPU cores and 219 GPU cards, IHEP's computational resources are indispensable for cutting-edge high-energy physics research [1]. However, the institute's extensive computing and storage infrastructure and large user base generate a significant volume of daily system logs, alerts, malfunctions, and user support tickets [2].

Such a scenario underscores the necessity for sophisticated and intelligent operational management strategies to tackle the intricate demands of system maintenance effectively. In response, the IHEP Computing Center has pioneered several innovative solutions, including implementing a Helpdesk platform to streamline operational issues and manage requests more efficiently. Despite these advances, maintaining continuous operations management for the computing cluster and promptly addressing user queries with a limited staff remains daunting.

Developing a question-answering (QA) system to assist operational staff with daily tasks and basic user inquiries is crucial to mitigate these challenges. By enabling natural language interactions with a large model for rapid problem resolution and professional document queries, such a system promises to significantly alleviate the operational staff's workload, thereby enhancing overall efficiency.

The recent surge in advancements in large language models, such as ChatGPT [3], ERNIE [4], and Llama [5], has significantly propelled the field of Natural Language Processing (NLP) forward, particularly in its ability to generate text that closely mirrors human writing. These models have demonstrated exceptional performance across various applications, from engaging in casual conversations to addressing standard queries [6]. However, their application in specialized domains, especially for handling domain-specific queries, must be improved by their limited exposure to specialized texts during the pre-training phase. This limitation frequently results in less-than-optimal performance in scenarios that demand a deep understanding of domain-specific knowledge. Employing targeted strategies for secondary pre-training and fine-tuning can substantially improve the ability of these models to address questions within specialized domains.

Although secondary pre-training and fine-tuning are effective in imbuing models with new domain knowledge, these processes are notably resource-intensive, inefficient, and risk catastrophic forgetting as models need help assimilating new information at lower learning rates. On the other hand, an increase in the learning rate might compromise the model's performance in different areas. Consequently, the model's training must be approached with caution. The Retrieval-Augmented Generation (RAG) [7] technology offers a promising solution by providing additional contextual information from vector knowledge database, enabling large language models to adapt

PoS(ISGC2024)034

more effectively to diverse tasks and domains, enhancing their performance in specific subtasks [8]. Moreover, it facilitates ongoing learning and updates by retrieving the latest information, ensuring that models continue to improve over time without requiring frequent and extensive retraining.

This paper delves into the potential of leveraging large language models within the operational domain of IHEP. By compiling daily operational logs and Helpdesk QA datasets and conducting fine-tuning on the Xiwu [9], which is specialized for the HEP domain, we have equipped the model to adeptly handle queries related to the computing cluster's usage and operations in HEP experiments. Furthermore, we have developed a RAG framework that bolsters model performance by integrating external knowledge, thereby enabling the querying of local databases for system manuals and documents to enhance the accuracy and reliability of answers.

2. Related Works

2.1 Intelligent Question-answering System

In 1950, Turing and others proposed a Turing Test to determine whether a machine could think [10]. This test is considered the earliest prototype of QA systems. QA systems are typically defined in academia as NLP applications aimed at automatically understanding questions posed by users in natural language and providing accurate, concise answers from one or more text sources. QA systems attempt to mimic the human ability to answer questions, requiring an understanding of the intent and complexity of the question and the ability to retrieve, understand and refine the correct answer from a vast amount of information resources.

Early researchers mainly focused on using computational linguistics techniques to improve the performance of answer systems. Until the end of the twentieth century, the rapid development of the Internet brought a massive amount of online textual material, ushering QA systems into the era of text. Nowadays, with the development of GPU computing power, the richness of datasets, and the rapid advancement in deep learning, pre-trained language models such as BERT [11], GPT [12], and T5 [13] have shown powerful semantic representation capabilities and are often used to build QA systems [14].

2.2 Large Language Model

Large Language Models, encompassing tens or hundreds of billions of parameters, primarily utilize the Transformer architecture and are trained on extensive text datasets. Notable examples include GPT-3 [15], PaLM [16], QWEN [17], and Llama [5]. These models markedly enhance the scaling of the model, the diversity of the pre-training data, and the computational resources required compared to conventional machine learning models [6].

