Kalpa Publications in Computing
Volume 22, 2025, Pages 923–932
Proceedings of The Sixth International Conference on Civil and Building Engineering Informatics

Research on A Methodology of AI-Driven Substation Layout Design Based on Large Language Model

Jianyong Shi[1], Zeyu Pan[1], Longfei Pan[2], and Junpeng Hu[1]
[1] Department of Civil Engineering, School of Ocean and Civil Engineering, Shanghai Jiao Tong University, Shanghai, China.
[2] Sichuan Research Institute, Shanghai Jiao Tong University, Shanghai, China.
shijy@sjtu.edu.cn, panzeyu@sjtu.edu.cn, plf_email@163.com, hu-j.p@sjtu.edu.cn

Abstract

In the design stage of substation projects, there is an enduring task for designing an optimal layout for equipment under various constraints from functional, geometric, and economic aspects. In traditional practices, professionals need to recursively adjust the positions of involved objects in the specialized working spaces to meet the requirements of different projects and to comply with design codes. It is highly dependent on professional skills and understanding of regulatory documents. To streamline this process, we propose an AI-driven substation layout design approach using large language model (LLM). This approach exploits the capacity of large language model by converting the task of generating the full layout plan into generating sequences of the positions of involved equipment. We have finetuned two models based on a base model Llama3.1-7B with two auto-generated datasets under professional guidance. One dataset consists of the input requirements and output layout scheme, while the other is augmented through chain-of-thought (CoT) to elicit the underlying language model to retrieve optimal capacity with the consideration of specific design constraints. To implement the output scheme, an automated procedure is further developed to translate the scheme into the corresponding layout plan and information models.

1 Introduction

The substation equipment layout task is a comprehensive and complex project. The specific contents involved include but are not limited to determining the type and quantity of equipment required for the substation and their precise placement. At the same time, the design scheme needs to be optimized to achieve the goals of maximizing layout accuracy, optimizing costs, and maximizing safety and

I. Hkust, J. Cheng and Y. Yantao (eds.), ICCBEI 2025 (Kalpa Publications in Computing, vol. 22), pp. 923–932

reliability. The task requirements encompass ensuring the stable operation of the power system and adhering to design specifications, while constraints involve a myriad of factors, including technical standards and legal regulations. To complete this task, designers must not only have solid professional knowledge and rich engineering experience, but also have a deep understanding and mastery of the specific requirements of the task, various constraints in the design process, and related engineering prior knowledge to ensure the rationality and feasibility of the layout plan.

Although a variety of design software has been developed in the field of substation equipment to improve design efficiency, current design technology still relies mainly on manual operation. This results in a lot of repetitive work when dealing with many standard design tasks, which not only consumes a lot of money, but also leads to errors and deviations caused by human factors, as well as some unavoidable problems. The high degree of manual intervention in the design process has affected efficiency and accuracy to a certain extent.

In response to the above problems, this study proposed an artificial intelligence-driven substation layout design method using a large language model. This study introduced artificial intelligence technology and focused on the semantic expression of substation layout drawings, model pre-training and fine-tuning, semantic conversion and other technical aspects. The use of artificial intelligence technology to generate drawings improves the efficiency of substation design; the use of procedural and modular ideas to generate design drawings reduces human intervention and improves the accuracy of the design. For professional designers, it significantly reduces the workload and further improves the efficiency of substation layout design, which is of great significance to the development of power grid engineering.

Studies on automatic layout generation have appeared several times in literature (Agarwala et al., 2011). Recent approaches to layout generation consider both unconditional generation (Arroyo et al., 2021) and conditional generation in various setups, such as conditional inputs of category or size (Gupta et al., 2021, Jiang et al., 2022), relational constraints (Jiang et al., 2022, Kikuchi et al., 2021), element completion (Gupta et al., 2021), and refinement (Jiang et al., 2022). Some attempt at solving multiple tasks in a single model (Kikuchi et al., 2021, Kong et al., 2022).

