Proceedings of the Creative Construction Conference (2024)
Edited by: Miroslaw J. Skibniewski & Miklós Hajdu
https://doi.org/10.3311/CCC2024-175

NAMED ENTITY RECOGNITION FOR CONSTRUCTION DOCUMENTS BASED ON FINE-TUNING OF LARGE LANGUAGE MODELS

Junyu Zhou[1], Zhiliang Ma[2]
1 Tsinghua University, Beijing, China, zhou-jy21@mails.tsinghua.edu.cn
2 Tsinghua University, Beijing, China, mazl@tsinghua.edu.cn

Abstract
Named Entity Recognition (NER) is a necessary task for automatic processing of construction documents. In traditional methods, machine learning has been used, but they rely on large high-quality datasets that are manually made and costly to obtain. Therefore, this paper proposes a method of NER based on fine-tuning of Large Language Models (LLMs) for information extraction of construction documents. Firstly, low-quality datasets are semi-automatically generated from national standards, professional qualification textbooks, input method editor lexicons, including a generation-type dataset, a tagging-type dataset, and a question-answering dataset. Then, the above datasets are used to fine-tune an LLM for NER of structural elements to obtain optimal parametric conditions for fine-tuning. Finally, the optimal conditions are used to fine-tune the LLM and the latter was evaluated manually based on an established dataset and evaluation rules. The accuracy and completeness of the method are significantly improved compared to the LLM before fine-tuning, proving that the method works well. The research contributes to providing a more efficient method for automatic processing of construction documents.

Peer-review under responsibility of the scientific committee of the Creative Construction Conference 2024.

Keywords: construction documents, large language model, named entity recognition.

1. Introduction

Construction documents, such as construction organization plans, construction logs and safety reports, are important assets of construction enterprises for reuse in the projects of building construction. On the one hand, construction enterprises need to manage hundreds and thousands of projects, and each project has hundreds and thousands of construction documents. As the number of projects increases, managing so much documents can become challenging. Especially, retrieving specific documents from a vast collection can be very time-consuming and laborious. On the other hand, as projects are finished, knowledge of construction accumulates at both project and enterprise levels. Currently, because there is a lack of efficient methods for the retrieval of construction documents, it is difficult for the enterprises to utilize knowledge from the past effectively by either directly referring to documents or using big data technologies. To address the challenges, researchers have begun to apply Natural Language Processing (NLP) to automate or semi-automate information extraction of construction documents.

As a fundamental task for NLP, Named Entity Recognition (NER) is used to automatically extract named entities of specific categories from text, such as personal names, place names, proper nouns. Through NER, computers can identify important information or research objects from sentences expressed in natural language. Extracted named entities are normally critical not only for tasks such as text classification, text filtering, and establishment of lexicons, but also for other complex analysis or calculations. For example, Kim et.al [1] carried out NER about causes, losses and locations of accidents from safety reports to estimate potential construction risks. Zheng et. al [2] conducted NER about the constrained elements, content and preconditions in building fire protection standards for compliance check automatically. Ko et. al [3] did NER about changes, objects and losses from construction changes documents to infer potential construction changes in current project.

Existing commonly used NER methods are mainly based on statistics, logical rules, and machine learning [4]. However, applying these methods to construction documents still relies on large amounts of manual works including designing statistic features and logical patterns or rules, as well as manually tagging datasets for machine learning. Among them, the most promising category of methods is that based on machine learning, such as Bi-LSTM and CRF, but they are costly because they require high-quality datasets. For example, to establish a high-quality dataset using BIO tags, manual or specifically designed algorithm are required to accurately tag the Beginning (B), the Inside (I) words of named entities and words Outside (O) of named entities in sentences. Different B and I tags are needed to distinguish different types of named entities. If there comes a new type of named entity, new datasets and retraining of models are needed. Additionally, because various departments are involved in construction, the sources and types of named entities are complex, while open access datasets are rare and difficult to reuse. For Chinese NER, datasets are even harder to find. This situation hinders the use of NER method based on machine learning in the construction up to now.

LLMs has provided powerful new tools for NER and reduced the datasets requirements for training models [5]. LLMs represent large deep learning language models that have undergone extensive pre-training [6], typically with billions or even hundreds of billions of parameters. By full fine-tuning or parameter efficient fine-tuning, LLMs can perform various NLP tasks such as NER, question- answering, and text classification. Since 2023, generative LLMs, represented by GPT-4 (Generative Pre-training Transformer-4), have shown impressive performance. Currently, commonly used LLMs are all generative LLMs [7]. They are expected to generate answers corresponding to prompts, and can perform NER simply by modifying the prompts without fine-tuning. If fine-tuned, they are expected to achieve much better results in specialized domains. In addition, it has been indicated that LLMs fine-tuning worked even using low-quality datasets, including datasets semi-automatically generated based on logical rules [8], datasets automatically generated by other LLMs [9], and even datasets generated by the same LLMs before fine-tuning [10, 11].