LLMs excel in parsing natural language more accurately and producing high-quality textual outputs tailored to specific contexts. A unique aspect of LLMs is their "emergent abilities"—capabilities that manifest exclusively in large-scale models. These abilities differentiate LLMs from earlier pre-trained models, enabling them to manage various downstream tasks adeptly without bespoke parameter modifications. Through Instruction Tuning, LLMs can adapt to new tasks autonomously, showcasing robust generalization capabilities. Furthermore, unlike smaller models, LLMs can navigate complex multi-step reasoning tasks, employing strategies like the Chain of Thought (CoT) to tackle challenging problems effectively [18].

2.2.1 Llama2

Llama is an open-source foundational language model developed by Meta, featuring a range of 7B to 65B parameters [5]. It is built upon the transformer architecture and incorporates advanced techniques such as pre-normalization strategies [19], SwiGLU activation function [20], and RoPE rotary position encoding [21]. The model is trained on a publicly available dataset comprising 1.4 trillion tokens. Llama has achieved superior performance levels by selecting a subset of weights in conjunction with a vast dataset.

Building upon Llama, Llama2 has significantly expanded its training dataset to 2 trillion tokens and has implemented several enhancements to boost its performance [22]. These improvements include enhanced data cleaning, refined data mixing methods, and utilizing over 40% of tokens for training. Additionally, Llama2 has doubled the context length, providing a broader scope for understanding and generating text. Furthermore, the introduction of grouped-query attention (GQA) in larger models, such as the 70B model, significantly enhances scalability, making Llama2 a more robust and efficient AI model.

2.2.2 Vicuna

Researchers investigating Vicuna [23] discovered that advanced large language models like GPT-4, through comprehensive training and alignment with human preferences, consistently align with human responses in question-answering tasks. To this end, they compiled dialogue data from ShareGPT.com, segmented the extensive conversations into shorter segments compatible with the model's maximum context capacity, and developed training datasets of varying sizes to refine the LLaMA base model.

Upon completing the refinement, the authors employed state-of-the-art LLMs, including GPT-4, as benchmarks to assess the performance of various models. This approach offers a quick and uniform evaluation method while substantially lowering labor expenses. Testing revealed that Vicuna could rival GPT-4 in specific contexts at a reduced cost, marking a notable enhancement from the original LLaMA model.

2.2.3 Xiwu

Xiwu is a large language model tailored for the high-energy physics domain developed by the IHEP research team [9]. They introduced an innovative approach named "Seed Fission Technology", which enables the swift generation of question-and-answer pairs pertinent to specific scientific fields. This technique begins with a singular seed topic and produces diverse and comprehensive question-and-answer pairs through specially designed chatbots (Newbie, Expert, and Checker). This process created over 1000 high-quality pairs within the high-energy physics domain. Human assessments were conducted using 5100 prompts to evaluate the efficacy of Xiwu. These assessments compared its performance against other models, including Vicuna-13B and ChatGPT. The findings from these evaluations demonstrated that Xiwu significantly outperformed its counterparts in handling question-and-answer tasks related to high-energy physics. Xiwu utilized a corpus that contained extensive information on IHEP and major scientific facilities during its training, making it an ideal foundation model for training to address users' questions better.

2.3 Innovative Contributions to Large Language Models for Operational Maintenance in High-Energy Physics

The technology underpinning traditional question-answering systems exhibits significant limitations in developing an assistant for the IHEP, primarily due to data confidentiality requirements that preclude Internet searches for a vast array of user queries, coupled with the Internet's scant information on IHEP and its experiments. Moreover, these systems often cater to specific fields, whereas IHEP's interdisciplinary research spans physics, chemistry, nuclear technology, mechanics, and computer science, presenting complex challenges that defy simple expert system solutions. Consequently, establishing a localized, robust dialogue capability and a comprehensive knowledge base for operational maintenance within IHEP is crucial. Building upon prior research, we present a series of innovative advancements to enhance the performance and precision of the question-answering system for operational maintenance in high-energy physics. Our primary contributions include:

