LLM-Based Design of Complex Industrial Systems: A Case Study of Heating System Design

CAI Xin[1], *LYU Hongqiang*[2], *XU Ran*[1], *WANG Bo*[3], *WANG Qi*[3], *WANG Heyun*[3], *LIU Xuejun*[1*]

1. College of Artificial Intelligence, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, P. R. China;
2. College of Aerospace Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, P. R. China;
3. Qingdao Kaineng Environmental Protection Technology Co., Ltd., Qingdao 266300, P. R. China

(Received 12 September 2025; revised 19 December 2025; accepted 20 March 2026)

Abstract: Modular design of complex engineering systems is a universal technology for rapid system design in the modern industrial field. Generating assembly schemes for modular engineering systems in industrial production is the core of achieving automation in system design. Traditional system design methods based on human experience suffer from low efficiency and poor adaptability. To enable the automatic assembly of modules for engineering systems, the performance matching and correlation between modules need to be accurately identified. Currently, general large language models, with their strong language analysis capabilities, have been applied to multiple industries. However, limited by data barriers in specific industrial field, their application in assembly of system modules is rare. Therefore, constructing high-quality domain data, fine-tuning professional large language models, and developing domain frameworks for automatic design of modular system constitute an important research direction to be explored at present. Based on the Qwen2.5 open-source large model, we utilize industrial system knowledge to construct a component library and propose a large language model for design of complex industrial system (DCI-LLM). Through the proposed low-rank adaptation (LoRA)-freeze-based "local-collaborative" fine-tuning method which combines local module and global knowledge of the system, DCI-LLM automatically generates system composition schemes and produces the related engineering drawings given system design requirements. We use the scheme design of heating systems as an example to verify the effectiveness of the proposed framework. Experimental results show that the fine-tuned DCI-LLM model achieves accuracy rates of 93.4% and 89.3% in answering questions about module knowledge and global knowledge, respectively. Moreover, scores from professional engineers indicate that DCI-LLM has practical application potential in scheme design of complex modular systems. Our work demonstrates that LLMs have significant application prospects in the field of automatic scheme design for complex industrial systems.

Key words: large language model (LLM); low-rank adaptation (LoRA) fine-tuning; freeze fine-tuning; complex system design; system engineering

CLC number: TP391.413 **Document code:** A **Article ID:** 1005-1120(2026)02-0251-24

0 Introduction

Modular design is a universal technology for rapid system design and has been widely applied in the design of complex engineering systems in the modern industrial field. In the process of scheme design and assembly of modular systems, the current design workflow faces many challenges. On one hand, engineering system design usually requires combining numerous modules within the system into a complex whole, and manually designing the assembly schemes of these modules is not only time-consuming but also error-prone. On the other hand, with the increasing complexity of products, it takes

*Corresponding author, E-mail address: xuejun.liu@nuaa.edu.cn.

How to cite this article: CAI Xin, LYU Hongqiang, XU Ran, et al. LLM-based design of complex industrial systems: A case study of heating system design[J]. Transactions of Nanjing University of Aeronautics and Astronautics, 2026, 43(2): 251-274.

http://dx.doi.org/10.16356/j.1005-1120.2026.02.007

a great deal of time for manual design in engineering drawing. As early as the 1990s, "expert systems" were put forward to perform automatic design in industrial engineering. Although the application of "expert systems" has gradually decreased since then, the role of other artificial intelligence algorithms in various engineering design tasks has continued to develop. In recent years, with the rise of large language models, advanced artificial intelligence (AI) technologies have been increasingly applied in industrial engineering design.

Large language models (LLMs) are advanced AI systems built based on the deep learning Transformer architecture, which acquire language rules through training on massive text data. With their ultra-large-scale parameters and complex neural network structures, they possess strong natural language understanding and generation capabilities, enabling them to accomplish complex language tasks such as multi-turn conversational interaction, logical reasoning, and generative writing. In practical applications, LLMs are widely used in fields such as intelligent customer service, intelligent writing, and knowledge question-answering (QA). Representative models such as OpenAI's GPT series[1-3] and Google's Bard[4-5] have revolutionized teaching models in the education field and optimized marketing strategies in the commercial field, demonstrating great potential to reshape industrial ecosystems. Therefore, how to apply large language models in engineering system design to fulfil knowledge QA of system design, and improve design efficiency and module assembly accuracy, has become an important research direction to be explored.

Although large language models have demonstrated strong natural language processing capabilities and changed the vertical application paradigms in multiple fields, general large language models such as SMoE[6] and ChatGLM[7] have not yet been widely applied TO the industrial field. In addition, the scarcity of domain data and the difficulty in obtaining high-quality annotated data in professional fields[8-9], coupled with the fact that models rely on general corpus for training, lead to deviations in the understanding of professional terminologies. This results in problems of inaccuracy and lack of credibility in the application of large language models in industrial sub-fields. To fully utilize the powerful reasoning ability of LLMs, it is necessary to conduct training of customized knowledge to achieve their professional performance in specific domains[10].

At present, some work has constructed customized datasets in industrial sub-fields and fine-tuned LLMs to fulfill the automatic design of some industrial systems. For example, Zhu et al.[11] systematically elaborated on the method of using large language models for creative conceptual design. Edwards et al.[12] developed the modular program Sketch-2-Prototype, which uses generative artificial intelligence to quickly and parallelly create design prototypes. Wang et al.[13] developed an intelligent maintenance assistant for aircraft maintenance technicians to help them better carry out aircraft inspection and maintenance. Wang et al.[14] used large language models to realize interaction with other data-driven processes in larger-scale intelligent autonomous systems. Lin et al.[15] proposed PE-GPT, which integrates multiple engineering application models and designs software through intelligent agents to optimize electronic design. Although these applications cover multiple aspects in the industrial production process, they either focus on parameter optimization of a single aspect or are confined to the partial stages of the design process. The modeling of deep coupling relationships between system modules is still lack of in-depth investigation.

We intend to apply large language models to the production and manufacturing process of the downstream industrial field, especially in the scheme design of the modular engineering systems. The knowledge of complex systems consists of global knowledge and local module knowledge. Global knowledge is systematic and it centers on the overall function of the system, integrating various modules, collaborative relationships, and operational logic. It includes the core functions of each module to explain their roles in the system. In contrast, the local module knowledge focuses on the attributes and operational details of individual modules and does not involve inter-module connections. Usually,

the global knowledge is composed of partial local module knowledge and system composition knowledge. We believe the knowledge characteristics of complex engineering systems can be discovered by exploring the deep coupling relationships between complex system modules. With the obtained module interaction knowledge, we can therefore perform the automatic design of complex systems and the automation of producing engineering drawings of the designed system. By utilizing the ability of large language models to learn the implicit coupling relationships between modules in existing data, and integrating this knowledge through the fine-tuning technology of pre-trained large language models, we hope to improve the ability of large language models to design complex systems in the industrial field, provide better system design suggestions for engineers, and ultimately optimize enterprise downstream production processes.

In this paper, we propose a large model framework for scheme design of complex modular industrial systems, i.e. a large language model for design of complex industrial system (DCI-LLM), based on Qwen2.5[16]. We first construct a module dialogue database of complex system knowledge, which contains a large amount of internal module information and global system knowledge, including design parameters, working conditions, etc. The creation of this database aims to provide large language models with subdivided professional engineering knowledge, enabling them to generate more accurate and professional responses to questions of system design. Using the module knowledge and global knowledge in the complex system data, we design a "local-coordination" strategy for fine-tuning of DCI-LLM to assist engineering designers in the scheme design of complex modular industrial systems. Local fine-tuning focuses on the learning of module working condition knowledge, while coordinated fine-tuning helps to mine coupling relationships between modules with the guidance of global knowledge. Finally, the automation of complex system engineering drawings is fulfilled through post-processing program integration, assisting engineers in efficiently completing system design. We use the heating system knowledge database of a Chinese company to construct a dialogue database, so as to verify the effectiveness of the proposed DCI-LLM framework in the design of modular heating system. The main contributions of this paper are summarized as follows:

(1) We propose a domain-specific large language model framework, i.e., DCI-LLM, for the automated design of modular engineering systems. This framework achieves, for the first time, an end-to-end automated design process from natural language requirements to complete engineering drawings. Compared with existing research, DCI-LLM breaks through the limitation that traditional large language models only focus on text interaction or intermediate code generation. By integrating PyCAD script generation capabilities, it builds a complete automated process of "requirements—scheme—drawings".

(2) We propose a data-feature-driven "local-coordination" fine-tuning strategy, and provide a possible way to apply large language models in specific industrial domain with modular data features through the collaborative mechanism of in-depth learning of module knowledge and mining of global coupling relationships.

(3) The feasibility and effectiveness of the proposed model are verified through engineering instances of the design of heating systems, and the experimental results show that the proposed framework has practical application potential for automatic engineering design in industrial sub-fields.

(4) The constructed large model framework for scheme design of complex modular industrial systems provide methodological reference for other application scenarios of large language models in the industrial field in terms of technical path and implementation strategies.

1 Related Work

1.1 Domain-specificlarge language models

Large language models have been applied to various fields such as finance and healthcare. In the financial field, LLMs can parse market financial reports, financial news, and user inquiries to generate personalized investment advice. For instance, Mor-

gan Stanley has deployed GPT-4[17] to assist analysts in quickly extracting key data from financial reports. The AI customer service of China Merchants Bank can accurately understand customers' financial needs through multi-turn conversations, and Ploutos[18] is used for interpretable stock trend prediction. It generates accurate and interpretable reasons for stock predictions by integrating multi-modal data. In the medical field, Singhal et al.[19] have explored how to use LLMs to implement medical QA systems that are close to the level of medical experts. Under the "Zaoji" fine-tuning framework, Sun et al.[20] obtained MedGLM.H, a domain-specific large model dedicated to answering common medical questions in hepatobiliary departments by fine-tuning ChatGLM, which has shown excellent performance on question banks of multiple medical exams. DoctorGLM[21] has built a Chinese medical dialogue database and adopted various technologies to train LLMs. In the legal field, Chatlaw[22] uses a mixture of experts (MoE) model and a multi-agent system to enhance the reliability and accuracy of AI-based legal services. DISC-LawLLM[23] adopts a legal syllogism prompting strategy, constructs a supervised fine-tuning dataset in China's judicial field, establishes a DISC-LawLLM with legal reasoning capabilities, and enhances LLMs through a retrieval module to strengthen the model's ability to acquire and utilize external legal knowledge. Intern-LM-Law[24] is specially designed to solve various legal issues related to Chinese law, capable of answering standard legal questions and analyzing complex real-world legal situations.