As an important part of generative models, language models have achieved remarkable achievements in the current development. The advent of the Transformer architecture, grounded in deep learning and large-scale pre-trained models, has revolutionized natural language understanding and generation by language models. These models play a key role in natural language processing tasks such as text classification, machine translation, and question-answering systems. From automated document writing to the initial conception of design schemes, language models have demonstrated their powerful text generation capabilities. Their application in the field of natural language processing has penetrated into multiple levels of scheme design. In scheme design tasks, these models are widely used to generate design instructions, technical specifications, and even directly participate in the preliminary layout of design schemes. However, this paper considers exploring the generation of substation equipment layout schemes from another perspective, that is, transforming the traditional task goal from directly generating a plane layout scheme diagram to generating a form of equipment metadata sequence. The core idea of this method is to use the semantic comprehension and inference capabilities of the language model to generate a series of metadata sequences representing the substation equipment layout by encoding equipment attributes and layout rules, and initially have the ability to control the design scheme generation process according to contextual information such as design requirements and design constraints. This provides designers with a new design idea and tool to improve the flexibility and automation level of the design while ensuring the rationality of the layout.

This paper uses a LLM to perform AI-driven substation layout design. By collecting substation layout drawings, key information is extracted from the drawings, a big language model fine-tuning dataset is constructed, and the big language model is fine-tuned to output semantic design drawing information. Finally, the semantic information is converted into layout drawings through a semantic conversion tool to complete the intelligent layout of the substation.

2 Method

Large language model (LLM) is a natural language processing model based on deep learning technology with billions to hundreds of billions of parameters. It is pre-trained with massive data and can understand and generate natural language text. It has versatility, contextual understanding and natural fluency. It can perform multiple language tasks including text generation, translation, question answering, etc. It is an important breakthrough in the current field of natural language processing. The Meta Llama 3.1 multilingual large language model (LLMs) collection is a collection of pre-trained and instruction-adjusted generative models with three sizes (text input/text output): 8B, 70B and 405B. The instruction-adjusted plain text models (8B, 70B and 405B) of Llama 3.1 are optimized for multilingual conversation use cases and outperform many existing open source and closed chat models in common industry benchmarks (meta-llama). Since 70B and 405B cannot run on consumer-grade graphics cards, this experiment uses llama3.1-8B-instruct as the base model. It has about 8 billion parameters and can complete tasks well according to user instructions. It can also run on consumer-grade graphics cards and can popularize the method in this article on a large scale.

Fine-tuning (Zhao et al., 2023) is a machine learning technique based on a pre-trained model. It aims to optimize the performance of the model and adapt it to new application scenarios by training it with a small amount of task-specific datasets. This process involves adjusting some parameters of the model, usually retaining most of the learned general features and modifying only the top layer or output layer to match the requirements of the specific task. Fine-tuning can effectively utilize the knowledge of the pre-trained model, reduce training costs, and achieve better performance on the target task. Common fine-tuning methods include full model fine-tuning (Lv et al., 2023), frozen bottom layer fine-tuning (Zheng et al., 2024), prompt tuning, and low-rank adaptation (LoRA) (Hu et al., 2021). Full model fine-tuning allows the entire model to be adjusted. Its advantage is that it can make full use of task-specific data, but its disadvantage is that it consumes a lot of computing resources and is prone to overfitting. Frozen bottom layer fine-tuning only adjusts the top-level parameters of the model, saving resources and reducing the risk of overfitting, but it may not fully capture task-specific features. Prompt fine-tuning affects the model output by modifying the input without updating the model parameters. It is computationally efficient, but its applicability may be limited. LoRA simulates parameter updates by introducing low-rank matrix decomposition, which has the advantages of high computational efficiency and low resource consumption, while keeping most of the parameters of the pre-trained model unchanged, effectively avoiding catastrophic forgetting. Comprehensive analysis shows that the LoRA fine-tuning method can efficiently adapt to new tasks while maintaining the advantages of pre-training. It is a resource-efficient and robust fine-tuning strategy.

This paper is dedicated to exploring the application of artificial intelligence in substation layout design. By utilizing the powerful capabilities of the big language model, a new substation layout design method is realized. We first systematically organize the existing substation design drawings, and use the prompt word technology of the big model to construct a design scheme dataset rich in thinking logic chain in combination with specific design conditions and specification constraints. On this basis, we use the LoRA fine-tuning technology to customize the big model to obtain LoRA weights that are more suitable for substation layout design. Then, we deploy the fine-tuned LoRA together with the big model as a local service, and develop a plug-in compatible with Revit software to provide users with an intuitive interactive interface. Users input key parameter information such as the number of equipment, and the plug-in encapsulates these parameters and constructs them into prompt words, and sends requests to the local big model service. After the server parses the returned design scheme, it presents it to the user intuitively, thus realizing the automation and intelligence of substation layout design.