- We meticulously collected and organized a substantial corpus from the IHEP Computing Center's Helpdesk platform, comprising over 4,500 high-quality, multi-turn dialogues. We achieved this through careful data cleaning and applying advanced LLM scoring techniques. This dataset augments research resources for IHEP's operational maintenance and is valuable for model training and validation.
- The development of a new operational maintenance large model, OMAI, leveraging the cutting-edge Xiwu model in high-energy physics. By pre-training and fine-tuning with the Helpdesk QA dataset and actual maintenance dialogues, OMAI integrates domain-specific knowledge and exhibits enhanced question-answering capabilities, offering a more innovative, more efficient operational maintenance solution.
- Incorporating Retrieval-Augmented Generation (RAG) technology bolsters the OMAI model's capabilities, enabling real-time access to various documentation and logs. It enhances the model's accuracy and reliability, providing more effective operational maintenance support.

In conclusion, this paper's contributions represent a technical leap forward and hold considerable promise for practical application. By leveraging the Helpdesk QA dataset and the innovative OMAI model, we offer a novel intelligent solution for high-energy physics operational maintenance, poised to boost operational efficiency and user satisfaction significantly.

3. Method

3.1 Helpdesk QA dataset

Despite the broad knowledge base of large language models, they may need help to precisely answer specific questions within certain domains. To mitigate this limitation, collecting and fine-tuning conversational datasets from targeted fields can significantly enhance the models' adaptability and accuracy in these specialized areas.

In developing a training dataset for the IHEP operations and maintenance, we compiled over 12,000 tickets and 25,000 question-and-answer pairs from the Helpdesk platform. The data cleaning phase aimed to automate user query responses, potentially replacing the need for operations and maintenance personnel. Initial steps involved removing tickets requiring on-site services and employing regular expressions to eliminate non-textual elements like HTML tags, CSS styles, and JavaScript scripts from the data. Additionally, we filtered out non-generalizable question-and-answer pairs using keyword matching. Due to the stringent permission protocols in stable computing environments, we excluded requests necessitating operations and maintenance staff intervention, such as password resets, server reboots, and disk recoveries. The final step involved anonymizing the dataset by erasing private contact information, leaving only operationally relevant emails and public contacts.

ChatGPT-3.5 was used to process public dialogues post-cleaning, and Vicuna-13B handled sensitive internal communications. The dataset corpus underwent further refinement and evaluation through Prompt Engineering, focusing on removing challenging unformatted text elements. The evaluation criteria were established as follows:

Table 1: Quality Assessment Criteria.

Score Range	Description
90-100	Exceptional quality, clear, coherent, highly informative, and universally applicable.
70-89	Good quality, mostly clear and coherent, contains useful information and is fairly universal.
50-69	Average quality, some clarity and coherence issues; information is somewhat useful but may not be universally applicable.
Below 50	Poor quality, lacks clarity and coherence, information is not very useful or is too specific to be universally applicable.

We filtered out conversational data with scores above 75 through automated scoring, forming our Helpdesk QA dataset. This process improved the dataset's quality and ensured the effectiveness and specificity of model training.

3.2 Model

In our study, we utilized the Xiwu-13B training base model, developed from the Vicuna-13B model, an extension of the Llama model, refined with datasets tailored to human preferences. Figure 1 shows that the Llama model incorporates a Transformer-based, Decoder-only architecture. This design simplifies the architecture and concentrates on generation tasks, increasing training efficiency. The model employs RMSNorm for normalization, calculating a scaling factor via the root mean square method to equilibrate features of varying scales, thus enhancing the model's stability and generalization capabilities. This normalization technique significantly improves the model's training stability and convergence rate.

Furthermore, the model integrates Rotary Positional Encoding (RoPE) for the query (Q) and key (K) vectors within its self-attention mechanism, enabling precise positional information capture and augmenting the model's expressive capacity. Incorporating a causal mask ensures that text generation is sequentially dependent, adhering to the causality principle in language production.