Therefore, this paper proposes a method of NER based on fine-tuning of LLMs for information extraction of construction documents. Firstly, low-quality datasets are semi-automatically generated from national standards, professional qualification textbooks and input method editor lexicons, including generation-type datasets, tagging-type datasets, and question-answering datasets. Then, the above datasets are mixed to fine-tune an LLM called ChatGLM3-6B to obtain optimal parametric conditions for fine-tuning. Finally, the LLM that has been fine-tuned under the optimal conditions was evaluated manually based on an established dataset and evaluation rules. The accuracy and completeness of the method are significantly improved compared to the LLM before fine-tuning to validate the effectiveness of the proposed method.

2. Related works

2.1. NER based on traditional machine learning models

NER based on traditional machine learning models can be mainly categorized into 3 types [12].

- Sequence tagging based NER. For example, the Bi-LSTM-CRF proposed by [13]. The input of this model are natural language sentences, and the output is a sequence of tags for each word. For instance, when using BIO tags with N types of named entities, the model predicts the probability of each word being tagged as one of $2N + 1$ possible tags (N types of B tags, N types of I tags, and the O tag) and tag the word with the highest probability. For model training, high-quality datasets with BIO tags are a must.
- Position extraction and text classification based two-stage NER. For example, the Biaffine-NER proposed by [14]. This method enumerates possible positions of named entities in the sentences, and then classify the words at these positions belongs to which type of named entities. This method

can handle nested named entities well but has high computational complexity, making it difficult to manage long named entities.

- Generation based NER. For example, Yan et. al [15] used the generative model BART for NER. The model takes natural language sentences as input and generates outputs composed of named entities and separating tags. This method has similar dataset requirements to sequence tagging based NER but does not require the model to output words outside named entities.

Currently, for construction documents, NER based on traditional machine learning models are generally sequence tagging based. For instance, Moon et. al [16] used RNN to recognize named entities such as bridge components and damage causes from inspection reports. Jeon et. al [17] employed BERT to recognize 23 types of named entities in building MEP maintenance reports. Tang et. al [18] compared models including CRF, Bi-LSTM, and Bi-LSTM+CRF to recognize named entities related to cost-estimation from construction punch list.

2.2. NER based on LLMs

In such fields as biology, chemistry, medicine, and literature, researchers have already applied LLMs to NER. According to the form of prompts, NER based on LLMs can be divided into tagging-type NER and generation-type NER. Prompts of the former describe the definitions of the named entities, provide a target sentence, and instruct an LLM to output the target sentence with separation tags or special symbols added before and after the named entities. Prompts of the latter describe the definitions of the named entities, provide a target sentence, and instruct the LLM to directly output the named entities.

Some researchers have not fine-tuned the LLMs. For example, Jung et. al [19] did prompt engineering on ChatGPT-4 for NER of biochemical named entities, achieving a precision of 0.836 in tests on five scientific papers. Yang et. al [20] compared the performance of ChatGPT with traditional machine learning models like Bi-LSTM and LSTM+CRF in NER of diseases, finding that ChatGPT outperformed traditional models on the CMeEE and cMedQANER datasets, with precision of 0.72 and 0.84. Tsitseklis et. al [21] used ChatGPT for tagging-type NER of named entities related to biological museum exhibits, achieving a precision of 0.92. Hu et. al [22] compared the performance among GPT-4, LLaMA, and ChatGLM on generation-type NER for classical Chinese named entities, with GPT-4 showing the best, achieving a F1 score (the harmonic mean of precision and recall) of 0.65.

Some researchers fine-tuned the LLMs. For example, Zhao et. al [23] fine-tuned GPT-3 for generation-type NER of organic chemical named entities, achieving a precision of 0.81. Keloth et. al [24] fine-tuned LLaMA for tagging-type NER of diseases, biochemicals, and genes, finding that LLMs fine-tuned with 3 NER datasets performed better on recognizing all 3 types of named entities than LLMs fine-tuned with a single dataset. Luo et. al [25] fine-tuned Qwen-7b using 140 medicine NLP datasets, enhancing its performance across various NLP tasks, achieving a precision of 0.76 in generation-type NER.

It seems that in biology, chemistry and medicine, LLMs without fine-tuning can perform comparably to traditional machine learning models. Moreover, LLMs that have been fine-tuned with high-quality datasets exhibit stronger on NER in specialized domains, but the cost to establish is also big while some of the datasets are available publicly. For example, for the medicine NER dataset CMeEE, 32 experts participated in the tagging, 2.2 million words are tagged, and the dataset has been downloaded over 20,000 times [26].

In contrast, there is a lack of open access high-quality datasets about the building construction. In most case, researchers started their research by creating their own high-quality datasets [1, 2, 3, 16, 17, 18]. Consequently, NER based on LLMs from other fields are not easily applicable to construction documents. Additionally, when using low-quality datasets for fine-tuning, it remains to be found out which prompt format is suitable.

3. Methodology

3.1. Establish low-quality datasets semi-automatically

This paper focuses on structural elements, because it frequently appears in construction documents and are commonly concerned by researchers for code compliance check [2], cost estimation [18], etc.

This paper establishes 3 datasets: a generation-type dataset, a tagging-type dataset, and a question-answering dataset. The generation-type and tagging-type dataset are NER datasets, used for fine-tuning an LLM to enhance its NER capabilities. Examples of data included in these datasets are shown in Figures 1 and 2. The question-answering dataset is used for the fine-tuning of the LLM jointly with the NER datasets. An example of its data is shown in Figure 3. Generally, to fine-tune the LLM to perform NER, NER datasets alone is sufficient, like research in [23, 24]. However, considering that using diverse data from various sources for different tasks like NER, question-answering, text classification, and relation extraction might better improve LLMs in specialized domains [25, 27], this paper also employs a question-answering dataset jointly with the NER datasets to test if it can improve the LLM on NER.