There are two datasets, one of which includes input requirements and output layout plans. The input requirements include the range, device name, and size of the required layout. The output layout plan includes the device name, size, and layout information (the distance of the device from the x-axis and y-axis, and the orientation). These two fields can be regarded as the input and output of the model during fine-tuning. System prompts are also required during fine-tuning. Construct the system prompts "You are now an experienced architectural designer. Use your design experience and your design talents to think step by step to get a design plan. Design plan 1 is expressed in standard markdown json format." The first fine-tuning dataset is obtained as shown in Figure 1.


```
"instruction": "Please provide a detailed design, use your design experience, and create a design that answers the user's
question.",
"input": "Within an area of 6.1m x 17.5m, arrange the following equipment: a 12.8m x 1.5m basic wall, two 8.8m x 0.6m
devices, two 1.725m x 1.08m workstations, two 2m wide doors, and a 1m x 1m window. Additionally, there are 1.15m x 3.5m
columns in the space. Requirements: cabinet placement along the wall, equipment placement perpendicular to the wall,
people and equipment must face each other, 0° faces 0°, 2° faces 2°, 3° faces 3°. Please ensure a passage width of no less
than 2m and keep the placement margins at 0.2m. Provide your design solution based on the given json format.",
"output": "{
  \"Design\": [
    {\"Column\": \"Position\": \"-200mm, 200mm\", \"Dimensions\": [\"1.75, 6.1\"], \"Orientation\": \"0°\", \"Name\":
\"Column\", \"Clearance\": \"1\"},
    {\"Dimensions\": [\"12.8, 1.5\"], \"Orientation\": \"3°\", \"Name\": \"Basic Wall\", \"Position\": [\"1.725, 108\"],
\"Clearance\": \"0.2\"},
    {\"Dimensions\": [\"8.8, 0.6\"], \"Orientation\": \"1°\", \"Name\": \"Device\", \"Position\": [\"7.25, 108\"],
\"Clearance\": \"5.2\"},
    {\"Dimensions\": [\"8.8, 0.6\"], \"Orientation\": \"1°\", \"Name\": \"Device\", \"Position\": [\"2.0, 18\"],
\"Clearance\": \"1\"},
    {\"Dimensions\": [\"2.0, 1.8\"], \"Orientation\": \"0°\", \"Name\": \"Workstation\", \"Position\": [\"12.525\"],
\"Clearance\": \"5.1\"},
    {\"Dimensions\": [\"1.15, 3.5\"], \"Clearance\": \"2.6\"}, {\"Clearance\": \"15.850\", \"Name\": \"Window\"}
  ]
}"
```


Figure 1: Basic fine-tuning dataset.

Another dataset is based on the previous dataset and integrates design specification constraints and chain of thought (COT) technology (Wei et al., 2022), aiming to improve the design accuracy of the model and reduce model hallucination phenomena.

Based on the above two data sets, this paper designed four experimental schemes. The first one is to use no fine-tuning + One-Shot case + design requirements + adding design constraints in Prompt. The second one is also in the form of no fine-tuning + Few Shot case + design requirements + design constraints. The third one is a fine-tuning model + design requirements for the combination of design requirements and design solutions. The fourth one is a fine-tuning model + design requirements for the combination of cot + sft (design requirements + intermediate design process + design solution). The experimental results show that the layout solution obtained by the cot + sft solution is the best. The specific technical method is shown in Figure 2.

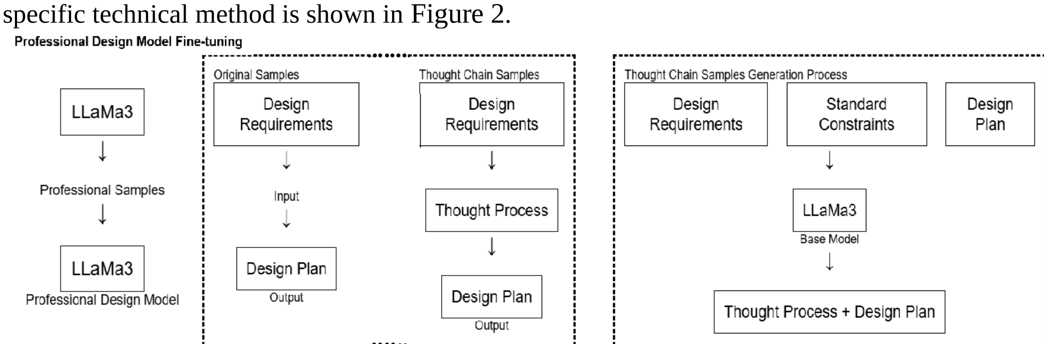

Figure 2: Schematic diagram of the cot dataset construction process.