PoS(ISGC2024)034

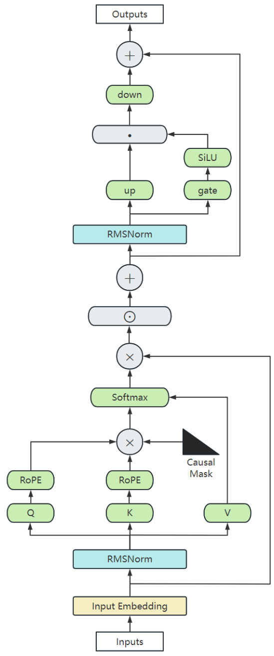

Figure 1: Architecture of Llama.

Llama2 introduces the Group query attention mechanism to optimize memory usage, markedly reducing computational and memory demands and boosting operational efficiency. After self-attention processing, a residual connection re-links the output to the input, which, after passing through a feedforward neural network (FFN) that includes two linear layers and an activation function (e.g., SiLU [24]), enhances non-linear feature detection. This process culminates in a Final RMSNorm normalization, producing a logit probability array for token prediction based on strategic selection.

The Llama model architecture excels in its streamlined, efficient design, focusing on generating tasks through a Decoder-only framework, simplifying training, and accelerating the process. Adopting RMSNorm and innovative positional encoding techniques bolsters the model stability and expressive capability, facilitating superior generalization and efficiency in large-scale dataset processing. Its meticulously designed attention mechanisms and feedforward networks adeptly capture and articulate complex language features, rendering it highly effective in natural language processing tasks.

3.3 Retrieval-Augmented Generation

Figure2 shows that we designed an efficient processing pipeline based on Retrieval-Augmented Generation. The design of this pipeline aims to optimize the information retrieval and generation process through advanced technical means, thereby improving the system's overall performance and response quality.

During the initial phase of the process, we utilize an Unstructured Loader to preprocess the collected documents in various formats and convert them into a text format. This step is crucial for the streamlined processing of data and subsequent operations. Afterward, sliding window

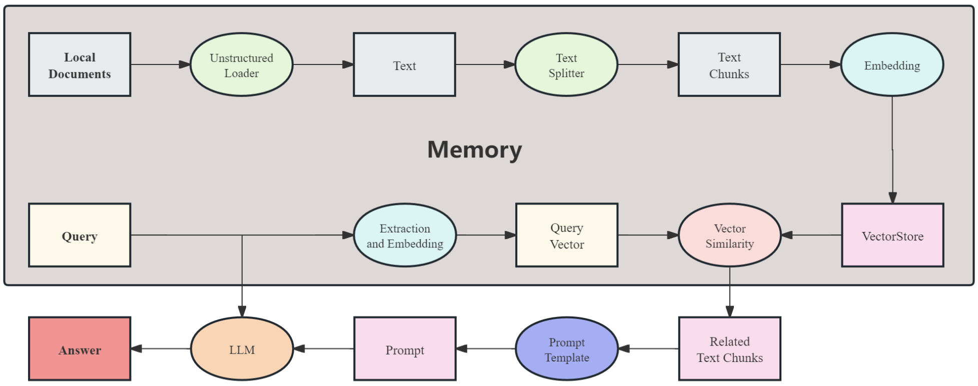

Figure 2: Retrieval-Augmented Generation pipeline.

technology is employed to slice the text for more efficient processing and analysis of the text data. It ensures the information is intact and the system's data processing capabilities are enhanced.

The next step involves using a vector embedding model to transform the slices above text into vector form and store them in a vector database. It is an essential step in enabling efficient information retrieval. Vectorizing text can significantly accelerate the similarity query process. Upon receiving a query request from the user, our system employs information extraction techniques to extract critical information, such as search keywords or paragraphs from the user's question. This information is then converted into a vector $\boldsymbol{v}$ using vector embedding technology. The system then conducts a similarity search for this vector in the vector database to locate document content most likely to provide helpful answers to the user's question.