The application of large language models in the industrial field is permeating from auxiliary decision-making to core production processes, optimizing manufacturing processes through natural language processing, multi-modal fusion, and knowledge reasoning. Pang et al.[25] proposed a novel framework of LLMs-guided deep reinforcement learning (LG-DRL) to solve decision-making problems in autonomous vehicles. Lin et al.[15] proposed PE-GPT, a multi-modal large model customized for power electronic design, which introduced a hybrid framework of meta-heuristic algorithms, model libraries, and simulation resource libraries into LLM agents. Zhu et al.[11] proposed to apply large language models in creative conceptual design tasks. They conducted creative generation experiments using GPT-2 and GPT-3 models for specific design problems respectively. Fine-tuning technology was used to conduct customized training on GPT-2 and few-shot learning was introduced for GPT-3 to optimize the generation process. The experimental results demonstrate that few-shot trained GPT-3 shows a lower tendency to generate repeated concepts compared to GPT-2, but its controllability is lower than that of the fine-tuned GPT-2. Edwards[12] proposed Sketch-2-Prototype, which used generative artificial intelligence to quickly and parallelly create prototypes. Firstly, the visual LLM model receives sketch images and generates text descriptions. Then, the descriptions are input into a conditional diffusion model for image generation. Finally, the images are input into 3D modeling tools to generate 3D modeling images. Wang et al.[13] developed an intelligent maintenance assistant for aircraft maintenance technicians. This study adopts a fine-tuned GPT model, and its fine-tuning process integrates aircraft structure ontology and selected maintenance logs. This work demonstrated that training data helps inject domain-specific and hierarchically-structured knowledge into large language models, thereby reducing ambiguity caused by similar component names in different sub-components of the aircraft. In the context of the combination of LLMs and digital twins (DTs), Wang et al.[14] integrated language models with larger-scale intelligent autonomous systems. The proposed industrial-grade large model (Industrial-GPT) was built into the multi-level digital twin architecture of a zinc smelting intelligent factory. The system connects to the automation system through a digital twin interface and configures an LLM-based intelligent agent to describe the technical details in the digital twin.

These large language models focus on different segments in industrial scenarios. Although these applications cover multiple aspects such as industrial decision-making, design, operation and maintenance, and production, they either focus on parame-

ter optimization of a single aspect or are confined to partial segments of the design process, and have not explored deep coupling relationships such as performance matching and correlation of working condition between modules in complex industrial systems. To explore the deep coupling relationships between modules, it is necessary to design training strategies adapted to the characteristics of industrial systems, including constructing a professional dialogue database containing module association knowledge, developing a fine-tuning framework thatintegrates local module knowledge and global system constraints, etc. Based on these considerations, DCI-LLM captures the module composition rules in complex systems through a hybrid fine-tuning strategy, and utilizes engineering design tools to fulfill a full-process closed loop from "coupling relationship reasoning" to "automated generation of engineering drawings", so as to fill the application blanks of existing industrial large language models in the field of complex system design.

1.2 Parameter-efficient fine-tuning methods

The application of large language models needs to conduct model training with customized knowledge in specific domains to achieve professional performance. However, training domain-specific LLMs from scratch is an extremely challenging and costly task. It not only requires massive computational resources but also demands guidance and optimization from AI algorithm engineers with profound domain knowledge. For many institutions, especially small enterprises or research institutions, such costs and complexities are often unaffordable. Therefore, fine-tuning based on general LLMs to adapt to specific domain needs has become a more feasible and economical option.

In recent years, parameter-efficient fine-tuning (PEFT) techniques, such as low-rank adaptation (LoRA)[26], adapter tuning[27], and prompt tuning[28], have made breakthrough progress. These techniques have significantly reduced the computational resources and training cycles required for fine-tuning large language models, achieving much higher efficiency compared to traditional full-parameter fine-tuning. Aghajanyan et al.[29] proposed that pre-trained models had an extremely small intrinsic dimension, meaning there existed some very low-dimensional parameters whose fine-tuning could achieve the same effect as fine-tuning in the full parameter space. Inspired by work on intrinsic dimensions, Hu et al.[26] argued that there is an "intrinsic rank" in the parameter update process of LLMs, and lightweight fine-tuning of large language models could be achieved by fine-tuning this "intrinsic rank of large language models". Lester et al.[28] found that soft prompts, learned through backpropagation, could integrate signals from multiple labeled examples. Unlike GPT-3's discrete text prompts, this end-to-end learning method far surpasses GPT-3's few-shot learning. Moreover, when the model scale reaches billions of parameters, it can narrow the gap with fine-tuning methods that adjust all model weights. Hu et al.[27] proposed the LLM Adapters framework, which integrated multiple adapters and open-source LLMs, and could be used for PEFT in different tasks. Empirical studies have explored the impact of adapter-related factors on method design, and experimental results show that PEFT using this framework on small-scale models (7 billion) can achieve performance equivalent to or even better than large-scale models (175 billion) in zero-shot reasoning for two types of reasoning tasks arithmetic reasoning and common-sense reasoning. When building domain-specific large language models, current strategy is to screen domain-related professional knowledge from public knowledge bases as training data to perform parameter-efficient fine-tuning on general large language models.

Existing studies have shown that fine-tuning highly specialized large language models based solely on public knowledge bases is difficult to achieve the desired performance improvement. This is mainly because public knowledge bases have insufficient coverage depth in professional domains, and there is a significant gap between the completeness and accuracy of their knowledge content and the professional standards of domain experts. Therefore, data processing and management during the fine-tuning and pre-training stages of large language models are ex-

tremely important[30-31]. Using professional documents and data to fine-tune large language models has gradually become a feasible path to improve the professional performance of models. However, due to the requirements of standardized operating procedures, professional documents not only contain a large number of professional terminologies but also have special format specifications, making them unable to be directly used for model fine-tuning. These documents need to undergo preprocessing such as data cleaning, format conversion, and annotation to meet the fine-tuning needs of large language models.

By fine-tuning, general models can be transformed into domain-specific LLMs, significantly improving their accuracy and credibility in the domain. This approach also saves time and resources while maintaining the flexibility to adapt to changes in domain requirements. In addition, using different fine-tuning strategies for different application scenarios to fine-tune large language models is also a key consideration in model fine-tuning. For example, during model training with knowledge of complex industrial systems in this paper, the LoRA fine-tuning is used to enable the model to learn local knowledge, and the Freeze fine-tuning is introduced to learn global knowledge while retaining the model's ability to discover local knowledge.

2 DCI-LLM

2.1 DCI-LLM framework

We propose a large model framework, DCI-LLM, for scheme design of complex modular industrial systems based on Qwen2.5. The framework of DCI-LLM consists of three stages: Data processing, hybrid fine-tuning, and automatic design of complex systems, as shown in Fig.1. Knowledge data of complex systems undergo data processing and extraction to generate the data for large model

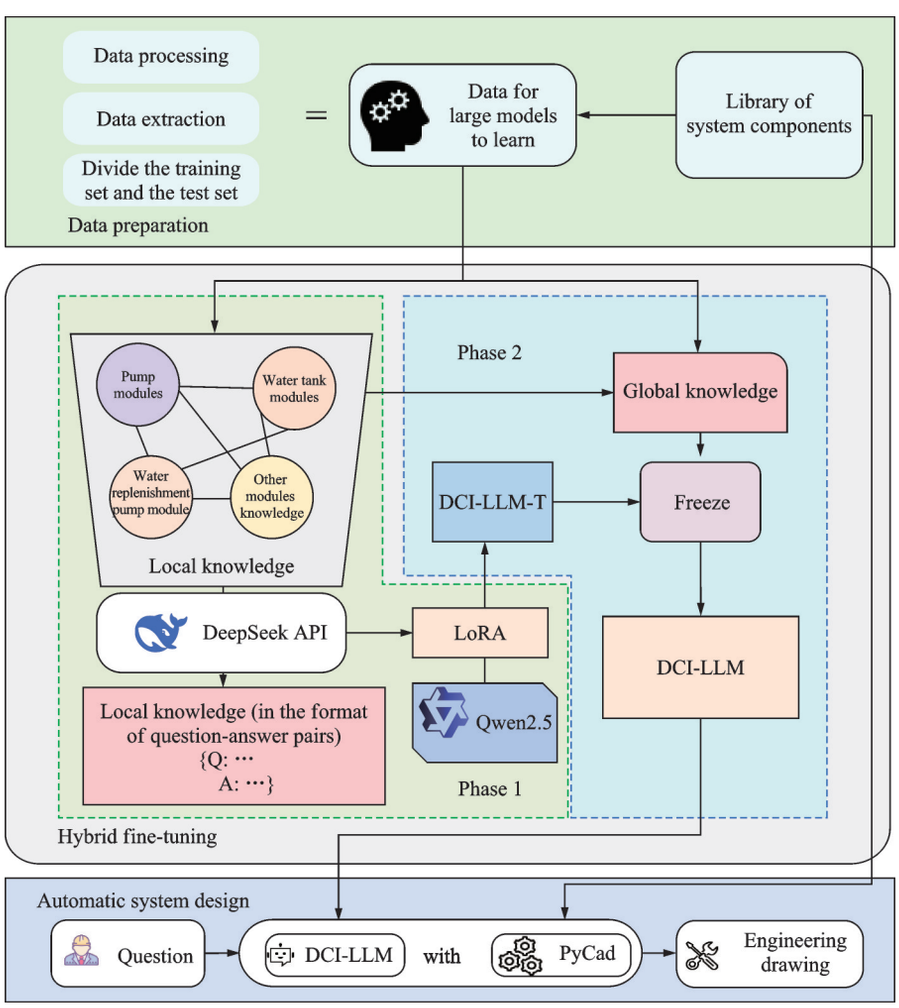

Fig.1 Framework of DCI-LLM

training and build the library of system components. The component library corresponding to module knowledge will be used in subsequent system scheme drawing. The detailed construction of the domain knowledge base will be explained in Section 3.2 in combination with specific application cases. The hybrid fine-tuning of DCI-LLM, namely the "local-collaborative" fine-tuning, includes internal module knowledge training (local training) and global system knowledge training (collaborative training). The "local-collaborative" fine-tuning of DCI-LLM corresponds to the process of mechanical engineers learning the composition principles of the entire system, that is, first learning local module knowledge and then progressing to system composition principles. We divide the model's fine-tuning process into two phases. In the first phase of fine-tuning, we use the detailed knowledge of internal modules and generate QA data to perform LoRA fine-tuning on Qwen2.5-7B. The model fine-tuned after the first phase can answer basic questions about the internal modules of the system, which is named as DCI-LLM-T, but it is prone to hallucinations when answering questions involving the global system. In the second phase of fine-tuning, we process the global knowledge data and perform Freeze fine-tuning on the DCI-LLM-T model. After the two-phase fine-tuning, the trained DCI-LLM can answer system-related questions while ensuring the ability to answer basic questions about internal modules of the system. In the final stage of automatic system design, DCI-LLM can generate the final engineering drawings according to the answers to the system questions asked by users. The final engineering drawings only need engineers to verify their completeness and correctnes.

2.2 Fine-tuning and inference process of DCI-LLM

The knowledge data of complex modular systems include global system knowledge and module knowledge, where global knowledge is composed of part of the local module knowledge and system composition knowledge. Global knowledge is a systematic knowledge that centers on the overall function of the system, integrating various modules, collaborative relationships, and operational logic. It contains the core functions of each module to explain their roles in the system. In contrast, local knowledge focuses on the attributes and operational details of individual modules and does not involve inter-module connections. We adopt Qwen2.5 as the backbone model in our work. Considering the combinatorial characteristics of modular knowledge data, the backbone model is fine-tuned through a "local-coordination" approach to achieve domain knowledge mining. By capturing the global combinatorial features and coupling relationships of the system, and in combination with post-processing procedures, engineering drawings can be automatically generated, thereby assisting engineers in producing system drawings. "Local fine-tuning" enables the model to learn module knowledge under specific working conditions, while "coordinated fine-tuning" leverages global knowledge to guide the model in mining the coupling relationships between internal modules of the system, thus discovering the underlying knowledge about the composition laws of the system under specific working conditions.