The standardized design scheme for substation layout refers to providing a set of preset layout schemes for substation equipment based on standards and specifications, including different types such as modular, compact and partitioned designs. These schemes convert the design intent into a serialized representation of the equipment layout, that is, convert the type, size, location and orientation of the equipment into a data token sequence using a large language model, thereby realizing the conversion from the traditional image generation problem to the determination of the location and orientation data of the equipment bounding box in the plane. This process not only simplifies the design complexity, but also makes it possible for automated layout.

Since the current large language model has a good understanding of data in JSON format, we consider converting the substation into a semantic string in JSON format. Converting the substation layout drawing into a JSON file is a complex but important process, which can achieve structured storage and convenient interaction of data. Figure 3 shows the JSON file content of the drawing information interpreted by the fine-tuned large model.

"Design Plan Description": "Please arrange the following devices within a 9.8m x 17.1m rectangular area:
12.8m x 1.5m device 1,
11.9m x 0.8m device 2,
1.6m x 0.6m device 3,
7.2m x 0.6m device 4, two 2.4m x 0.8m devices connected to No. 2,
two 2.4m x 0.8m devices connected vertically to the above-mentioned devices,
two 0.9m x 2.0m devices connected to the above-mentioned devices, and two 2m x 2m platforms!
12.8m, 0.8m and 0.9m devices are placed horizontally,
1m x 2.8m devices are placed vertically.
Layout principle: people and devices are arranged symmetrically.
Devices face forward, people face forward.
Two represent the negative direction, three represent the positive direction."

"Device Layout":

- "Name": "Basic platform, offset -200mm_200"
  - "Length": "17.1"
  - "Width": "9.8"

- "Name": "Device 1"
  - "Length": "12.8"
  - "Width": "1.5"

- "Name": "Device 2"
  - "Length": "7.3"
  - "Width": "3.3"

We use the big language model to output the thought chain process, that is, given the design

Figure 3: Semantic information of drawings in json format.

conditions, design constraints, and design results, let the big language model output the intermediate process of the design. The design constraints sorted out based on the existing data are as follows:

The width (clearance) of various channels in the high-voltage distribution room should not be less than the following:

When the switch cabinet is arranged in a single row, in the channel classification, the clear width of the maintenance channel behind the cabinet should not be less than 0.8m; in the operation channel, if the switch cabinet is fixed, the clear width of the operation channel should not be less than 1.5m, and if the switch cabinet is a push type, the clear width of the operation channel should not be less than the length of a single vehicle + 1.2m; and the clear width of the channel leading to the explosion-proof interval should not be less than 1.2m.

When the switch cabinet is arranged with switch cabinets on both sides and face to face, in the channel classification, the clear width of the maintenance channel behind the cabinet should not be less than 0.8m; in the operation channel, if the switch cabinet is fixed, the clear width of the operation channel should not be less than 2m, and if the switch cabinet is a push type, the clear width of the operation channel should not be less than the length of two vehicles + 0.9m; and the clear width of the channel leading to the explosion-proof interval should not be less than 1.2m.

When the switch cabinet is arranged with switch cabinets on both sides and back to back, in the channel classification, the net width of the maintenance channel behind the cabinet should not be less than 1m; in the operation channel, if the switch cabinet is fixed, the net width of the operation channel should not be less than 1.5m, and if the switch cabinet is hand-push, the net width of the operation *channel should not be less than the length of the single vehicle + 1.2m; and the net width of the channel leading to the explosion-proof interval should not be less than 1.2m.*

When the length of the high-voltage distribution room exceeds 7m, two doors should be opened and arranged at both ends. The height of the handling door of the GC-1A (F) type high-voltage switch cabinet is 2.5~2.8m and the width is 1.5m.

When the power supply enters from the back of the cabinet and an isolating switch and its manual operating mechanism need to be installed on the wall behind the cabinet, the net width of the channel behind the cabinet should not be less than 1.50m; when the protection level of the back of the cabinet is IP2X, it can be reduced to 1.30m.