Once potentially relevant document content has been retrieved, the system combines it with a pre-prepared template to create a complete prompt. This step further enhances the accuracy and relevance of the answer by structuring the information prompts.

Finally, the user's original query is merged with the generated prompt containing potential answer information and submitted to an LLM for processing. It enables the system to generate more precise and comprehensive answers that cater to the user's query needs.

In summary, the processing pipeline designed in this study, based on Retrieval-Augmented Generation, significantly improves the efficiency and quality of information retrieval and question answering through refined data processing, intelligent information retrieval, and efficient question-answering generation mechanisms, providing users with more accurate and rapid information services.

4. Experiment

4.1 Dataset

We collected over 12,000 tickets and 25,000 question-and-answer pairs from the Helpdesk platform. By utilizing regularization techniques, we removed parts of the text not in natural language and content that involved privacy. At the same time, we also excluded tickets that required on-site services or administrator intervention through classification and keyword exclusion. Finally, we designed a set of scoring criteria that allowed large models to score the quality of the question-and-answer pairs, thereby filtering out 3,000 high-quality and universally relevant multi-turn dialogue question-and-answer pairs for the Helpdesk QA dataset.

4.2 Experiment Setup

Table 2: Experimental Environment.

Component	Configuration
CPU	Intel(R) Xeon(R) Gold 6330 CPU @ 2.00GHz *2
Memory	500G
GPU	NVIDIA Tesla A800 (80G) *4
OS	AlmaLinux 9.2

Our experimental setup is detailed in Table 2, where we utilized the open-source Xiwu and Fastchat projects as the frameworks for fine-tuning and inference within our expansive model. The Torchrun utility, which facilitates single-node multi-GPU parallelism within PyTorch, enabled us to conduct model training across four A800 GPUs. Additionally, we employed CPU Offload technology during training to mitigate the issue of excessive GPU memory consumption. Transferring specific computations to the CPU allowed for the liberation of GPU memory, thereby enhancing memory utilization efficiency. Xiwu-13B served as the foundation for our experiments, performed on the Helpdesk QA dataset over three epochs, with the training parameters delineated in Table 3. The parameters were carefully selected and tuned to optimize the training process and improve the performance of the OMAI model.

Table 3: Main Training Parameters.

Parameter	Value
Epochs	3
Per Device Training Batch Size	4
Per Device Evaluation Batch Size	32
Evaluation Steps	1500
Save Steps	1500
Learning Rate	2e-5
Warmup Ratio	0.04
FSDP	full_shard auto_wrap offload
FSDP Transformer Layer Class to Wrap	LlamaDecoderLayer
Model Maximum Length	4096

4.3 Construction of the RAG

In this study, we processed and integrated various document resources into a unified format. Initially, we transformed a wide array of document contents, including documents and handbooks, into

JSON format through a meticulously designed conversion process. Subsequently, we transformed data and uploaded it to a vector database within our bespoke RAG Pipeline.

Upon receiving a query request, our system activates a specialized worker unit, "hepai/hai-vector," an intelligent module engineered to process and interpret user queries efficiently. This module excels in identifying keywords within the query and converting them into the vector by embedding, a pivotal step for precise document retrieval via similarity matching in the vector space. The system then integrates the retrieved document content into a pre-designed template prompt, enhancing the response's coherence and naturalness while directing the content's style and focus.

The response, refined by the template prompt, is then processed by our inference model on the OMAI platform for final analysis, ensuring the delivery of a tailored answer that fulfills the user's query.