The fine-tuning and reasoning process of DCI-LLM is shown in Fig.2. Knowledge distillation is first carried out using the knowledge of internal modular components involved in complex systems. Professional documents are processed to meet the data and format requirements for fine-tuning. The source model is fine-tuned with a certain amount of training data. In the first stage, the model is fine-tuned using professional knowledge data of each module in the system, so that the model can learn specific knowledge of each module under various working conditions. The model is then fine-tuned using global system knowledge to minimize model hallucinations, enabling the model to learn the overall global knowledge of the system under various working conditions. Finally, a large model for assisting complex system engineering drawing is obtained. With such a model, engineers only need to put forward questions on system design, and the answers of specific module knowledge and global system knowledge under various working conditions are

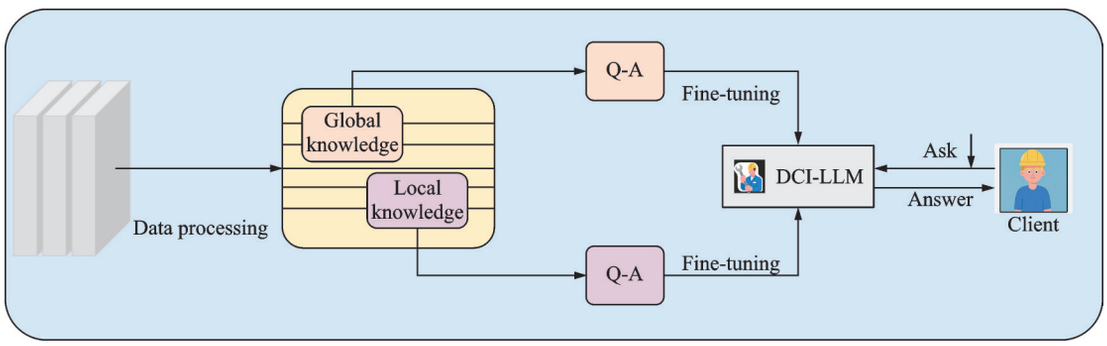

Fig.2 Fine-tuning and inference process of DCI-LLM

then generated, as well as the final system engineering diagram.

2.3 "Local-collaborative" fine-tuning

2.3.1 Fine-tuning based on local module knowledge

Through low-rank decomposition technology, only a small number of parameters need to be fine-tuned for LoRA, significantly reducing memory requirements. Full-parameter fine-tuning of a 7B model requires more than 80 GB of memory, while LoRA only needs 10—15 GB. This makes it possible to fine-tune large language models on consumer-grade graphics cards. The efficiency of this fine-tuning approach also enables rapid switching between multiple tasks. In addition, in small data scenarios (1×10^3—5×10^3 samples), LoRA performs particularly well, as it can converge quickly and retain the general capabilities of the pre-trained model, avoiding knowledge forgetting. Moreover, these characteristics make LoRA the preferred solution for adapting large language models to vertical fields. Therefore, we use LoRA for fine-tuning module knowledge in the first stage.

During training, LoRA freezes all parts of the pre-trained model and only trains the newly added low-rank matrix parameters (i.e., adapter parameters). Fig.3 shows the training logic of LoRA compared with the full-parameter fine-tuning strategy. In the scenario of "full-parameter fine-tuning" on the left side of Fig.3, the parameters are divided into two parts, shown as

$$\boldsymbol{W}\in\mathbf{R}^{d\times d},\Delta\boldsymbol{W}\in\mathbf{R}^{d\times d} \tag{1}$$

where $\boldsymbol{W}$ is the pre-trained weight, $\Delta\boldsymbol{W}$ the incremental weight, and d the hidden layer dimension of large models. Full-parameter fine-tuning involves updating the weights based on the original pre-trained weights, shown as

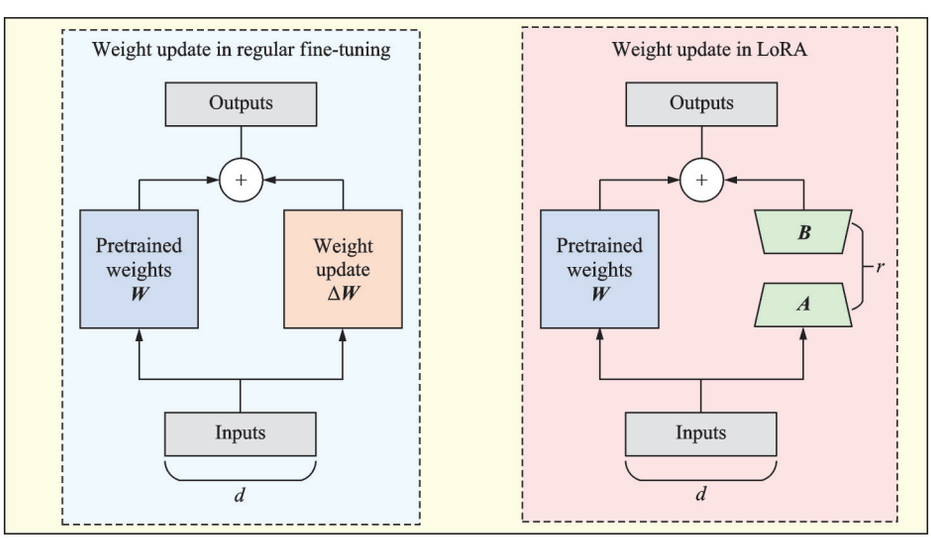

Fig.3 Full-parameter fine-tuning vs. LoRA fine-tuning

$$\text{Output} = (\boldsymbol{W} + \Delta\boldsymbol{W}) \times \text{Input} \quad (2)$$

The right side of Fig.3 represents the LoRA fine-tuning scenario. In LoRA, the incremental weights $\Delta\boldsymbol{W}$ in full-parameter fine-tuning are approximated by matrix $\boldsymbol{A}$ and matrix $\boldsymbol{B}$. The output of LoRA is expressed as

$$\text{Output} = \boldsymbol{W} \times \text{Input} + a/r \times \boldsymbol{BA} \times \text{Input} \quad (3)$$

where $\boldsymbol{A} \in \mathbf{R}^{r \times d}$, $\boldsymbol{B} \in \mathbf{R}^{d \times r}$, a is the scaling coefficient, r denotes the rank of the low-rank matrix, and the number of model fine-tuning parameters d^2 is reduced from d^2 to $2 \times r \times d$. We perform Gaussian initialization for matrix $\boldsymbol{A}$ and zero initialization for matrix $\boldsymbol{B}$. Zero initialization ensures that the model training starts from 0 without introducing noise, while Gaussian initialization provides directions for random exploration in the early stage of model training.

The rank represents the information content of a matrix. If a certain dimension in the matrix can always be linearly derived from the remaining dimensions, this dimension of information is redundant for the model. Theoretically, there is also redundant information in the incremental weights of full-parameter fine-tuning $\Delta\boldsymbol{W}$. Therefore, we can perform singular value decomposition on $\Delta\boldsymbol{W}$ to find the corresponding low-rank matrices $\boldsymbol{A}$ and $\boldsymbol{B}$. Since $\Delta\boldsymbol{W}$ is unknown, $\boldsymbol{W}$ and $\Delta\boldsymbol{W}$ represent old and new knowledge, it is impossible to perform singular value decomposition on $\Delta\boldsymbol{W}$. Thus, we consider taking r as a hyperparameter of the model and making the model learn matrices $\boldsymbol{A}$ and $\boldsymbol{B}$.

For the hyperparameter a, we initialize it to r that is set for the model the first time. Here, r is a relatively large value, which can cover new knowledge while making $\boldsymbol{BA}$ as close as possible to $\Delta\boldsymbol{W}$. Therefore, when r is smaller, the low-rank matrix represents new knowledge more concisely but may miss information. At this time, a/r is relatively large, amplifying the impact of new knowledge on the model during forward propagation. When r is larger, the information contained in the low-rank matrix is richer, and it is more similar to $\Delta\boldsymbol{W}$. In the training and updating (training dynamics) of the algorithm, gradient updates only optimize matrices $\boldsymbol{B}$ and $\boldsymbol{A}$. In inference deployment, $\boldsymbol{W}' = \boldsymbol{W}_0 + \boldsymbol{BA}$, where $\boldsymbol{W}_0$ and $\boldsymbol{W}'$ represent the original pre-trained weight matrix of the model and the fine-tuned weight matrix, respectively. The weights are merged to eliminate inference delays.

2.3.2 Fine-tuning based on the global knowledge

After the first fine-tuning stage, DCI-LLM-T has already mastered the knowledge of internal system modules relatively well. It can answer some questions related to module knowledge that engineers encounter during the system design and engineering drawing processes, and show the reasonable accuracy. To retain DCI-LLM-T's performance in the task of answering questions about internal system module knowledge and reduce the hallucinations of the large model when answering questions about global system knowledge, we choose to introduce global system knowledge in the second stage of fine-tuning, freeze the base layers of the model, and only allow the parameters of the last five layers to be updated. For the frozen parameters ω_i ($i \leqslant k$), the update is $\omega_i^{(t+1)} = \omega_i^t$. For the parameters involved in fine-tuning ω_j ($j \leqslant k$), the update is

$$\omega_j^{(t+1)} = \omega_j^t - \eta \frac{\partial L}{\partial \omega_j} \quad (4)$$

where t represents the number of iterations, η the learning rate, and L the loss function. They are updated according to the gradient descent method.

During the fine-tuning process, for each frozen parameter ω_i, its gradient descent is zero, i.e.

$$\frac{\partial L}{\partial \omega_j} = 0 \quad (5)$$

Therefore, these parameters will not be updated during backpropagation. For parameters that need to be updated, we calculate their gradients and update their values. The advantages of the Freeze fine-tuning method are mainly reflected in the following two aspects. Firstly, this method can effectively retain the powerful representation learning ability of the pre-trained model. Secondly, it only needs to fine-tune a small number of parameters to achieve adaptation to specific tasks, and this method shows excellent applicability in scenarios where training data is limited or computing resources are

constrained. The DCI-LLM model after Freeze fine-tuning not only maintains excellent performance in the task of answering questions about internal system module knowledge, but also significantly improves its ability to handle problems related to global knowledge. The DCI-LLM model simulates the process of mechanical engineers learning the composition principles of the entire system. That is, it first learns local module knowledge and then progresses to system composition principles, by constructing a domain knowledge base and adopting the "local-collaborative" hybrid fine-tuning technology, which includes internal module knowledge training and global system knowledge training. Specifically, in the first stage of fine-tuning, LoRA fine-tuning is performed on Qwen2.5-7B to answer basic questions about internal system modules. In the second stage of Freeze fine-tuning, the model learns global system knowledge data, enabling it to answer system-related questions. The data characteristic of this study—that global knowledge incorporates partial local knowledge—also explains why LoRA-Freeze requires a smaller volume of data to achieve the same accuracy. This is because the model has already mastered module knowledge during phased fine-tuning, and thus expends less effort in learning the module knowledge embedded within global knowledge. The verification of the fine-tuning methods is presented in Section 4.