The built prompts are as follows:

Please combine the design requirements: {requirment} and the design constraints: {limit}. Please briefly quote the design constraints, give the design steps step by step, and reason to get the final design result: {result}. Please note: only output the intermediate thinking process, do not add the final design result.

The above prompt words do not allow the big language model to add the final design results because the output of the big language model will have hallucinations. In order to avoid this phenomenon, the final design results can be concatenated with strings after generation.

After using the local big model to build a local service, input the data set and prompt words into the big model to obtain a thinking chain data set with design logic and design constraints.

Based on the fine-tuning dataset and large model, the method for LoRA fine-tuning is as follows: First, replace the key layers in the large model with LoRA layers, which contain trainable low-rank matrices to reduce the number of fine-tuning parameters. Next, freeze the original parameters of the large model and only train the LoRA layer and classification layer. Use an appropriate optimizer (such as AdamW) and learning rate scheduler to fine-tune the model with the training dataset, while monitoring the performance of the validation set to adjust the hyperparameters. After fine-tuning, use the test set to evaluate the model performance to ensure that the LoRA fine-tuned model performs well on specific tasks. This paper uses the llama factory fine-tuning framework (Llama-Factory) to implement lora fine-tuning. The fine-tuning training loss and validation loss images are shown in Figure 4.

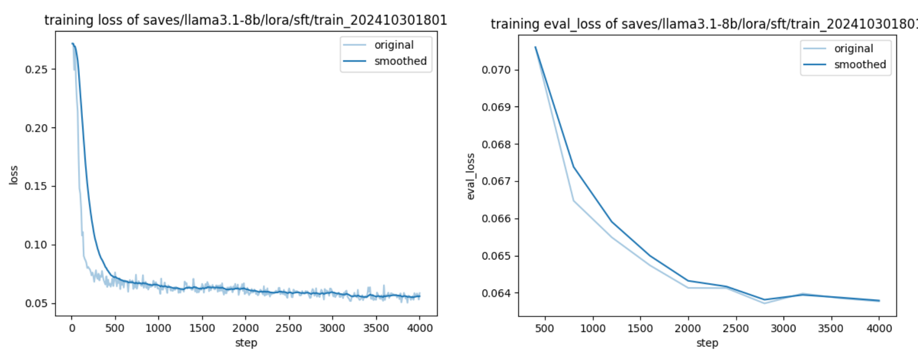

Figure 4: A fine-tuning training and validation loss image.

Judging from the training loss graph, the LoRA fine-tuning process of the Llama 3.1B model showed some remarkable characteristics. The loss value dropped sharply at first, showing the model's ability to quickly learn and adapt to new tasks. As the training progressed, the rate of decline of the loss value gradually slowed down and entered a relatively stable stage, indicating that the model is gradually approaching the optimal solution. Despite this, the loss value remains at a low level, reflecting the model's ability to continuously optimize during training. Overall, the results of this LoRA fine-tuning are satisfactory, and the model has shown stable learning ability and continuous improvement potential during training.

3 Results

After fine-tuning the large language model, the model and LoRa weights are deployed locally using the VLLM framework (vllm-project), and the large model is accessed using the API. The results show that the model has preliminary layout capabilities after fine-tuning.

Using the big model Agent technology, we build a drawing tool agent that can call the Revit API, which can convert JSON data into drawings in RVT format. For equipment information, we intelligently create corresponding family instances in Revit based on the data in JSON. At the same time, we process layout information, which involves creating elements such as rooms and areas, and setting their boundaries and properties. Figure 5 and Figure 6 show the results of the case model output.