4.4 Experimental Results

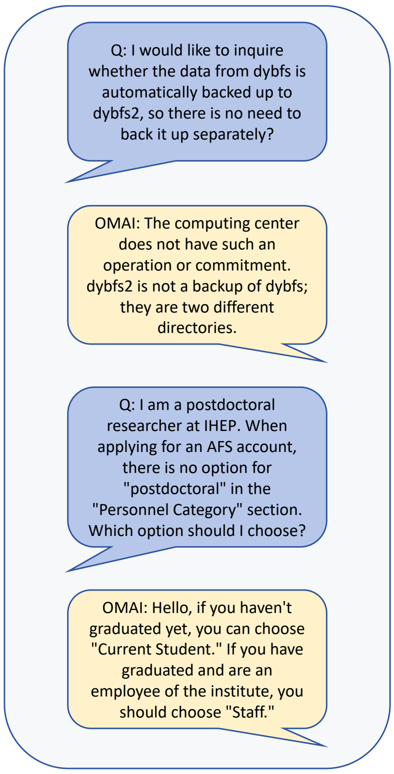

Figure 3: Answer by OMAI

Using manual methods, we constructed a dataset of the 50 most frequently asked internal Helpdesk inquiries at IHEP during the testing phase. The OMAI model can directly address these questions through fine-tuning and leveraging the knowledge embedded in its internal weights. Comparing OMAI with ChatGPT, we found that OMAI's response quality is significantly superior to ChatGPT. For the 50 most common questions, OMAI achieved an effective response rate of over 90%, as shown in Figure3. This data can effectively reduce the workload of operations personnel, allowing them more time and energy to handle on-site requests and higher-level faults.