Therefore, this framework can provide the final scheme of system design based on the answers to the questions asked by users, and automatically generate the final system engineering drawings by calling drawing tools, significantly improving the design efficiency of traditional manual operations.

3 Illustration of Heating System

To verify the effectiveness of the proposed DCI-LLM framework, we apply DCI-LLM to the system design of heating systems produced from a Chinese company. In this section, we will first introduce the background knowledge of heating system. The construction of knowledge base for heating system is then explained. Finally, the evaluation metrics of the proposed method are presented.

3.1 Heating system

The overall function of the heating system is to achieve efficient generation, transmission, and distribution of heat, as well as the control of system pressure and temperature, through the collaborative operation of multiple modules, thereby providing users with a stable supply of heat energy. A typical heating system basically contains six modules. Among them, the boiler (BOI) module, as the core heat source, is responsible for transferring the heat generated by fuel combustion to the circulating water to produce high-temperature hot water. The water pump (WP) module drives the water to circulate in the system through mechanical power, ensuring that heat can be continuously delivered to various heat-using terminals. The plate heat exchanger (PHE) module is used to perform heat exchange between different loops. The elevated water tank (EWT) module mainly plays the role of pressure stabilization and water replenishment, providing a stable pressure reference for the system through the static pressure formed by its own height, and replenishing water when the system water decreases due to evaporation or leakage. The pressure stabilizing expansion tank (PSET) module is used to absorb the expanded volume of water in the system after heating, preventing damage to pipelines or equipment due to excessive pressure, and at the same time assisting in maintaining stable system pressure. The pressure stabilizing and water replenishing pump (PSWR) module automatically starts working when the system pressure is lower than the set value, replenishing water into the system to maintain normal operating pressure. These six modules are connected through pipelines and control devices, and operate in coordination based on the connection relationships shown in Table 1. The value "1" in Table 1 indicates "direct connection" and "0" the "not direct connection". Fig.4 shows the real devices of modules BOI and WP. According to the system requirements that each module needs to be adapted to other related modules, every module has a specific number of configuration choices. The functions and the system configuration options of each module are shown in Table 2.

Table 1 Connection between modules of the heating system

Module	BOI	WP	PHE	EWT	PSET	PSWR
BOI		1	0	0	0	0
WP	1		1	1	0	1
PHE	0	1		0	0	0
EWT	0	1	0		0	1
PSET	0	0	0	0		1
PSWR	0	1	0	1	1	

(a) BOI (b) WP

Fig.4 True devices and composition of BOI and WP modules in the heating system

Table 2 Description of functions and number of configurations for each module in the heating system

Module name	Function	Number of module configuration
BOI	Transfer the heat generated by combustion to the circulating water to produce high-temperature hot water	18
WP	Drive water to circulate in the system and continuously transfer heat to various heat-using terminals	12
PHE	Fulfill heat exchange between different loops	8
EWT	Perform the functions of pressure stabilization and water replenishment	6
PSET	Absorb the expanded volume of water in the system after being heated, while maintaining stable system pressure	10
PSWR	Replenish water into the system to maintain normal operating pressure	10

In the design of a heating system, given the design requirements engineering designers first determine the parameter configuration of the boiler based on the input design parameters of heat load and temperature difference. They select the corresponding boiler and its connectors to form the BOI module, and then determine the matching parameters of the connected water pump module according to the parameter requirements of this module. Next, they need to select the corresponding WP module accordingly, and so on, to determine the configuration parameters of each module and their composition. Finally, based on these module selections, the engineering drawings of the system are manually produced and the engineering quotation information is issued. The entire process relies heavily on the human experience of engineering designers, and requires manually producing engineering drawings and quotations, which is highly complex. Consequently, the traditional heating system design has a long cycle, and involves a lot of repetitive work. Therefore, we apply the proposed framework for complex system scheme design, DCI-LLM, to automate the design of heating systems. By learning system design knowledge from existing commercial schemes, it can automatically complete the scheme design of the heating system and producing engineering drawings. Engineering designers only need to manually review the designed scheme for the verification of completeness and correctness. This framework is expected to greatly improve the efficiency of heating system design and optimize the production process of the enterprise.

3.2 Construction of heating system knowledge base

The construction of the design knowledge base for heating system mainly comes from the commercial heating system schemes of a Chinese company, and the data processing flow is shown in Fig.5. These datasets include the internal module design parameters, working conditions, and global knowledge of the heating system, where the global knowledge consists of partial module knowledge and macroscopic knowledge of the entire system. The API interface of Deepseek[32] is used to build QA data according to the scheme knowledge, and the training data is in the form of "question-answer". A total of 1 770 QA data related to heating system schemes

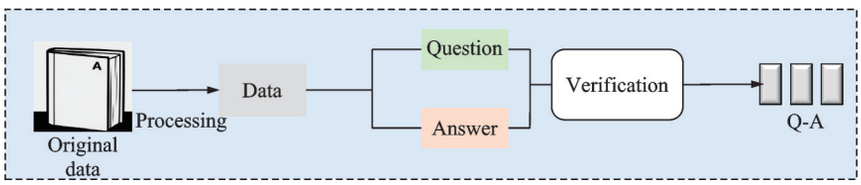

Fig.5 Data processing workflow for heating system case studies

has been sorted out. Among them, there are about 1 070 QA data related to heating system modules, and about 700 QA data related to global knowledge of heating systems. For specific module details, such as the inner diameter of the water pump and the position of the water tank, these overly precise data have very low value for DCI-LLM learning. Therefore, when processing scheme data, it is necessary to eliminate these overly detailed but not very important data. It should be noted that due to the limited scheme data at present, in order to make DCI-LLM's answers more professional and accurate, more data are still needed.

Fig.6 shows the process of extracting and converting complex industrial system knowledge data into training knowledge for large language models. The original unstructured text data is finally transformed into "Q-A" structured text data suitable for model training. Fig.7 illustrates the compositional characteristics of global knowledge and local knowledge. It can be seen from the figure that global knowledge contains some detailed content of local knowledge. Fig.8 describes the composition of the data in detail. Local knowledge consists of detailed module knowledge, and global knowledge is composed of part of the local knowledge and system knowledge. The complex system knowledge is divided into a training set and a test set, where the training set is composed of 940 pieces of local knowledge and 400 pieces of global knowledge. The test set contains 130 pieces of local knowledge and 130 pieces of global knowledge.

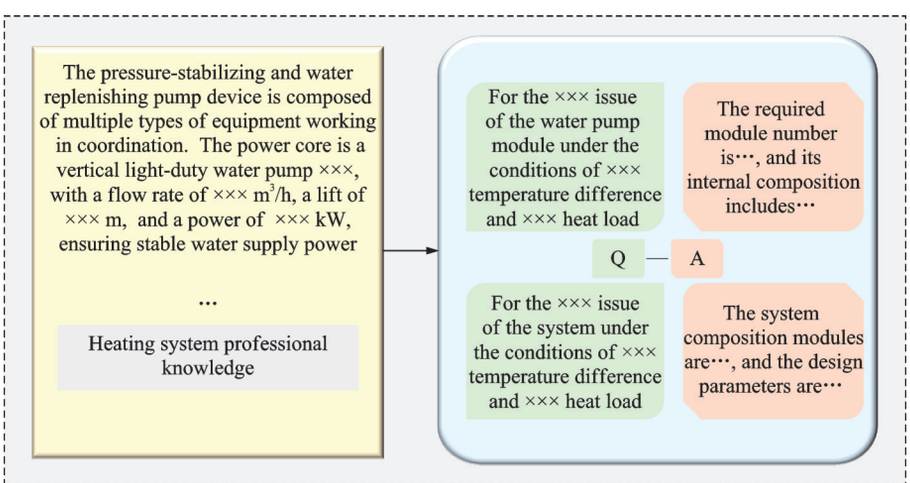

Fig.6 Details of data transformation

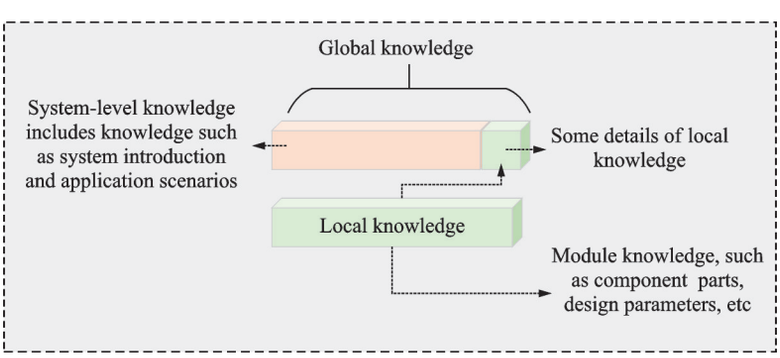

Fig.7 Composition of global knowledge and local knowledge

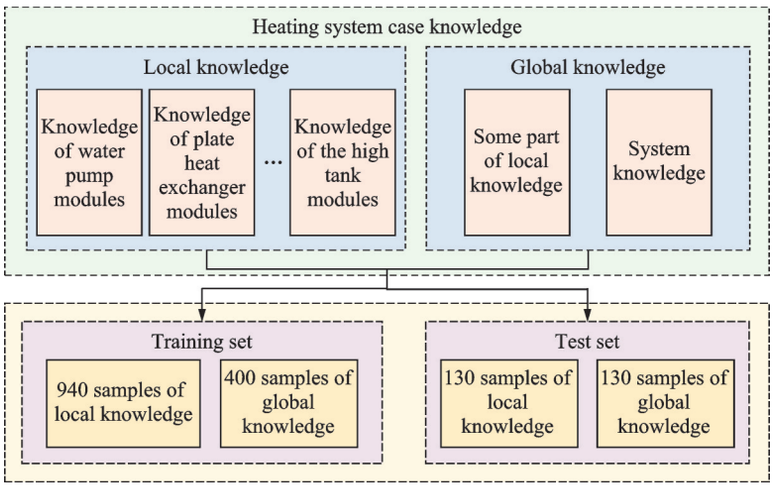

Fig.8 Dataset composition

In this paper, the query part of the heating system case knowledge is uniformly structured as questions "Q", and the corresponding query results are used as their answers "A". In addition, in view of the professionalism of the heating system scheme knowledge data, the rewriting of some engineering terminologies is also an important part of data processing. To comply with the principle of data confidentiality, the data processing task is manually completed by the authors of this paper.

The dataset introductions for some modules are shown in Tables 3,4.