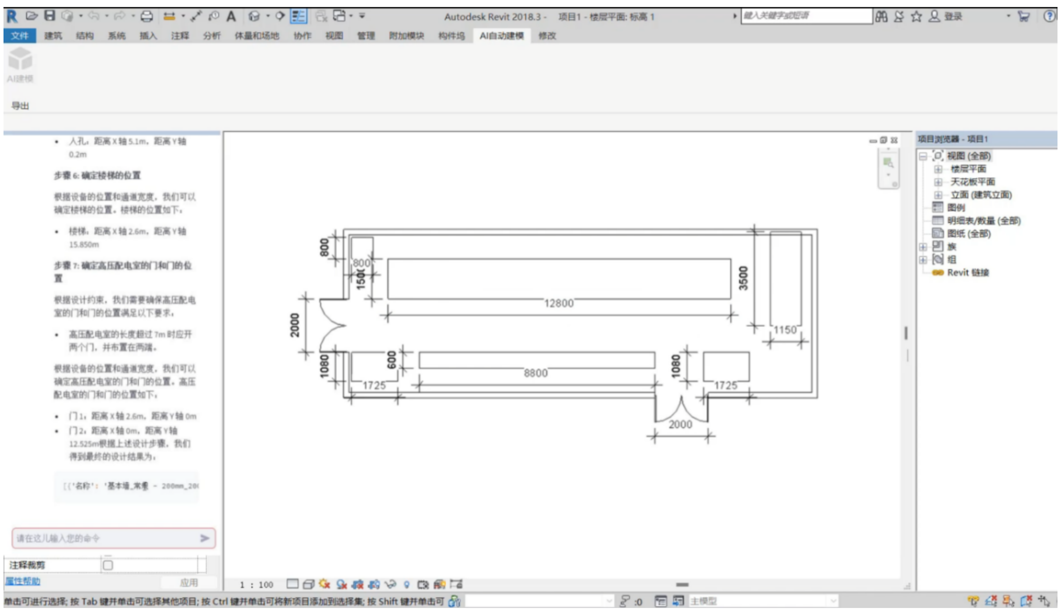

Figure 5: Layout drawing after a model is exported.

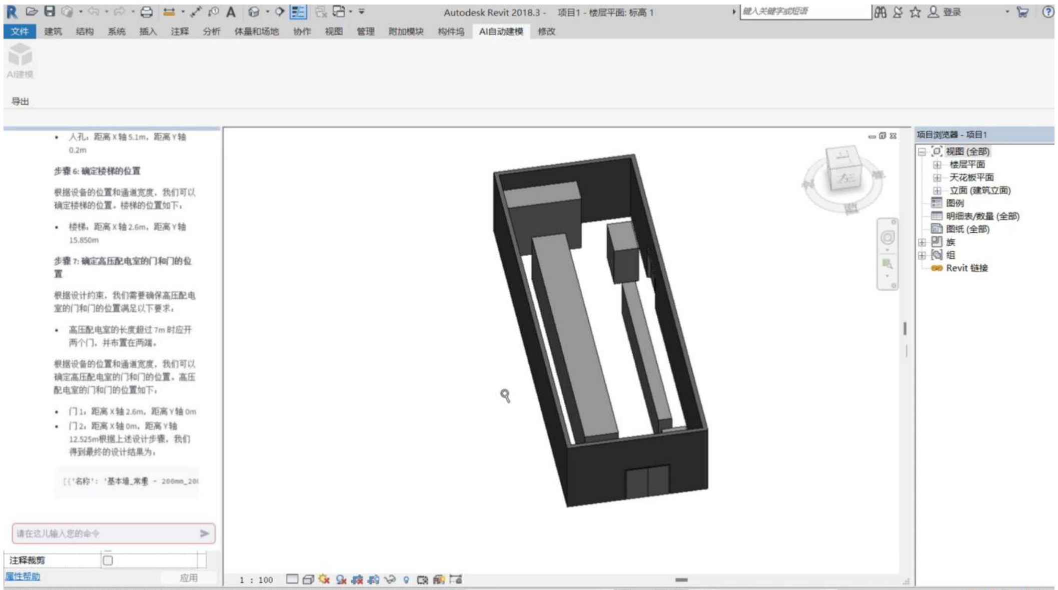

Figure 6: Layout drawing after a model is exported (3D).

Determine the evaluation indicators and display the comparison results. The four contents mentioned in the previous article are used for the comparison experiment. The first is to use no fine-tuning + One-Shot case + design requirements + adding design constraints in the prompt. The second is also in the form of no fine-tuning + Few Shot cases + design requirements + design constraints. The third is a fine-tuning model + design requirements for the combination of design requirements and design solutions. The fourth is a fine-tuning model + design requirements for the combination of (design requirements + intermediate design process + design solution). Display the generation results of the floor plan and the information model under different solutions.

Evaluation index: defines whether the format generated by the model satisfies json as J. If yes J = *true*, otherwise J = *false*.