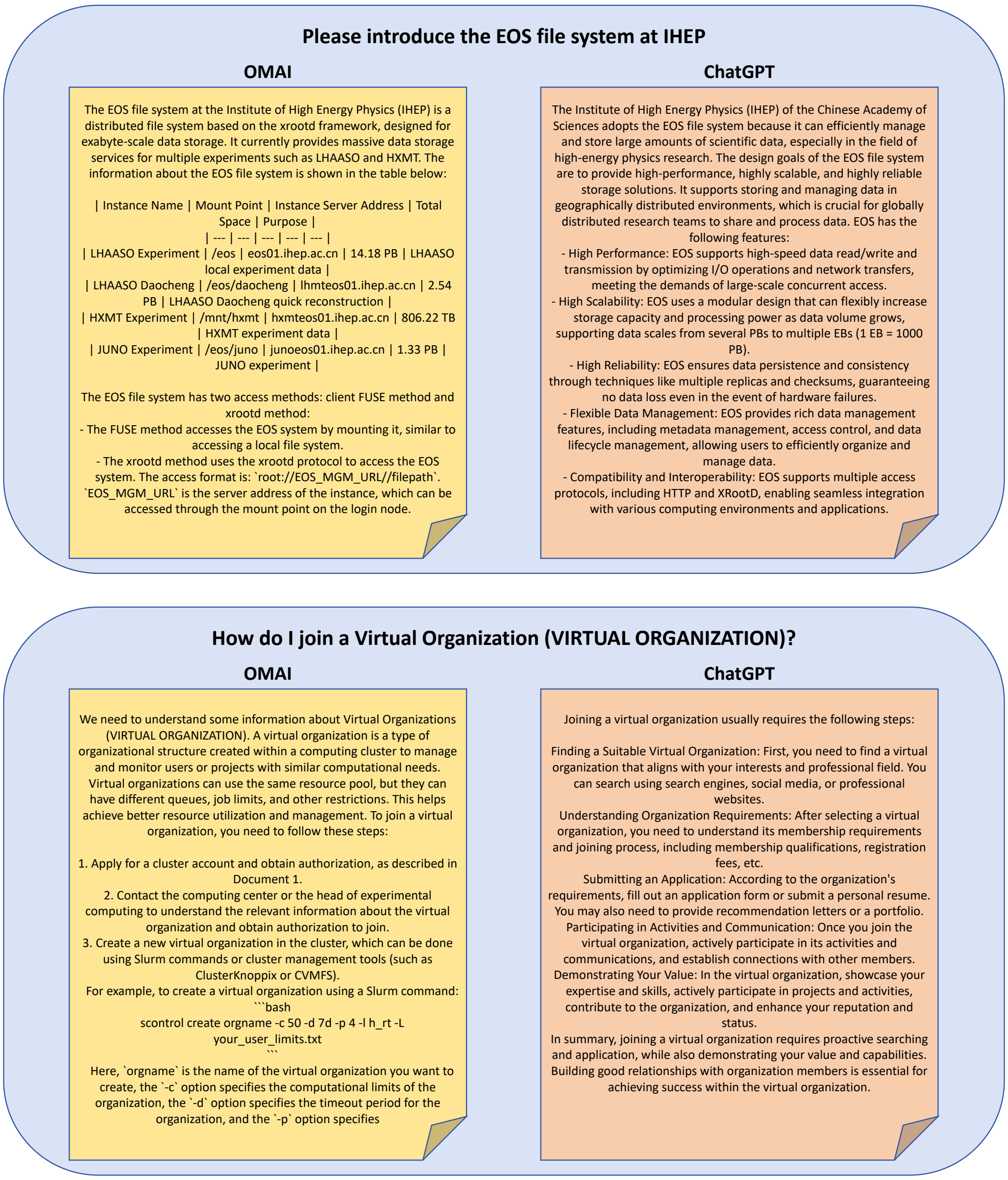

Figure 4: ChatGPT and OMAI answer comparison.

In contrast, when consulting ChatGPT with the same questions, it either refuses to respond or generates ineffective answers. That is because most of our questions come from the daily operation and maintenance of IHEP's internal computer systems. ChatGPT has yet to learn from our business and lacks the operational knowledge of IHEP's computer systems, making it unable to understand the users' needs and provide accurate and practical answers.

Additionally, we designed thirty possible common question queries based on the content of documents and handbooks, as shown in Figure4. Leveraging the knowledge of the vector database, OMAI can effectively respond to user questions, guide users through logging in and other routine operations on the computing cluster, and address users' basic knowledge queries about the computing cluster. When the same questions are asked to ChatGPT, although it can also provide correct answers, it fails to address the users' actual needs and only offers general suggestions. Furthermore, OMAI is often more closely aligned with the user's usage scenarios than ChatGPT, and ChatGPT's understanding of specific terms may be based on something other than professional domains, leading to potentially incorrect responses.

5. Conclusion

In conclusion, the OMAI model, specifically designed for the IHEP operation and maintenance sector, has exhibited outstanding performance after fine-tuning and adopting the Retrieval-Augmented Generation. This model efficiently resolves user queries by integrating a vector database and RAG technology and facilitates an intuitive interface for novices and professionals. This interface allows for effective navigation during the search and comprehension of documents and handbooks, ensuring users receive precise and straightforward answers. Such advancements have significantly reduced the time investment required across all user groups. Moreover, the innovative retrieval process introduced by RAG grants access to related texts across a broad spectrum of documents, overcoming the limitations of manual search efforts that often need to be revised due to the sheer volume and diversity of available sources. This capability substantially enhances the practical utility of the OMAI model.

6. Future Work

Even though OMAI currently demonstrates exceptional performance in automatically responding to users' daily queries and has alleviated the burden on operational staff by querying a variety of documents and manuals in the vector database using RAG technology, there is still room for improvement in automatically completing specific user requests (such as account unlocking, password resetting) and in handling user on-site service tickets. In the future, we plan to develop a series of new features, including but not limited to:

- **Request Automation Processing:** Design and implement a mechanism that allows OMAI to automatically forward user requests that require administrator privileges to administrators for approval. This process will use advanced permission verification and request distribution algorithms to ensure security and efficiency.
- **Automatic Correction and Completion of User Queries:** Implement a feature that enhances OMAI's ability to understand and correct common misconceptions in user queries, such as confusing disk space with memory space. It will involve advanced natural language understanding algorithms capable of contextually interpreting and auto-correcting user statements to ensure accurate issue resolution and information provision.
- **Intelligent Handling of On-site Service Requests:** Develop a module that enables OMAI to automatically ask for and collect possible fault condition information when it recognizes a user's request for on-site service. This mechanism will be based on natural language processing technology to optimize the preparation work of operational staff.

PoS(ISGC2024)034

- **System Monitoring and Automated Maintenance:** By fine-tuning the operating system and file system log data, enabling OMAI can monitor the machine operation status, perform anomaly detection, and automatically provide intelligent alerts and solutions to users and administrators, and even directly repair faults.