Table 3 Local module knowledge datasets

Serial number	Input condition	Selection result	Supplementary explanation
1	Heat load: 135 kW Temperature difference: 10 ℃ Demand type: Water pump	Water pump module 1	The demand is "water pump". The heat load of 135 kW meets the condition "120 kW$<Q_h\leqslant$240 kW", here Q_h represents the heat load, and the temperature difference of 10 ℃ conforms to "8—12 ℃", so it matches water pump module 1. The rated flow rate of this module is 30—50 m^3/h, and the head is 25—35 m.
2	Heat load: 740 kW Temperature difference: 20 ℃ Demand type: Water pump	Water pump module 4	The demand is "water pump". The heat load of 740 kW meets the condition "700 kW$<Q_h\leqslant$1 000 kW", and the temperature difference of 20 ℃ conforms to "18—22 ℃", so it matches water pump module 4. The rated flow rate of this module is 180—250 m^3/h, and the head is 45—55 m.
3	Heat load: 4 550 kW Temperature difference: 20 ℃ Demand type: Water pump	Water pump module 7	The demand is "water pump". The heat load of 4 550 kW meets the condition "4 500 kW$<Q_h\leqslant$6 000 kW", and the temperature difference of 20 ℃ conforms to "18—22 ℃", so it matches water pump module 7. The rated flow rate of this module is 1 100—1 500 m^3/h, the head is 60—70 m, and the motor power is 200 kW.
4	Heat load: 10 680 kW Temperature difference: 20 ℃ Demand type: Make-up pump	Make-up pump module 12	The demand is "make-up pump". The heat load of 10 680 kW meets the condition "8 500 kW$<Q_h\leqslant$11 000 kW", and the temperature difference of 20 ℃ conforms to "18—22 ℃", so it matches make-up pump module 12. The rated flow rate of this module is 2 100—2 800 m^3/h, the head is 70—80 m, the motor power is 315 kW, and the pressure stability accuracy is ±0.03 MPa.
5	Heat load: 503 kW; Temperature difference: 10 ℃; Demand type: Make-up pump	Make-up pump module 8	The demand is "make-up pump". The heat load of 503 kW meets the condition "500 kW$<Q_h\leqslant$1 000 kW", and the temperature difference of 10 ℃ conforms to "8—12 ℃", so it matches make-up pump module 8. The rated flow rate of this module is 40—80 m^3/h, the head is 30—40 m, the motor power is 15 kW, and the pressure fluctuation range is less than orequal to ±0.02 MPa.

Table 4 Global system knowledge datasets

Serial number	Input condition	Selection result	Supplementary explanation
1	Heat load: 190 kW Temperature difference: 10 ℃ (conventional temperature difference) Scheme 1 (Boiler 1 + Water pump 1 + Make-up pump 1 + Heat exchanger 1 + Water tank 1 + Expansion tank 1)	Selection matching: 190 kW ∈ (120 kW < Q_h ≤ 240 kW), 10 ℃ ∈ 8—12 ℃ (conventional temperature difference), matching Scheme 1. Core module details: Boiler 1: Thermal power 240 kW ≥ 190 kW, outlet water temperature 95 ℃; Water pump 1: Flow rate 30—50 m^3/h (covering the 38 m^3/h demand under 190 kW); Make-up pump 1: Flow rate 10—20 m^3/h (covering the 8 m^3/h make-up water demand); Heat exchanger 1: Heat exchange area 20 m^2 (heat exchange capacity ≥ 250 kW ≥ 190 kW); Water tank 1: Volume 1.5 m^3; Expansion tank 1: Volume 50 L	Core module details: Boiler 1: Thermal power 240 kW ≥ 190 kW, outlet water temperature 95 ℃; Water pump 1: Flow rate 30—50 m^3/h (covering the 38 m^3/h demand under 190 kW); Make-up pump 1: Flow rate 10—20 m^3/h (covering the 8 m^3/h make-up water demand); Heat exchanger 1: Heat exchange area 20 m^2 (heat exchange capacity ≥ 250 kW ≥ 190 kW); Water tank 1: Volume 1.5 m^3; Expansion tank 1: Volume 50 L. Core coupling relationships: Boiler 1 → Water pump 1 → Heat exchanger 1 (heat transfer chain): Hot water from Boiler 1 is delivered to Heat exchanger 1 via Water pump 1, and the flow rate of Water pump 1 matches the thermal power of Boiler 1 and the area of Heat exchanger 1; Make-up pump 1 → Water tank 1 → Water pump 1 (make-up water stabilization chain): Make-up pump 1 supplies water to Water tank 1, and Water tank 1 stabilizes the flow for Water pump 1; Make-up pump 1 → Expansion tank 1 (pressure regulation chain): Make-up pump 1 compensates for leakage, and Expansion tank 1 compensates for expansion, collaborating to stabilize pressure. Scenario adaptation: Total floor area ≤ 20 m^2, suitable for small and medium-sized residential communities with an area of 50 000—100 000 m^2.
2	Heat load: 235 kW Temperature difference: 20 ℃ (large temperature difference) Scheme 9 (Boiler 1 + Water pump 1 + Make-up pump 7 + Heat exchanger 1 + Water tank 1 + Expansion tank 1) Selection matching: 235 kW ∈ (120 kW < Q_h ≤ 240 kW), 20 ℃ ∈ 18—22 ℃ (large temperature difference), matching Scheme 9	Core module details: Boiler 1: Thermal power 240 kW ≈ 235 kW, outlet water temperature 110 ℃; Water pump 1: Flow rate 30—50 m^3/h (fine-tuned to adapt to the 21 m^3/h demand under 235 kW); Make-up pump 7: Flow rate 20—35 m^3/h (adapted to the pressure demand under large temperature difference); Heat exchanger 1: Heat exchange area 20 m^2 (heat exchange capacity ≥ 480 kW ≥ 235 kW); Water tank 1: Volume 1.5 m^3; Expansion tank 1: Volume 50 L	Core coupling relationships: Boiler 1 → Water pump 1 → Heat exchanger 1 (heat transfer chain): High-temperature water (110 ℃) from Boiler 1 is delivered via Water pump 1, adapting to the large temperature difference heat exchange of Heat exchanger 1; Make-up pump 7 → Water tank 1 → Water pump 1 (make-up water stabilization chain): The flow rate of Make-up pump 7 matches the volume of Water tank 1 to ensure the inlet pressure of Water pump 1; Make-up pump 7 → Expansion tank 1 (pressure regulation chain): Make-up pump 7 achieves high-precision pressure stabilization (±0.03 MPa), collaborating with Expansion tank 1 to compensate for volume changes under large temperature difference. Scenario adaptation: Total floor area ≤ 22 m^2, suitable for small and medium-sized industrial parks with an area of 50 000—100 000 m^2.

Continued

Serial number	Input condition	Selection result	Supplementary explanation
3	Heat load: 610 kW Temperature difference: 20 ℃ (large temperature difference) Scheme 11 (Boiler 3 + Water pump 3 + Make-up pump 8 + Heat exchanger 4 + Water tank 1 + Expansion tank 2)	Selection matching: 610 kW ∈ (500 kW$<Q_h\leqslant$700 kW), 20 ℃ ∈ 18—22 ℃ (large temperature difference), matching Scheme 11. Core module details: Boiler 3: Thermal power 700 kW$\geqslant$610 kW, outlet water temperature 110 ℃; Water pump 3: Flow rate 80 — 120 m^3/h (frequency conversion adapted to the 55 m^3/h demand under 610 kW); Make-up pump 8: Flow rate 40 — 80 m^3/h (covering the 32 m^3/h make-up water demand); Heat exchanger 4: Heat exchange area 50 m^2 (heat exchange capacity$\geqslant$1 200 kW$\geqslant$ 610 kW); Water tank 1: Volume 1.5 m^3; Expansion tank 2: Volume 150 L	Core coupling relationships: Boiler 3 → Water pump 3 → Heat exchanger 4 (heat transfer chain): The large thermal power of Boiler 3 matches the large flow rate of Water pump 3 and the large heat exchange area of Heat exchanger 4, realizing large heat load transfer; Make-up pump 8 → Water tank 1 → Water pump 3 (make-up water stabilization chain): Make-up pump 8 is linked with Boiler 3 to supply water in advance, ensuring the large flow operation demand of Water pump 3; Make-up pump 8 → Expansion tank 2 (pressure regulation chain): The large volume of Expansion tank 2 compensates for volume expansion under large heat load, collaborating with Make-up pump 8 to maintain system pressure. Scenario adaptation: Total floor area $\leqslant$ 35 m^2, suitable for large residential communities with an area of 300 000—500 000 m^2 or medium-sized industrial parks with an area of 100 000—200 000 m^2.

3.3 Evaluation indicators

We evaluate the performance of DCI-LLM using evaluation metrics of large model fine-tuning effect to compare the score rate and accuracy of DCI-LLM with those of other fine-tuned models. Precision (Pre) and bilingual evaluation understudy (BLEU)[33] scores are adopted as performance indicators to evaluate the model's performance in knowledge question-answering after fine-tuning. The BLEU score is an evaluation metric used to assess the quality of machine-generated answers by AI models. It gives a score based on the degree of matching between the model-generated results and the answers in the test set, which is calculated as follows

$$\mathrm{BLEU}=\mathrm{BP}\times\exp\left(\sum_{n=1}^{N}\omega_n\log p_n\right) \tag{6}$$

$$\mathrm{BP}=\begin{cases}1 & l_{\mathrm{m}}\geqslant l_{\mathrm{r}}\\ \mathrm{e}^{1-\frac{l_{\mathrm{r}}}{l_{\mathrm{m}}}} & l_{\mathrm{m}}<l_{\mathrm{r}}\end{cases} \tag{7}$$

where p_n is the n-gram precision, ω_n the weight of the n-gram, N the maximum n-gram order, l_{m} the length of the machine translation result, and l_{r} the length of the reference translation. N-gram precision refers to the degree of overlap between the n-grams in the machine translation result and those in the reference translation. BP is the short sentence penalty, which mainly serves to prevent machine translation from generating excessively short translations. Without the short sentence penalty, the model might tend to generate very short sentences to improve other evaluation metrics. BLEU value ranges from 0 to 1. The closer it is to 1, the higher the translation quality. However, the BLEU score evaluation can only ensure whether the answers of the fine-tuned model are close to the standard answers. Therefore, the BLEU score is used to measure the professionalism of the model's answers to questions, but it cannot directly indicate the accuracy of the model's answers to internal and global questions about the heating system.

Accuracy is the ratio of the number of correctly predicted entities to the total number of predicted entities, which is used to measure the accuracy of the fine-tuned model in answering questions related to the internal modules and global aspects of the heating system. Unlike natural language response tasks that only require a simple distinction between posi-

tive and negative examples, the system composition under given working conditions is deterministic, so there exists an exact answer for the large model's prediction results. Thus, accuracy can be adopted to evaluate the performance of the proposed method. This metric is used to measure the accuracy of the fine-tuned model in answering questions pertaining to the internal modules and global aspects of the heating system, with its calculation formula given as follows

$$\text{Accuracy} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{8}$$

where TP represents the number of samples correctly predicted by the model, and FP the number of samples incorrectly predicted by the model. The value of accuracy ranges from 0 to 1, with a value closer to 1 indicating higher accuracy.