Because LLMs can only produce low quality of NER datasets [28], this study employs a semi-automatic method to establish the generation-type and tagging-type datasets to achieve better fine-tuning results. Nonetheless, both generated datasets are still considered low-quality datasets because they are created using computer systems.

The process of establishing the generation-type and tagging-type datasets involves the following 4 steps.

- Step 1 Collect raw corpus: 289 Chinese national standards related to structures, 4 professional qualification textbooks, and 3 building construction lexicons from Chinese input method editors are collected as the raw corpus.
- Step 2 Process raw corpus: Data cleaning was performed on the raw corpus, including text deduplication, removal of special symbols, and elimination of tables. The standards and textbooks were segmented into sentences, resulting in the establishment of standard corpus, textbook corpus, and lexicon corpus.
- Step 3 Establish named entity set: Using a regex-based extraction algorithm, approximately 3,700 named entities were extracted from the lexicon corpus and the "Terms and Symbols" sections of the standards corpus. After manual screening, 335 structural elements named entities were retained, forming a named entity set.
- Step 4 Create datasets: A string matching algorithm was used to detect the presence of the named entities included in the named entity set within the sentences of the standards corpus. If a named entity was found, the sentence was used to create positive case data; otherwise, it was used to create negative case data. These data were then automatically used to create the generation-type and tagging-type datasets by using specific templates. Each dataset contained 6,729 positive cases and 6,000 negative cases, with approximately 2.5 million words covering 303 named entities from the named entity set. It deserves to explain that the initial amount of negative cases in the standards corpus was about 1,000 times that of the positive cases. So, only a similar magnitude of negative cases to positive cases are randomly sampled.

In establishing the question-answering dataset, this research utilized the question-answering dataset mining feature based on ERNIE-3.0 provided by Baidu Qianfan AI Cloud Platform [29]. The standards corpus and textbook corpus were input into ERNIE-3.0 in the form of segmented sentences to automatically generate the question-answering dataset. It is noteworthy that the textbook corpus was not used when creating the NER datasets because it was reserved for validation dataset. The reason for choosing this platform is that Baidu, the platform provider, is a leading company with a self-developed Chinese LLM. Using an excellent LLM with hundreds of billions of parameters like ERNIE-3.0 through its platform is cost-effective and does not require additional hardware or software. A question-answering dataset containing 63,597 data are generated. The generated questions by using an LLM might have conceptual errors, and the answers might be inaccurate or incomplete, thus this dataset is also low-quality.

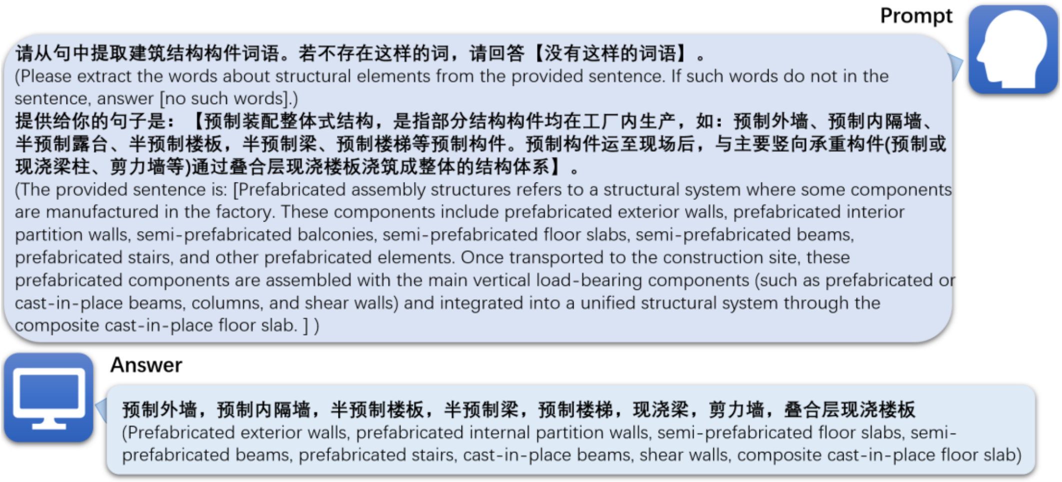

Fig. 1. Example of data included in generation-type dataset.