We will adopt a series of evaluation methods, including user satisfaction surveys and operational efficiency indicators, to measure the effectiveness of these new features. At the same time, we foresee technical and security challenges in implementing these features and plan to address these issues through continuous technological innovation and security assessments. Ultimately, these efforts will help further enhance OMAI's performance, bringing greater convenience and efficiency to users and operational staff.

7. Acknowledgments

This work was supported by the National Natural Science Foundation of China (No. 12075268), the Informatization Plan of the Chinese Academy of Science (No. CAS-WX2022SF-0104), and the "From 0 to 1" Original Innovation Project of IHEP (No. E3545PU2) and Youth Innovation Promotion Association CAS (No. 292021000091).

References

[1] Shi Jingyan, Huang Qiulan, Wang Lu, LI Haibo, DU Ran, Jiang Xiaowei, HU Qingbao, Zheng Wei, Yan Xiaofei, and Zhang Xuantong. Distributed data processing platform of national high energy physics data center. *Frontiers of Data and Computing*, 4(1):97, 2022.

[2] Xuantong Zhang, Xiaomei Zhang, Haibo Li, Yujiang Bi, Hao Hu, and Haofan Wan. Distributed Data Management System at IHEP. *PoS*, ISGCHEPiX2023:027, 10 2023.

[3] OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, and Florencia Leoni Aleman. Gpt-4 technical report, 2024.

[4] Yu Sun, Shuohuan Wang, Shikun Feng, Siyu Ding, Chao Pang, Junyuan Shang, Jiaxiang Liu, Xuyi Chen, Yanbin Zhao, Yuxiang Lu, Weixin Liu, Zhihua Wu, Weibao Gong, Jianzhong Liang, Zhizhou Shang, Peng Sun, Wei Liu, Xuan Ouyang, Dianhai Yu, Hao Tian, Hua Wu, and Haifeng Wang. Ernie 3.0: Large-scale knowledge enhanced pre-training for language understanding and generation, 2021.

[5] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models, 2023.

[6] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. A survey of large language models, 2023.

[7] Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. Retrieval-augmented generation for large language models: A survey, 2024.

[8] Jasper Xian, Tommaso Teofili, Ronak Pradeep, and Jimmy Lin. Vector search with openai embeddings: Lucene is all you need. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, WSDM '24, page 1090–1093, New York, NY, USA, 2024. Association for Computing Machinery.

[9] Zhengde Zhang, Yiyu Zhang, Haodong Yao, Jianwen Luo, Rui Zhao, Bo Huang, Jiameng Zhao, Yipu Liao, Ke Li, Lina Zhao, et al. Xiwu: A basis flexible and learnable llm for high energy physics. *arXiv preprint arXiv:2404.08001*, 2024.

[10] Alan M. Turing. Computing machinery and intelligence. *Mind*, 59(October):433–60, 1950.

[11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.

[12] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training, 2018.

[13] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67, 2020.

[14] Yuan-Jie Yao, Yi-Guang Gong, Jia Liu, Chuang Xu, and Dong-Liang Zhu. Survey on intelligent question answering system based on deep learning. *Computer Systems Applications*, 32(04):1–15, 2023.

[15] Luciano Floridi and Massimo Chiriatti. Gpt-3: Its nature, scope, limits, and consequences. *Minds and Machines*, 30:681–694, 2020.

[16] Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*, 2023.

[17] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.

[18] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models, 2023.

[19] Ruibin Xiong, Yunchang Yang, Di He, Kai Zheng, Shuxin Zheng, Chen Xing, Huishuai Zhang, Yanyan Lan, Liwei Wang, and Tie-Yan Liu. On layer normalization in the transformer architecture, 2020.

[20] Noam Shazeer. Glu variants improve transformer, 2020.

[21] Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding, 2023.

[22] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, et al. Llama 2: Open foundation and fine-tuned chat models, 2023.

[23] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena, 2023.

[24] Stefan Elfwing, Eiji Uchibe, and Kenji Doya. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural networks*, 107:3–11, 2018.

PoS(ISGC2024)034