4 Experimental Results

We conduct four experiments to verify the effectiveness and domain expertise of DCI-LLM after "local-collaborative" fine-tuning, including the evaluation of QA performance with local and global knowledge fine-tuning, the verification of the effectiveness of "local-collaborative" fine-tuning, the advantages of the "local-collaborative" fine-tuned model, and the evaluation of the professionalism and applicability of the fine-tuned DCI-LLM in answering local and global questions about heating systems. And, we use a demo of a system design case to show the practical usage and advantage of our proposed method.

4.1 Evaluating QA performance with local/global knowledge

In the knowledge QA task for the design of industrial heating systems, the model needs to be capable of processing both local module knowledge and global system knowledge simultaneously. The choice of the base model directly affects the performance after fine-tuning. Due to the differences in pre-training data and model architectures, different LLMs may exhibit significant deviations in the learning of domain-specific knowledge. In the experiments of this section, under the premise of adopting the "local-collaborative" fine-tuning (LoRA-Freeze) strategy, seven mainstream base models including Qwen2.5, LLaMa3.2[34], DeepSeek R1, etc., are compared to verify their performance differences in the QA tasks of heating system with local and global knowledge. As LLM technology is developing rapidly, it is difficult to find the best model ever for DCI-LLM. We therefore select some typical versions of LLMs to provide a general approach for source model selection in practice. This helps determine the optimal base model suitable for knowledge modeling of industrial heating systems, thus offering a model selection pipeline for other complex systems in automatic design. We divide the heating system local and global QA data into a training set and a test set. The training set consists of 940 pieces of local module knowledge and 400 pieces of system global knowledge. 260 heating system questions (covering both local and global knowledge, 130 each) excluding the training set are randomly selected as the QA test set.

We evaluate the performance of various fine-tuned models using BLEU score and accuracy. The evaluation results for using local and global knowledge are shown in Tables 5 and 6, respectively. The BLEU score only indicates that the model-generated answers are reliable in terms of the degree of matching with the reference answers in the test set. For real accuracy, further evaluation by professionals is required (see Section 4.4). As can be seen from Tables 5 and 6, after LoRA-Freeze, the performance rankings of base LLMs in module knowledge and global knowledge tasks are consistent. Qwen2.5 takes a significant lead with a BLEU score of 0.85 and an accuracy of 93.4% in module knowledge, and a BLEU score of 0.79 and an accuracy of 89.3% in global knowledge, followed by DeepSeek R1 and LLaMa3.2. All models exhibit the results that module knowledge processing ability is better than global knowledge, reflecting that global knowledge involves cross-module associations and is more difficult to be discovered. The BLEU score is positively correlated with the accuracy rate, verifying the effectiveness of this indicator in preliminary evaluation. However, considering that a lower BLEU

score may be associated with the situation where the model's answer style is different from that of the standard answer, BLEU cannot be used to fully measure the professionalism of the model. The evaluation of professionals still needs to be considered for comprehensive assessment. From the results found in this section, Qwen2.5 is the preferred base model for QA tasks in the heating system domain.

Table 5 Performance of various base LLMs with "local-collaborative" fine-tuning (local knowledge)

Base LLM	Fine-tune method	Hard-ware environment	BLEU	Accuracy/%
Qwen2.5	LoRA-Freeze	RTX 8000 (48 GB)	0.85	93.4
LLama3.2	LoRA-Freeze	RTX 8000 (48 GB)	0.78	88.3
DeepSeek R1	LoRA-Freeze	RTX 8000 (48 GB)	0.81	90.7
Gemma3	LoRA-Freeze	RTX 8000 (48 GB)	0.74	82.1
Mistral	LoRA-Freeze	RTX 8000 (48 GB)	0.77	83.3
Yi	LoRA-Freeze	RTX 8000 (48 GB)	0.69	75.7
LLava	LoRA-Freeze	RTX 8000 (48 GB)	0.67	73.6

Table 6 Performance of various base LLMs with "local-collaborative" fine-tuning (global knowledge)

Base LLM	Fine-tune method	Hard-ware environment	BLEU	Accuracy/%
Qwen2.5	LoRA-Freeze	RTX 8000 (48 GB)	0.79	89.3
LLaMa3.2	LoRA-Freeze	RTX 8000 (48 GB)	0.70	83.1
DeepSeek R1	LoRA-Freeze	RTX 8000 (48 GB)	0.74	84.3
Gemma3	LoRA-Freeze	RTX 8000 (48 GB)	0.68	77.5
Mistral	LoRA-Freeze	RTX 8000 (48 GB)	0.63	71.4
Yi	LoRA-Freeze	RTX 8000 (48 GB)	0.66	75.9
LLava	LoRA-Freeze	RTX 8000 (48 GB)	0.58	68.2

4.2 Validation of effectiveness of "local-collaborative" fine-tuning

In the knowledge modeling of complex industrial systems, the traditional single-stage fine-tuning (LoRA-Only) method struggles to integrate fragmented local information into system-level global cognition due to its lack of learning the implicit coupling relationships between modules. This easily leads to poor performance of the model when dealing with system-level complex problems. Since global knowledge is crucial for integrating local information, how to effectively utilize it to guide model learning remains an unresolved challenge. An important goal of the DCI-LLM model is to address this challenge. In this section, experiments are conducted to compare the performance of the single-stage LoRA fine-tuned model (DCI-LLM-T) based solely on local module knowledge with the "local-collaborative" fine-tuned model (DCI-LLM) guided by fused global knowledge in the global knowledge QA task of heating system. This comparison aims to verify the necessity of global knowledge guidance for the model to integrate fragmented module information and improve reasoning ability, as well as the effectiveness of the "local-collaborative" fine-tuning strategy in strengthening the model's global cognition of the system through the LoRA-Freeze method.

The proposed DCI-LLM model effectively integrates global and local knowledge through "collaborative" fine-tuning for improving accuracy of generated answers. We conduct experiments to verify the performance of DCI-LLM-T, which only fine-tunes local module knowledge through LoRA-Only, and compare it with DCI-LLM, a LoRA-Freeze model guided by global knowledge, in QA tasks of global knowledge. The comparison results are shown in Table 7. As can be seen from the table, DCI-LLM-T has a BLEU score of 0.37 and an accuracy rate of 52.4%, while DCI-LLM has these two indicators increased to 0.79 and 89.3% respectively, showing a significant performance improvement. The experiments indicate that the "local-collaborative" fine-tuning strategy, by incorporating local details in Lo-

RA-Freeze fine-tuning and using global knowledge to guide model learning, enables the model to capture implicit coupling relationships between modules, significantly improving its ability to understand and generate global knowledge of industrial systems. This verifies the effectiveness of the strategy in enhancing the model's reasoning ability for system knowledge. The core reason is that single-stage local fine-tuning is difficult to integrate fragmented module information, while "local-collaborative" fine-tuning achieves an upgrade in cognition from the module level to the system level.

Table 7 Evaluation of the effectiveness of the "local-collaborative" fine-tuning framework (global knowledge questions)

LLM	Fine-tune method	Knowledge	BLEU	Precision/%
DCI-LLM-T	LoRA-Only	Local knowledge	0.37	52.4
DCI-LLM	LoRA-Freeze	Local and global knowledge	0.79	89.3

4.3 Comparison of fine-tuning performance between Local-Collaborative and LoRA-Only

In the knowledge modeling of complex industrial systems, global knowledge is crucial for enhancing the model's ability to handle system-level problems. The experiments in Section 4.2 have verified that introducing global knowledge guidance improves model performance. However, there may be differences in the efficiency of utilizing global knowledge between the two fine-tuning strategies, LoRA-Freeze and LoRA-Only. The former integrates information through a collaborative mechanism of "local knowledge learning + global knowledge guidance", while the latter learns both local and global knowledge in a single-stage manner. The purpose of the experiments in this section is to compare their performance in system-level QA tasks under the premise that both strategies take global knowledge guidance into account, with a focus on verifying whether "local-collaborative" can achieve the same answer accuracy as "LoRA-Only" with a smaller amount of global data. This will reveal its advantages in the efficiency of global knowledge utilization and provide a basis for model fine-tuning with limited global data in industrial scenarios.

Fig.9 shows the performance comparison between the LoRA single-stage fine-tuned model and the "local-collaborative" fine-tuned model with global knowledge considered in both strategies. It can be seen from Fig.9 that as the amount of global data increases, both fine-tuning strategies gradually improve in the accuracy of system-level question-answering, with the overall performance of each model showing an upward trend. More data help enhance model performance, while the model based on "local-collaborative" fine-tuning requires a smaller amount of knowledge compared with the "LoRA-Only" strategy, to achieve the same accuracy. This result verifies that under the premise of including global knowledge guidance in both methods, the "local-collaborative" strategy is significantly superior to "LoRA-Only" in terms of global knowledge utilization efficiency, making it particularly suitable for scenarios with limited data in industrial settings. The reason is that "LoRA-Only" learns both local and global knowledge simultaneously in a single stage, which easily leads to knowledge confusion and requires a large amount of data for repeated reinforcement to learn global patterns, resulting in low efficiency in data utilization. In contrast, the "local-collaborative" approach adopts a hierarchical learning method of "local first, then global". It first learns detailed module knowledge, and then specifically discovers the knowledge association between local modules and the system, enabling it to accu-

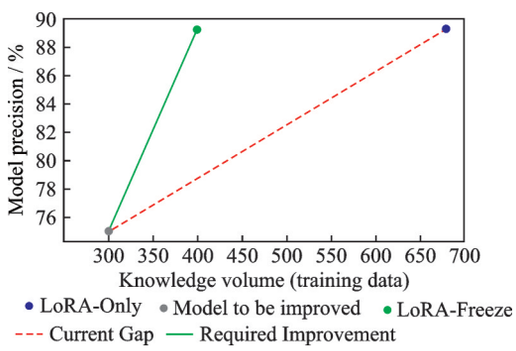

Fig.9 Performance comparison of models with "LoRA-Freeze" and "LoRA-Only" fine-tuning framework

rately capture the core principles in global knowledge. Thus, it establishes effective global cognition even with limited data, giving the advantage of data utilization efficiency.

4.4 Evaluation by professional designers

We expect DCI-LLM to be able to answer common engineering design questions about heating systems and provide professional opinions for engineering designers. In addition, since BLEU has certain limitations in evaluating the professionalism of model answers, it is necessary to conduct human evaluation on the answers generated by DCI-LLM, in term of applicability and professionalism. We invited two professional engineers from a heating system company to conduct ten rounds of dialogues with DCI-LLM. The dialogues included global knowledge questions and local knowledge questions, and the generated answers were manually evaluated. Fig.10 shows the scoring results of the two engineers on the answers generated by DCI-LLM (with a scoring range of 0—10 points). It can be seen that both engineers scored more than six points in both applicability and professionalism indicators, which proves that DCI-LLM has reached a certain professional level in the dialogue tasks at the current stage. This provides ideas for training more professional industrial-level large language models

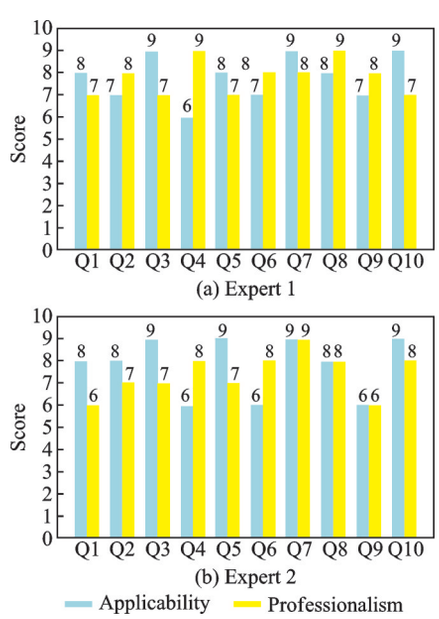

Fig.10 Engineers' ratings on the response of DCI-LLM in terms of applicability and professionalism

in vertical fields. By fine-tuning general large language models using processed professional text data, customized large language models for specific engineering industries can be obtained, and their cost is affordable for most small and medium-sized enterprises.