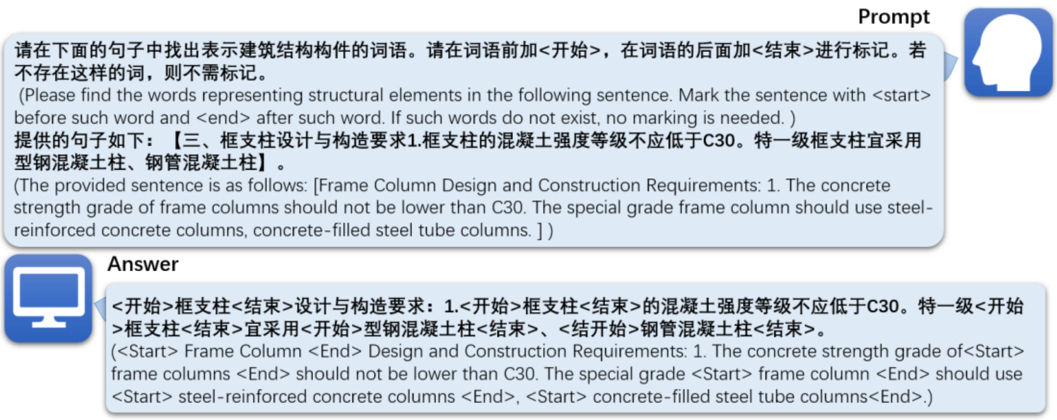

Fig. 2. Example of data included in tagging-type dataset.

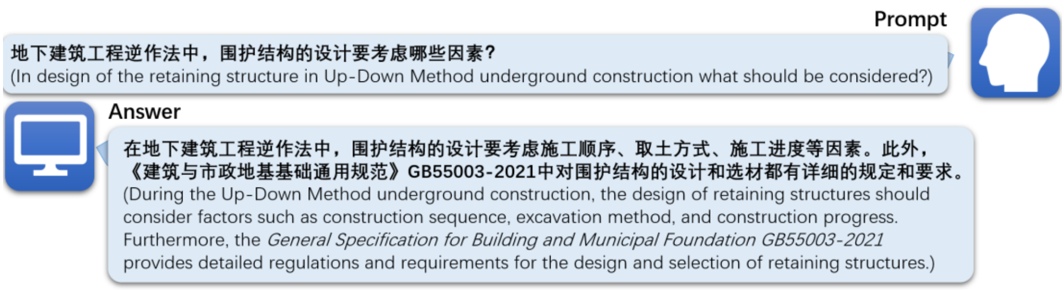

Fig. 3. Example of data included in question-answering dataset.

3.2. LoRA fine-tuning on ChatGLM3-6B

ChatGLM3 [30] is an open-source pre-trained LLM released by Zhipu AI and the KEG lab at Tsinghua University. ChatGLM3-6B, a version of ChatGLM3 with 6.2 billion parameters, is smaller in size, easier to deploy, and ranks first among general open-source LLMs under 10 billion parameters in Chinese NLP tasks according to C-Eval [31]. Therefore, this research chose ChatGLM3-6B as the base LLM.

Methods for fine-tuning of LLMs include normally full fine-tuning, prefix tuning, LoRA, and adapters [32]. LoRA (Low-Rank Adaptation), proposed in 2021, is a parameter efficient fine-tuning method [33]. For instance, when the full fine-tuning needs a parameters update matrix of size $s \times d$, LoRA approximates this by training two smaller matrices of size $s \times r$ and $r \times d$, in which $r \ll \min(s, d)$. This significantly reduces the number of parameters fine-tuned. The LoRA method enables individual researchers to fine-tune LLMs without excessive computational resources, thus reducing time and computational costs. This study chose LoRA because it is early-established and effective, with many open-source Chinese LLMs like ChatGLM, Qwen, and LLaMA-Chinese natively supporting LoRA fine-tuning.

In this research, the open-source code of ChatGLM3-6B was downloaded, deployed locally, and its pre-trained weights were loaded. This required 14GB of disk and a RTX 4090D GPU with 24GB of VRAM for LoRA fine-tuning. The average fine-tuning time per data was between 1 to 2 seconds.

3.3. Fine-tuning experiments

This research conducted 3 groups of fine-tuning experiments to obtain optimal conditions for fine-tuning.

- Group 1: This group varied the ratio of data used for fine-tuning from the generation-type and tagging-type datasets, aiming to compare which dataset composition is more beneficial for NER. This group used 6000 positive cases for fine-tuning, with ratios of 0%, 20%, 40%, 50%, 60%, 80%, and 100% from the tagging-type dataset, and the remainder from the generation-type dataset.
- Group 2: This group varied the ratio of positive and negative cases used for fine-tuning, aiming to explore how to set the negative cases to reduce hallucinations and prevent the LLM from fabricating hallucination when positive cases are much smaller than negative cases. The hallucination refers to the LLMs generating seemingly reasonable but incorrect outputs. This group used the generation-type dataset, with 6000 positive cases and 0, 500, 1000, …, 3000 negative cases.
- Group 3: This group varied the ratio of data used for fine-tuning from the generation-type and question-answering datasets, aiming to explore whether it is possible to use easily accessible question-answering dataset to replace the more difficult-to-obtain NER dataset. This group used a total of 6,000 data in which the data from the question-answering dataset varied from 0 to 5,000 in increments of 1,000, and the remain are from the generation-type dataset.

After determining the optimal fine-tuning condition in each group, this research combined the conditions to fine-tune ChatGLM3-6B and tested its capabilities on NER of structural elements in textbook corpus.