Define the number of devices in the layout conditions as n, the number of devices output by the model as n', and the accuracy of the number of devices generated is as Equation (1):

$$T_n = \frac{|n - n'|}{n} \tag{1}$$

Define the *x-axis* range of the device i to be placed as $[x_{i1}, x_{i2}]$, the y-axis range as $[y_{i1}, y_{i2}]$, the *x-axis* and *y-axis* coordinates of device i predicted by the model are x'_i and y'_i, then the device range accuracy is as Equation (2):

$$S = \frac{\sum_1^n x'_i \subset [x_{i1}, x_{i2}] \& y'_i \subset [y_{i1}, y_{i2}]}{n} \tag{2}$$

The following Table 1 compares some of the methods used in the study:

Methods	Indicator J	Number accuracy T_n	Device range accuracy S
One-shot+Requirements+Constraints	False		
Few-shot+Requirements+Constraints	true	85.1%	15.7%
Sft+Requirements+Constraints	true	95.3%	85.2%
Sft+Requirements+Constraints	true	98.2%	95.1%

Table 1: Comparison of methods.

From the evaluation results, we can see that the Sft+cot+demand+constraint solution outperforms the other solutions in all aspects, followed by the Sft+demand+constraint solution. Although the Few-shot+demand+constraint solution can complete the task, its range accuracy needs to be improved. The One-shot+demand+constraint solution seems to be the least ideal and needs further improvement or adjustment.

4 Conclusions

In the field of substation layout design, the use of large language models to achieve AI-driven scene intelligence has far-reaching significance. It can improve the level of design automation, reduce human errors, and significantly improve design efficiency. However, the existing image generation-based solution is incapable of dealing with specific design specifications and engineering constraints, and lacks the necessary flexibility and controllability. The method proposed in this paper achieves a significant improvement in design efficiency and precise control of design details by converting the design scheme into a token sequence consisting of the positioning and orientation data of the equipment bounding box. Nevertheless, this method is limited in its applicability to the diversity of substation types and plane forms. In the future, it is necessary to expand and fine-tune the data set to enhance its adaptability. At the same time, exploring more reasonable layout serialization forms and effectively converting design ideas in actual engineering projects into data that can be understood by the model are also key directions for improvement.

Acknowledgements

This project is funded by the Shanghai Municipal Electric Power Company's technology project "Research on Key Technologies for AI Intelligent Design of 10kV Substations" SGTYHT/24-JS-001.

References

Agarwala, M., Li, W., & Berthouzoz, F. (2011). *Design principles for visual communication*. Communications of the ACM, 54(4).

Arroyo, D. M., Postels, J., & Tombari, F. (2021). *Variational transformer networks for layout generation*. In CVPR.

Gupta, K., Achille, A., Lazarow, J., Davis, L., Mahadevan, V., & Shrivastava, A. (2021). *LayoutTransformer: Layout generation and completion with selfattention*. In ICCV.

Jiang, Z., Sun, S., Zhu, J., Lou, J.-G., & Zhang, D. (2022). *Coarse-to-fine generative modeling for graphic layouts*. In AAAI.

Kikuchi, K., Simo-Serra, E., Otani, M., & Yamaguchi, K. (2021). *Constrained graphic layout generation via latent optimization*. In ACM MM.

Kong, X., Jiang, L., Chang, H., Zhang, H., Gong, H., & Essa, I. (2022). *BLT: Bidirectional layout transformer for controllable layout generation*. In ECCV.

LLAMA 3.1. Retrieved from github website: https://github.com/meta-llama/llama-models/blob/main/models/llama3_1/MODEL_CARD.md

Zhao, W. X., et al. (2023). *A survey of large language models*. arXiv preprint, 2303.18223.

Lv, K., et al. (2023). *Full parameter fine-tuning for large language models with limited resources*. arXiv preprint, 2306.09782.

Zheng, Y., et al. (2024). *Llamafactory: Unified efficient fine-tuning of 100+ language models*. arXiv preprint, 2403.13372.

Hu, E. J., et al. (2021). *LoRa: Low-rank adaptation of large language models*. arXiv preprint, 2106.09685.

Wei, J., et al. (2022). *Chain-of-thought prompting elicits reasoning in large language models*. Advances in Neural Information Processing Systems, 35, 24824–24837.

LLAMA-Factory. Retrieved from github website: https://github.com/hiyouga/LLAMA-Factory

VLLM. Retrieved from github website: https://github.com/vllm-project/vllm