4.5 Comparison with MLP-based expert systems

We constructed an MLP-based expert system as a comparative baseline. The architecture of this system was specifically optimized for the task of predicting heating system design parameters. It should be noted that since the core parameter of the boiler module is thermal load, this core parameter of the boiler module is not involved in the training of the multi-layer perceptron. Through comparative experiments on test samples, significant differences were observed between the two methods in terms of prediction accuracy and output richness: Mean absolute percentage errors (MAPEs) for the core parameters of each module in the MLP system are as follows: 13.2%, 10.6%, 11.8%, 11.9%, 13.9%, respectively. Comparison results are shown in Fig.11.

Traditional expert systems can only output fixed parameter vectors without any explanatory notes, essentially functioning as "black-box calculators" that require secondary interpretation. In contrast, DCI-LLM can output knowledge-rich solutions in natural language, including principle explanations and selection rationales, and directly drive PyCAD scripts to generate engineering drawings. This leap from "rigid data prediction" to "flexible knowledge generation and delivery" enables DCI-LLM to support conversational and iterative design exploration, achieving end-to-end automation from requirements to preliminary design deliverables and fundamentally reshaping the human-machine collaborative design process. Therefore, DCI-LLM represents a higher level of automation direction—the automation of cognitive processes—filling a critical gap in the full-process intelligence of traditional design tools. Information on LLM versions, API endpoints and model checkpoints is shown in Table 8.

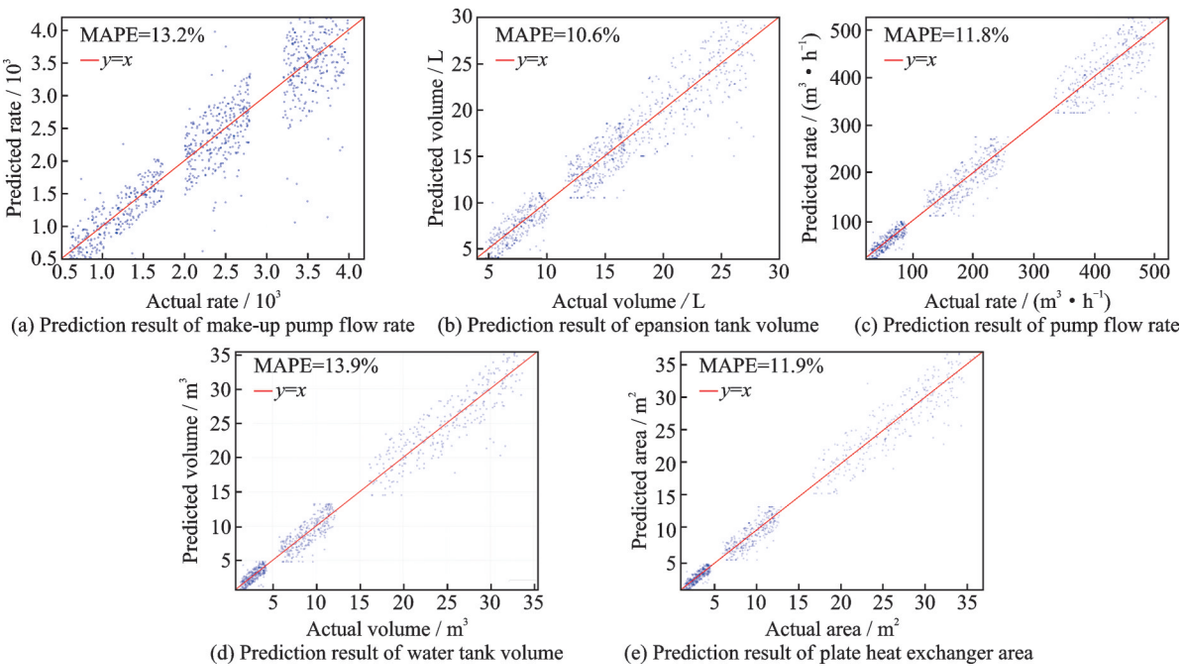

Fig.11 Core parameter prediction results of the MLP module

Table 8 LLM version API endpoint and model checkpoint information

Model	Version	URL
Qwen	Qwen2.5	https://github.com/QwenLM/Qwen/tree/main/Qwen2.5
LLaMa	LLaMa3.2	https://ai.meta.com/resources/models-and-libraries/llama-downloads/
DeepSeek	DeepSeek R1	https://github.com/deepseek-ai/DeepSeek-R1
Gemma	Gemma3	https://huggingface.co/google/gemma-7b
Mistral	Mistral-7B-Instruct-v0.3	https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.3
Yi	Yi	https://huggingface.co/liuhaotian/llava-v1.5-7b

4.6 A demo of heating system design

This section aims to demonstrate the application of the entire large model system in complex engineering system design through an actual scheme design case in specific industrial scenarios, and verify whether it can effectively assist designers in completing the design process of engineering drawings in real business scenarios. Specifically, through the interaction between engineers and the system, the accuracy of module recommendation by the model under specific working conditions and the standardization of automatic scheme drawing are evaluated to verify the practicality and reliability of the DCI-LLM system. This section uses a specific demonstration case to verify the performance of the model. Fig.12 shows a QA demonstration for obtaining internal modules of the system. After consultation, engineers got informed of which modules should be selected for system scheme drawing under the current working conditions, as well as knowledge such as the design parameters of modules and the system. Fig.13 shows the heating system diagram finally drawn by the DCI-LLM model according to the consultation results. Engineers only need to verify the generated system engineering drawings manually. This demonstration case shows that the proposed DCI-LLM model automatically outputs the reasonable module composition of the heating system and the detailed engineering design drawings of the system through interaction with system designers, which improves the efficiency of system design, and reduces design costs, thus significantly optimizing the production workflow of enterprises. Table 9 presents the design parameters of the key component modules predicted by DCI-LLM in the current heating system.

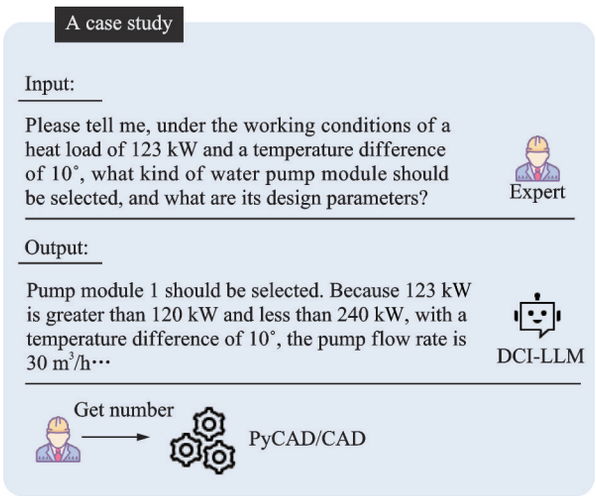

Fig.12 A QA case study for automatic design of the heating system

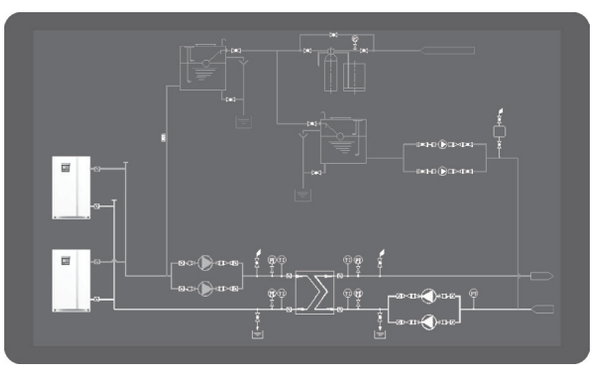

Fig.13 Final system engineering diagram after consultation for the demonstration case

Table 9 Important module parameters of the demo of heating system suggested by DCI-LLM

Module name	Parameter value
BOI/kW	120
WP/(m^3•h^{-1})	30
PHE/m^2	3.16
EWT/m^3	0.1
PSET/L	300
PSWR/(m^3•h^{-1})	0.5

5 Conclusions

To assist the scheme design of complex modular engineering system, this paper proposes a large model framework, DCI-LLM, by integrating professional domain knowledge with downstream engineering scenarios and imitating how engineers learn the composition principles of complex systems. On the basis of constructing system knowledge base, a "local-collaborative" training method based on LoRA fine-tuning technology is designed to fine-tune the Qwen2.5-based industrial dialogue large model that assists system engineering design, so as to fulfill the automatic design of complex modular systems. The performance of DCI-LLM is verified in the scenario of heating system design, and it shows certain accuracy in answering questions about module details and global knowledge of heating systems, which verifies the performance of the proposed large model framework. Compared with the traditional LoRA single-stage fine-tuning, the proposed "local-collaborative" fine-tuning method requires a relatively smaller amount of knowledge on heating system. In addition, with the help of the proposed large model framework, engineers can generate heating system engineering drawings in one stop, without the need to manually draw engineering diagrams. Our work provides a reference for automatic design of other complex systems in the industrial field.

Although DCI-LLM has shown a certain professional level in the experimental verification and can answer some professional questions about heating systems, due to the incomplete and limited amount of data used for fine-tuning training and the fact that this work is still in the early stage of research, the responses generated by the current DCI-LLM model cannot be fully relied upon in practical applications. We aim to improve the accuracy and domain expertise of the responses generated by DCI-LLM in future work, so that it can be deployed in the downstream manufacturing industry of heating systems. In the future, we will carry out the following work to improve the performance of DCI-LLM: (1) We will continuously conduct tests in the downstream industrial design stage, collect test results to enrich scheme knowledge data; (2) we will use the updated industrial scheme data to design a knowledge graph, which is connected to DCI-LLM in the form of an external knowledge base, so that DCI-LLM can not only improve accuracy but also answer questions about real-time updated professional knowledge of new heating systems; (3) we will con-

sider to include traditional machine learning or deep learning models for predicting design parameters of certain internal modules of heating systems, in order to improve the generalization of DCI-LLM; (4) the proposed method will be extended to conduct the automatic scheme design of complex modular systems in other industrial fields for enhancing the production efficiency in broader industrial areas.

References

[1] RADFORD A, NARASIMHAN K. Improving language understanding by generative pre-training[EB/OL]. [2025-09-10]. https://www.semanticscholar.org/paper/Improving-Language-Understanding-by-Generative-Radford-Narasimhan/cd18800a0fe0b668a1cc19f2ec95b5003d0a5035.