The fine-tuning set the same hyperparameters, which have been validated to prevent overfitting and catastrophic forgetting, and the fine-tuning loss function has stabilized. Specifically, the fine-tuning was conducted with 2 epochs, a learning rate of 0.00005, a cosine learning rate decay strategy, and 8 gradient accumulation steps. For LoRA method, the LoRA rank was set to 8, the LoRA alpha was set to 16 recommended in [33], and a dropout of 0.1. Furthermore, 10% of the data from the fine-tuning dataset was reserved as the test set. Testing was conducted every 200 steps during fine-tuning, and the best-performing LLM was retained. This facilitates comparison under different conditions.

3.4. Evaluation of controlled experiments

Due to the lack of high-quality data sets, this research evaluated the fine-tuning effect of the LLM by manual scoring. Referring to a literature [34], the standard of manual scoring is designed, which includes 3 dimensions: completeness, accuracy and stability. The completeness refers to whether the LLM answers the question comprehensively, which reflects the recall. The accuracy refers to whether the LLM answers correctly, which reflects the precision. The stability refers to whether the answer of the LLM can be understood or produces no harmful information, reflecting whether the LLM generates hallucination or errors. Each dimension was assigned values of 0, 0.5, or 1, and specific scoring criteria were designed as follows, considering the requirements of the NER task.

- Completeness: A score of 0 indicates that less than 20% of the named entities were recognized in the positive cases, or that negative cases were mistakenly identified as positive. A score of 1 indicates that more than 80% of the named entities were recognized in the positive cases, or that negative cases were correctly identified. Other cases receive a score of 0.5.

- Accuracy: A score of 0 indicates that none of the recognized named entities are correct. A score of 1 indicates that all recognized named entities are correct. Other cases receive a score of 0.5.
- Stability: A score of 0 indicates that the named entities recognized by the LLM are all hallucinations that do not correspond to the prompts, or that loop words occur. A score of 1 indicates that all named entities recognized can be found in the prompts. Other cases receive a score of 0.5.

The evaluation consists of 3 steps. First, the textbook corpus is used to generate a validation dataset of generation-type. Totally, 35 positive cases are manually selected containing the most named entities and the most total words of named entities, and 15 negative cases are randomly sampled. The prompts were answered manually to create the validation dataset. Then, the fine-tuned LLM was used to answer the prompts in the validation dataset. For the responses, the word selection strategy used in generating text by the LLM followed the default setting of ChatGLM3-6B, where the threshold for the Top-p sampling strategy was set to 0.7 and the coefficient for the temperature sampling strategy was set to 0.95. Finally, manual comparisons were made between the human and LLM answers, and scores were assigned.

4. Evaluation

The results of Group 1 are shown in Tables 1 and 2. The comparison between Tables 1 and 2 reveal that, under the premise of fine-tuning with positive case data, the performance of the LLM in completeness and accuracy is generally higher in tagging-type NER compared to generation-type NER. It is worth noting the data in the first row of Table 1, where the LLM tends to adopt a conservative strategy, providing answers with just one short named entity, but with high accuracy. For instance, regarding the named entities ‘sheet pile retaining wall’, ‘interlocking piles retaining wall’ and ‘gravity retaining wall’ in a sentence, the LLM only tags ‘retaining wall’, leading to high accuracy but the lowest completeness. This strategy also results in higher accuracy in the negative case data, as it can directly answer sentences without any tag. As for the generation-type NER, the LLM failed to recognize negative cases (0 completeness). For example, in the last row of Table 2, among the answers to the 15 negative cases, 11 are with an incorrect named entity, such as "chimney" and "rolled material," and 4 are hallucinations with a named entity that is correct but not mentioned in the provided sentence, such as ‘concrete component’ when neither ‘concrete’ and ‘component’ were not mentioned. This group demonstrates that negative case data is indispensable.

Table 1. Results of Group 1 (fine-tuned LLM preforms tagging-type NER).

Fine-tuning condition	Average in positive case data			Average in negative case data		
	completeness	accuracy	stability	completeness	accuracy	stability
6000 tagging-type&0 generation-type	**0.386**	**0.971**	1.000	0.933	0.867	0.000
4800 tagging-type&1200 generation-type	0.353	0.897	1.000	0.813	0.875	0.938
3600 tagging-type&2400 generation-type	0.300	0.700	1.000	0.438	0.531	0.875
3000 tagging-type&3000 generation-type	0.314	0.686	0.971	0.250	0.344	1.000
2400 tagging-type&3600 generation-type	0.257	0.514	0.986	0.281	0.344	0.800
1200 tagging-type&4800 generation-type	0.271	0.514	0.629	0.063	0.125	0.688

Table 2. Results of Group 1 (fine-tuned LLM preforms generation-type NER).

Fine-tuning condition	Average in positive case data			Average in negative case data		
	completeness	accuracy	stability	completeness	accuracy	stability
4800 tagging-type&1200 generation-type	0.414	0.914	0.914	**0.000**	0.167	0.800
3600 tagging-type&2400 generation-type	0.414	0.929	0.986	**0.000**	0.333	0.600
3000 tagging-type&3000 generation-type	0.371	0.929	0.986	**0.000**	0.200	0.800
2400 tagging-type&3600 generation-type	0.400	**0.943**	0.971	**0.000**	0.267	0.667
1200 tagging-type&4800 generation-type	0.414	0.914	0.971	**0.000**	0.433	0.600
0 tagging-type&6000 generation-type	**0.429**	0.914	0.957	**0.000**	0.333	0.533

The results of Group 2 are shown in Figure 4. It shows that by using negative case data equivalent to only 1/12 of the positive case data, the fine-tuned LLM start to recognize negative cases, thus avoiding hallucinations. The accuracy and stability of the LLM in positive case data can be improved with negative case data when negative cases do not exceed 1/3 that of positive ones. The best results are achieved when it is 1/4. However, when it exceeds 1/3, the completeness reduced comparing to when no negative case data is added. This is because the negative case data is also of low-quality, containing some sentences that should have been considered positive cases because they contain named entities not in established named entity set. With an increase in negative cases, the influence of its low-quality is amplified, making the LLM more likely to misidentify positive cases as negative cases.

The results of Group 3 are shown in Figure 5. It shows that the use of question-answering dataset can indeed improve the performance of the LLM on completeness. The best performance is achieved when the amount of data from the question-answering dataset is equivalent to 1/2 of that from the generation-type dataset. This implies that in cases when researchers severely lack NER datasets, using question-answering datasets for fine-tuning can help the LLM learn named entities indirectly. However, the data used for fine-tuning still needs to be primarily composed of NER datasets (no less than 1/2 of the total), as otherwise, the LLM may tend to recognize the more common and simpler named entities in the question-answering dataset, reducing the completeness.

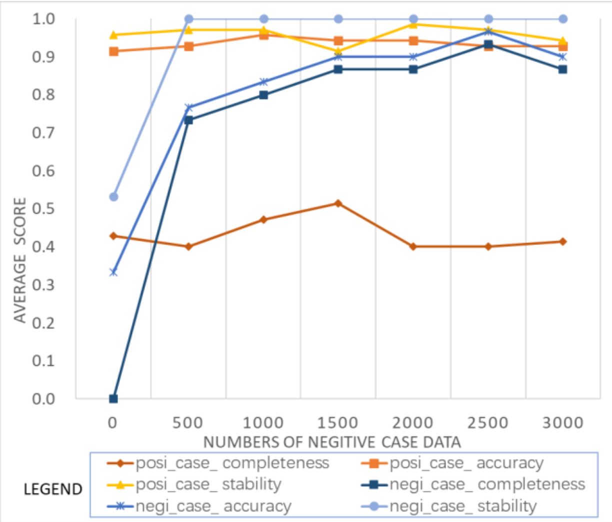

Fig. 4. Results of Group 2 (fine-tuned LLM preforms generation-type NER).

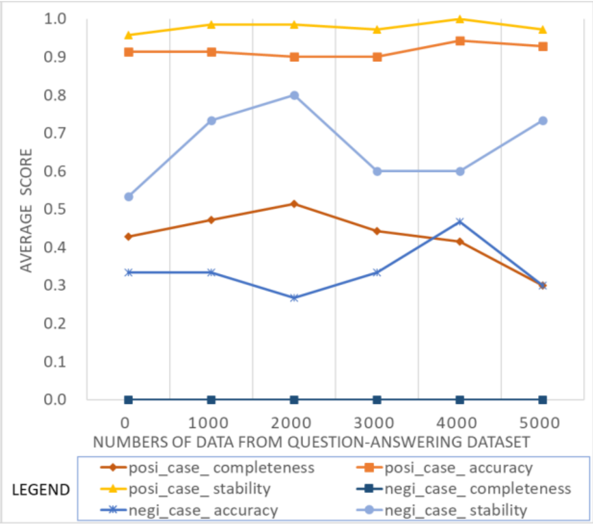

Fig. 5. Results of Group 3 (fine-tuned LLM preforms generation-type NER).

Based on the optimal conditions for fine-tuning from the above 3 groups of experiments, as a controlled experiment, this research fine-tuned ChatGLM3-6B using 10,500 data, including 6,000 positive cases and 1,500 cases from the generation-type dataset, and 3,000 data from the question-answering dataset.

The results show a significant improvement compared to the LLM before fine-tuning, as shown in Table 3. Examples of the answer from the LLM before and after fine-tuning are shown in Tables 4 and 5. It shows that before fine-tuning, the LLM could hardly distinguish structural elements, building materials, and element attributes, and it generated hallucinations in response to negative cases. After fine-tuning, the completeness for positive cases improved by 40%, the accuracy nearly doubled, and the stability remained consistent. For negative cases, completeness improved by 11%, accuracy improved by about 3 times, and completeness improved by 15%. It indicates that the proposed method significantly improves the recognition accuracy and completeness compared to the LLM before fine-tuning.

Table 3. Results from LLM before and after fine-tuning.

Fine-tuning condition	Average in positive case data			Average in negative case data		
	completeness	accuracy	stability	completeness	accuracy	stability
After	**0.514**	**0.943**	**0.971**	**0.800**	**0.833**	**1.000**
Before	0.368	0.309	0.971	0.067	0.200	0.867

Table 4. Example of positive case from LLM before and after fine-tuning (translated from Chinese).