[2] LIU Y, HAN T, MA S, et al. Summary of ChatGPT-Related research and perspective towards the future of large language models[J]. Meta-Radiology, 2023, 1(2): 100017.

[3] BAHRINI A, KHAMOSHIFAR M, ABBASIMEHR H, et al. ChatGPT: Applications, opportunities, and threats[C]//Proceedings of 2023 Systems and Information Engineering Design Symposium (SIEDS). Charlottesville, VA, USA: IEEE, 2023: 274-279.

[4] NYBERG E P, NICHOLSON A E, KORB K B, et al. BARD: A structured technique for group elicitation of Bayesian networks to support analytic reasoning[J]. Risk Analysis, 2022, 42(6): 1155-1178.

[5] QIN H, JI G P, KHAN S, et al. How good is google bard's visual understanding? An empirical study on open challenges[J]. Machine Intelligence Research, 2023, 20(5): 605-613.

[6] JIANG A Q, SABLAYROLLES A, ROUX A, et al. Mixtral of experts[EB/OL]. (2024-01-08)[2025-05-19]. https://arxiv.org/abs/2401.04088.

[7] GLM T, ZENG A. ChatGLM: A family of large language models from GLM-130B to GLM-4 all tools[EB/OL]. [2025-05-19]. https://arxiv.org/abs/2406.12793.

[8] YANG J, WANG Z Q, LIN Y B, et al. Global data constraints: Ethical and effectiveness challenges in large language model[EB/OL]. (2024-06-17)[2025-05-19]. https://arxiv.org/html/2406.11214v1.

[9] WETTIG A, GUPTA A, MALIK S, et al. QuRating: Selecting high-quality data for training language models[EB/OL]. (2024-02-17) [2025-05-19]. https://arxiv.org/abs/2402.09739.

[10] SINGHAL K, AZIZI S, TU T, et al. Large language models encode clinical knowledge[J]. Nature, 2023, 620(7972): 172-180.

[11] ZHU Q, LUO J. Generative pre-trained transformer for design concept generation: An exploration[J]. Proceedings of the Design Society, 2022, 2: 1825-1834.

[12] EDWARDS K M, MAN B, AHMED F. Sketch-2 Prototype: Rapid conceptual design exploration and prototyping with generative AI[J]. Proceedings of the Design Society, 2024, 4: 1989-1998.

[13] WANG P, KARIGIANNIS J, GAO R X. Ontology-integrated tuning of large language model for intelligent maintenance[J]. CIRP Annals, 2024, 73(1): 361-364.

[14] WANG H, WANG C, LIU Q, et al. A data and knowledge driven autonomous intelligent manufacturing system for intelligent factories[J]. Journal of Manufacturing Systems, 2024, 74: 512-526.

[15] LIN F, LI X, LEI W, et al. PE-GPT: A new paradigm for power electronics design[J]. IEEE Transactions on Industrial Electronics, 2025, 72(4): 3778-3791.

[16] YANG A, YANG B S, ZHANG B C, et al. Qwen2.5 technical report[EB/OL]. (2024-12-19)[2025-05-19]. https://arxiv.org/abs/2412.15115.

[17] OPENAI, ACHIAM J, ADLER S, et al. GPT-4 technical report[EB/OL]. (2023-03-15)[2025-05-19]. https://arxiv.org/abs/2303.08774.

[18] TONG H, LI J, WU N, et al. Ploutos: Towards interpretable stock movement prediction with financial large language model[EB/OL]. (2024-03-18)[2025-05-19]. https://arxiv.org/abs/2403.00782.

[19] SINGHAL K, TU T, GOTTWEIS J, et al. Toward expert-level medical question answering with large language models[J]. Nature Medicine, 2025, 31(3): 943-950.

[20] SUN Liping, TONG Zilong, QIAN Qian, et al. Two-phases fine-tuning of professional large language model via clinical data[J]. Application Research of Computers, 2024, 41(10): 2906-2910.

[21] XIONG H, WANG S, ZHU Y, et al. DoctorGLM: Fine-tuning your Chinese doctor is not a Herculean task[EB/OL]. (2023-04-03)[2025-05-19]. https://arxiv.org/abs/2304.01097.

[22] CUI J, LI Z, YAN Y, et al. Chatlaw: A multi-agent

collaborative legal assistant with knowledge graph enhanced mixture-of-experts large language model[EB/OL]. (2023-06-28)[2025-05-19]. https://arxiv.org/abs/2306.16092.

[23] YUE S, LIU S, ZHOU Y, et al. LawLLM: Intelligent legal system with legal reasoning and verifiable retrieval[C]//Proceedings of Database Systems for Advanced Applications. Singapore: Springer, 2024: 304-321.

[24] FEI Z, ZHANG S, SHEN X, et al. InternLM-Law: An open-sourced chinese legal large language model[C]//Proceedings of the 31st International Conference on Computational Linguistics. Abu Dhabi, UAE: ACL, 2025: 9376-9392.

[25] PANG H, WANG Z, LI G. Large language model guided deep reinforcement learning for decision making in autonomous driving[EB/OL]. (2024-12-24)[2025-05-19]. https://arxiv.org/abs/2412.18511.

[26] HU E J, SHEN Y, WALLIS P, et al. LoRA: Low-rank adaptation of large language models[EB/OL]. (2021-06-07). https://arxiv.org/abs/2106.09685.

[27] HU Z, WANG L, LAN Y, et al. LLM-adapters: An adapter family for parameter-efficient fine-tuning of large language models[C]//Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing. Stroudsburg, PA, USA: ACL, 2023: 5254-5276.

[28] LESTER B, AL-RFOU R, CONSTANT N. The power of scale for parameter-efficient prompt tuning[C]//Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. Punta Cana, Dominican Republic: ACL, 2021: 3045-3059.

[29] AGHAJANYAN A, GUPTA S, ZETTLEMOYER L. Intrinsic dimensionality explains the effectiveness of language model fine-tuning[C]//Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers). [S.l.]: ACL, 2021: 7319-7328.

[30] WANG Z, ZHONG W, WANG Y, et al. Data management for training large language models: A survey[EB/OL]. (2023-12-04). https://arxiv.org/abs/2312.01700.

[31] XU X, WU Z, QIAO R, et al. Data-centric AI in the age of large language models[EB/OL]. (2024-06-24)[2025-05-19]. https://arxiv.org/abs/2406.14473.

[32] DAI D, DENG C, ZHAO C, et al. DeepSeekMoE: Towards ultimate expert specialization in mixture-of-experts language models[C]//Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Bangkok, Thailand: ACL, 2024: 1280-1297.

[33] PAPINENI K, ROUKOS S, WARD T, et al. BLEU: A method for automatic evaluation of machine translation[C]//Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics. Philadelphia, Pennsylvania, USA: ACL, 2002: 311-318.

[34] TOUVRON H, LAVRIL T, IZACARD G, et al. LLaMA: Open and efficient foundation language models[EB/OL]. (2023-02-27)[2025-05-19]. https://arxiv.org/abs/2302.13971.

Authors

The first author Mr. **CAI Xin** received his Bachelor of Science degree in management science and engineering from Chang'an University in Xi'an, China in 2023. He is currently pursuing his M.S. degree at College of Artificial Intelligence, Nanjing University of Aeronautics and Astronautics. His research interests include artificial intelligence and large-scale models.

The corresponding author Prof. **LIU Xuejun** works in College of Artificial Intelligence, Nanjing University of Aeronautics and Astronautics. Her research focuses on artificial intelligence and its applications.

Author contributions Mr. **CAI Xin** designed the research methodology, conducted the experiments, performed data analysis, and prepared the initial manuscript. Prof. **LYU Hongqiang** and Mr. **XU Ran** were responsible for validating the research methods. Mr.**WANG Bo**, Mr. **WANG Qi**, and Mr. **WANG Heyun** were responsible for data collection. Prof. **LIU Xuejun** provided research supervision and revised the manuscript. All authors commented on the manuscript draft and approved the submission.

Competing interests The authors declare no competing interests.

(Production Editor: ZHANG Huangqun)

基于大语言模型的复杂工业系统设计:以供暖系统设计为例

蔡 鑫[1],吕宏强[2],许 冉[1],王 波[3],王 琦[3],王鹤云[3],刘学军[1]

(1.南京航空航天大学人工智能学院,南京 211106,中国;2.南京航空航天大学航空学院,南京 210016,中国;
3.青岛凯能环保科技有限公司,青岛 266300,中国)

摘要:针对当前模块化工程系统方案设计与装配流程中存在的人工设计效率低、易出错、工程制图耗时久,以及通用大语言模型在工业领域因领域数据稀缺、专业术语理解偏差而导致应用不准确、可信度不足等问题,本文将大语言模型应用于下游工业生产制造尤其是模块化工程系统方案设计中,挖掘复杂系统模块间的深度耦合关系,实现复杂系统自动化设计与工程制图自动化。本文基于通义千问2.5(Qwen2.5)构建面向复杂模块化工业系统方案设计的大模型框架(Large language model for design of complex industrial system,DCI-LLM),搭建涵盖模块内部信息与全局系统知识的模块对话数据库,并设计"局部协同"微调策略,使模型充分学习模块工况知识、深度挖掘模块间耦合关系,同时结合后处理程序集成,实现复杂系统工程制图全流程自动化。本文依托国内某企业供热系统知识数据库构建对话数据集开展验证实验。实验结果表明,微调后的DCI-LLM在模块知识问答、系统全局知识问答任务中的准确率分别可达93.4%、89.3%,模型专业认知能力优异;同时经专业工程师综合评测,该模型具备良好的实际工程应用价值。实际测试证实,DCI-LLM框架可有效实现从自然语言需求到完整工程图纸的端到端自动化设计,突破了传统大模型仅聚焦文本交互或中间代码生成的应用局限。本文研究揭示了基于数据特征驱动的大模型微调策略可有效适配模块化工业领域设计任务,所提框架与方法可为大语言模型在工业领域其他场景的落地应用提供可行技术路径与实施策略参考,在各类工业子领域自动化工程设计工作中具备广阔的应用前景与实用价值。

关键词:大语言模型;低秩自适应微调;冻结微调;复杂系统设计;系统工程

研究亮点:

1.本文面向模块化工程系统自动化设计,提出一种领域专用大语言模型框架DCI-LLM。该框架实现了从自然语言需求到完整工程图纸的端到端自动化设计流程。与现有研究相比,DCI-LLM突破了传统大语言模型仅局限于文本交互或中间代码生成的短板。通过融入PyCAD脚本生成能力,构建了"需求—方案—图纸"全链路自动化设计流程。

2.本文提出一种数据特征驱动的局部协同微调策略,通过模块知识深度学习与全局耦合关系挖掘的协同机制,为大语言模型在具备模块化数据特征的特定工业领域落地应用提供了可行路径。

3.依托供暖系统工程设计实例,验证了所提模型的可行性与有效性。实验结果表明,该框架在工业细分领域工程自动化设计中具备实际应用潜力。

4.本文采用构建的复杂模块化工业系统方案设计大模型框架,从技术路径与实施策略层面,为大语言模型在工业领域的其他应用场景提供了方法学参考。