Prompt	Please extract the words related to structural elements from provided sentence. If there is no such word, please answer [There is no such word.]. The provided sentence is: [Figure 13.4.20 Connection between Reinforced Secondary Beam and Main Beam (2) Reinforced Angle Steel Beam Support Flat Steel Belt Figure 13.4.21 Connection between Reinforced Frame Beam and Frame Column (1) Section Four Reinforcement of Cast-in-Place Beams (2) Connection with Transmission Flat Steel Belt or Steel Bar: At the frame node, a transmission flat steel belt or steel bar can be used to bypass the outer side of the column, and welded at the top of the reinforced beam with the reinforced flat steel belt; welded at the bottom of the reinforced beam with the reinforced angle steel, the cross-sectional area of the transmission flat steel or steel bar should not be less than the cross-sectional area of the flat steel belt or angle steel on the reinforced beam, and the influence of its bending angle on the tension should be considered. The bending slope should not exceed 1:3.]
Answer from LLM before fine-tuning	The structural elements are: reinforced secondary beam, main beam, reinforced angle steel beam support, transmission flat steel belt, steel bar, frame node, reinforced frame beam, frame column, cast-in-place beam, transmission flat steel belt or steel bar, reinforced flat steel belt, angle steel, bending angle, bending slope.
Answer from LLM after fine-tuning	Frame beam, secondary beam, main beam, angle steel beam, cast-in-place beam, frame column.
Result explanation	The correct answer: secondary beam, main beam, frame beam, frame column, cast-in-place beam. It shows that before fine-tuning, the accuracy was low. After fine-tuning, the accuracy improved significantly.

Table 5. Example of negative case from LLM before and after fine-tuning (translated from Chinese).

Prompt	Answer the structural elements in provided sentence. If there is no such word, please answer [There is no such word.]. The provided sentence is: [(2) Public buildings mainly refer to buildings for various public activities, including office buildings, cultural and educational buildings, scientific research buildings, medical buildings, commercial buildings, etc.].
Answer from LLM before fine-tuning	Public building structural elements include: Pillar: A structural element that supports the building, usually used for bearing weight. Wall: The enclosing structure of a building, including load-bearing walls and non-load-bearing walls. Floor Slab: A structural element that connects the upper levels of a building to support the weight above. Beam: A structural element that connects the upper levels of a building to support the weight above. Roof Truss: A structural element that connects the upper levels of a building to support the weight of the roof.
Answer from LLM after fine-tuning	There is no such word.
Result explanation	The correct answer: there is no such word. It shows that before fine-tuning, the LLM generated hallucinations and fabricated named entities that did not exist in the sentence. After fine-tuning, the negative cases were correctly identified.

5. Conclusions

To overcome the problem of high-quality dataset requirements for traditional NER methods on construction documents, this research proposes an NER method for construction documents based on fine-tuning of LLMs, in which only low-quality datasets are required. Evaluation of fine-tuning experimental results have demonstrated the effectiveness of the method, with the NER completeness improved by 40% and accuracy increased approximately 2 times compared to that before fine-tuning.

It is expected that by using techniques such as few-shot learning and Retrieval-Augmented Generation (RAG), NER could be more accurate. As future research steps, we will combine these techniques to make it easier to perform NER in specialized domains with limited datasets based on LLMs. Additionally, we will validate the transferability of this NER method and explore whether the excellent general capabilities of LLMs can help to recognize other types of named entities not tagged in the dataset.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (Grant No. 52378305).

References

[1] T. Kim, S. Chi. “Accident case retrieval and analyses: using natural language processing in the construction industry”, *J. Constr. Eng. M.*, vol. 145, no. 3, 2019, DOI: https://doi.org/10.1061/(ASCE)CO.1943-7862.0001625.

[2] Z. Zheng, Y. Zhou, X. Lu, et. al. “Knowledge-informed semantic alignment and rule interpretation for automated compliance checking”, *Automat. Constr.*, vol. 142, 2022, DOI: https://doi.org/10.1016/j.autcon.2022.104524.

[3] T. Ko, H. D Jeong, G. Lee. “Natural language processing–driven model to extract contract change reasons and altered work items for advanced retrieval of change orders”, *J. Constr. Eng. M.*, vol. 141, no. 11, Nov. 2021, DOI: https://doi.org/10.1061/(ASCE)CO.1943-7862.0002172.

[4] L. Lui, D. Wang. “A review on named entity recognition”, *JCSSTI*, vol. 37, no. 3, pp. 329-340, Nov. 2018. (in Chinese)

[5] D. Xu, W. Chen, W. Peng, et al. “Large language models for generative information extraction: a survey”, 2023, *arXiv: 2312.17917*.

[6] A. J. Thirunavukarasu, D. S. J. Ting, K. Elangovan, et al. “Large language models in medicine”, *Nat. Med*, vol. 28, no. 8, pp. 1930-1940, Jul. 2023, DOI: https://doi.org/10.1038/s41591-023-02448-8.

[7] H. Yan, Y. Liu, L. Jin, et. al. “The development, application, and future of LLM similar to ChatGPT”, *J. Imag. Graph.*, vol. 28, no. 9, pp. 2749-2762, Dec. 2023. (in Chinese)

[8] G. Iker, A. Begoña, A. Javier, et. al. “This is not a dataset: a large negation benchmark to challenge large language models”, 2023, *arXiv: 2310.15941.*

[9] Y. Yu, Y. Zhuang, J. Zhang, et. al. “Large language model as attributed training data generator: a tale of diversity and bias”, 2023, *arXiv: 2306.15895.*

[10] Y. Wang, K. Yeganeh, S. Mishra, et al. “Self-instruct: aligning language models with self-generated instructions”, 2023, *arXiv: 2212.10560.*

[11] S. Wang, X. Sun, X. Li, et. al. “GPT-NER: Named entity recognition via large language models”, 2023, *arXiv: 2304.10428*.

[12] Z. Wang, Q. Zhou, J. Zhao, et. al. “A knowledge-enhanced medical named entity recognition method that integrates pre-trained language models”, *2023 IEEE Int. Conf. MedAI*, Beijing, China, 2023, pp. 296-301.

[13] Z. Huang, W. Xu, K. Yu. “Bidirectional LSTM-CRF models for sequence tagging”, 2015, *arXiv: 1508.01991*.

[14] J. Yu, B. Bohnet, M. Poesio, et al. “Named entity recognition as dependency parsing”, 2020, *arXiv:2005.07150.*

[15] H. Yan, T. Gui, J. Dai, et. al. “A unified generative framework for various NER subtasks”, 2021, *arXiv:2106.01223*.

[16] S. Moon, S. Chung, S. Chi. “Bridge damage recognition from inspection reports using NER based on recurrent neural network with active learning”, *J. Perform. Constr. Fac.*, vol. 34, no. 6, Dec. 2020, DOI: https://doi.org/10.1061/(ASCE)CF.1943-5509.0001530.

[17] K. Jeon, G. Lee, S. Yang, et. al. “Named entity recognition of building construction defect information from text with linguistic noise”, *Automat. Constr.*, vol. 143, Sep. 2022, 143. DOI: https://doi.org/10.1016/j.autcon.2022.104543.

[18] S. Tang, H. Liu, M. Almatared, et. al. “Towards automated construction quantity take-off: an integrated approach to information extraction from work descriptions”, *Buildings*, vol. 12, no. 3, Mar. 2022.

[19] S.J. Jung, H. Kim, K.S. Jang, et. al. “LLM based biological named entity recognition from scientific literature”, *2024 IEEE Int. Conf. BigComp*, 2024, pp. 433-435, DOI: https://doi.org/10.1109/BigComp60711.2024.00095.

[20] B. Yang, X. Sun X, J. Dang, et. al. “Named entity recognition method of large language model for medical question answering system”, *FCST*, vol. 17, no. 10, pp. 2389-2402, 2023. (in Chinese)

[21] K. Tsitseklis, G. Stavropoulou, S. Papavassiliou, et. al. “Custom named entity recognition VS ChatGPT prompting: a paleontology experiment”, *2024 PACET*, 2024, pp. 82-85.

[22] D. Hu, G. Sun, L. Liu, et. al. “Evaluation of ancient Chinese natural language understanding in large language models based on ACHNLU”, *Wisdom, Well-Being, Win-Win: 19th Int. Conf.*, 2024, pp. 3-18.

[23] X. Zhao, K. Langlois, J. Furst, et. al. “When LLM meets material science: an investigation on MOF synthesis labeling”, *2023 IEEE Int. Conf. BigData*., 2023, pp. 6320-6321.

[24] V.K. Keloth, Y. Hu, Q. Xie, et. al. “Advancing entity recognition in biomedicine via instruction tuning of large language models”, *Bioinformatics*, vol. 40, no. 4, Mar. 2024.

[25] L. Luo, J. Ning, Y. Zhao, et. al. “Taiyi: a bilingual fine-tuned large language model for diverse biomedical tasks”, *J. Am. Med. Inform. ASSN*, 2024, DOI: https://doi.org/10.1093/jamia/ocae037.

[26] N. Zhang, M. Chen, B. Zhen, et. al. “CBLUE: a Chinese biomedical language understanding evaluation benchmark”, 2022, *arXiv: 2106.08087.*

[27] C. Zhou, P. Liu, P. Xu, et. al. “LIMA: Less is more for alignment”, 2023, *arXiv:2305.11206.*

[28] B. Bhavya, P.T. Isaza, Y. Deng, et. al. “Exploring large language models for low-resource it information extraction”, *2023 IEEE Int. Conf. on Data Mining Workshops*, 2023.

[29] Baidu. “Baidu Qianfan platform ModelBuilder”, Baidu Qianfan AI platform. https://qianfan.cloud.baidu.com/. (in Chinese)

[30] A. Zeng, X. Liu, Z. Du, et. al. “GLM-130B: an open bilingual pre-trained model”, 2023, *arXiv: 2210.02414.*

[31] Y. Huang, Y. Bai, Z. Zhu, et. al. “C-Eval: A multi-level multi-discipline Chinese evaluation suite for foundation models”, 2023, *arXiv:2305.08322.*

[32] J. He, C. Zhou, X. Ma, et. al. “Towards a unified view of parameter-efficient transfer learning”, 2022, *arXiv: 2110.04366.*

[33] E.J. Hu, Y. Shen, P. Wallis, et. al. “LoRA: Low-Rank Adaptation of large language models”, 2021, *arXiv:2106.09685*.

[34] O. David, X. Wu, S. Visweswaran, et. al. “Enhancing large language models for clinical decision support by incorporating clinical practice guidelines”, 2024, *arXiv: 2401.11120.*